

Capturing interpretational uncertainty of depositional environments with Artificial Intelligence

ATHANASIOS NATHANAIL

*A thesis submitted in fulfilment of the requirements
for the degree of Doctor of Philosophy*

Institute of GeoEnergy Engineering
School of Energy, Geoscience, Infrastructure and Society
Heriot-Watt University

June 2023

The copyright in this thesis is owned by the author. Any quotation from the thesis or use of any of the information contained in it must acknowledge this thesis as the source of the quotation or information.

ABSTRACT

Geological interpretations are always linked with interpretational and conceptual uncertainty, which is difficult to elicit and quantify, often creating unquantified risks for understanding the subsurface. The complexity and variability of geological systems may lead geologists to analyse the same data and arrive at different conclusions based on their subjective interpretations, personal expertise, or biases. In order to address the associated uncertainty, it is valuable to consider multiple plausible interpretations of outcrop data and acknowledge the degree of ambiguity associated with each interpretation. By examining a diverse range of outcrop analogues, it becomes possible to derive multiple potential geological interpretations and identify variations within and across depositional systems.

This thesis proposes a new AI system that learns valuable geological information from surface data (outcrop images), transfers this knowledge to the fragmented data of the subsurface (core data), and finally, links all the extracted information with the geological literature to produce plausible interpretations of the depositional environment based on a single outcrop image. To identify patterns and geological features within image data, three Supervised Learning Computer Vision techniques were employed: Image Classification, Object Detection, and Instance Segmentation. Natural Language Processing was utilised to extract geological features from textual information from heritage geological texts, thus complementing the analysis. Lastly, a custom Neural Network was deployed to assimilate the gathered information into meaningful sequences, apply geological constraints to these sequences, and generate multiple plausible interpretational scenarios, ranked in descending order of probability.

The results of this study demonstrate that combining approaches from different areas of Artificial Intelligence within cross-disciplinary workflows under the umbrella of a broader AI system holds significant potential for subsurface characterization, better risk analysis, and potentially enhancing decision-making under uncertain conditions during subsurface exploration stages.

DEDICATION

In Loving Memory of My Beloved Father,

Alexandros Nathanail

This thesis is dedicated to the artistic spirit, kindness, patience, and strength of my dear father, whose presence in my life was an endless source of inspiration.

Throughout my journey, he was my guiding light, instilling in me the values of nobleness, humbleness, perseverance, wisdom, and dedication.

Although he is no longer physically with us, his love, his beautiful music, and his encouragement continue to resonate within me, shaping my pursuit of knowledge and pushing me to reach new heights.

I am forever grateful for the time we shared and the invaluable lessons he taught me.

Dad, this is for you.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude and appreciation to all those who have supported me throughout my doctoral journey and contributed to the successful completion of this thesis. This research endeavor would not have been possible without their guidance, encouragement, and assistance.

I extend my sincere gratitude to my supervisors, Dr. Dan Arnold and Prof. Vasily Demyanov, for their invaluable guidance, mentorship, and unwavering support. Their expertise, insightful feedback, and scholarly guidance have played a pivotal role in shaping the research direction, enhancing the quality of this thesis, and nurturing my intellectual growth. I would also like to thank Dr. Andy Gardiner and Dr. Helen Lewis for their geological expertise and input in sedimentology and structural geology.

I am deeply thankful to my thesis committee members, Professors Mikhail Kanevski and Dorrik Stow, for their insightful comments, constructive suggestions, and critical evaluation of this research. Their expertise and scholarly input have immensely enriched the quality and rigor of this work.

I extend my heartfelt appreciation to my fellow researchers, Bastian Steffens, Quentin Corlay, Chao Sun, colleagues, and peers, especially Petar Kotchev. There were numerous stimulating discussions, valuable insights, and camaraderie that have greatly contributed to the intellectual environment within which this research flourished. Their willingness to engage in scholarly debates and exchange of ideas has been instrumental in shaping my research perspective.

Being part of the CDT for Oil & Gas, put together by Prof John Underhill, was also a pleasure. Many thanks and appreciation to Lorna Morrow and Anna Clark for organising fantastic field trips and training courses that allowed me to see so many different locations around the UK and Europe. I truly enjoy the collaborative community that came with the CDT, and it is great to see that through the duration of this project, I have built a network with my colleagues that extends beyond our PhDs and made friends all over the UK.

I would like to express my gratitude to NERC National Productivity Investment Fund (NPIF) and Heriot-Watt University for their financial support and the opportunity to pursue this research. Their investment in my academic journey has played a pivotal role in facilitating the successful completion of this thesis.

I am deeply indebted to my beloved mother, Maria Karvela, for her unwavering support, understanding, and encouragement throughout this challenging endeavor. Her priceless

help, love, patience, and belief in my abilities have been my constant source of motivation and inspiration.

Lastly, I would like to express my gratitude to all those individuals who, in one way or another, have contributed to my academic and personal growth. Although it is impossible to mention everyone individually, your support and encouragement have been invaluable to me.

Thank you all for your immeasurable contributions and for being a part of my academic journey.

Sincerely,

Athanasios Nathanail

DECLARATION STATEMENT



Submission of Thesis Declaration by Supervisor

Strictly Confidential

Candidate

Name (<i>in capitals</i>):	ATHANASIOS NATHANAIL	HW Person ID:	H00284491
School:	IGE	Degree Sought:	(G117-PEG) PhD Petroleum Geoscience

Declaration

I believe the thesis to contain the sole work of the candidate, with other work adequately referenced. Yes No

The thesis has been checked using an approved plagiarism detection application e.g. Turnitin. Yes No

I believe the thesis to be in a suitable presentational form and is ready for examination.* Yes No

* *For guidance on the format for presentation please refer to:*
<https://www.hw.ac.uk/students/studies/examinations/thesis.htm>

If you do not believe the thesis to be in a suitable form and should not be examined, or if you have any concerns about the thesis being submitted, please give details below:

Primary Supervisor

Print Name:	DAN ARNOLD	Date:	01/06/2023
Signature:		School:	IGE

Notes

This form should be completed after the thesis has been bound. The thesis must conform in layout, binding and presentation to the requirements prescribed by the Senate. The completed form must be lodged with the Student Service Centre at the time when the thesis is submitted.

TABLE OF CONTENTS

Capturing interpretational uncertainty of depositional environments with Artificial Intelligence	i
ABSTRACT	ii
TABLE OF CONTENTS	i
1. CHAPTER 1 – INTRODUCTION	1
1.1 High-Level Thesis Workflow	4
1.2 Thesis Outline and Chapters Objectives.....	10
2. CHAPTER 2 – GEOLOGY and ARTIFICIAL INTELLIGENCE FUNDAMENTAL CONCEPTS for this THESIS	14
2.1 Introduction	14
2.2 Geology Fundamental Background.....	14
2.2.1 Sedimentology	14
2.2.2 Outcrop Interpretation.....	15
2.2.3 Interpretation of Depositional Environments.....	16
2.3 Artificial Intelligence and Machine Learning Fundamental Background.....	18
2.3.1 Human Learning vs. Machine Learning in Learning from Outcrops.....	18
2.3.2 ML Important Components.....	22
2.3.3 Natural Language Processing.....	23
2.3.4 Metrics to Evaluate Computer Vision Models.....	24
2.3.5 Pre-trained Models and Backbones.....	25
2.3.6 Overview of Common Benchmark Datasets for Computer Vision Models 27	
2.4 Relevant Work on outcrop data using Computer Vision Methods	28
3. CHAPTER 3 - METHOD OVERVIEW	33
3.1 Introduction	33
3.2 Computer Vision Methods & Workflow	34

3.2.1	Image Classification.....	38
3.2.2	Object Detection.....	39
3.2.3	Image Segmentation Overview	42
3.3	Common Metrics Used for the Evaluation of Computer Vision Models.....	49
3.3.1	Confusion Matrix	49
3.3.2	Intersection over Union (IoU).....	50
3.3.3	Average Precision	50
3.3.4	Mean Average Precision (mAP)	50
3.4	Computer Vision Models & Model Backbones	52
3.4.1	Residual Networks: Resnet18	53
3.4.2	Residual Networks: Resnet 50	53
3.4.3	Residual Networks: Resnet 101	54
3.4.4	VGG 16	54
3.4.5	VGG 19	54
3.4.6	DarkNet 53	54
3.4.7	CSPDarknet53.....	55
3.5	Natural Language Processing (NLP).....	55
3.5.1	Document Processing and Information Extraction	56
3.6	Custom Neural Network to Interpret the Geology	59
3.7	Chapter’s Conclusions.....	61
4.	CHAPTER 4 - DATASETS DESCRIPTION.....	63
4.1	Introduction	63
4.2	Dataset-building workflow for Computer Vision applications in Geology	63
4.3	Data Collection.....	65
4.4	Data Cleaning and Preprocessing.....	66
4.4.1	Data Cleaning.....	66
4.4.2	Data Preprocessing.....	69
4.5	Data Augmentation.....	73

4.5.1	The Importance of Data Augmentation.....	73
4.5.2	Limitations of Data Augmentation.....	74
4.5.3	Data Augmentation Techniques in Computer Vision	75
4.6	Data Annotation	81
4.6.1	Annotation Types in Computer Vision	83
4.6.2	Data Labeling Approaches.....	85
4.6.3	Boundary Recognition	85
4.6.4	Bounding Boxes Annotations	86
4.6.5	Polygon Annotations.....	88
4.6.6	How to Encapsulate the Scale of the geological features	89
4.7	Geologic Datasets for Image Classification	90
4.7.1	Image Classification Datasets (Part 1)	92
4.7.2	Image Classification Datasets (Part 2).....	94
4.8	Geologic Dataset for Object Detection.....	98
4.9	Geologic Datasets for Instance Segmentation.....	102
4.9.1	Instance Segmentation Yolact with Mixed Labels (Dataset 9).....	103
4.9.2	Instance Segmentation (Dataset 10A & 10B)	105
4.10	Chapter’s Conclusion.....	107
5.	CHAPTER 5 - THE USE OF SKETCHES TO IMPROVE THE IMAGE CLASSIFICATION OF SEDIMENTARY STRUCTURES.....	109
5.1	Introduction	109
5.2	Previous Use of Sketches in Machine Learning.....	110
5.3	Proposed Workflow for the Classification of Geological Images.....	111
5.3.1	Dataset Selection.....	112
5.3.2	Dataset Split	112
5.3.3	Model Training.....	112
5.3.4	Feature Extraction	113
5.3.5	Model Testing and Evaluation	114

5.4	The Custom Model for Geological Image Classification.....	114
5.4.1	Part One: Establish Method, Dataset, and Model Parameters	115
5.4.2	Part Two: Build a robust geological image classifier	128
5.5	Chapter’s Conclusions.....	162
6.	CHAPTER 6 - IDENTIFYING MULTIPLE GEOLOGICAL FEATURES USING OBJECT DETECTION ON OUTCROP AND FOSSIL IMAGES.....	164
6.1	Introduction	164
6.2	The YOLOv6 Model	165
6.2.1	The YOLOv6 Backbone Architecture	167
6.2.2	Metrics for YOLOv6 Evaluation	168
6.3	Workflow for YOLOv6 applied to Geology	169
6.3.1	Step 1: Input Images and Preprocessing	169
6.3.2	Step 2: Dataset annotation.....	170
6.3.3	Step 3: Feature Extraction.....	171
6.3.4	Step 4: Anchor Box Selection & Object Detection.....	171
6.3.5	Step 5: Non-Maximum Suppression	172
6.3.6	Step 6: Output Image	172
6.4	Training and Testing of the YOLOv6-S Model on Geology	173
6.4.1	Experiment 1: Object Detection of Sedimentary Structures on Outcrop Images (Dataset 8).....	173
6.4.2	Experiment 2: Object Detection of Fossils (Dataset 11).....	182
6.4.3	Experiment 3: Object Detection of Sedimentary Structures on Core Images	190
6.5	Conclusions	198
7.	CHAPTER 7 - LEARNING COMPLEX GEOLOGICAL PATTERNS FROM OUTCROP DATA (2D IMAGES) BY USING IMAGE ANALYSIS WITH INSTANCE SEGMENTATION	201
7.1	Introduction	201
7.2	The Yolact model	202

7.3	Methodology (YOLACT).....	204
7.3.1	Step 1: Dataset Selection.....	205
7.3.2	Step 2: Dataset annotation.....	206
7.3.3	Step 3: Instance Segmentation model training.....	207
7.3.4	Step 4: Results (evaluation & inference)	209
7.4	Chapter’s Findings & Discussion.....	211
7.4.1	Experiment 1: Application of the default YOLACT model on outcrop data 212	
7.4.2	Experiment 2: Recommended Improvements for the default YOLACT (DarkNet53) model.....	223
7.4.3	Experiment 3: Comparative study: YOLACT (ResNet101) vs (cDarkNet53) models.....	244
7.4.4	Experiment 4: Application of the YOLACT (cDarkNet53) model on core images 256	
7.5	Chapter’s Conclusions.....	260
7.5.1	Experiment 1 Conclusions	260
7.5.2	Experiment 2 Conclusions	260
7.5.3	Experiment 3 Conclusions	261
7.5.4	Experiment 4 Conclusions	262
7.6	Summary of the Three Computer Vision Methods	262
8.	CHAPTER 8 - INTERPRETING MULTIPLE DEPOSITIONAL ENVIRONMENTS BASED ON AVAILABLE DATA AND KNOWLEDGE WITH NLP AND NEURAL NETWORKS	265
8.1	Introduction	265
8.2	Natural Language Processing (NLP).....	265
8.3	Custom Artificial Neural Network and Graphical User Interface to Interpret the Geology.....	270
8.3.1	Custom Artificial Neural Network.....	270
8.3.2	Streamlit Graphical User Interface.....	274

8.4	Chapter's Results & Discussion	275
8.4.1	Test Case 1: Interpret the Depositional Environment from Outcrop Images 276	
8.4.2	Test Case 2: Interpret the Depositional Environment from Core Images 280	
8.4.3	Test Case 3: Interpret the Depositional Environment from Sedimentary Logs 283	
8.5	Chapter's Conclusions.....	291
9.	CHAPTER 9 - SUMMARY, CONCLUSION, AND FUTURE WORK	294
9.1	Summary & Conclusions.....	294
9.2	Challenges & Recommendations for Future Work	303
10.	References	307

CHAPTER 1 – INTRODUCTION

Due to the sparseness of data and the complexity of geology, uncertainty in subsurface reservoirs means that for a given data set, there are many ways to interpret this data, leading to different predictions about the future behavior of the reservoir. Geologists rely on their prior knowledge of surface rocks and depositional systems to interpret the subsurface, determining the environment of deposition and the distribution of rock facies, such as sandstone and mudstone, as well as their associated properties, including permeability and porosity. Regarding most reservoirs, sedimentologists are typically the experts best equipped to describe this uncertainty, as they specialize in studying the sedimentary rocks that make up most reservoir rocky types. Sedimentology is critical for comprehending subsurface uncertainty, as it extrapolates past environmental conditions from present Earth systems, allowing sedimentologists to reconstruct past environments and comprehend the distribution of reservoir facies. The data extracted from the subsurface, whether in the form of core samples or well logs, represent fragmented knowledge of the geology as they only cover a very small portion of the subsurface's lateral extent. These types of data come from the drilled wells and provide valuable information but only for the specific location (Bond, et al., 2007) and close proximity of the wells, while the space between wells remains highly uncertain.

The key challenge for geologists, and the problem I tackle in this thesis, is the existence of multiple interpretations of the same data, leading to interpretational uncertainty. Geologists cannot easily generate diverse interpretations because they can only rely on the rocks they have observed in the field or learned about through reading. Their confidence in interpreting geological processes or events is restricted by data availability or the limitations of current theories and models. As a result, different interpretations of the same data may exist, leading to conflicting conclusions and interpretations (Bond, et al., 2007). While adding more subsurface data, such as core, well logs, and seismic data, leads to reduced uncertainty, acquiring subsurface data is prohibitively expensive, thus limiting the amount of available data. As a result, significant uncertainty will always be present.

Bond et al. (2007) documented this interpretational uncertainty phenomenon by giving seismic data to a number of geologists to interpret. The range of interpretations of a single dataset highlighted the variability of outcomes due to differences in individuals'

experiences and, in doing so, quantified the uncertainty inherent in seismic interpretation (Bond, et al., 2007). According to Bond et al. 2007, when geoscientists interpret such datasets, they must rely upon their previous experience and apply a limited set of geological concepts to describe the data. This data is used to create geological framework models and to estimate the geology or other reservoir properties. All the components of such frameworks carry some uncertainty due to the incompleteness of the geologic data. This data incompleteness is often associated with psychological biases such as availability and anchoring. The geologist's prior experience, which is limited to a subset of all the geology, biases the interpretation of the rocks as it relies on information that comes readily to mind.

Bond et al. (2007) also identified two types of uncertainty related to interpretation: conceptual uncertainty and interpretational uncertainty. Conceptual uncertainty refers to the degree of ambiguity or lack of clarity in the underlying geological concepts or models used to interpret the data. It reflects the level of understanding and agreement among geoscientists regarding the geological framework applied to interpret the data (Bond, et al., 2007). Interpretational uncertainty, on the other hand, refers to the range of possible interpretations of the data given a specific geological concept or model. It reflects the degree of subjectivity and variability arising from differences in individual experience, expertise, and personal biases (Bond, et al., 2007). Geological interpretations are always linked with interpretational and conceptual uncertainty, which is difficult to elicit and quantify, often creating unquantified risks for understanding the subsurface (Randle, et al., 2019).

To better handle interpretational uncertainty, we would ideally allow geologists to learn from a wider array of outcrops to remove some psychological biases. Outcrops refer to visual exposure of rocks, a common type of surface data that sedimentologists examine to understand the geology. The outcrops are laterally extensive and information-rich, containing various sedimentological features such as sedimentary structures, different lithology types, fossils, and other structural elements. The assemblages and combinations of these features into facies provide crucial information about the depositional system and the process that shaped it. Geologists often use outcrops to develop realistic models and understand the uncertainty associated with subsurface data. By observing a range of outcrop analogues, it is possible to derive a range of possible outcomes between wells and inspire the development of a more robust model ensemble. This approach allows for

identifying variations within and between depositional environments to enable a more accurate interpretation of the subsurface space between wells.

This thesis demonstrates a novel approach to dealing with interpretational uncertainty by developing an AI system able to learn valuable geological information from surface data (outcrop images), link this knowledge to the fragmented data of the subsurface (core data), and finally, interpret the depositional environment. To accomplish this, the approach taken is analogous to that of a geologist, who relies on surface observations to infer the subsurface geology. When presented with a piece of data, geologists tend to develop a limited number of concepts, with one concept being more prominent than the others. On the other hand, this AI system can provide a broader range of possible concepts and ideas for the same data because the model has learned from a larger pool of data and data combinations.

Specifically, my thesis showcases a Supervised AI system that uses a novel combination of Computer Vision, Natural Language Processing, and Neural Networks to observe rocks, extract geological knowledge from a corpus of geological data, and embed this knowledge into a custom Neural Network model that combines all the information as a human geologist would into comprehensive interpretations. This approach represents the first attempt to identify the geological depositional environment using only two-dimensional images of sedimentary rocks to generate multiple interpretations, ranked according to the probability of each scenario, to capture the uncertainty. The goal was to create a system that learns from outcrops to apply this knowledge to new outcrop locations and extend this knowledge to other valuable geologic data, including core samples from the subsurface where interpretation is more challenging. Computer Vision algorithms were trained to analyse and segment images of outcrops and automatically learn, identify, and extract features such as rock textures and classify different types of sedimentary structures and lithology types. However, knowledge incorporation into computer vision and machine learning models has been challenging due to the multiple forms of knowledge representation. To make this feasible, NLP is used to elicit expert knowledge from the corpus of geological publications. By creating a customized Neural Network that utilises the results of both Computer Vision and Natural Language Processing networks, it is possible to generate several different interpretations to predict the likelihood of an outcrop being formed by various depositional environments. The cumulative effect of observing multiple outcrops is to expand the spectrum of scenarios

based on geological evidence and fully explore ideas available for interpretation. While the quality of ML-generated output may not surpass that of a human, it can be a valuable tool in providing an indicative overview of geological features.

1.1 High-Level Thesis Workflow

This Ph.D. aims to create a new AI-based system for interpreting the rock record by rapidly creating a range of plausible geological concepts that fit the observations without violating the principles of geology. The depositional environment interpretations are based on information extracted from the tested, two-dimensional outcrop images and domain knowledge. The manuscript demonstrates the development of a framework where Computer Vision and Machine Learning techniques are combined to mimic the human expert's approach (Figure 1-1). Creating such a system is not meant to replace the human geologist but rather provide a tool to help the geologist to come up with multiple interpretational scenarios faster while automating the bulk of manual work that entails aggregating information from outcrop data and identifying depositional structures and features.

The human geologists' approach entails a number of steps the geologists follow in interpreting the outcrop. They separate the collection of observations from interpretation. In practice, they tend to start interpreting as they go, but the idea is to collect data and observations first and then interpret. This involves examining the outcrop from different perspectives and distances, starting with a broad overview to identify key features such as sedimentary structures and structural elements, followed by a closer examination to identify the lithology and other features such as grain size and shape, which may require the use of a hand lens. With that said, outcrop interpretation is a multiscale problem in which geologic features and data may be collected across different scales, ranging from centimeters to tens of meters. The information gathered is then recorded in sketches and sedimentary logs, which serve to communicate findings to other geologists, and are subsequently analysed and interpreted. Importantly, returning to the rocks after fieldwork can be difficult and costly, underscoring the need for a thorough and systematic approach during the initial data collection and interpretation stages. To summarize the above, geologists start by observing the various geological features present in the outcrop, including lithology, sedimentary structures, and structural elements. Then they define the ordering and distribution of the geological features and form an initial hypothesis.

According to their expertise and potentially additional information, the geologists improve their understanding of the outcrop, enhancing their initial hypothesis. Finally, they combine all the pieces of information together to form an interpretation of the depositional environment. This process is depicted with a simplistic workflow in Figure 1-1.

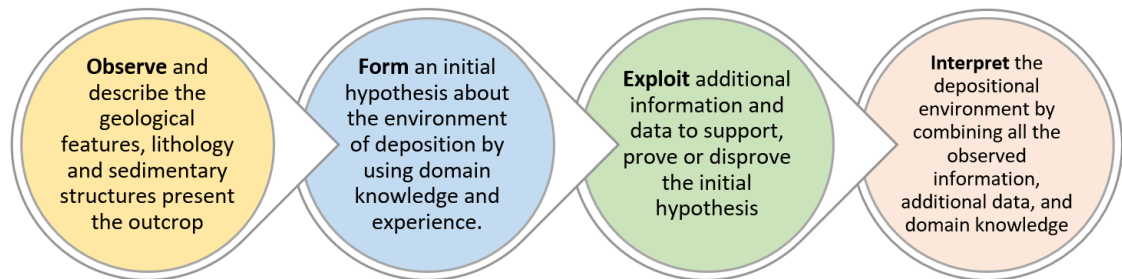


Figure 1-1: Concise Geologist's steps for interpreting the depositional environment from an outcrop (Vitor Abreu 2022, CDT course).

Figure 1-2 outlines the steps the proposed AI system can follow in interpreting the depositional environment from an outcrop, mimicking the geologists' methodology. The extraction and analysis of geological features, such as lithology, sedimentary structures, and fossil assemblages, provide further evidence about the depositional environment as well as the geological history of the outcrop. The AI workflow I developed consists of five different models (Figure 1-3), three Computer Vision models to extract all the geological features from the outcrop images, an NLP model to mine text from established literature, and finally, a Neural Network model to form geological interpretations by putting together all the available information. This process is illustrated in Figure 1-2.

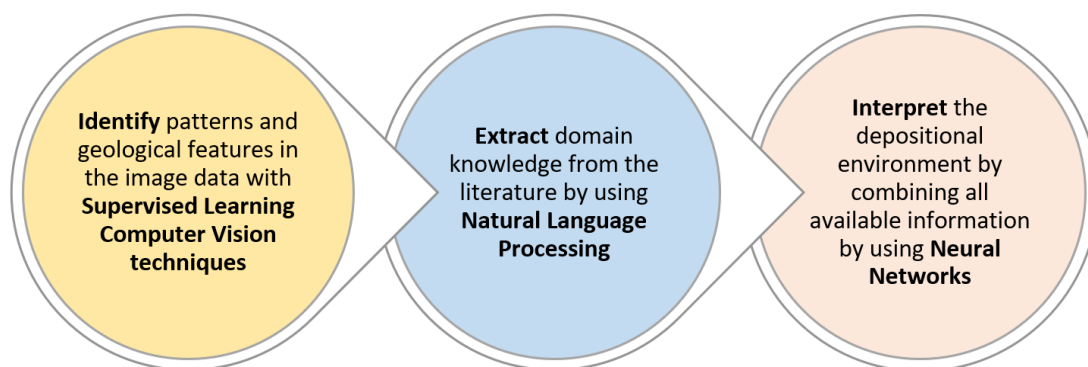


Figure 1-2: AI system steps for interpreting the depositional environment from an outcrop.

The first step utilises three Supervised Learning Computer Vision techniques -- Image Classification, Object Detection, and Instance Segmentation -- to identify patterns and geological features in the image data. Image Classification is used to classify single fossils and sedimentary structures from close-up images. Classification is good on individual features when they take up the whole image and can be enhanced using sketches and augmentation. Object Detection identifies and localizes multiple sedimentary structures and fossils from 2D outcrop images at different scales, from a few centimeters to several meters. Instance Segmentation identifies, localizes, and clearly defines the shape and boundaries of multiple sedimentary structures and estimates lithology types from 2D outcrop images at different scales. This method allows for the identification of geological features such as erosive bases, truncations, and injections, adding important geological context. Our AI system uses all of these three Computer Vision methods to identify patterns and learn what geological features look like, as progressively, each one of them adds more information.

In the second step, Natural Language Processing is used to extract information from the literature and the established interpretations to link the previous observations to the different types of depositional environments. NLP is used to extract knowledge of feature combinations associated with depositional environments. This is a way to utilise domain knowledge and transform it into a text format that a Neural Network can use. NLP captures domain knowledge by mining the printed text on geology by extracting keywords from the sedimentology domain. All the extracted text is cleaned from unimportant words, keeping only words that are relevant to the depositional environment

interpretation. The result is automatically arranged into a CSV file, which is suitable for use by a Neural Network model.

Lastly, the Interpretation/Synthesis step follows, in which the NLP results are embedded into a Neural Network to predict the depositional environment by identifying features and finding the best fit of the label combinations to the NLP results. The custom Neural Network model processes and combines all the information from the previous steps to generate multiple scenarios and develop a comprehensive understanding of the depositional environment.

The detailed breakdown of the previous workflow (Figure 1-2) is explained in Figure 1-3.

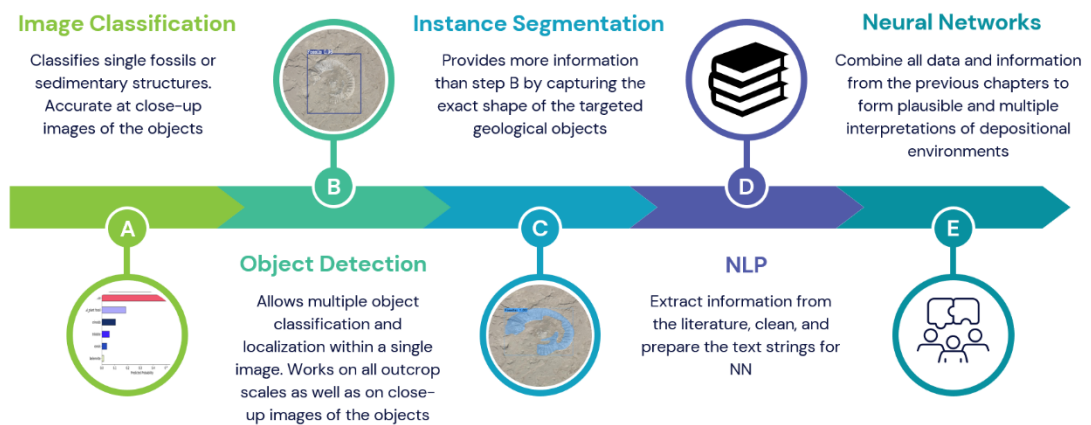


Figure 1-3: Detailed AI system steps for interpreting the depositional environment from an outcrop.

Steps A, B, and C provide all the visual evidence extracted from the outcrop and core images. Step D extracts domain knowledge from established literature. Finally, Step E combines all data from steps A-C with the geological rules and facies from Step D to generate multiple geological interpretations, ranked according to their probabilities.

As already mentioned, geology is a multiscale problem, meaning that sedimentological features can be present under different scales, and their correct recognition affects the prediction of the depositional environment. While a geologist can easily learn and recognize the significance of scale and identify features accordingly, the same is not valid for a non-human (machine). The Computer Vision models are trained on images including various objects similar to the ones we are trying to predict. For instance, if we want a model to predict soccer balls, we will train it with examples of soccer balls. Now,

since a soccer ball is specific and almost always looks the same in terms of shape and patterns, it can be recognized easily at all scales, from a few centimeters to meters. For a human geologist, the scale of the geological features is itself a differentiating factor. For instance, if a geologist is asked to classify an image of planar bedding and a planar lamination, the distinction between the two is almost impossible without a scale reference. The machine does not automatically understand the concept of scale and its importance as a differentiating factor. Therefore, adapting the scale to the CV models is one of the greatest challenges throughout this project. To address this, two steps were followed; Train the CV models with images belonging to various scales and embed the dimensionality of the features with the annotations of the images. As shown in Chapter 7, an accurate adaptation of scale during the annotations improved the performance of the segmentation model's predictions by allowing more correct label assignments to the geological features of the outcrop.

Figure 1-4 illustrates how the described AI system Figure 1-3 uses all five different models to interpret the depositional environment when given a test outcrop image. The image is tested against the three computer vision models sequentially. Then the predicted labels are input manually into a Graphical User Interface (GUI), and the system generates a set of potential geological scenarios regarding the depositional environment of the examined outcrop image in a matter of seconds. This breadth of interpretations can capture the interpretational uncertainty and can ultimately lead to enhanced decision-making by unlocking the more viable options and scenarios for the geologist to investigate further with reservoir modeling and simulations.

This thesis will contribute to the holistic understanding of the role of AI in geology, specifically in clastic sedimentology and outcrop interpretation, and provide a foundation for further research and development in this field.

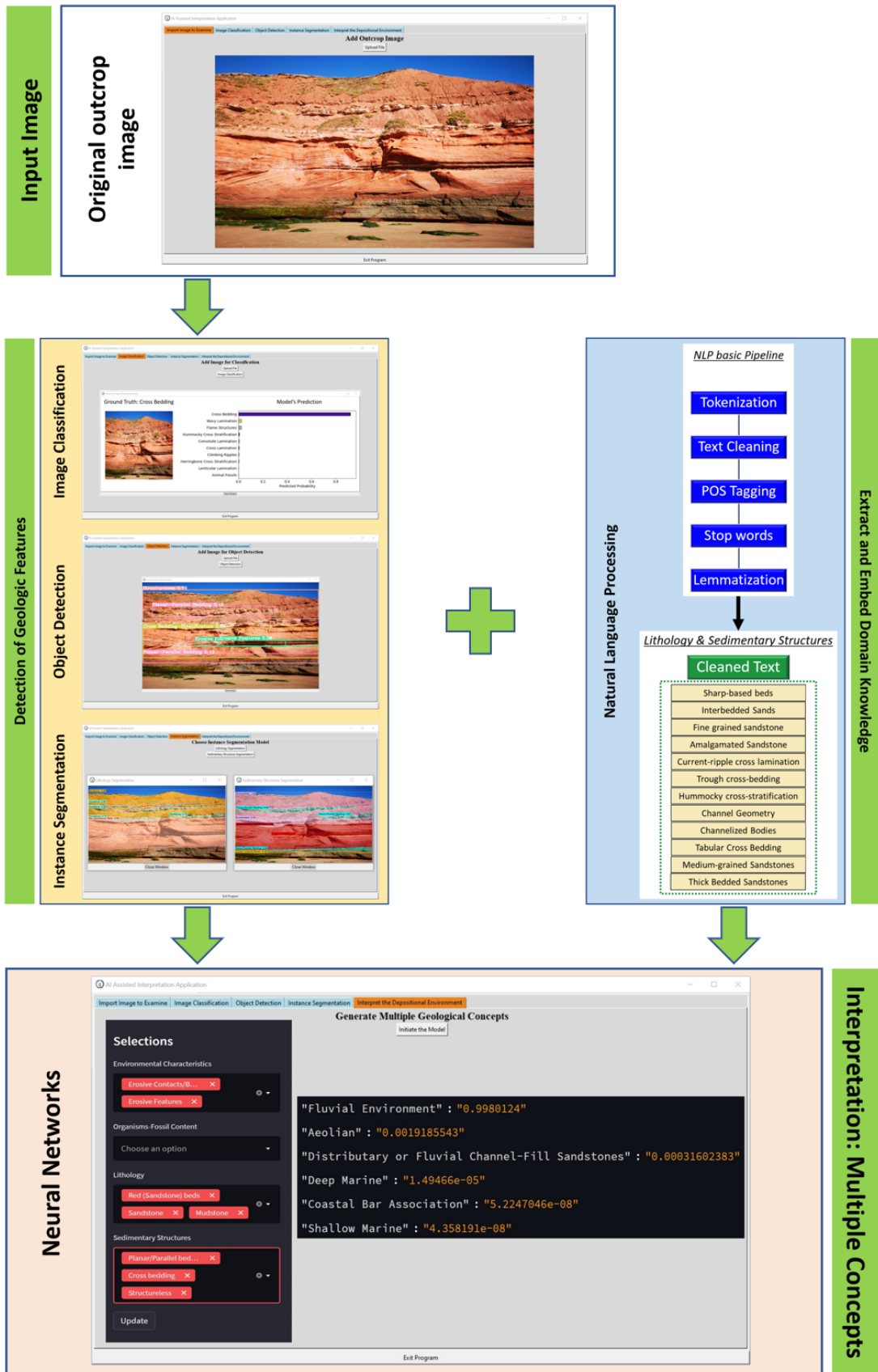


Figure 1-4: Overview of thesis workflow with example results.

1.2 Thesis Outline and Chapters Objectives

The thesis comprises the following structure:

Chapter 2: Geology and Artificial Intelligence Fundamental Concepts for this Thesis

This chapter explores relevant research regarding the uncertainty in geological interpretation and presents recent methods and applications of CV and ML on outcrop data. By providing a comprehensive overview of the challenges associated with geological interpretation and the methods used so far to address them, this chapter will contribute to a better understanding of the nature of this thesis by setting the ground for its novelty.

Furthermore, this chapter offers a theoretical foundation of Machine Learning and its relevant aspects applied in this project, highlighting the components and use of Convolutional Neural Networks (CNNs) - a type of deep learning algorithm commonly used in Machine Learning applications for image and pattern recognition tasks.

Chapter 3: Method Overview

This chapter describes the three Supervised Learning Computer Vision methods - Image Classification, Object Detection, and Instance Segmentation - employed in this thesis for geological feature extraction, as well as the implementation of an AI component utilising Natural Language Processing and Artificial Neural Networks to embed domain knowledge into the developed AI system. The reasoning behind the choice of each method will be discussed, as well as why I chose only to use Supervised Machine Learning approaches and not Unsupervised Learning. Lastly, this chapter briefly overviews the benchmark datasets commonly used to train and evaluate computer vision models, often used in research and academic settings. Their relevance and contribution to this project will also be mentioned.

Chapter 4: Datasets Description

This chapter provides a detailed description of my dataset-building workflow for Computer Vision applications in geology. It highlights the importance of each workflow component during the dataset-building process, describes all the outcrop datasets used in this thesis, and explains the choices for each dataset and the geology depicted within each one.

Chapter 5: The Use of Sketches to Improve the Image Classification of Sedimentary Structures

This chapter demonstrates how Image Classification can be utilised to classify single geological structures and fossils. The Classification model demonstrates how the use of a blended dataset, consisting of 2D outcrop images and simplified sketches of geological structures, makes improvements in predictions of sedimentary structures. The addition of sketches enhances the data quality by improving the accuracy and completeness of the dataset, boosting the model's learning capability to identify complex sedimentary structures and fossils. According to the chapter's findings, such a blended dataset improves the model's geological learning. It results in fewer misclassifications and higher test accuracy of the model predictions of the sedimentary structures and the fossils at hand. The model assigns a class for each tested image, then compares each prediction to the ground truth. The chapter ends with a discussion of how this method can help us extract visual evidence from an outcrop, which is helpful for a geologist to form an interpretation. In addition, the main drawbacks of this method are discussed, leading to the next chapter on Object Detection.

Chapter 6: Identifying Multiple Geological Features Using Object Detection on Outcrop and Fossil Images

This chapter tackles the problem of identifying and localizing multiple sedimentary structures & fossils from 2D images with Object Detection. The Object Detection model uses an annotated image dataset as an input, consisting of outcrop and fossil images. The model assigns a bounding box around each object present in each image of the test data. A bounding box is a rectangular frame or box used in object detection that encloses an object of interest in an image or video. The bounding box aids in identifying the limits of the object within the image and serves as a spatial reference for the object's location. According to the chapter's findings, the presented geological Object Detector is successful at predicting the geology at different scales compared to the Image classification model, which was useful only for close-up images of outcrops and fossils. The chapter ends with a discussion of how this method can help us extract visual evidence from an outcrop, which is helpful for a geologist to form an interpretation. In addition, the main drawbacks of this method are discussed, leading to the next chapter on Instance Segmentation.

Chapter 7: Learning Complex Geological Patterns from Outcrop Data (2D Images) by Using Image Analysis with Instance Segmentation

This chapter demonstrates the effectiveness of Instance Segmentation in accurately delineating the boundaries of various sedimentary structures and lithology types in 2D images/video, alongside their recognition and localization. The segmentation model assigns a mask and a bounding box around each object present in each image of the test data, segmenting each outcrop image twice, once for lithology estimation and once for sedimentary structure identification. A mask, used in instance segmentation, is a binary picture composed of pixels that pinpoint an object's precise size, shape, and placement within an image or video. The instance segmentation algorithm can discriminate between different items in the same image thanks to the mask, which gives each pixel of the object a distinct colour value. The instance segmentation algorithm creates the mask by classifying and localizing each object within the image or video frame using a combination of object detection and image segmentation algorithms. Instance segmentation contours the object present within the bounding boxes and forms the mask. According to the chapter's findings, the geological Instance Segmentation model outperforms the Object Detection model in Chapter 6 in accurately predicting the geology across varying scales. Instance Segmentation also provides more detailed information regarding the shape and location of each geological object and enables the estimation of lithology by using the masks. Furthermore, applying this model to real-time data makes it a novel approach and a valuable tool used in the field for outcrop segmentation on the fly. The chapter discusses how this method can help us extract visual evidence from an outcrop, which is helpful for a geologist to form an interpretation. In addition, a final comparison of the three computer vision algorithms is presented, highlighting how each tool complements the other and why it is necessary to use all three tools to move to the final chapter. The final chapter of the results demonstrates how we can incorporate all the collected visual observations from outcrop data and combine them with domain knowledge to generate multiple interpretations of the depositional environment.

Chapter 8: Interpreting Multiple Depositional Environments Based on Available Data and Knowledge with NLP and Neural Networks

This chapter presents how Natural Language Processing and Neural Networks can be combined to create multiple geological concepts based on the published available data (textual interpretations of outcrops) and the collected observations from the outcrops.

Such interpretations and conceptual scenarios can reduce the uncertainty on a range of subsurface problems, including oil and gas reservoirs, geothermal systems, and energy storage sites. The use of NLP assists in extracting information from the established literature, from already published interpretations, and facies assemblages. Furthermore, it can help edit and manipulate the text into a form that can be used as input into a Neural Network. The Neural Network will then combine all this domain knowledge with the results of the previous three chapters, which provided visual evidence from the outcrops. According to the chapter's findings, the presented Neural Network successfully forms multiple interpretations of the depositional environment based on the input provided by the user/geologists. By utilising a user interface, the user can select various inputs from an extensive list of features, which the network will take into account and generate multiple scenarios, each with a probability assigned. This model can be used as a valuable interpretation and teaching tool in the field for outcrop interpretation on the fly. The chapter ends with a discussion of how the use and combination of these four different ML/AI methods can integrate and form a complete AI system performing outcrop interpretation in a matter of seconds, providing potentially new insights about the geology, reducing uncertainty tied with geological interpretation, and complement decision making.

Chapter 9: Summary, Conclusion, and Future Work

This chapter covers the main findings and workflows developed in this thesis while summarizing how all the individual results and workflows can be combined into a fully functional workflow for geological concept generation. It presents the challenges related to this project and highlights a few points on how it can be potentially improved further.

CHAPTER 2 – GEOLOGY and ARTIFICIAL INTELLIGENCE

FUNDAMENTAL CONCEPTS for this THESIS

2.1 Introduction

The present chapter is structured into three fundamental sections that aim to provide a comprehensive understanding of the topics addressed in this thesis. The first section encompasses relevant background information concerning the geology and fundamental concepts that are utilised in this research. The second section is dedicated to providing detailed insights into the various aspects of Artificial Intelligence that are leveraged in this study. Lastly, the third section expounds on the current state of research concerning the application of Computer Vision methods in processing outcrop image data. By delineating the contents in this manner, this chapter aims to provide an overview of the underlying themes of this thesis, thereby facilitating the reader's comprehension of the subsequent chapters.

2.2 Geology Fundamental Background

Sedimentology, outcrop interpretation, depositional environment, and uncertainty are all essential components of geology that play a crucial role in understanding the Earth's subsurface. This thesis explores these concepts in detail and proposes strategies for capturing uncertainty in the interpretation of depositional environments.

2.2.1 Sedimentology

Sedimentology is a subfield of geology that involves the study of sedimentary rocks, which form through the accumulation and cementation of particles and minerals derived from weathering, erosion, and transportation of pre-existing rocks (Nichols, 2009). Sedimentologists employ a multidisciplinary approach to investigate the physical, chemical, and biological characteristics of sediments and sedimentary rocks to understand the processes that govern their formation and the environmental conditions under which they were deposited (Boggs, 2013).

The significance of sedimentology lies in its ability to provide valuable insights into Earth's history, including the evolution of the planet, the development of the atmosphere and oceans, and the evolution of life. Sedimentary rocks serve as archives of Earth's surface environments, preserving information about past changes in climate, sea level,

and tectonic activity over millions of years (Reading, 1996). Furthermore, sedimentary rocks offer crucial clues about the location and composition of natural resources, such as oil, gas, and minerals.

Beyond its fundamental contributions to geology, sedimentology has important practical applications in fields such as engineering, environmental science, and resource exploration. For instance, sedimentological studies are employed in oil exploration and production to better understand reservoir properties and fluid flow behavior (Nichols, 2009). In addition, sedimentology plays a critical role in geological hazard assessment, particularly with regard to understanding the behavior of landslides and floods (Boggs, 2006). Lastly, sedimentological research is leveraged in environmental science to track the transport and fate of contaminants in water and soil systems (Reading, 1996).

2.2.2 Outcrop Interpretation

An outcrop is a visible exposure of bedrock or rock layers at the Earth's surface that has been exposed due to erosion, tectonic uplift, or weathering (Wicander & Monroe, 2012). Outcrops can vary in size, from small and isolated exposures to large and continuous rock formations that can extend over several kilometers. By examining outcrops, geologists can gain a better understanding of the geologic history of an area, including the deposition and deformation of rock layers, the history of geological events, and the tectonic history of a region (Wicander & Monroe, 2012).

Geological outcrop interpretation is an essential component of geology, which involves studying the exposed rock layers or geological formations at the Earth's surface, to understand the geological history and structure of an area (Schwartz & Tracy, 2020). The process of geological outcrop interpretation requires careful observation and analysis of the physical features of the rocks, as well as any fossils or other evidence of past environmental conditions that may be present.

The primary objective of geological outcrop interpretation is to gain a better understanding of the geological history of a given area. This requires identifying the age, composition, and forces that have acted upon the rocks over time, such as erosion, tectonic activity, and volcanic activity (Schwartz & Tracy, 2020). Outcrop interpretation can also provide valuable information about the geological structure of an area, including the presence of faults, folds, and other geologic features that may be relevant to various fields, such as engineering or resource exploration.

Fossils are also crucial for geological outcrop interpretation as they provide information about the organisms that lived in the past and the environmental conditions that existed at the time (Prothero & Schwab, 2013). For example, marine fossils in a rock layer may indicate that the area was once covered by a shallow sea, while plant fossils may suggest that the area was once a forested region.

Technological advancements have made it easier for geologists to interpret geological outcrops. Digital imaging and mapping tools can create detailed 3D models of outcrops, enabling better visualization of rock layers and structural features.

2.2.3 Interpretation of Depositional Environments

In geology, a depositional environment refers to the conditions and processes that existed in a particular location when sedimentary rocks were being deposited. The depositional environment of sedimentary rocks can be attributed to various environmental conditions, such as marine or freshwater environments, desert or glacial environments. The characteristics of depositional environments are determined by specific physical, chemical, and biological conditions that influenced sediment supply, transport, and depositional processes at the time of deposition (Reading, 1996). For instance, sedimentary rocks deposited in a marine environment may contain marine fossils, while those deposited in a desert environment may contain cross-bedding, dunes, and evaporite minerals.

Geologists use a variety of methods to identify depositional environments, including the study of sedimentary structures, fossils, and geochemical signatures. Sedimentary structures such as cross-bedding, ripple marks, and mud cracks can provide evidence of the types of depositional processes that occurred in a particular environment. Moreover, the study of lithology, which focuses on the physical and chemical characteristics of rocks, complements sedimentology by providing a detailed understanding of rock composition, grain size, and texture. Fossils can also reveal important information about the organisms that inhabited the environment and help to determine the age of the sedimentary rocks.

The understanding of depositional environments is vital in many areas of geology, including petroleum geology, where the identification of reservoir rocks and the prediction of their distribution and quality depend on the comprehension of the depositional environments in which they were formed. Similarly, the identification of

mineral resources, such as coal or iron ore, may depend on understanding the depositional environments in which they were deposited (Tucker, 2001). Therefore, the study of depositional environments is an essential aspect of geology.

However, there is often uncertainty associated with the interpretation of depositional environments. This uncertainty arises from a variety of factors, including the preservation of sedimentary features, the complexity of depositional processes, and the limited amount of data available for interpretation (Davies, 2011). Geological interpretations are always linked with interpretational and conceptual uncertainty, which is difficult to elicit and quantify, often creating unquantified risks for understanding the subsurface (Randle, et al., 2019).

This uncertainty can lead to different interpretations of the same depositional environment, and the degree of uncertainty can vary depending on the quality and quantity of data available, as well as the expertise of the interpreter (Davies, 2011). As a result, geologists must often make interpretations based on incomplete or ambiguous data. In some cases, uncertainty can be reduced through the use of additional data sources, such as geophysical surveys or laboratory analyses of sediment samples (Miall, 2015). In other cases, uncertainty may persist, and different interpretations may need to be considered and evaluated in order to arrive at the most plausible explanation (Miall, 2015).

The interpretation of depositional environments is also important because it can help us to understand the distribution and characteristics of sedimentary rocks in the subsurface. Subsurface geology is often characterized by significant uncertainty, particularly in areas where direct observations are limited. Thus, geologists must rely on indirect methods to infer the properties and characteristics of the rocks beneath the surface (Boggs, 2013). These methods include techniques such as seismic surveys, well logging, geological modeling, and outcrop interpretation, with the latter being the main focus of this thesis.

Uncertainty in subsurface geology can arise from a number of factors, including the complexity of the geological processes that have occurred over time, the limitations of the available data, and the inherent uncertainty associated with making predictions about complex geological systems (Wellmann & Caumon, 2018). For example, the accuracy of seismic surveys is limited by factors such as the resolution of the survey and the presence of noise and other sources of interference (Li & Dehler, 2019). Similarly, well logs can provide detailed information about the properties of rocks in the vicinity of the

wellbore, but their accuracy can be affected by factors such as drilling fluid invasion, borehole washout, and tool calibration (Serra, 1984). The data extracted from the subsurface, whether in the form of core samples or well logs, represent fragmented knowledge of geology as they only cover a very small portion of the subsurface's lateral extent (Miall, 2015). These types of data come from the drilled wells and provide valuable information but only for the specific location and close proximity of the wells, while the space between wells remains highly uncertain. However, it is important to recognize that uncertainty will always be present to some degree in both surface and subsurface geology and that multiple interpretations may need to be considered in order to arrive at the most plausible explanation (Miall, 2015).

2.3 Artificial Intelligence and Machine Learning Fundamental Background

Artificial Intelligence (AI) and Machine Learning (ML) are rapidly evolving fields with significant potential to revolutionize various aspects of modern society. ML and AI are concerned with the development of algorithms and systems that enable computers to learn from data and make predictions or decisions based on that learning. These technologies have already made an impact in fields such as finance, healthcare, and transportation and are increasingly being applied in the natural sciences as well. In this section, our goal is to provide a comprehensive understanding of the fundamental concepts and techniques that underlie these powerful tools for data analysis and prediction. Specifically, we use a combination of AI, ML, and Computer Vision (CV) methods to deal with interpretational uncertainty.

2.3.1 Human Learning vs. Machine Learning in Learning from Outcrops

Human learning and machine learning (ML) differ in several ways. Humans can learn from a variety of sources, such as experience, intuition, and reasoning, and can generalize knowledge to new situations. In contrast, machines learn by processing large amounts of data through algorithms and do not have the capacity for intuition or reasoning that humans possess. While human learning is typically slow and requires extensive training, machines can learn rapidly from large datasets.

The acquisition of knowledge is a fundamental process that enables individuals, including humans and other living organisms, to adapt and respond effectively to their environment. Human beings acquire knowledge through diverse techniques, ranging from naturalistic observations to specialized educational settings, that involve various degrees of

complexity. Learning is characterized by a transformation of ideas and information structures in the human mind. It is widely believed that learning involves introducing changes in the learning system, which enhance its performance. These changes are deemed adaptive since the system improves in its execution of a given set of tasks when they are repeated. It is notable that much of the knowledge that individuals possess is implicit and is not readily available in a formalized or textual format (Polanyi, 1966) that can be understood by a computer program.

This is the reason it is not easy to write a program for a computer to do many tasks that we humans do so easily, such as understanding spoken sentences, images, languages, or driving a car. The concept of learning pertains to training computing machines to acquire knowledge and teach themselves, as proposed by Chowdhary (Chowdhary, 2020). These machines are designed to emulate human decision-making processes, which are based on principles that promote survival and reproduction. However, unlike humans, machines do not have social interactions to consider. In the case of self-driving cars, the quality of decision-making is measured by how closely they can mimic the behavior of trained human drivers (Chowdhary, 2020). Thus, the effectiveness of machine learning algorithms is evaluated based on their ability to make decisions that approximate human decision-making processes.

Human geologists employ their expertise and practical skills to interpret the geological features exposed in an outcrop. To characterize a new outcrop, they frequently resort to using outcrop analogues, drawing on their prior knowledge of other interpreted outcrops (Almklov, et al., 2011). Geologists typically use a combination of learning by induction, analogy, and deductive reasoning when interpreting geological features exposed in an outcrop. This involves drawing on their prior knowledge and experience to identify patterns and similarities with other interpreted outcrops (i.e., analogues) to infer the geological processes that have acted in that location. Learning by induction involves generalizing from specific examples to form a more abstract understanding of a particular phenomenon while learning by analogy involves using knowledge gained from one context to make inferences about another similar context. Deductive reasoning uses general principles or rules to draw specific conclusions or predictions. It is a top-down approach to learning, where learners start with a general concept or hypothesis and then use logical reasoning to arrive at specific conclusions (Williamson, 2002). By combining

these types of learning and reasoning, geologists are able to form a more complete understanding of the geological features they are interpreting.

This arises naturally in the human mind but would require separate techniques for a machine to assimilate both types of learning into a machine learning model. It should be emphasized that the knowledge necessary to arrive at an appropriate decision in a given situation is not always explicit enough to be encoded into a computer program. Intuitive knowledge is frequently acquired through a process of learning from examples and practice, which cannot be converted into a well-defined sequence of instructions as in a computer program (Chowdhary, 2020)].

The methods employed in this thesis encompass various branches of AI and adhere to a supervised learning approach, as depicted in Figure 2-1. These techniques comprise Computer Vision (CV), Machine Learning (ML), Natural Language Processing (NLP), Deep Learning (DL), and Artificial & Convolutional Neural Networks (ANN & CNN). Research in these AI subfields has yielded substantial advancements across a range of applications, including image recognition, speech recognition, natural language comprehension, and autonomous driving.

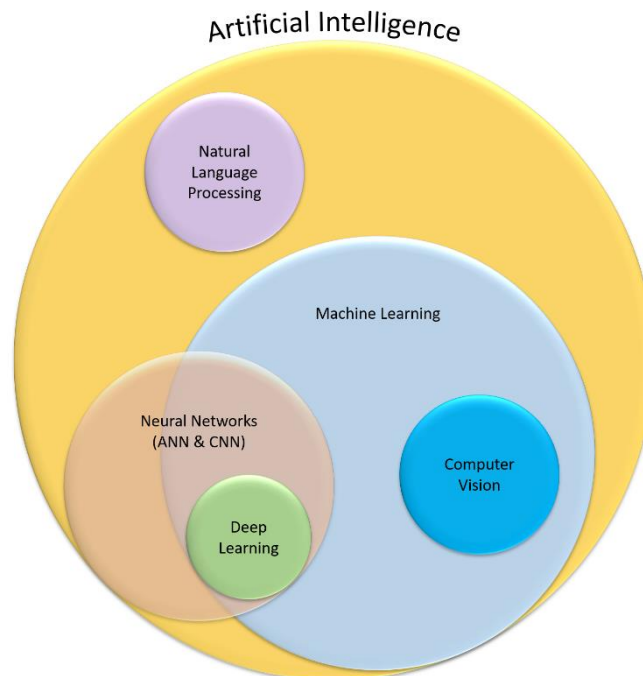


Figure 2-1: Key AI Components used in this thesis

This thesis utilises learning by analogy and induction, as well as inductive and deductive reasoning, to mirror the techniques employed by geologists in their work. These human learning and reasoning techniques can be translated into machine language through supervised learning. Though most current artificial systems focus on any single learning strategy, one may expect that machine learning research will give increasing attention to a multi-strategy approach, one which is close to the human learning system (Chowdhary, 2020). Learning by analogy is exemplified by Computer Vision, as the Machine Learning model is fed with a set of image examples and prior knowledge encoded in the images via annotations. An instance of learning by induction is illustrated in Chapter 7, where the Instance Segmentation model assimilates geological patterns from outcrops and transfers this learning to core data. Furthermore, the combination of Natural Language Processing and the Neural Network, presented in Chapter 8, is an instance of embedding inductive and deductive reasoning into a neural network. By integrating all these methods into the AI system outlined in this thesis, the geologists' approach to interpreting outcrops can be emulated.

The field of Machine Learning is dedicated to the development of computational theories for learning processes and the creation of learning machines. As learning is a fundamental component of intelligent behavior, the concerns and objectives of Machine Learning are central to the larger discipline of Artificial Intelligence. Machine Learning may be basically classified as Supervised or Unsupervised Learning. There are more categories of machine learning, such as Semi-Supervised, Self-Supervised, and Reinforced learning, but their explanation and application are not provided here as they were not used in this thesis.

Supervised learning is a type of machine learning in which an algorithm is trained on a labeled dataset to make predictions or classify new unlabeled data based on patterns identified in the training data (Alpaydin, 2010; Sathya & Abraham, 2013). In machine learning, labeled data refers to a dataset that has been pre-assigned with class labels or target variables. The labels are typically provided by annotators/human experts who manually assign the correct classification or category to each data point in the training dataset. Labeled data is essential for training and evaluating supervised learning models, which learn to predict the correct label or output variable based on the input features or variables. The model learns from the labeled examples and tries to generalize to unseen examples during the testing or inference stage. The quality and quantity of the labeled

data play a critical role in determining the accuracy and performance of the machine learning model. On the other hand, unlabeled data refers to a dataset that does not have any pre-assigned labels or categories. Unlike labeled data, which is already annotated with class labels or target variables, unlabeled data only contains the raw input features or variables that the machine learning model will use to discover patterns or relationships in the data.

Unsupervised Learning is a type of machine learning in which an algorithm is employed to extract structure or patterns from unlabeled data without external guidance (Alpaydin, 2010; Sathya & Abraham, 2013).

2.3.2 ML Important Components

Computer vision is a field of study that focuses on enabling computers to extract meaningful information from digital images or videos, emulating the human visual system's ability to perceive and understand visual content (Szeliski, 2010). It encompasses various tasks such as image recognition, object detection, image segmentation, and scene understanding. Computer vision algorithms employ techniques from computer science, mathematics, and signal processing to analyse visual data, recognize patterns, and make inferences about the visual world (LeCun, et al., 2015). This discipline finds applications in diverse areas, including autonomous vehicles, surveillance systems, medical imaging, robotics, and augmented reality.

Deep learning focuses on training artificial neural networks with multiple layers to automatically learn hierarchical representations from raw input data through multiple layers of neurons enabling it to capture intricate patterns and dependencies, making it well-suited for high-dimensional and unstructured data tasks (Bengio, et al., 2013). It has gained significant attention and popularity due to its ability to process large amounts of data and achieve state-of-the-art performance in various domains, including image classification, object detection, speech recognition, natural language understanding, and machine translation (LeCun, et al., 2015). Its breakthroughs have been fueled by the availability of large-scale datasets, advances in computing power, and the development of specialized hardware, such as graphics processing units (GPUs), that expedite the training process. Deep learning leverages the concept of artificial neural networks, which are computational models inspired by the architecture and operation of biological neural networks in the human brain (Bengio, et al., 2013).

ANNs are typically composed of fully connected layers (interconnected nodes), where each neuron is connected to every neuron in the previous and next layers. Each neuron performs a mathematical transformation on its input and transmits the outcome to the subsequent layer. Through the process of backpropagation, the neural network adjusts the weights assigned to the connections between neurons, aiming to minimize the disparity between predicted outputs and actual outputs, thereby enhancing the network's performance over time (Goodfellow, et al., 2016). However, they are limited in their ability to handle high-dimensional inputs, such as images, due to the sheer number of parameters that need to be learned.

CNNs, on the other hand, is a specialized ANN variant, designed to handle high-dimensional input data, such as images and videos (Krizhevsky, et al., 2012; Goodfellow, et al., 2016). They do this by using a series of convolutional layers, which apply filters to the input image to extract spatial features, followed by pooling layers, which downsample the feature maps to reduce their size. CNNs are specifically designed for image processing tasks and typically have more convolutional layers than traditional ANNs. This allows CNNs to efficiently learn and classify complex patterns in image data, making them particularly well-suited for tasks such as image classification, object detection, and instance segmentation, which are the three main CV methods employed in this thesis to analyse outcrop images and extract geological features.

2.3.3 Natural Language Processing

NLP facilitates computer understanding, interpretation, and generation of natural language text, and the relevant research has contributed to the development of chatbots, virtual assistants, and other AI systems capable of human interaction through natural language (Young, et al., 2018; Manning, et al., 2008). NLP models are sophisticated as they employ a diverse range of methodologies, including Machine Learning, statistical analysis, and linguistic theories, to meticulously examine, comprehend, and generate human language data (Jurafsky & Martin, 2023). The scope of human language data encompasses an extensive array of textual, spoken, or written information that individuals generate through diverse modes of communication, such as dialogues, documents, books, articles, transcripts, and similar sources. This corpus of language data serves as the foundation for training NLP models, enabling the acquisition and comprehension of linguistic patterns, semantics, grammar, and contextual meanings. Consequently,

language comprehension is enhanced, facilitating the development of language-oriented applications.

The intricacy of NLP models can be attributed to several factors. Language itself is a complex entity characterized by nuances, ambiguity, cultural references, idiomatic expressions, and context-dependent meanings (Chowdhary, 2020). Furthermore, the integration of deep learning architectures into NLP introduces further complexity, requiring the inclusion of multiple layers, connections, and numerous parameters. These elements enable the model to comprehend the structural, grammatical, and syntactical aspects of language, including the acquisition of rules governing sentence construction and adaptation to contextual variations. As digital data continues to rise at an unprecedented rate, the need to extract valuable insights from unstructured text data has become more critical (Bird, et al., 2009). As a result, NLP has found numerous applications in diverse fields such as healthcare, finance, education, and customer service.

The capabilities and applications of NLP (Liddy & Liddy, 2001) extend well beyond the scope of this project, thus not being explored further within this context. In this thesis, Document Processing and Information Extraction techniques are employed to extract text from the geologic literature, including publications, books, journals, and validated interpretations of the geology.

2.3.4 Metrics to Evaluate Computer Vision Models

According to the model and methods used for Computer Vision and Machine Learning problems, there are always metrics to evaluate the model's performance. In this thesis, three metrics were used to assess the previously described methods: Confusion Matrix for Image Classification, Intersection over Union (IoU), and Mean Average Precision (mAP) for Object Detection and Instance Segmentation.

A Confusion Matrix is used to assess the performance of a classification model by comparing the predicted and actual values of a set of test data (Ting, 2017). It provides a detailed summary of the model's performance by categorizing the results into true positives, false positives, true negatives, and false negatives. The IoU metric is used to assess the overlap of predicted bounding boxes and ground truth bounding boxes, providing insight into the model's ability to detect and segment objects accurately (Rezatofghi, et al., 2019). The intersection of the bounding boxes represents the overlapping region between the predicted and ground truth bounding boxes, while the

union represents the total area covered by both bounding boxes (Rosebrock, 2016). Average Precision (AP) is a common evaluation metric in Computer Vision tasks like Object Detection and Instance Segmentation (Everingham, et al., 2010; Russakovsky, et al., 2015). It measures the quality of the model's predictions, specifically how well the predicted bounding boxes or masks overlap with the ground truth objects. Finally, the mean Average Precision (mAP) metric is used to evaluate the model's accuracy in detecting and segmenting multiple objects, providing an average of the precision and recall across all objects and classes (Everingham, et al., 2010). These metrics can help data scientists and machine learning practitioners understand how well a model performs, make informed decisions on how to improve it, and compare the performance of different models. Both the IoU and mAP were used to evaluate the accuracy of the Object Detection and Instance Segmentation models presented in Chapters 6 and 7, respectively.

2.3.5 Pre-trained Models and Backbones

According to (Hosna, et al., 2022), transfer learning is a process of using knowledge from one domain to improve learning performance in another domain. Transfer learning, often considered a part of supervised learning, involves training a model using labeled examples to learn a mapping between inputs and outputs.

In transfer learning, a pre-trained model is first trained on a large dataset for a related task, using supervised learning, before being fine-tuned on a smaller, more specific dataset for a different task (Hosna, et al., 2022). During the fine-tuning stage, the pre-trained model is further trained using labeled examples in a similar way to traditional supervised learning. The labeled examples used for fine-tuning are typically specific to the target task, and the goal is to adapt the pre-trained model to perform well on the target task using the limited labeled data available. Therefore, transfer learning can be seen as a form of Supervised Learning where a pre-trained model is used as a starting point for learning a related task. This makes transfer learning an indispensable technique for custom projects and applications where custom datasets are needed.

A pre-trained model is a deep neural network that has been trained on a large dataset for a specific task, such as image classification or language modeling. A backbone, on the other hand, is the core architecture of a deep neural network that provides the main feature extraction capabilities of the model (Goodfellow, et al., 2016). By using pre-trained models as backbones, ML practitioners can leverage the knowledge and representations

learned by these models on large datasets, and fine-tune them for specific tasks with smaller datasets, resulting in more efficient and accurate models (Kornblith, et al., 2018). Pre-trained models are commonly used in Computer Vision tasks because they have been trained on large-scale image datasets and are capable of extracting features from images that can be used to recognize objects, as well as classify and segment images. The training of such models not only requires vast amounts of data and information to train but also utilises resources and computing power not accessible to everyone. Using a pre-trained model can save time and computational resources, as it is not necessary to train a model from scratch. One technique used to utilise pre-trained models in computer vision is transfer learning (Goodfellow, et al., 2016).

When a neural network is trained from scratch, without using a pre-trained model, its initial layers are able to identify very simple features, for instance, a straight or slanted line (Hosna, et al., 2022). As the deeper layers are trained, the model can identify more sophisticated features. For example, the 2nd layer can now identify basic shapes such as squares and circles. Layer 3 can identify complex patterns, and finally, the deeper layers can extract intricate features such as human faces, animals, etc. A useful solution to the lack of data issue is to use a pre-trained model such as ResNet50 and then retrain the final layers using the custom data, in our case, geological outcrop images. For any classifier, regardless of the classification task, the initial layer will always detect straight lines (Hosna, et al., 2022). It is not wise to train all the layers every single time we create a neural network. It is only the final layers of the chosen network that we need to train, as only these will learn and identify classes specific to a problem.

There are many pre-trained models to choose from depending on the type of problem at hand, and for geologic image analysis, several pre-trained models are employed, each with its own strengths and limitations. In this thesis, the ResNet18, ResNet50, ResNet101 (He, et al., 2016), VGG 16, and VGG 19 (Simonyan & Zisserman, 2014) pre-trained models were used for Image Classification, YOLOv6-S (Li, et al., 2022) for Object Detection, and YOLACT (Bolya, et al., 2019) for Instance Segmentation.

Model backbones are the feature extraction networks used in deep learning models for computer vision tasks. There are pre-built backbone architectures available, or one can create a custom backbone based on the problem they want to solve. ResNet (He, et al., 2016) and VGG (Simonyan & Zisserman, 2014) are popular deep neural network architectures that have achieved state-of-the-art results in a wide range of computer vision

tasks, including image classification, object detection, and segmentation. ResNet uses residual connections to allow information to bypass intermediate layers and be directly passed from one layer to another. This helps to address the problem of vanishing gradients that can occur in very deep networks. On the other hand, VGG relies on stacking many convolutional layers to learn increasingly complex features. ResNet is typically much deeper than VGG, with versions that have more than 100 layers. ResNet-18, ResNet-50, and ResNet-101 are specific variants of the ResNet architecture with 18, 50, and 101 layers, respectively, that have become popular baseline architectures for many computer vision tasks. The deeper architecture of ResNet allows it to capture more complex features but also makes it more difficult to train and requires more computational resources. In general, ResNet has been found to outperform VGG on many computer vision tasks, particularly when the networks are very deep. However, the performance difference between the two models may not be significant on simpler tasks or when the number of layers is relatively small.

2.3.6 Overview of Common Benchmark Datasets for Computer Vision Models

Computer Vision has witnessed rapid growth in recent years, partly due to the availability of large datasets that are utilised to train computer vision algorithms. The size and quality of these datasets have improved over time thanks to advancements in data collection and storage technology. The creation of datasets such as ImageNet (Deng, et al., 2009), MNIST (Deng, 2012), CIFAR-10 (Krizhevsky, et al., 2009), CIFAR-100 (Krizhevsky, et al., 2009), PASCAL VOC (Everingham, et al., 2010), and COCO (Lin, et al., 2015) has been instrumental in the progress made in computer vision research. The COCO dataset is recognized as an excellent benchmark dataset and is widely used for training and comparing the performance of multiple computer vision algorithms. It serves two primary purposes, providing a valuable resource for training computer vision models and serving as a benchmark for comparing the performance of multiple computer vision algorithms (Samuel, 2021). However, it is important to note that each custom model is built and fine-tuned for different data and tasks, and a model's performance on COCO should not be the sole determinant of its effectiveness. Additionally, the accuracy of benchmark datasets varies based on the type of data being analysed, and it is crucial to approach the evaluation of a model's performance with a skeptical mindset. Data annotation, which is going to be explained in detail in Chapter 4, is an essential part of

these datasets, and computer vision annotation tools are used to convert raw image data into labeled images for training the machine learning models.

2.4 Relevant Work on outcrop data using Computer Vision Methods

In the context of geological outcrops, computer vision models can be used to identify and analyse key features of the rock formations, such as layers, fractures, and sedimentary structures. Interpreting outcrops is a segmentation problem, where geologists aim to identify diagnostic features to form a comprehensive interpretation. The combination, arrangement, and scale of these diagnostic features are important for the understanding of the depositional environment.

Vasuki et al. (2017) introduce the Interactive Lithological Boundary Detection (ILBD) method, designed specifically for mapping lithological boundaries in complex geological images based on colour similarity (Vasuki, et al., 2017). The ILBD method utilises an initial over-segmented image and user inputs to accurately delineate boundaries of multiple lithological units in exposed rock surface images. Their experimental results demonstrated the successful separation of lithologies in visually intricate rock surface images using the ILBD method. In Figure 2-2, an example of the Interactive Lithological Boundary Detection (ILBD) method is shown.

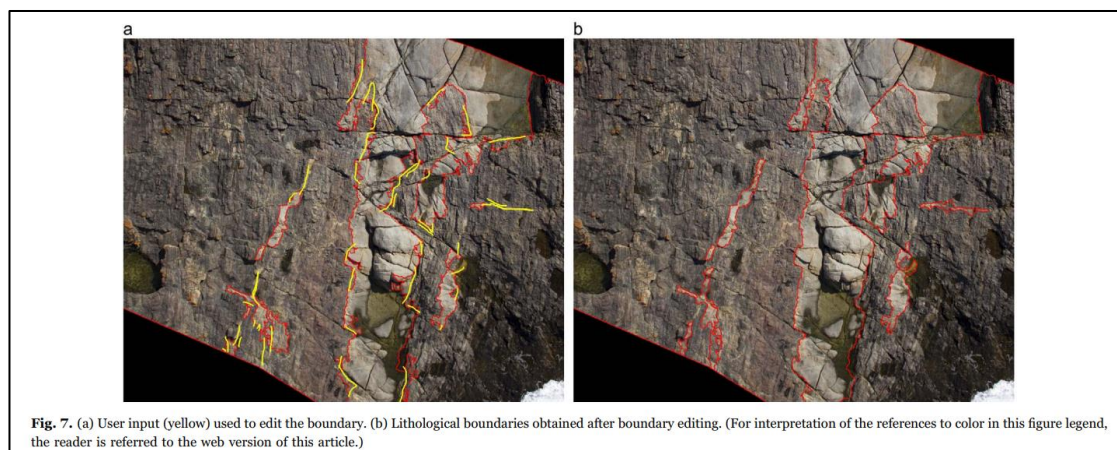


Figure 2-2: Example of lithological boundary detection by Vasuki et al. (2017).

Birgenheier et al. 2019 used transfer learning to address a suite of geologic interpretation tasks (Birgenheier, et al., 2020). They used two different base models, MobileNet V2, and Inception V3, to classify microfossils, core images, petrographic photomicrographs, and rock and mineral hand sample images. In Figure 2-3, an example of the classification

process of microfossils is shown. In the implementation they used, the model classifies any image as one of the seven learned classes – even if the image is clearly not a fossil. This highlights the importance of domain expert intervention.

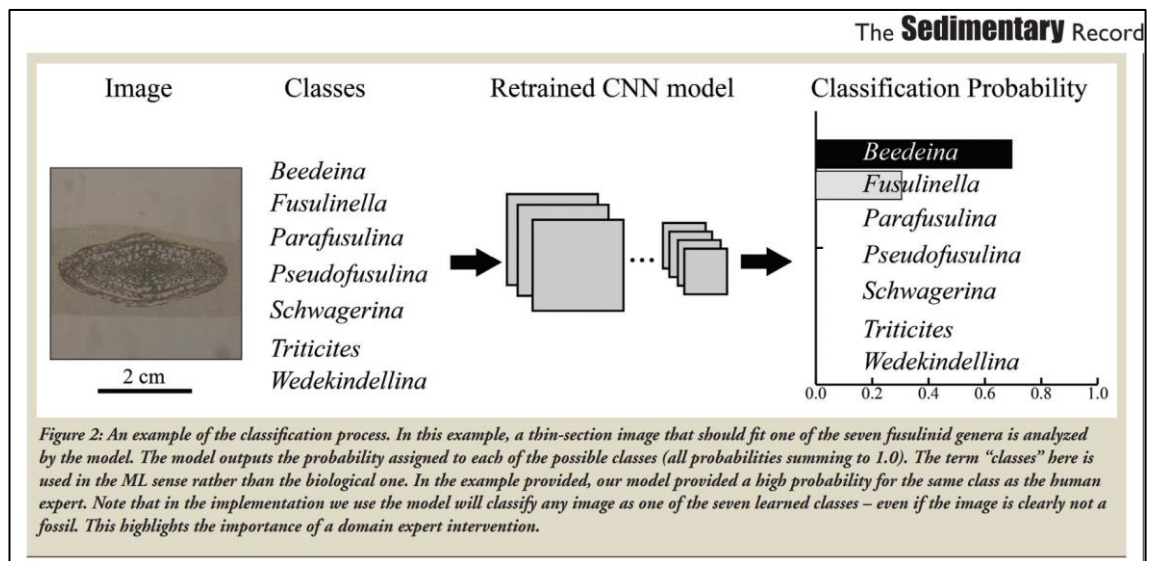


Figure 2-3: Fossil Image Classification by Birgenheier et al., 2019.

Various segmentation techniques, mostly unsupervised ones, have been previously applied to outcrop data. Figure 2-4 presents two clustering algorithms used on outcrop images, namely K-means clustering (Francis, et al., 2014) and Point Cloud Segmentation (Anders, et al., 2016).

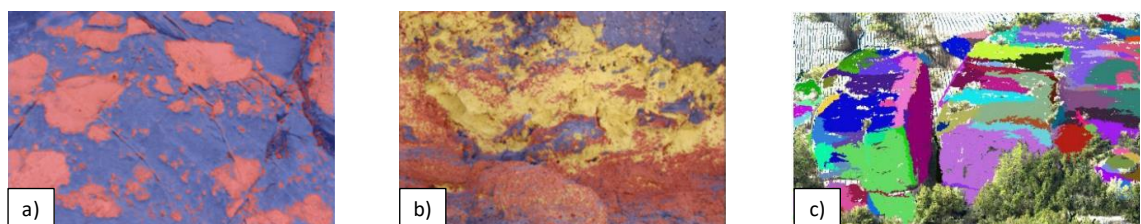


Figure 2-4: Unsupervised Segmentation techniques, a) and b) K-means Clustering (Francis, et al., 2014) and c) Point Cloud Segmentation (Anders, et al., 2016).

The examples in Figure 2-4 are segmented images with a) and b) using 2D outcrop images while c) uses photogrammetry data, but in both cases, the goal is to apply unsupervised segmentation to outcrop data, segmenting the geological features based on colour and texture.

According to Francis et al. (2014), the number, type, and spatial distribution of facies are crucial for understanding the geological context, while the orientation and position of contacts between facies provide insights into the formation of rocks. Therefore, spatial relationships between different geological materials are as important as the facies themselves in interpreting an outcrop (Francis, et al., 2014).

Francis et al. (2014) developed a Computer Vision algorithm using Unsupervised Machine Learning to identify geological contacts in a single image and segment rock outcrop images based on geological units. The algorithm exploits the visual differences between units on either side of a visible geological contact, such as colour, albedo, or texture, to extract multiple types of visual information at low computational cost and separate pixels belonging to different geological units. A vector clustering technique, such as k-means, was then used to group pixel vectors and assign them to their respective geological units.

The work of A.D. Pascual (2019) demonstrates the use of CNNs to classify a set of nine different types of rock images, along with dataset augmentation (Pascual, 2019). The application of CNNs has been extended to classify natural scene images of rocks, which present a more complex dataset. The task is simplified into a binary classification problem, distinguishing breccia and non-breccia. Finally, the rock image classifier was deployed in a lightweight and portable device, such as an iPad, to create a system that geologists could take into the field.

Previous research conducted by Dunlop et al. (2006) has explored the detection and classification of rock images within natural scenes (Dunlop, 2006). They approached the task in two steps: firstly, detecting the presence of rocks and segmenting them from the surrounding image, and secondly, classifying the specific type of rock in the segmented image. Their dataset consisted of 8 coloured images with a resolution of 2048 x 1536. Each image depicted around 15 rocks arranged on a bed of sand, aiming to simulate a natural rock environment as shown in Figure 2-5.

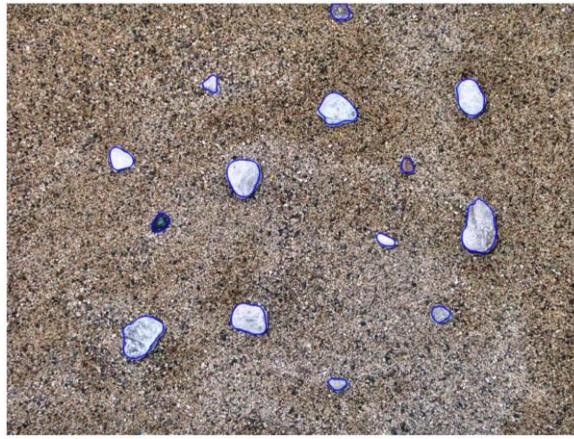


Figure 2-5: Individual rocks segmented from the original image taken from Dunlop et al. (2006).

Kwok et al. 2018 conducted a pilot study using deep learning techniques to enhance the efficiency of identifying rock outcrops on a large scale for landslide susceptibility analysis (Kwok, et al., 2018). The objective was to develop a methodology that combines Convolutional Neural Networks (CNNs) with remote sensing techniques. The aim was to generate a detailed map of rock outcrops across the entire territory of Hong Kong at a very high resolution. The developed algorithm takes into account the spatial relationship, texture, and spectral signature of pixels in remote-sensing images. The identification of rock outcrops utilised orthophotos from 2012 and 2015, supplemented by SPOT satellite images from 2015 and airborne LiDAR data from 2010. This approach resulted in the creation of a rock outcrops map with a spatial resolution of five meters. The primary goal was to improve landslide susceptibility analysis by accurately identifying and mapping rock outcrops on a territorial scale.

Malik et al. (2022) utilised high-resolution photographs obtained from a sedimentological investigation to explore an alternative approach for multi-rock identification using machine learning (Malik, et al., 2022). They applied two advanced segmentation models, namely U-Net and LinkNet, to accurately identify various rock types within digital photographs. Specifically, they segmented the sandstone, mudstone, and background classes in a dataset comprising 102 self-collected images from a field in Brunei Darussalam by employing four pre-trained networks (Resnet34, Inceptionv3, VGG16, and Efficientnetb7) as backbones for both segmentation models (Figure 2-6).

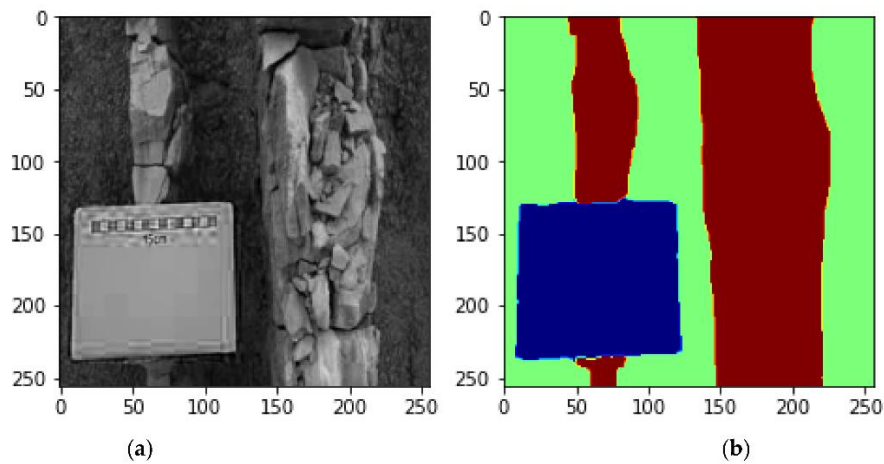


Figure 2-6: A multi-rock identification using machine learning by Malik et al. (2022). An example of (a) an original image and (b) its labels—background (blue), mudstone (wine), and sandstone (green).

To the best of our knowledge, no study has applied instance segmentation to solve the multi-rock classification problem using digital photographs of outcrops containing multiple rock types, nor has investigated the interpretation of the depositional environment from outcrop images by employing an AI/ML workflow. This thesis demonstrates a novel approach to dealing with interpretational uncertainty by developing an AI system able to learn valuable geological information from surface data (outcrop images), link this knowledge to the fragmented data of the subsurface (core data), and finally, interpret the depositional environment. Specifically, my thesis showcases a Supervised AI system that uses a novel combination of Computer Vision, Natural Language Processing, and Neural Networks to observe rocks, extract geological knowledge from a corpus of geological data, and embed this knowledge into a custom Neural Network model that combines all the information as a human geologist would into comprehensive interpretations.

The application of computer vision models on geological outcrops is a rapidly evolving area of research, with many potential applications in fields such as geology, mining, and engineering. As technology continues to advance, it is likely that computer vision will play an increasingly important role in the analysis and interpretation of geological data.

CHAPTER 3 - METHOD OVERVIEW

3.1 Introduction

This chapter describes the five methods used in this thesis to develop the AI system described in Chapter 1. The justification for selecting these methods and for opting to use only Supervised Machine Learning approaches instead of Unsupervised Learning will also be outlined.

For geological feature extraction from outcrop images, three Supervised Learning Computer Vision methods were employed: Image Classification, Object Detection, and Instance Segmentation. It was found that Image Classification provides information about what types of sedimentary structures/fossils occur in the outcrop. Object Detection shows where the particular features occur within the outcrop. Finally, Instance Segmentation allows the estimation of lithology and defines the shape and size of each feature. Starting with image classification and then moving on to object detection and instance segmentation is a natural progression in computer vision tasks as each method builds upon the previous one. By using Image Classification, one can establish a baseline understanding of the contents of an image. Object Detection then adds an additional layer of information by identifying and localizing objects within the image. Lastly, Instance Segmentation provides the most detailed understanding of the objects within the image by segmenting them into individual pixels. Starting with the simplest method and building up to the more complex one, we can ensure that we have a solid foundation before tackling more advanced techniques. It also allows us to gradually increase the complexity of our analysis as we become more familiar with each technique.

The result of the Computer Vision methods used is a list of labels describing sedimentary structures, fossils, and lithology types. An AI component is needed to combine all these individual pieces of information into meaningful sequences, assisting in the interpretation of an outcrop's geology. The geological knowledge required to make plausible combinations of the features can be extracted from the geological literature with the help of Natural Language Processing (NLP). The NLP model used was a custom-built model aiming to extract certain geological words or combinations of these words, such as lithology types, sedimentary structures, or fossils, that exist in the examined geological publications.

To embed domain knowledge with the results from the aforementioned CV methods into the AI system, I created a custom Neural Network to combine all the information and produce plausible interpretations of depositional environments.

3.2 Computer Vision Methods & Workflow

The Computer Vision methods employed in this thesis aimed to extract visual information from 2D outcrop images and identify various geological features such as lithology types, sedimentary structures, and fossils. To achieve this, Supervised Learning was deemed more appropriate as it provides labeled predictions for the geological features, unlike unsupervised methods that only provide clusters without interpretability and context of what each cluster in the image resembles. Geologists approach outcrops in a supervised way, relying on prior knowledge and experience to interpret geological features, similar to how Supervised Learning models leverage prior knowledge to generalize and make predictions.

All methods used in this thesis follow a Supervised Learning approach, meaning that the learning process requires some intervention from the user and prior knowledge. Supervised learning uses a training set containing manually tagged examples of sedimentological features to induce a classifier that can classify new data. Unsupervised learning extracts geological patterns or structures from unlabeled data without external guidance or a labeled dataset.

To ensure the proper functioning of the entire AI system, the supervised learning approach is necessary. This method provides valuable information about geological features through labeling, which the final neural network utilises as input for making interpretations. While supervised methods segment and label objects, unsupervised approaches only segment geological features without providing details about each segment. As demonstrated in the unsupervised results in section 3.2.3.2, commonly used unsupervised methods are inadequate at differentiating and extracting the essential information from outcrop images and therefore were not deemed appropriate for this thesis.

Supervised Machine Learning and Computer Vision have the potential to assist geologists in outcrop interpretation by providing automated and data-driven approaches to geologic feature recognition and extraction. However, “problems in geosciences have several

unique challenges that are seldom found in traditional applications, requiring novel problem formulations and methodologies in machine learning” (Karpatne, et al., 2019).

One major challenge is the variability and complexity of geological features as well as their scale. Geologic features can be highly variable in appearance, size, orientation, and context, making it difficult to design algorithms that can reliably identify and interpret them. To address this challenge, several datasets were compiled, each with different complexity and geological features, to train the CV models presented in this thesis and were tested for their competence. Furthermore, to reflect the appropriate scale of the geological features for Object Detection and Instance Segmentation (Chapters 6 and 7), annotations were utilised to depict the dimensions of the annotated geological features accurately. A meticulous approach to the image annotation process allows for precise representation and analysis of the geological features, contributing to a more comprehensive understanding of their characteristics and spatial relationships.

The lack of available and good-quality (section 4.3) training data adds to the above challenge. Machine Learning algorithms for Image Classification, Object Detection, and Instance Segmentation require large and diverse datasets to learn and generalize from, but collecting and annotating geological data can be time-consuming, expensive, and subjective. Moreover, the data's quality and consistency can affect the algorithms' accuracy and robustness. Earlier research has emphasized that CNN models require a large amount of data to be trained from scratch, and a considerably deeper network is necessary to achieve superior performance (LeCun, et al., 2015), (He, et al., 2016). Pre-training CNNs on domains with readily available data have been used to overcome this issue. Using such pre-trained models and transfer learning, the acquired knowledge is transferred to geology, where data collection is challenging and costly.

Pre-trained models are commonly used in Computer Vision tasks because they have been trained on large-scale image datasets. These models can extract features from images that can be used to recognize and classify objects and also segment images. For example, the pre-trained model ResNet50, was trained on the ImageNet dataset, which contains millions of labeled images with one thousand object categories. By using transfer learning, I took the pre-trained ResNet50 model and fine-tuned it for image classification and instance segmentation tasks in order to identify types of sedimentary structures, fossils, and lithology with a smaller dataset of images labeled with geological features. The model learned to recognize the geological features and performed each task with high

accuracy without requiring as much training time or data as it would if it were trained from scratch.

In this thesis, three supervised Computer Vision methods were used to extract geological features from 2D outcrop images utilising the pre-trained models ResNet18, ResNet50, ResNet101, VGG 16, and VGG 19 for Image Classification, YOLOv6-S (Li, et al., 2022) for Object Detection, and YOLACT (Bolya, et al., 2019) for Instance Segmentation. The rationale behind each choice is discussed in the corresponding sections of the chapter describing each method. Each method is progressively more complex but also provides more information about the geological features.

The proposed workflow for the Computer Vision aspect of this thesis is shown in Figure 3-1. This workflow was configured based on the findings from Chapters 5,6, and 7. When the test image is a close-up image (meaning that it doesn't show the entire outcrop), it should pass through steps A-C, as the application of all three methods will provide the maximum amount of information. If the image captures the outcrop in its entirety, then it should pass only through steps B and C.

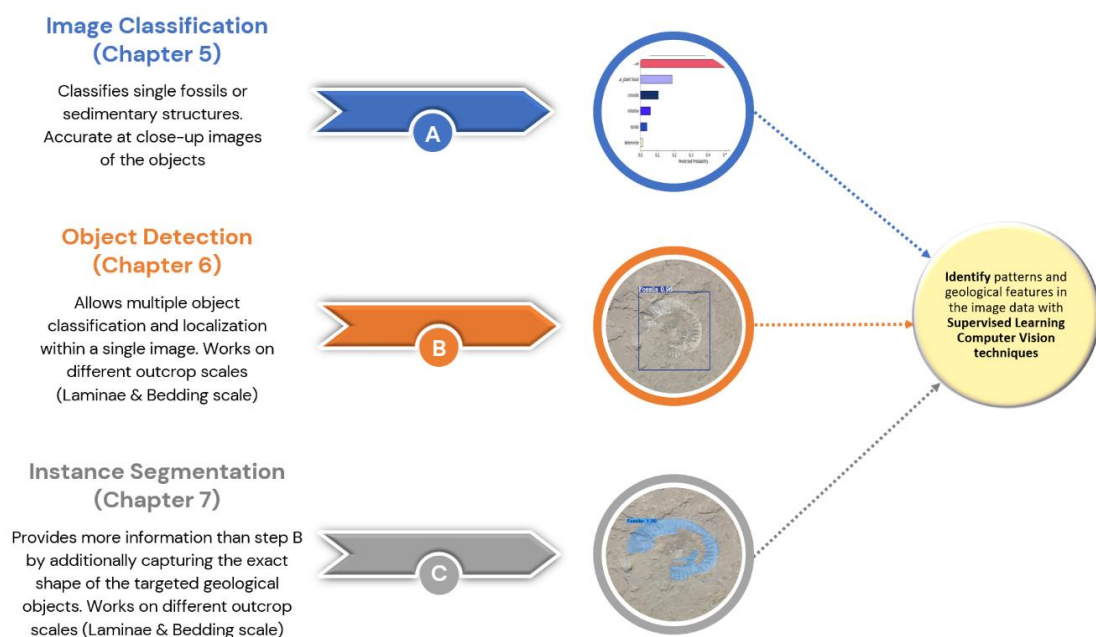


Figure 3-1: Workflow summarizing the use of three Computer Vision methods.

Image Classification is applied to categorize single fossils or sedimentary structures based on their appearance in the outcrop. This method works well for close-up images of objects but is not useful for entire outcrop images. According to my findings, attempting

to classify an entire outcrop as one sedimentary structure is incorrect and unhelpful. However, when used on zoomed-in images, image classification can differentiate between similar-looking structures or fossils. This is valuable in cases where object detection and segmentation models fail or when additional information is needed. For instance, if Object Detection or Instance Segmentation identifies a fossil and does not distinguish what type of fossil, Image Classification can be used to make that differentiation. Image classification models are relatively easy to train and can handle a large number of labels and classes in datasets. This makes them computationally efficient and accessible on both CPU and GPU.

Object Detection takes image classification a step further by allowing multiple object classification and localization within a single image. This method is more suitable for outcrop images of various scales, unlike Image Classification, which only works well on close-ups of an outcrop. Object Detection algorithms identify the geological objects in the image and mark their location with bounding boxes, providing exact coordinates for each object. However, the algorithm only classifies the object located in the middle of the bounding box and may not recognize any additional objects within it.

Instance Segmentation goes beyond Object Detection by assigning masks around the geological objects, capturing their shape within the bounding boxes. This method provides the most comprehensive information about the objects' class/label and location within the image or outcrop. While bounding boxes may miss certain objects, masks capture the precise geometry and shape of each object within the bounding boxes. In geology, it is important to know the exact boundaries between sedimentary structures and lithology types, as the boundaries make a clear distinction between the geological features and their arrangement in the outcrop. The features, their combinations, and arrangement give essential information about the depositional environment interpretation.

Figure 3-2 visually summarizes the key differences between Image Classification, Object Detection, and Instance Segmentation, the three Computer Vision methods used in this thesis.

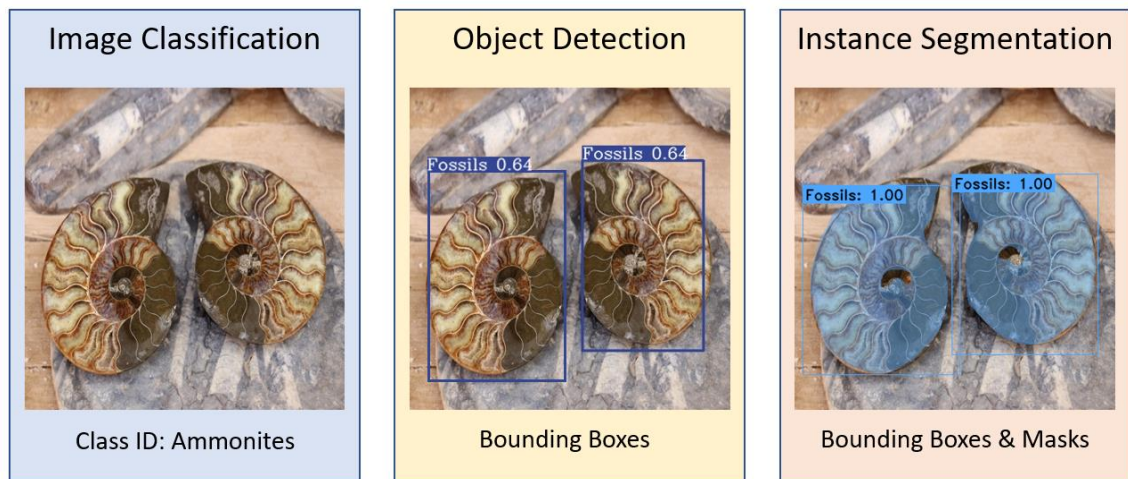


Figure 3-2: Differentiation between the three common Supervised Learning Computer Vision methods.

To obtain a complete visual characterization of an outcrop, all three methods should be used in increasing complexity. Starting with image classification and then moving on to object detection and instance segmentation is a natural progression in computer vision tasks as each method builds upon the previous one. Image Classification provides information about what types of sedimentary structures/fossils occur in the outcrop, Object Detection shows where the particular features occur within the outcrop, and finally, Instance Segmentation allows the estimation of lithology and defines the shape and size of each feature. Although Instance segmentation is the most complex and more descriptive method compared to the other two, all three methods should be used to extract geological information from the images because each method provides a different level of information about the geological features.

This approach, combining the three CV methods, can provide much information about the geological features present in the examined outcrop, saving a lot of time and effort for the human geologist.

3.2.1 Image Classification

Image classification is the process of assigning an object to a pre-existing group or class based on observed attributes. In the context of outcrop images, it can be used to recognize the presence of objects similar to the geological features the model was trained with. This process involves training a model using labeled examples to help it identify specific target classes or objects within an image.

The classification is performed at a high level across the entire image. The classification process considers the entire image as a whole and makes a general decision about what objects or regions are present in the image. This type of classification is often used for tasks such as image-level classification or scene classification, where the goal is to classify the image as a whole rather than identifying individual objects or regions within the image. Among the various classes represented in an image, the image classification method classifies the image according to the most prominent. Thus, Image Classification classifies an outcrop image based on the prominent geological feature. This in geology is not ideal as, most of the time, multiple sedimentary structures and geological features are present in the outcrops. However, it can be very useful if we are to differentiate between two similar fossils or sedimentary structures. The results of the developed geological image classifier are described in Chapter 5.

3.2.2 Object Detection

Object Detection systems analyse new images by comparing them to already stored objects within trained models to identify if any of these features are present in the new images and locate them (Lazebnik, et al., 2006). Such systems can perform both recognition and localization tasks, meaning they can identify the object present in the image and determine its location in the image or world coordinate system (Redmon, et al., 2016). In simpler words, Object detection identifies if the particular object occurs in the image and how similar the occurrence is to the particular object entity. It identifies the presence, location, and number of one or more entities in an image and labels them accurately. In such a task, the goal is to get an algorithm to predict the class and bounding box location of each instance in our image. When compared to image classification, where we only classify an image according to the predominant class, Object Detection introduces the concept of Image Localization, which helps us identify the class and location of multiple objects in the given image. This way, we can predict the location along with the class for each object by forming bounding boxes around each object. To apply this method to outcrop data, we trained an Object Detection model to predict fossils and sedimentary structures at various outcrop scales (Chapter 6).

There are different types of Object Detection systems, classified according to the problems they solve. Simple recognition tasks involve identifying fully un-occluded 2D objects that appear in a uniform background with controlled lighting conditions (no

reflections or shadows) (LeCun, et al., 2004). However, object recognition becomes difficult when the background is highly textured, lighting conditions are unknown or uncontrolled, there are too many objects in the scene, the number of objects in storage is very large, or when some objects may be touching and occluding other objects. The latter is the case for geology, as outcrop images are often very information-rich, containing multiple sedimentary structures and lithology layers, various lighting, shadows, and noisy background (e.g., vegetation), making it challenging for a model to identify all the geological features.

3.2.2.1 Feature Selection and Object Detection

Feature selection involves identifying the attributes of an image that aid in recognizing the object within the image. Object detection utilises features, such as patterns and shapes, to identify the relevant object(s) in the image (Sun, et al., 2004). Hence, it is crucial that the selected features provide compelling evidence of the object in question.

To effectively differentiate objects in the feature space, it is essential that the visual features be distinctive. The feature space plays a critical role in many machine learning algorithms, as it provides a way to represent data in a structured format that can be used for classification, clustering, regression, and other tasks.

The selection of features is closely linked to how the objects are represented (Dubey, 2022). The feature space is constructed based on the selected features, and the choice of features can significantly affect the performance of machine learning algorithms. For example, if the selected features are not distinctive enough, the objects in the feature space may overlap and become difficult to differentiate, leading to poor classification or clustering results. On the other hand, if the selected features are too complex or not relevant to the task at hand, they may introduce noise and decrease the performance of the algorithm.

In image analysis, edge detection is frequently employed to identify changes in image intensities that correspond to object boundaries. This is because such boundaries are typically associated with marked changes in image intensities. Moreover, the intensity of the edges is less susceptible to variations in illumination levels. Therefore, algorithms that are designed to track object boundaries utilise edges as important representative features of the image.

The texture of an object can cause variations in the intensity of the light that is reflected from its surface, thereby conveying properties of the object's surface, such as its smoothness and regularity (Ali & Sharma, 2017). Generating descriptors for texture requires an additional processing step, and the selection of representative features of an object typically depends on the application domain.

In Object Detection, while object edges and texture are critical for contour-based representations, colour is another crucial feature for histogram-based representations. In terms of an object's appearance, combinations of features are helpful in determining what the object represents. The RGB colour space, for instance, which comprises the primary colours of red, green, and blue, is commonly employed for object representation. All the images used in this thesis use the RGB colour profile. The HSV (Smith, 1978), which incorporates the hue, saturation, and value dimensions, was used as a colour augmentation technique, as will be discussed in Chapter 4.

Although colour is often considered an important feature, a large portion of colour bands are sensitive to illumination variation. As a result, in situations where such an effect cannot be avoided, alternative features are used to model the object's appearance. This applies directly to outcrops as the collection of outcrop images heavily depends on the time of the day and the weather conditions under which the photos are taken. These will affect the light intensity, colour, and shadows in the images.

Object detection sometimes turns out to be challenging, and its application in geology is a good example of this challenge. Object detection struggles with complex shapes of objects, occlusion of objects partially or in full, articulated or nonrigid nature of objects, illumination changes in the scene, the need for real-time processing of the scenes, loss of information due to the projection of 3D world on a 2D images, and image noise. Outcrops include many of the aforementioned characteristics, such as noise in the images, variable illumination conditions, occluded objects, and very complicated shapes of objects, making their detection by machines a real challenge. According to the results of Chapter 6, Object Detection shows accurate results regarding the identification and localization of sedimentary structures and fossils from outcrop and core images. In some cases, where it fails, it can at least be helpful in pinpointing certain locations on the outcrop of geological interest by displaying sedimentary structures or fossils that can be further examined by image classification and instance segmentation.

In this project, the Object Detection model employed was the sixth version of the YOLO (Redmon, et al., 2016) family of models, called YOLOv6 (Li, et al., 2022). YOLOv6 is a CNN, state-of-the-art object detection model that was released in 2022 as an improved version of the popular YOLO (You Only Look Once) family of models. YOLOv6 is designed to be faster and more accurate than its predecessors, outperforming the previous versions through a streamlined architecture that reduces computational overhead and optimizes memory usage, allowing for fast and efficient processing. It can detect objects with high precision and recall even in complex and crowded scenes, with improved detection performance on a variety of object detection benchmarks. Lastly, its flexibility allows for easy customization and hyperparameter tuning to suit specific use cases and requirements by supporting multiple backbones and different configurations. All these reasons justify the selection of YOLOv6 as the most suitable model for geological Object Detection.

3.2.3 Image Segmentation Overview

Image Segmentation plays an essential role in localizing objects and defining their shape, making it one of the most commonly used techniques in Computer Vision (Haralick & Shapiro, 1992). It is a crucial step in developing intelligent systems that can interact with the environment and support human work. Image segmentation is widely employed in various fields, such as self-driving cars (Krizhevsky, et al., 2012), (Krizhevsky, et al., 2017), medical image analysis (including X-rays and dental imaging) (Li, et al., 2018), and satellite imagery, among others. Image segmentation can be classified into three main categories: Semantic Segmentation, Instance Segmentation, And Panoptic Segmentation.

Semantic Segmentation aims to classify image pixels into a set of categories without differentiating separate object instances. It segments all objects of the same class in the image, but it does not distinguish instances of the same class. For example, in medical scans, we can recognize all cancer cells, but we cannot differentiate one cell from another (Long, et al., 2015).

On the other hand, Instance Segmentation differentiates each instance in the visual input that belongs to the same class. It combines object detection and semantic segmentation to detect each object in the scene while precisely segmenting each instance. In cancer

cell segmentation tasks, it accurately predicts each cell's shape and distinguishes one cell from another (He, et al., 2017).

Finally, Panoptic Segmentation is a useful and important approach that connects semantic segmentation and instance segmentation. It performs semantic and instance segmentation on a given image and combines their outcomes into one image (Kirillov, et al., 2019). It enables holistic scene understanding, which is essential for intelligent systems to comprehend the visual scene both on the pixel-wise level and the class instance level. Panoptic segmentation categorizes scenes into "stuff" and "things," where "stuff" refers to background classes like sky or sidewalk, and "things" refer to particular instances of foreground classes such as pedestrians, cars, or cancer cells (Kirillov, et al., 2019), (Cordts, et al., 2016). An example of each of the described image segmentation methods can be found in Figure 3-3.

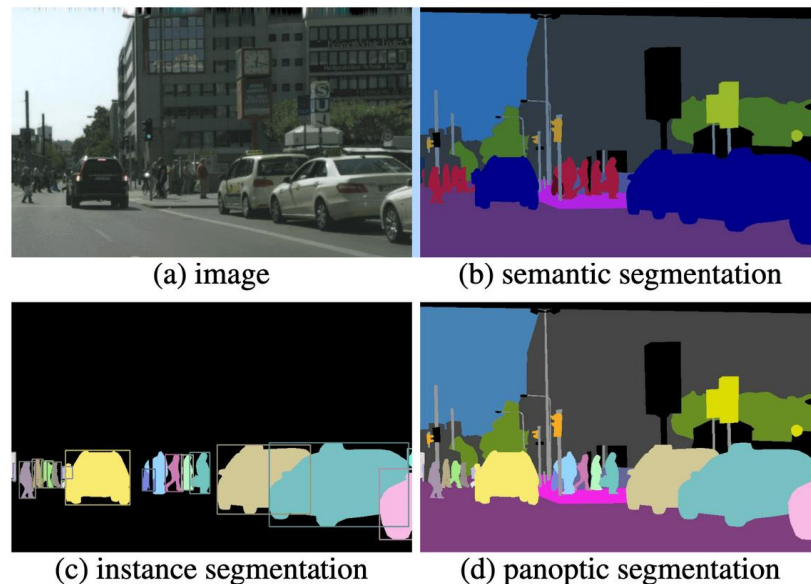


Figure 3-3: In this series of photos, (a) is the original image, and the others show three kinds of segmentation that can be applied in image annotation. In this example, the objects of interest are the cars and the people. Photo credit: Panoptic Segmentation, CVPR 2019.

In this thesis, out of the three methods, only Instance Segmentation was used because we want to identify each instance in the outcrop images that belongs to the same class and, at the same time, differentiate between the various classes present. Furthermore, we are particularly interested in extracting the exact shape of each geological object, their location in the images, and also the class they belong to. Panoptic Segmentation was also

considered at the beginning of this project. However, it was rejected early on as for outcrop description was not useful due to the addition of irrelevant information regarding the scene identification. The additional information this method provides, in this case, was acting as noise, and it did not help towards the final goal of outcrop interpretation. Using panoptic segmentation would also mean a lot more polygon masks, making the whole process computationally expensive and very time-consuming.

3.2.3.1 Instance Segmentation

The image segmentation technique enlisted for this thesis is Instance Segmentation, intended for the geological feature extraction from the outcrop images. It tracks and counts the presence, location, count, size, and shape of objects in an image. In order to apply this method to outcrop data, we divided the task into two parts: Lithology Segmentation and Sedimentary Structure Segmentation. Segmenting the outcrop in two parts allowed clear, interpretable, and accurate results. If the segmentation of both the lithology and sedimentary structures were grouped under one task and performed by a single model simultaneously, severe label misclassifications, mask overlapping, and poor mask fit would occur, as the results of Chapter 7 demonstrate. We constructed two identical models, one for each task, and optimized the segmentation workflow by testing various network configurations to ensure both speed and accuracy. The results of this application, described in Chapter 7, are particularly useful for fieldwork, as the models can be applied in real-time using a drone or another remote device for automated identification and characterization of an outcrop.

The most significant obstacle faced in the Instance Segmentation problem is effectively dealing with occluded objects of the same class. This involves accurately assigning pixels to their corresponding classes, separating various instances, and properly identifying overlapping instances.

To address this challenge, YOLACT (Bolya, et al., 2019) was used, which is a one-shot Instance Segmentation neural network performing simultaneously object classification and segmentation. Such a single-shot method is much faster than two-stage instance segmentation algorithms such as Mask-RCNN (He, et al., 2017), which first detects objects using bounding boxes and then applies segmentation head on the object proposals.

An example of a Two-stage instance segmentation architecture is illustrated in Figure 3-4.

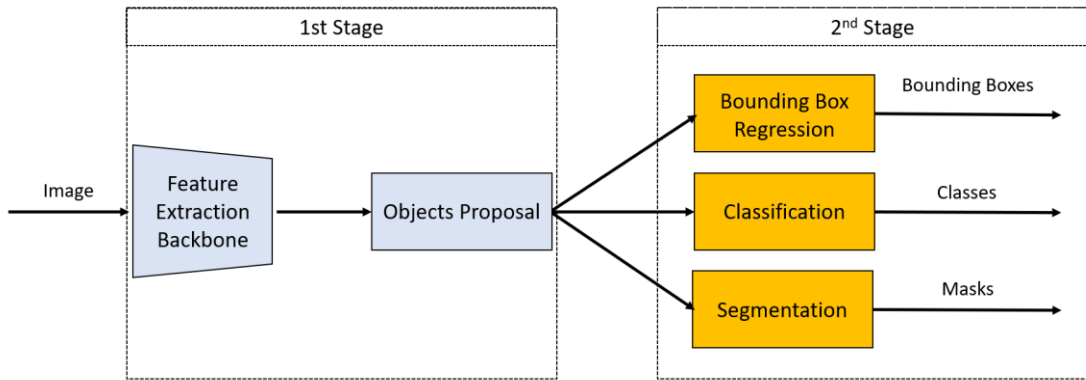


Figure 3-4: Example of a Two-Stage Instance Segmentation architecture.

Instance Segmentation can be divided further into two categories based on the architectures of the used models. First, detection-based instance segmentation or commonly referred to as a proposal-based method. To detect various objects in an image, this method utilises a detection network. Additionally, a segmentation head is run on each detected bounding box to obtain instance segmentation. This method runs the aforementioned steps sequentially, which is why it is called a Two-stage instance segmentation (He, et al., 2017).

An example of a One-stage instance segmentation architecture is illustrated in Figure 3-5.

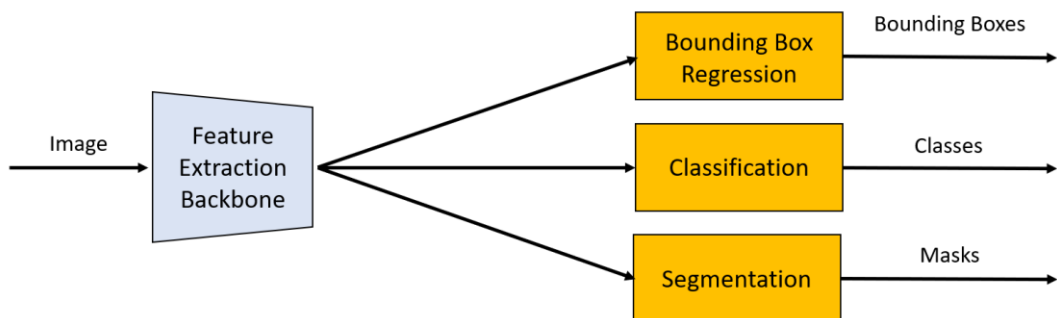


Figure 3-5: Example of a One-Stage Instance Segmentation architecture.

Single-shot or one-stage instance segmentation, also known as the proposal-free method, is a real-time approach that is typically much faster than detection-based methods because it uses two parallel processing branches, eliminating the need for explicit localization steps (Bolya, et al., 2019). The details of the YOLACT model are explained in detail in Chapter 7. The reason I chose YOLACT over the other models was that it provided a

good trade-off between prediction accuracy and inference speed, according to Bolya et al. (2019), as shown in Figure 3-6. Compared to other segmentation models, YOLACT allows real-time segmentation with inference speeds up to approximately 45 frames per second (fps). Although its mean average precision (mAP) score is lower than in other models, it is still the chosen model as its real-time capabilities make it a valuable tool for geologists as it can be used for real-time outcrop segmentation out in the field.

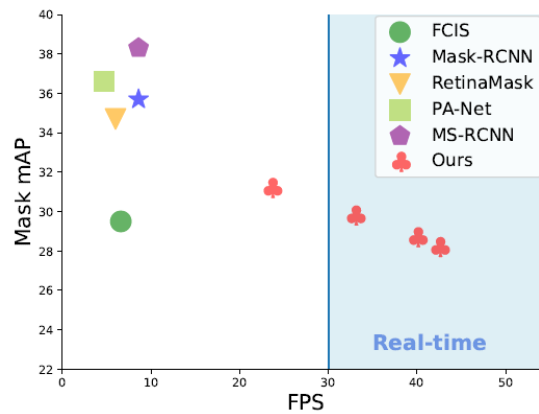


Figure 3-6: Speed-performance trade-off for various instance segmentation methods on COCO. To our knowledge, ours is the first real-time (above 30 FPS) approach with around 30 mask mAP on COCO test-dev (Bolya, et al., 2019).

3.2.3.2 Unsupervised versus Supervised Segmentation

This short section presents the application of unsupervised segmentation techniques, such as K-means clustering, when applied to an outcrop image. These results are compared with the results of the YOLACT model applied to the same outcrop image, as shown in Figure 3-7.

Unsupervised Segmentation

Supervised Segmentation

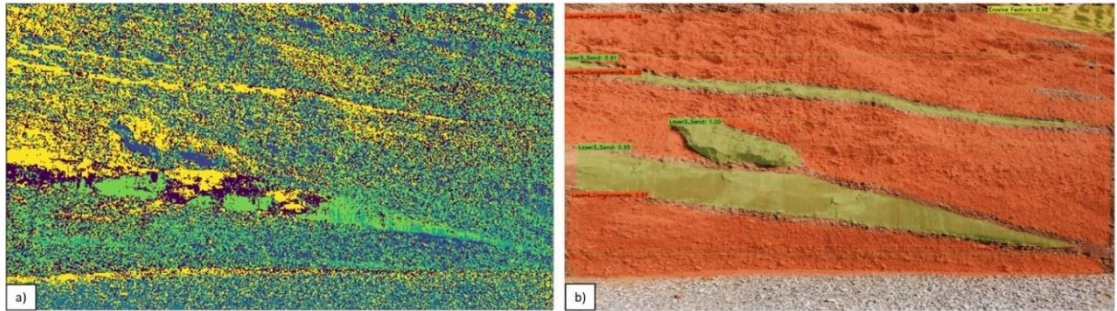


Figure 3-7: a) Unsupervised Segmentation by using the K-Means clustering algorithm vs. b) Supervised Segmentation by using the YOLACT Instance segmentation model.

In Figure 3-7, a comparison between supervised and unsupervised segmentation on the same outcrop image is presented. The unsupervised example in Figure 3-7a exhibits unnecessary complexity while still failing to produce clear segmentation results. The K-means algorithm was tested with 3 and 4 clusters, and in both instances, it produced unclear and not useful results. In contrast, the supervised segmentation in Figure 3-7b generates distinct segments of the outcrop by capturing the shape of geological features, assigning a probability to each prediction, and, most importantly, providing a label for each segment. Figure 3-7b is clearly segmented between the conglomerate (orange) occupying the majority of the outcrop image and the sandstone layers (green).



Figure 3-8: Example of the Test outcrop image used in Figures 7 and 9.

Figure 3-8 shows the original, unsegmented outcrop image for a clear comparison with the results from Figure 7. The orange/brown conglomeratic bed is the majority of the outcrop. Toward the bottom part of the outcrop, another sandstone bed (channel) intrudes

into the conglomeratic bed. The difference in the texture between the two beds is quite obvious to the human eye. Something noteworthy here is that the bottom grey part of the image does not belong to the outcrop; it is actually the beach, which is an irrelevant feature for the geologic interpretation. In other words, the grey part can be considered as noise. Now comparing Figure 3-8 with Figure 3-7b, it is obvious that the supervised segmentation does an excellent job of identifying, locating, and contouring the different geological features, excluding the beach part of the image. On the contrary, when Figure 3-8 is compared with Figure 3-7a, the unsupervised segmentation is not successful at distinguishing the geological features at all. The clusters are not clear, and also, the bottom grey part is also included in the segmentation, although it is a completely irrelevant feature and does not belong to the actual outcrop. From these results, it is clear that, for the objectives of this thesis, Supervised learning is more appropriate.

Edge detection, another unsupervised technique, was tested on the outcrop image of Figure 3-8. This method was expected to yield favorable outcomes when applied to outcrop images, based on how this method works, which, as shown from the results in Figure 3-9, does not hold true. Edge detection is well-known for providing reliable results at various image scales, and one would think that it would be a method particularly suited to geology, where sharp edges and thin structures are commonly produced by geological processes. Geological features can be complex, subtle, and highly variable, requiring a combination of visual, contextual, and analytical skills to decipher. Their interpretation often requires a combination of visual, contextual, and analytical skills.

Figure 3-9 shows the test outcrop image to which three Edge Detection techniques, such as Gradient-based methods (Saif, et al., 2016; Sobel & Feldman, 1973), LoG (Otsu, 1979), and Canny Edge (Canny, 1986), were applied. The unsupervised segmentation methods depicted in Figure 3-9a, b, and c fall short of providing sufficient information about the segments. These methods only separate the segments without assigning any labels or describing the objects, and as in Figure 3-7a, they fail to exclude irrelevant information.



Figure 3-9: Unsupervised Segmentation Edge Detection techniques, including a) Sobel, b) LoG, and c) Canny Edge detection.

The extraction of the geological features and prediction of their labels is necessary for the interpretation of geological features and in the proposed approach of this thesis. As demonstrated in Chapter 8, these labels serve as inputs to a custom neural network that integrates all visual evidence and forms geological concepts. Supervised learning can provide all these necessary labels and information about the geological features, while unsupervised learning fails to provide this information. For the above reasons, Supervised learning was deemed more appropriate to use in this thesis.

3.3 Common Metrics Used for the Evaluation of Computer Vision Models

According to the model and methods used for Computer Vision and Machine Learning problems, there are always metrics to evaluate the model's performance. In this thesis, three metrics were used to assess the previously described methods: Confusion Matrix for the Image Classification problems and Intersection over Union (IoU) along with Mean Average Precision (mAP) for the Object Detection and Instance Segmentation. The confusion matrix was used to determine which is the best-performing classifier for the image classification task presented in detail in Chapter 5. The Mean Average Precision (mAP) for the Object Detection and Instance Segmentation was used to evaluate the performance of the models described in Chapters 6 and 7 and monitor the models' improvements based on these scores. Both of these metrics will be used to evaluate the pattern classification results across the three CV methods.

3.3.1 Confusion Matrix

This metric is used to evaluate the accuracy of the Image Classification model presented in Chapter 5. The Confusion Matrix is used to evaluate the performance of the geological image classification model by comparing the predicted and actual values of a set of test

data. By using a confusion matrix, we can get a clear and comprehensive understanding of how well a classification algorithm is performing and make informed decisions about how to improve it. The confusion matrix provided information about which geological classes were predicted more accurately and which did not. It also acted as a guide on whether is necessary to improve the model's predictions based on the number of misclassifications and overall test accuracy.

3.3.2 Intersection over Union (IoU)

This metric is used to measure the extent to which the predicted bounding box of an object overlaps with the ground truth bounding box for object detection and instance segmentation tasks in Chapters 6 and 7. The value of IoU can be used to evaluate the accuracy of object detection algorithms and to determine the threshold for true positive predictions. IoU values range from 0 to 1, with a value of 1 indicating perfect overlap between the predicted and ground truth bounding boxes. In practice, a threshold value of the IoU is typically used to determine whether a prediction is considered a true positive or a false positive. The threshold value can vary depending on the specific task and dataset, and in this thesis, the IoU values considered were between 0.5 and 0.95.

3.3.3 Average Precision

Average Precision (AP) is used also in Chapters 6 and 7, as it is a common evaluation metric in Computer Vision tasks like Object Detection and Instance Segmentation (Everingham, et al., 2010), (Russakovsky, et al., 2015) measures the quality of the model's predictions, specifically how well the predicted bounding boxes or masks overlap with the ground truth objects. To calculate Average Precision, first, Precision and Recall values are calculated for different IoU thresholds. We vary the IoU threshold from 0.5 to 0.95 in steps of 0.05 and calculate Precision and Recall at each threshold.

3.3.4 Mean Average Precision (mAP)

The Mean Average Precision or mAP score is used to evaluate the overall performance of a model across multiple classes or categories. Essentially, it represents a summary statistic that provides a single score reflecting the overall accuracy of a system. The formula to calculate that score is based on the Confusion Matrix, Intersection over Union

(IoU), Recall, and Precision, which were previously mentioned in Chapter 2. The mAP scores are used to evaluate the accuracy of the Object Detection and Instance Segmentation models, YOLOv6 (Li, et al., 2022) and YOLACT (Bolya, et al., 2019), respectively, as presented in Chapters 6 and 7.

In object detection, the model is trained to identify and locate the geological objects in an image by drawing bounding boxes around them. The IoU (Intersection over Union) measures the overlap between the predicted and the ground truth bounding boxes.

In Instance segmentation, the model is trained to identify, locate and contour the shape of the geological objects in an image by drawing masks and bounding boxes around them. This time, the IoU (Intersection over Union) measures the overlap between the predicted and the ground truth masks and bounding boxes.

The mAP@0.50:0.95 metric is an extension of mAP that measures the AP at different IoU thresholds between 0.50 and 0.95, with a step size of 0.05. It calculates the AP for each object category separately and then computes the mean AP across all categories. The thresholds between 0.50 and 0.95 are used in this project because they are commonly used in object detection benchmarks such as COCO (Common Objects in Context) and PASCAL VOC (Visual Object Classes). It is considered to be a more comprehensive metric than mAP@0.50, which only measures the AP at a single IoU threshold of 0.50.

The mAP is a useful evaluation metric because it takes into account the performance of a system across all classes, not just a single class, and it is commonly used in benchmark datasets such as PASCAL VOC and COCO, which makes it a standard measure for evaluating the performance of Object Detection and recognition algorithms (Everingham, et al., 2010).

Geology does not have benchmark datasets to compare the mAP scores with. Therefore, the accuracy of detection and segmentation models refers to the formation and fit of masks and bounding boxes as well as to the label assignment per instance/geological object within every image. The model will predict every geological instance in every image and assign a confidence score for each prediction calculated by the mAP scores and localize each feature by drawing the masks and polygons around them. The model's confidence for predictions depends on mAP. The model is evaluated by mAP and geologists. If geologists validate the annotations, then the only criteria for model

evaluation are the mAP. There is a need to manually evaluate the results to ensure the model has learned what we need it to learn.

3.4 Computer Vision Models & Model Backbones

This section introduces the specific models and backbones used for the Computer Vision aspect of this thesis. The Computer Vision and CNN models used across various industries rely on specific architectures known as "model backbones." As mentioned earlier, backbones are feature extraction networks that are part of Deep Learning model architectures, while hyperparameters determine how the network is trained. For this thesis, seven pre-built backbones have been utilised for the Computer Vision models presented in Chapters 5, 6, and 7.

For Chapter 5 (Image Classification), a custom Image Classification model was built using the Pytorch framework (PyTorch, 2016), utilising five different backbones. The backbones used for this chapter are from the ResNet (He, et al., 2016) and VGG families (Simonyan & Zisserman, 2014), and more specifically, ResNet18, ResNet50, ResNet101, VGG16, and VGG19. The choice behind these 5 backbones can be justified as it was important to test the validity of my conclusions across different model architectures to ensure that my results were consistently accurate and, furthermore, to create a comparative study for these backbones.

For the Object Detection chapter (Chapter 6), the model that was used is called YOLOv6s (Li, et al., 2022), with the CSPDarknet53 as a backbone, a version of the DarkNet53 backbone. The YOLOv6 model has several different versions, each with a different level of computational complexity and accuracy (Li, et al., 2022). YOLOv6s (the "s" stands for "small") is the smallest and fastest backbone version of the YOLOv6 model, with the fewest number of layers and the lowest computational requirements.

The YOLOv6s model was selected because it has been optimized for faster inference speed, which makes it suitable for real-time applications and geological applications in the field. It has been shown to achieve state-of-the-art accuracy on various Object Detection benchmarks, which makes it suitable for applications that require high accuracy and fast inference. It is highly maintained, and the amount of research and available resources available to make it easy to use and customize for specific applications. Saying

that it can be fine-tuned on custom datasets, also making it suitable for use in the detection of geological objects from outcrop images.

For Chapter 7 (Instance Segmentation), the YOLACT model was used to segment the geology from outcrop images utilising three different backbones, ResNet101, DarkNet53, and my custom version of the DarkNet53 called ‘cDarkNet53’. The default DarkNet53 backbone was used to establish whether YOLACT has the potential to be used for this thesis, while the other two backbones were chosen to create a comparative study and improve the model’s robustness.

YOLACT (Bolya, et al., 2019) is a Single-shot or one-stage instance segmentation, also known as the proposal-free method. It is a real-time approach that is typically much faster than detection-based methods but shows slightly less accuracy. This model was chosen in this thesis as it has been thoroughly researched and tested with multiple benchmark datasets, as Bolya et al. (2019) show in their research, and has proven to be a robust model for various applications.

3.4.1 Residual Networks: Resnet18

ResNet18 is a specific variant of the Residual Network (ResNet) architecture. It is a relatively shallow version of ResNet, with only 18 layers, and has become a popular baseline architecture for many Computer Vision tasks. Its relatively small size and computational efficiency make it a good choice for applications with limited computational resources.

3.4.2 Residual Networks: Resnet 50

ResNet-50 is the second variant of the Residual Network (ResNet) architecture used in this thesis. It is a deeper and more complex version of ResNet compared to ResNet-18, with 50 layers. ResNet-50 has been shown to achieve state-of-the-art performance on a variety of Computer Vision tasks, including Image Classification, Object Detection, and Semantic Segmentation. It is deeper and more complex than ResNet-18, which can allow it to capture even more complex features and achieve higher accuracy, but it also requires more computational resources and may be more difficult to train. However, it is a popular choice for many Computer Vision tasks due to its strong performance and versatility.

3.4.3 Residual Networks: Resnet 101

To create even deeper ResNets, 101-layer ResNet was constructed by incorporating more 3-layer blocks (He, et al., 2016). ResNet-101 is the fourth variant of the Residual Network (ResNet) architecture, and it is a deeper and more complex version of ResNet compared to the variants previously described.

3.4.4 VGG 16

The construction of this network uses very tiny convolutional filters, something that, in some cases, might lead to loss of information if the input image size is reduced significantly. Thirteen convolutional layers and three fully connected layers make up the VGG-16.

VGG16 is a sizable network with about 138 million parameters in total. Even by today's high standards, it is a sizable network. The network is more appealing due to the simplicity of the VGGNet16 architecture, nevertheless.

3.4.5 VGG 19

The concept of the VGG19 model (also VGGNet-19) is the same as the VGG16 except that it supports 19 layers. The “16” and “19” stand for the number of weight layers in the model (convolutional layers). This means that VGG19 has three more convolutional layers than VGG16. VGG-16 has approximately 138 million trainable parameters, while VGG-19 has approximately 144 million trainable parameters.

3.4.6 DarkNet 53

Darknet53 is the convolutional neural network architecture used in Chapter 7 as the default backbone for the YOLACT model. The architecture is designed for use in object recognition, classification, and segmentation tasks, specifically for image and video processing. It is designed to be efficient, with a relatively small number of parameters compared to other popular neural network architectures, and it is flexible as it can easily be adapted and optimized for different use cases and datasets (Redmon & Farhadi, 2018). As the results of Chapter 7 demonstrate, the Darknet53 architecture yielded very good results for the segmentation of the sedimentological features from an outcrop image.

3.4.7 CSPDarknet53

CSPDarknet53 (Wang, et al., 2019), (Bochkovski, et al., 2020) is a convolutional neural network that was used in Chapter 6 as the backbone for YOLOv6S. It is designed to balance the number of parameters and computation while maintaining high accuracy in object detection tasks. Such CNN has been used as the backbone for YOLOv4, YOLOv5, and YOLOv6. YOLOv6s modifies this backbone by adding SPP (Spatial Pyramid Pooling) and PAN (Path Aggregation Network) modules to improve the network's ability to detect objects at different scales and locations. As mentioned earlier in this thesis, geology is a multiscale domain, and thus, the choice of YOLOv6S with the CSPDarknet53 as a backbone is the perfect candidate for the detection of geological features.

As described in Chapter 5, the backbones selected were Resnet18, Resnet50, Resnet101, VGG16, and VGG19. The selection criteria were the model depth and computational power. As we move from Resnet18 up to VGG19, the number of trainable parameters significantly increases. I needed to test how the geologic feature image classification improves or not based on the number of trainable parameters. The results of the application of a custom Image Classification model on outcrop images utilising the aforementioned backbones are presented in Chapter 5.

For the segmentation results shown in Chapter 7, the backbones used were Resnet101 and Darknet53. Instance Segmentation was performed mainly with the DarkNet53 and a modified version of it I called cDarkNet53, which is discussed in section 7.4.2 in Chapter 7. Section 7.4.3 in Chapter 7 shows the application of Instance Segmentation with the ResNet101 backbone and demonstrates a comparison between the results of the segmentation of outcrops with both backbones. Broadly speaking, the ResNet101 variant, as discussed above, is very good at Image classification, texture extraction, and pattern recognition. At the same time, DarkNet53 is good at locating and classifying objects fast, allowing real-time inference with a high number of frames per second (FPS).

3.5 Natural Language Processing (NLP)

This section briefly describes steps D and E from the high-level workflow (Figure 1-3) introduced in Chapter 1, presenting how Natural Language Processing (NLP) and Neural Networks (NN) can be combined to create multiple geological concepts based on the

published available data (textual interpretations of outcrops) and the collected observations from the outcrops.

Document Processing and Information Extraction are the two aspects of NLP employed for this project. As shown in Chapter 8, NLP proved to be a valuable tool for processing the bulk of texts published in the geological literature. Geology is a very information-rich domain; thus, an automated method to extract information from a large volume of texts was necessary. Furthermore, NLP assisted in transforming the extracted information into a clean text format (strings of text).

Natural Language Processing assists in extracting information from the established literature, from already published interpretations, and facies assemblages. Furthermore, it can help edit and manipulate the text into a form that can be used as input into the Neural Network described briefly in the next section and in detail in Chapter 8.

3.5.1 Document Processing and Information Extraction

This aspect of NLP is used in this thesis to mine text from the geologic literature, aiming to quickly scan pdf documents and locate specific geological keywords or expressions of interest based on a custom file containing these words. Such keywords include sedimentary structures, lithology, fossil types, facies assemblages, and types of depositional environments. The NLP identifies if these words or combinations of these words exist in the examined document, extracts them, and puts them into an Excel file, making them readable and in a clean text format.

A basic NLP pipeline was used to process the text from pdf files and extract only the necessary geological information, following the steps of Tokenization, Text Cleaning, POS Tagging, Stop words, and Lemmatization.

3.5.1.1 Tokenization

Tokenization is a fundamental process in natural language processing (NLP) that involves breaking down a text document into smaller units, known as tokens (Chaitanya, 2020). In Tokenization, the text in the chosen geological pdf document is first normalized, which involves converting all the characters to lowercase or uppercase, removing punctuation

marks, and handling special characters and symbols. After normalization, the document is split into tokens based on predefined rules or patterns (Manning, et al., 2008).

3.5.1.2 Text Cleaning

Text cleaning is the process of removing or correcting unwanted characters, words, or data from text data before fully processing it with Natural Language Processing (Chaitanya, 2020). This step reduces the number of characters in the document while keeping the desired geological terms.

In Natural Language Processing, a PhraseMatcher is a tool provided by the spaCy library for matching a list of phrases against a large text. The spaCy PhraseMatcher performs pattern matching very quickly and is highly customizable, allowing users to define their matching rules based on their specific use case. The 'en_core_web_sm' module from the spaCy library was used to help in the text cleaning. This module is a small English pipeline trained on written web text (blogs, news, comments), including vocabulary, syntax, and entities.

I developed a custom PhraseMatcher which takes geological keywords (e.g., cross-bedding, sandstone, ammonite, etc.) from the custom 'keywords' text file and matches them to the words from the geological pdf files allowing the user to efficiently find occurrences of specific phrases or terms within the text. The Text Cleaning step improves the accuracy of my NLP model by reducing the number of irrelevant features (words, numeric characters, etc.) that the model needs to consider.

3.5.1.3 POS Tagging

POS tagging stands for Part-of-Speech tagging, a fundamental task in Natural Language Processing that involves the assignment of a label to each word in a sentence corresponding to its respective part of speech, such as noun, verb, adjective, and adverb (Chaitanya, 2020), (Chowdhary, 2020). POS tagging aims to provide insight into a sentence's structure and meaning and enable machines to understand the grammatical relationships between words, thus, to precisely identify and extract the relevant geological terms from the text based on their grammatical categories.

3.5.1.4 Stop Words

Stop Words refer to words commonly used in a language but do not carry significant meaning in a text (Chowdhary, 2020). The spaCy library provides a built-in set file of 'stopwords.' Examples of Stop Words include "the," "a," "an," "in," and "of." Stop Words are typically removed from the text during the preprocessing stage in NLP to reduce the computational load, improve the accuracy of text classification, simplify the text, and highlight the more meaningful words, in this case, the geological terms such as sedimentary structures, fossils, and lithology types.

3.5.1.5 Lemmatization

Lemmatization is a natural language processing technique that involves reducing words to their base or dictionary form, known as the lemma (Chowdhary, 2020). Lemmatization helps standardize words and improve text analysis and information retrieval by grouping variations of the same word. By standardizing words, lemmatization improves text analysis and information retrieval by reducing the number of unique words in a text and grouping together variations of the same word. This reduces the sparsity of the data, which can improve the accuracy of machine learning models that rely on text data. For example, if the algorithm encounters in the text the terms “upward-fining,” “upwards fining,” or “fining-upward,” it will know to group these terms under a single term, “fining upwards.”

These aforementioned five components are used to form a custom NLP workflow to extract critical (to the depositional environment interpretation) geological terms and knowledge from the selected pdf files. This extracted knowledge is inherent and easy to use for a human geologist to interpret an outcrop. However, teaching a computer model to understand the meaning of different geological features and their arrangements, as well as what their presence or absence indicates in a geological setting, is a difficult task.

This custom workflow allows the geologist to quickly scan many pdf files from the geological literature to extract the necessary geological knowledge, utilised in the final step of the workflow by a neural network to interpret the geology. In this thesis, a total of 14 geological pdf files were used as a starting point, including several keywords and tables, in which (tables) part of these keywords are combined into sedimentological sequences that describe depositional environments. These files were processed by my

algorithm until the desired keywords were extracted and combined into a single Excel file, which was manually checked for grammar, spelling, and geological correctness of the results. The generated Excel file, including strings of text, is used by the neural network explained in the following section to embed the domain knowledge into my AI system.

3.6 Custom Neural Network to Interpret the Geology

The custom Neural Network model, described in detail in Chapter 8, combines all this domain knowledge extracted with the help of NLP and the Computer Vision results of Chapters 5, 6, and 7, which provided visual evidence from the outcrops. According to the Chapter 8 findings, the presented Neural Network successfully forms multiple interpretations of the depositional environment based on the input provided by the user. The user can choose different geological features from a long list compiled from all the aforementioned results through a customized Graphical User Interface. Then, the network analyses the inputs and produces several scenarios, each with an assigned probability.

The results from the three Computer Vision methods were manually translated to strings of text, which were added to the Excel spreadsheet generated by the NLP model. For clarity, an example is demonstrated in the following figure (Figure 3-10) and its description.

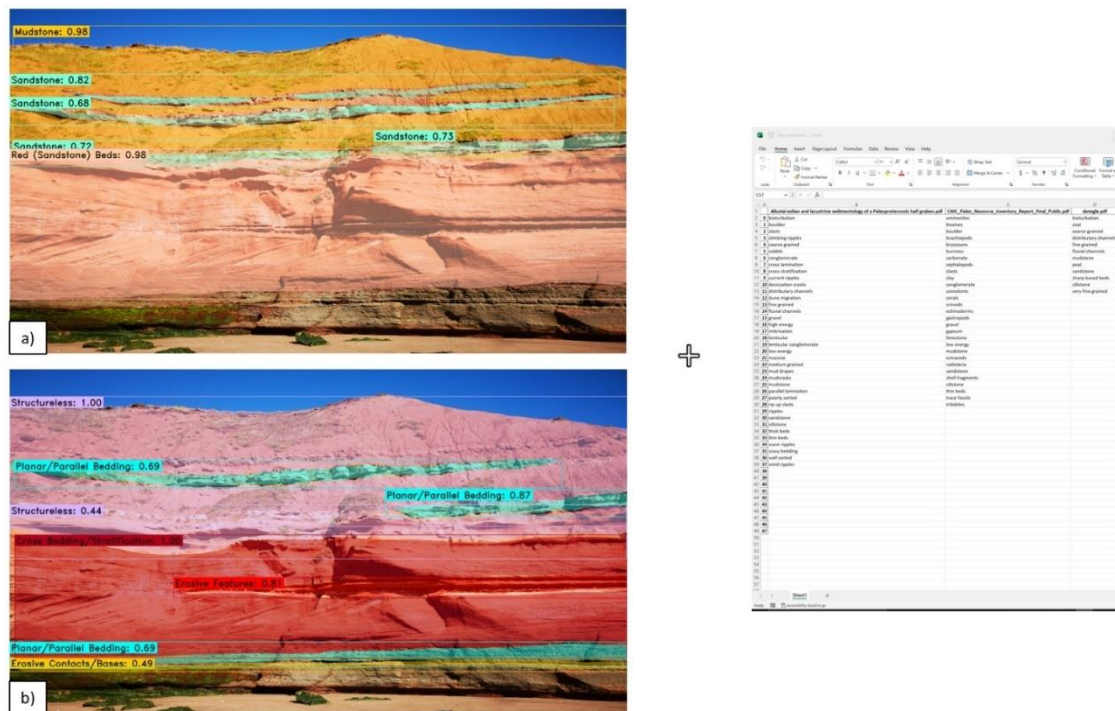


Figure 3-10: a) Segmentation of an outcrop's lithology b) Segmentation of an outcrop's sedimentary structures.

Figure 3-10 shows the predicted labels from the segmentation model presented in Chapter 7. Figure 3-10a shows the segmentation of the lithology, while Figure 3-10b shows the segmentation of the sedimentary structures of the outcrop. The unique geological labels from the segmentation model, according to Figure 3-10, are mudstone, sandstone, red sandstone beds, structureless, planar/parallel bedding, cross-bedding/stratification, erosive features, and erosive surfaces/contacts. These labels are manually added to the Excel spreadsheet generated by the NLP model previously described. This is only information from one outcrop, but the more outcrops the CV models are tested with, the richer the final list of geological features will be. This list of strings of text representing sedimentological features serves as the training data for the final neural network model. The model is trained on the text data and their potential combinations to produce depositional environment scenarios.

This model, once trained, is incorporated into a Graphical User Interface (GUI), which accepts input from the user, and according to the provided input and the combinations of geological features the NN model was trained with, the GUI generates possible interpretations of the geology. More specifically, the concatenations of the geologic keywords, such as sedimentary structures, fossils, and lithology types, yield multiple

depositional environment interpretations accompanied by a prediction probability. The probability for each prediction is based on the combination of the individual inputs, is independent of the other predictions, ranges from 0-1, and the final result is a list of possible interpretations ranked from the highest to the lowest probability. The results of this model and the model's specifics are all explained in detail in Chapter 8.

3.7 Chapter's Conclusions

The Computer Vision methods employed in this thesis aimed to extract visual information from 2D outcrop images or videos and identify various geological features such as lithology types, sedimentary structures, and fossils. To achieve this, Supervised Learning was deemed more appropriate as it provides labeled predictions for the geological features, unlike unsupervised methods that only provide clusters without interpretability and context of what each cluster in the image resembles.

Image Classification is a useful method for categorizing single fossils or sedimentary structures based on their appearance in close-up images, but it is not suitable for entire outcrop images. However, Image Classification can differentiate between similar-looking structures or fossils, and it can be used to complement object detection and segmentation models. Object Detection can identify and locate multiple objects in a single image using bounding boxes, while Instance Segmentation goes further by assigning masks around objects to capture their precise geometry and shape. In geology, it is important to know the exact boundaries between sedimentary structures and lithology types, and instance segmentation provides the most comprehensive information about object class/label and location within the image or outcrop.

The results for all the Computer Vision methods used on static images are shown in the corresponding chapters. Additionally, for the completeness of my research, I will upload a multimedia file demonstrating how the models work in real-time applications both for object detection and instance segmentation.

After collecting all the visual data with the CV models, Natural Language Processing (NLP) is used to mine additional geological labels, interpretations, and facies assemblages from established literature to enhance the model's understanding of all the individual pieces of information.

Finally, a custom neural network (NN) integrates all the available information, including observations from the outcrop and geological domain knowledge, to generate multiple interpretations of the depositional environment based on the outcrop images.

Table 3-1 is an overview table listing all models with their corresponding backbones used in this thesis, alongside the task each model was used for. The details of all the individual models as well as their results, can be found in the corresponding chapters of the thesis.

AI/ML Method	Task	Model	Backbone(s)	Results Chapter
Image Classification	Classify images according to sedimentary structures and fossil types	Custom	ResNet18, ResNet50, ResNet101, VGG16, VGG19	5
Object Detection	Identify and locate sedimentary structures and fossil types	YOLOv6s	CSPDarknet53	6
Instance Segmentation	Identify, locate, and segment sedimentary structures, fossil, and lithology types	YOLACT	Darknet53, cDarknet53, ResNet101	7
Natural Language Processing (NLP)	Extract keywords (sedimentary structures, fossil, and lithology types) from a bulk of geological publications	Custom	None	8
Deductive and Inductive Reasoning	Combine all the individual geological features into meaningful sequences to interpret the depositional environment	Custom	Custom	8

Table 3-1: Overview table, listing all models with their corresponding backbones used in this thesis, alongside the task each model was used for.

The use and combination of these five different ML/AI methods described in this chapter form a complete AI system performing outcrop interpretation in a matter of seconds, capturing uncertainty tied to geological interpretations, and complementing decision-making.

CHAPTER 4 - DATASETS DESCRIPTION

4.1 Introduction

This chapter provides a detailed description of my dataset-building workflow for Computer Vision model applications in geology. It highlights the importance of each workflow component during the dataset-building process, describes all the outcrop datasets used in this thesis, and explains the choices for each dataset and the geology depicted within each one.

All the Computer Vision models presented in this work have pre-trained weights with one or more of the benchmark datasets mentioned in Chapter 2, particularly the COCO, MS COCO, and ImageNet datasets. None of the herein benchmark datasets contain outcrop images or images of sedimentary structures and fossils, and thus, they are not suitable for outcrop interpretation, which is the primary goal of this project. For that reason, I had to build several datasets to train my CV models used in this thesis.

This chapter describes eleven datasets created to train the Supervised Computer Vision methods reviewed in Chapter 3, and their application is evaluated in Chapters 5, 6, and 7. All the datasets that were used to train and test the models presented in this thesis were manually assembled and annotated, as described at the end of this chapter).

4.2 Dataset-building workflow for Computer Vision applications in Geology

The generation of all datasets in this thesis followed a series of steps, which will be described in detail in the following sections. All the generated datasets, once compiled, are evaluated by a geologist and cross-checked with the literature to ensure their correctness. The workflow summarizing the dataset-building steps is shown in Figure 4-1.

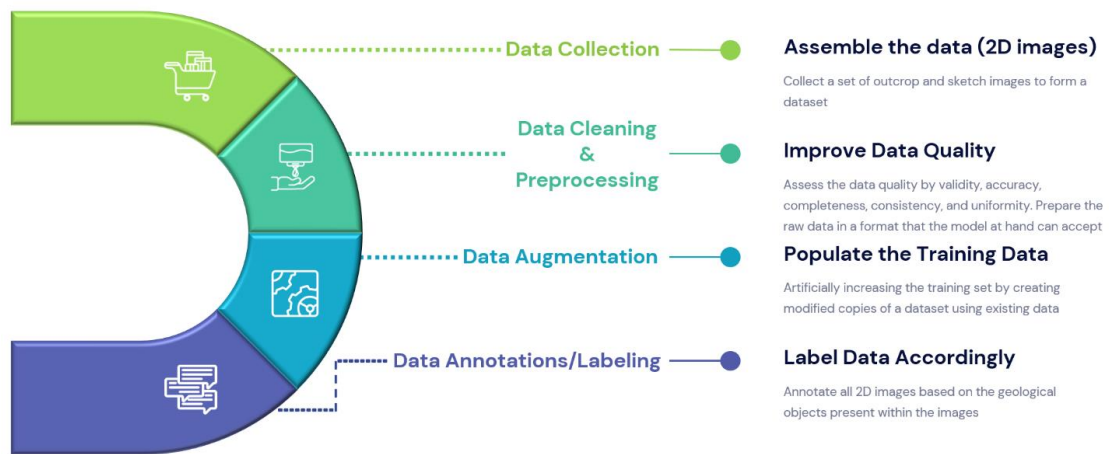


Figure 4-1: Geologic Dataset Building Workflow.

The first step is to collect outcrop images and make geological sketches depicting the desired geology. Additional images of core data and fossils may be incorporated into the datasets. The geological sketches carry some interpretational uncertainty, but this uncertainty is minimized early as they are cross-referenced with the published geological literature and expert geologists before their use from the ML models.

The second step, explained in section 4.4, is to clean the assembled dataset, meaning that the data needs to be assessed for their quality, as the better the quality of the data (section 4.4.1.3), the better the performance of the models. The data pre-processing follows to manipulate the data into a suitable format for the computer vision models to use, whether that is in terms of file type conversion, image size, colours, orientation, or other manipulation techniques.

Next, the augmentation step increases the number of samples in the training set while introducing greater variety in what the model sees and derives information from.

Lastly, as the dataset is meant for object detection and segmentation tasks, all the images in the dataset are annotated individually to help the algorithms learn the desired objects within the images. This is a good example of how human learning relates to supervised machine learning, where a human provides information to the algorithm to make it learn by looking at several examples of the desired features.

4.3 Data Collection

All my datasets contain only 2-dimensional (2D) images with dimensions of 512 x 512 pixels, one of the standard image sizes used for Computer Vision tasks. Using 2D images is a practical approach as there is a larger pool of data compared to 3D data to select from. Although 3D data (LiDAR) is very information-rich, 2D data is preferred as it is easier to collect, less expensive, and a wider variety and amount of 2D outcrop images are available.

All the Computer Vision models used in this thesis accept the input (training) images in a 1:1 ratio, 512 x 512 pixels. Other image sizes can also be utilised, but 512 x 512 was chosen due to computational resource limitations. Outcrop images are often of landscape orientation, leading to different aspect ratios, such as 3:2, 16:9, etc. In such cases, the landscape (rectangular) images are cropped appropriately into square-shaped (1:1 ratio) images to make them suitable for the ML models.

Different image sizes do not affect the scale of the geological features because, for each image in the training dataset, the scale of the geology is already known, either by its source, geologist, the literature, or a reference object. Thus, the scale of each image is incorporated into the models with the help of annotations and labeling, as will be explained later in this chapter.

All the image datasets created and described in this chapter focus on sedimentology. Identifying the key features and structures in geological outcrops can help us infer the likely processes and events during sedimentary deposition.

The criteria for selecting the images in the datasets were based on the Computer Vision task we were trying to tackle. The selected (training and validation) images were limited to four major depositional environments (section 4.9), depicting sedimentological features from the laminae and bedding scales. The majority of images in all the datasets included outcrop images displaying a variety of sedimentary structures, lithology types, and fossils.

The images used for the datasets include outcrop images and sketches of geological features. The outcrop images were collected from personal field trips, virtual outcrops (V3Geo) snapshots, and publications. All the sketches were generated based on established geological sketches from the geological literature. The generated sketches do

not resemble a particular outcrop but rather the patterns of the desired geological features that can be found in multiple outcrops or parts of a single outcrop.

4.4 Data Cleaning and Preprocessing

Dataset building has been an extensive and tedious task during this Ph.D. project as several custom geological datasets have been compiled and used, depending on the challenge trying to solve.

The geological data was selected based on the notion that quality data beats even the most sophisticated algorithms. The idea that quality data is more important than sophisticated algorithms is widely accepted in data science and is supported by various sources. Foster Provost and Tom Fawcett, the authors of a book called "Data Science for Business," state that "Even the best machine learning algorithms cannot overcome bad data. In fact, bad data is likely to degrade the performance of any predictive model" (Provost & Fawcett, 2013). An additional source to support the statement about the importance of data quality is a blog post by data scientist Cassie Kozyrkov on the Google Cloud Platform that quotes: "In practice, the quality of the data is often more important than the sophistication of the algorithm. A great algorithm applied to low-quality data will typically produce worse results than a simpler algorithm applied to high-quality data." (Kozyrkov, 2018).

Thus, without the use of good quality (section 4.4.1.3) and clean data (section 4.4.1), the models will produce misleading and incorrect results that can negatively impact decision-making processes. The results in chapters 5, 6, and 7 show that the amount of data is of less importance compared to the quality of data. According to the computer science community, to train state-of-the-art Computer Vision models, it is required to have datasets with thousands of images per class. Such a number of geological images would be impossible to gather; therefore, this thesis, particularly in Chapter 5, demonstrates how the addition of geological sketches blended with outcrop images in the training can help us partially overcome the data availability limitations.

4.4.1 Data Cleaning

Data cleaning refers to the process of preparing the available data for analysis by removing irrelevant or incorrect information. Certain data can have a negative impact on a model's performance as it might reinforce a negative/wrong/incorrect notion. Besides

removing incorrect or unnecessary data, data cleaning can also reduce the number of duplicates in the dataset and fix incorrect information within the training, validation, and test sets.

4.4.1.1 The Importance of Data Cleaning

Data cleaning is a necessary step before performing any further analysis on the dataset in question. In larger projects, multiple smaller datasets are combined to create a larger dataset. This leads to redundancies and duplication in the data, which may cause the model to learn inaccurate representations of the data and may impair the model's decision-making ability. At this stage, any duplicate images present in my datasets were discarded. If the data cleaning stage is skipped, the models will be trained on raw datasets that contain noise as information and could malfunction when new, clean data is supplied to them. Thereby, data cleaning is a vital part of any machine learning model pipeline (Géron, 2019) and should always be incorporated into the ML workflows.

4.4.1.2 Data Cleaning vs. Data Transformation

Data cleaning is often confused with data transformation, although these are two different things.

Data transformation deals with the conversion or transformation of raw data into a format that makes it easier for the model to process. During the data processing step, the incoming raw data go through the data cleaning step before any data transformation takes place. Typically, the initial data transformation involves normalization and standardization. Normalization changes the values of numeric columns in the dataset to ensure there is a common scale across the dataset before it is processed without losing information or altering/distorting the differences in the ranges of values. The normalization of an image consists in dividing each of its pixel values by the maximum value that a pixel can take. In this case, for 8-bit images, each pixel is divided by 255 (Nikhil, 2017). This step is usually performed automatically by the ML models when the image data is loaded into the model before training starts.

Standardization is applied to a dataset, whether these numeric values refer to pixel values if the dataset consists of images, or just numerical values. In other words, normalization

transforms the features in the dataset to be on the same scale, improving the performance and stability of the model.

4.4.1.3 Quality of Data

In this thesis, the term ‘quality of data’ is mentioned; therefore, it is important to establish what it means. All data types have distinct characteristics that can be used to determine their quality. The quality of data depends on five common characteristics; Validity, Accuracy, Completeness, Consistency, and Integrity (Ridzuan & Zainon, 2019). For this Ph.D. project, the choice of all image data heavily depended on the above five characteristics and the amount of available data.

To ensure data validity, modern techniques and constraints can be used to control how data is stored, and various constraints can be applied to forms and documents. These constraints include data type, range, unique, and cross-field validation, which help prevent inconsistencies resulting from incorrect data types or duplicates and ensure that multiple fields in the document correspond to each other.

Accuracy is a common term used to evaluate a model's performance and the feasibility and correctness of data used for a task. In the context of geological outcrop data, the datasets consist of 2D images or sketched interpretations showing lithology types, fossils, and sedimentary structures. The correctness of the data can be confirmed by cross-referencing with literature and expert geologists. At the same time, the feasibility can be validated by checking if certain geological features can coexist next to each other based on domain knowledge. Data displaying a mismatch can be easily rejected.

Completeness refers to the extent to which data used in the datasets is comprehensive. Often, missing fields and values in the data pose a significant challenge as they could significantly reduce the dataset’s size and variability. Incomplete data is inevitable, but it can be mitigated by carefully selecting the appropriate data for the task by applying the proper constraints.

Consistency refers to how the data responds to cross-checks with other fields in the dataset. In this case, ensuring the geologic features depicted in each image are consistently identified and labeled accordingly when cross-checked with the literature or geologists.

Visual data integrity assures that all labeled object classes can be tracked down and linked to a single source of ground truth.

Image datasets often display several duplicate images coming from geologists taking similar images of an outcrop. While for test or numerical data is easy to remove duplicates and condition the dataset by utilising algorithms or machine learning libraries, duplicate image data have to be manually evaluated and removed by the geologist. Although there are machine learning models that can identify image similarity, in geology, there are often images that look alike or depict the same information but are captured under different weather conditions and light. For Computer Vision models, such duplicates enhance the model's learning by acting as a data augmentation technique. On some occasions, there is noise present in the images. Noise refers to either the fuzziness of the pixels or irrelevant information concerning the objects of interest. For instance, vegetation can be counted as noise when considering an outcrop because it does not add any valuable information to the outcrop's interpretation.

4.4.2 Data Preprocessing

Data preprocessing is a standard initial procedure in deep learning frameworks that transforms raw data into a format that deep learning networks can handle. Preparing the data is a crucial process in the field of Machine Learning, as the precision of the data and the valuable insights that can be gained from it significantly impact the capability of ML models to learn. Hence, it is important that data are cleaned and processed before being fed into the model.

4.4.2.1 Data split: Train vs. Validation vs. Test set

For each dataset used in a Machine Learning or Computer Vision task, before being used to train a model, the data should be broken down into three sub-sets, the training, validation, and test sets. The training set refers to the part of the data used to train the model, the validation set used to validate the model's learning capability and further enhance the model's learning, and finally, the test set consisting of images unknown to the model, on which set the model is applied to make certain predictions (classification, object detection, and segmentation) (Goodfellow, et al., 2016).

The Training Set serves the purpose of instructing and enabling the model to grasp the concealed characteristics and patterns within the data. Throughout each epoch, the neural network architecture is repeatedly fed with the identical training data, facilitating the model's continuous acquisition of data features. It is essential for the training set to encompass a diverse range of inputs, ensuring comprehensive training across various scenarios, enabling the model to accurately predict future unseen data samples (Baheti, 2021).

The Validation Set is a distinct collection of data, separate from the training set, which serves the purpose of assessing and validating the performance of our model during the training process. This validation step provides valuable insights that assist in fine-tuning the model's hyperparameters and configurations. It can be compared to a critic, offering feedback on whether the training is progressing in the desired direction. While the model is being trained on the training set, simultaneous evaluation of the model occurs on the validation set after each epoch. The primary motivation behind splitting the dataset into a validation set is to prevent overfitting of the model. Overfitting refers to a situation where the model becomes exceptionally skilled at classifying samples within the training set but struggles to generalize and accurately classify unseen data (Baheti, 2021).

The Test Set is a separate collection of data that is utilised to assess the performance of the model once the training phase is complete. Its purpose is to provide an impartial evaluation of the final model's performance in terms of metrics such as accuracy and precision. This subset is primarily employed to demonstrate the model's effectiveness and to facilitate comparative predictions and the ultimate selection of the preferred model (Baheti, 2021).

The split of the data can be determined by taking into account a few main points. First, it depends on the number of samples/images included in the dataset. Then we need to consider how many images per class are available in the dataset, and the split should be done in such a way that there are samples/images for every class, both in the training and validation sets.

Different splits of the data assist in evaluating the model's performance. To optimize the model's performance, a machine learning model with a large number of tunable hyperparameters requires a larger validation set. If the number of hyperparameters is small, a small validation set is sufficient to validate the model.

The optimum data split percentage between the training, validation, and test sets is defined by the user's and model's needs. Choosing the split percentage according to the needs of the dataset/model and the task at hand is important as it will help the model perform at its best. Although there is no optimal split percentage, the widely used standard data splits that might be a good starting point for every project are shown in Figure 4-2. For this thesis and most datasets, the conventional 70-20-10 percentage was used, while in a few cases, the 80-10-10 split was used. Regarding the implementation of this split principle between the images and sketches, all the details are shown in section 4.7.

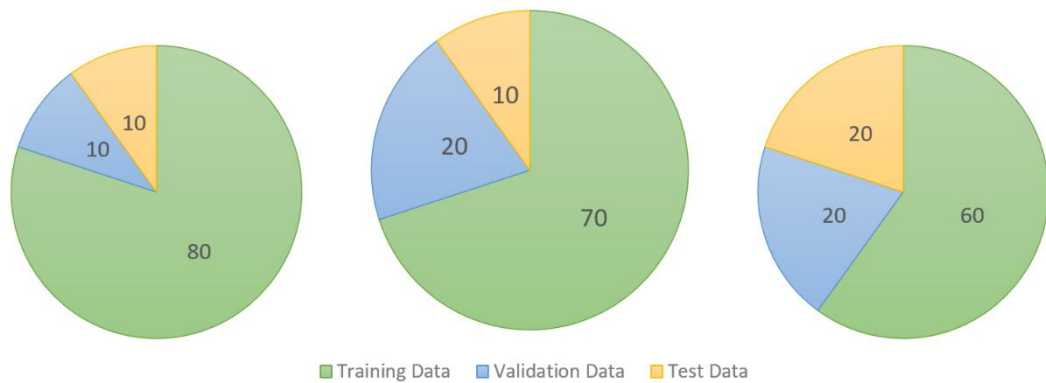


Figure 4-2: Standard data set split.

4.4.2.2 Three common pitfalls in the training data split

The training data split has three common pitfalls that need to be considered:

A. The first and most important pitfall is the quality of the data, affecting the model's performance and its ability to improve that performance based on its training steps and available data. The better the data quality the model is trained with, the better its performance. Even slight variations or errors in the training set might lead to significant errors in the model's performance. The way data is split can also have a significant impact on the model's performance. If the model is trained on too little data, it may overfit the training set and fail to generalize well to new data. Conversely, if the model is trained on too much data, it may underfit the training set and not learn the underlying patterns in the data. The test set is used to evaluate the model's performance on new, unseen data. If the

test set is not representative of the real-world data that the model will encounter, the model's performance on the test set may not reflect its performance in the real world.

B. The second and frequently occurring pitfall is the overfitting of the data. Overfitting occurs when the model memorizes specific patterns in the training data so well that it fails to classify and understand unseen data. Any noise in the training data is considered as new features learned by the model, which might result in the model's outperforming the training set while performing poorly in the validation and test sets. In other words, although it learns the features in the training dataset very well, it fails to generalize and performs equally well on unknown data. This is one of the most significant challenges in geology as the training dataset must be highly variable, including multiple examples of each class from several outcrops. The geological features are highly variable in terms of texture and shape and might look very different from outcrop to outcrop, as shown in Figure 4-3. Due to the data variability and complexity of the geology, data sparseness, and availability have posed a major challenge for this thesis. To mitigate these issues, emphasis was given to the data quality, meaning carefully selecting the images in each dataset, the use of pre-trained models, utilising transfer learning to account for the limited data, and finally, careful model tuning to improve and maximize the learning from the available training data.



Figure 4-3: Four examples of cross-bedding from different outcrops.

C. The last pitfall is the overemphasis on the metrics of the validation and test sets. The validation metric is the one forging the model's training path. After every epoch, the ML model is automatically evaluated on the validation set based on the validation metrics (accuracy and losses). Model's accuracy and losses during training act as a guide to modify the model's hyperparameters accordingly to improve the model.

In applying the Computer Vision methods described in Chapter 3 to the geological problem addressed in this thesis, relying solely on metrics to evaluate model performance is insufficient. Given the complexity of geology, it is essential to incorporate human geologist evaluation and feedback to ensure the development of robust models and obtain reliable results.

4.5 Data Augmentation

Data augmentation deals with expanding the size of the data by creating new data points from the existing ones, either through minor modifications or by using machine learning algorithms to generate new points in the latent space of the original data (Shorten & Khoshgoftaar, 2019).

Augmented data is derived from real images with slight geometric changes (such as flipping, translating, rotating, or adding noise) or colour modifications to diversify the training set.

Synthetic data is artificially generated data without using real-world images and is often produced using Generative Adversarial Networks. Due to data privacy concerns, synthetic data generation is becoming common for building datasets.

Augmented data is generally preferred over synthetic due to its similarity to real-world data, leading to more realistic datasets.

4.5.1 The Importance of Data Augmentation

Over the last few years, data augmentation techniques have gained a lot of traction as they are widely used in every state-of-the-art application of deep learning, such as Image Classification and Recognition, Object Detection, and Image Segmentation, among others. By providing more diverse datasets, especially new and diverse examples of the training data, the augmented data improves the performance and outcomes of the

deep/machine learning models. Furthermore, it often reduces the operating costs associated with data collection. Collecting and annotating domain-specific data is time-consuming and costly. Using augmentation techniques can provide a sustainable alternative for creating datasets by transforming and enhancing already existing datasets.

4.5.2 Limitations of Data Augmentation

Of course, as with every method, data augmentation has its own limitations. The first limitation is the cost of quality assurance of the augmented dataset because advanced research and development are required to build such synthetic datasets utilising advanced applications. Then, the verification of the augmented images needs a domain expert to validate the generated data, finding then optimal augmentation techniques that are applicable to the data and problem at hand. Lastly, the bias inherent to the original data also remains true in the augmented data.

When building a deep learning model to perform a classification task, for the model to differentiate between the tested images, it requires a lot of training data for all classes represented in the images. A Convolutional Neural Network (CNN) is well-suited for this task due to its ability to classify objects accurately regardless of orientation, size, illumination, or viewpoint. This is the basis of data augmentation, which is used to adapt the model to real-world scenarios where photos may vary in orientation, scale, brightness, and location.

Deep learning models like CNNs have many parameters that help learn complex features by analyzing numerous examples. The size and type of the input dataset significantly impact the performance of these models. Advanced computer vision models like Inception-V3, VGG, and RESNET have tens of millions of parameters, while NLP models like BERT have even more (340 million parameters). However, gathering a large amount of data is crucial for building a successful deep-learning model.

When abundant data is not available, data augmentation provides a solution that artificially increases the data quantity by using techniques to manipulate existing data. This addresses the challenge of limited data and enables the building of a more robust deep-learning model.

4.5.3 Data Augmentation Techniques in Computer Vision

Several data augmentation techniques transform the position or colours of an image. This distinction separates augmentation methods into two broader categories, Position Augmentation (Table 4-1) and Colour Augmentation (Table 4-2). The following table lists some of the most popular techniques.

Position Augmentation	
Technique Name	Description
Center Crop	Crops the given image at the center. Size is the parameter given by the user
Random Crop	Crop the given image at a random location
Random Vertical Flip	Vertically flips the given image randomly with a given probability
Random Horizontal flip	Horizontally flip the given image randomly with a given probability
Random Rotation	Rotate the image by some angle
Resize	Resize the size of the input image to a given size.
Random Affine	Random affine transformation of the image, keeping center invariant

Table 4-1: An Overview of Position Augmentation Techniques.

Color Augmentation	
Technique Name	Description
Brightness	One way to augment is to change the brightness of the image. The resultant image becomes darker or lighter compared to the original one
Contrast	Contrast is the degree of separation between an image's darkest and brightest areas. The contrast of the image can also be changed.
Saturation	Saturation is the separation between the colors of an image

Table 4-2: An overview of Colour Augmentation techniques.

While augmentation is broadly used for computer vision and deep learning models and projects, for the domain of geology, augmentation is not always helpful. Geologic image data are often very information-rich and contain a high level of complexity. In addition, sedimentary structures and geologic features are highly variable in appearance from outcrop to outcrop. As already shown in Figure 4-3: Four examples of cross-bedding from different outcrops., cross-bedding looks very different depending on the outcrop and the environment of deposition that the outcrop belongs to. Although it is the same structure, as will be demonstrated in Chapters 6 and 7, it is often misclassified. The reason for this is that its shape is not specific like common objects, like a cat or a soccer ball.

To further explain this statement, a few augmentation techniques were applied to an image of a cat, a soccer ball, and an outcrop with a prominent cross-bedding to demonstrate why not all augmentation techniques are applicable to geology.



Figure 4-4: Nine augmented images of a cat, with applied augmentation techniques such as rotation, flip, random zoom, and contrast/brightness adjustments.



Figure 4-5: Nine augmented images of a soccer ball, with applied augmentation techniques such as rotation, flip, random zoom, and contrast/brightness adjustments.

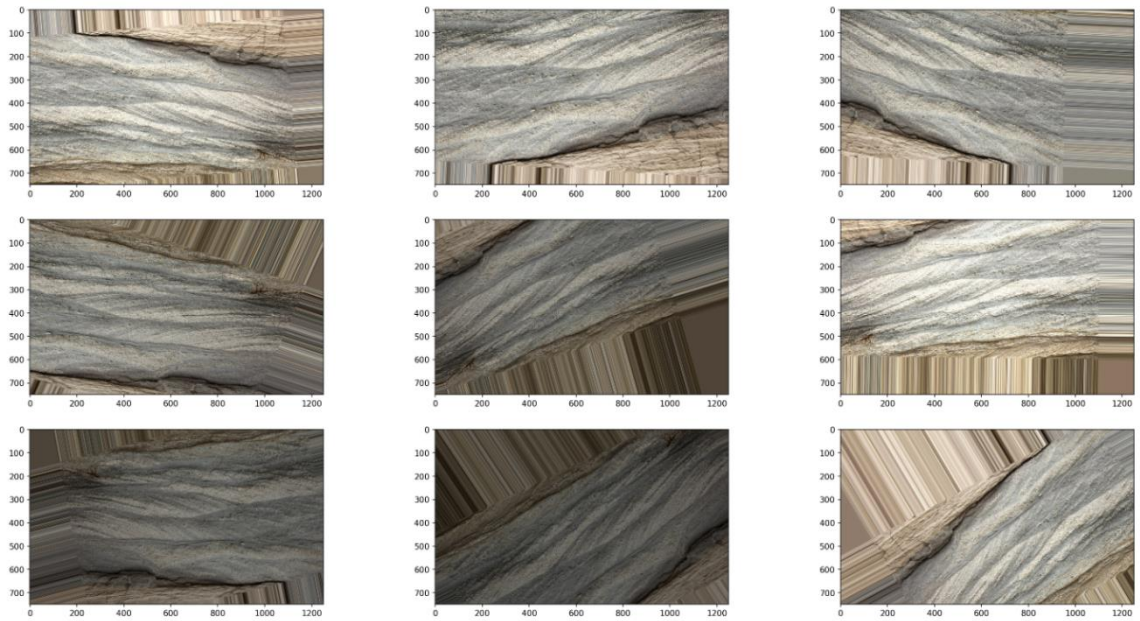


Figure 4-6: Nine augmented images of an outcrop image with a prominent cross-bedding, with applied augmentation techniques such as rotation, flip, random zoom, and contrast/brightness adjustments.

In all three figures (Figure 4-4, Figure 4-5, and Figure 4-6), both position and colour augmentation techniques were applied for two common objects and a sedimentary structure. Figures 4-4 and 4-5 show that regardless of the applied augmentation technique, one can still say what the object of interest is. In Figure 4-4, we can always say that the image's main object is the cat; the same holds true for the soccer ball in Figure 4-5. These two objects or classes can always be identified correctly as they are common objects. Even when some information is lost during the augmentation (e.g., a high level of zoom or cropping of the image), humans and ML models can still identify the objects accurately. This is not valid for the nine images in Figure 4-6 and Figure 4-7. Certain augmentation techniques may alter or result in images that do not make sense geologically, or the application of random cropping may result in the loss of information that may be important for describing the geologic feature.

Identifying sedimentary structures is much more complicated than identifying a cat or a ball; therefore, the augmentation of outcrop images should be handled carefully, applying only methods that do not alter the geology or violate its rules. Such methods are horizontal or vertical flip, a rotation from plus or minus 0-10 degrees, and colour adjustments. However, it always depends on the images in the dataset and the geology they include.

The augmentation techniques applied to outcrop images should be such that they do not alter or distort the geology. Figure 4-6 shows that some of the augmented examples of the cross-bedding demonstrate some extreme angles of dipping, which may be unrealistic, not necessarily wrong; however, it is less likely to occur in nature. Such examples that might be ambiguous or potentially confusing to the ML models were discarded during the manual evaluation of the augmentation results. In Figure 4-7, the images with the red bounding boxes display some geologically wrong examples, as cross-bedding cannot exist at such an angle.

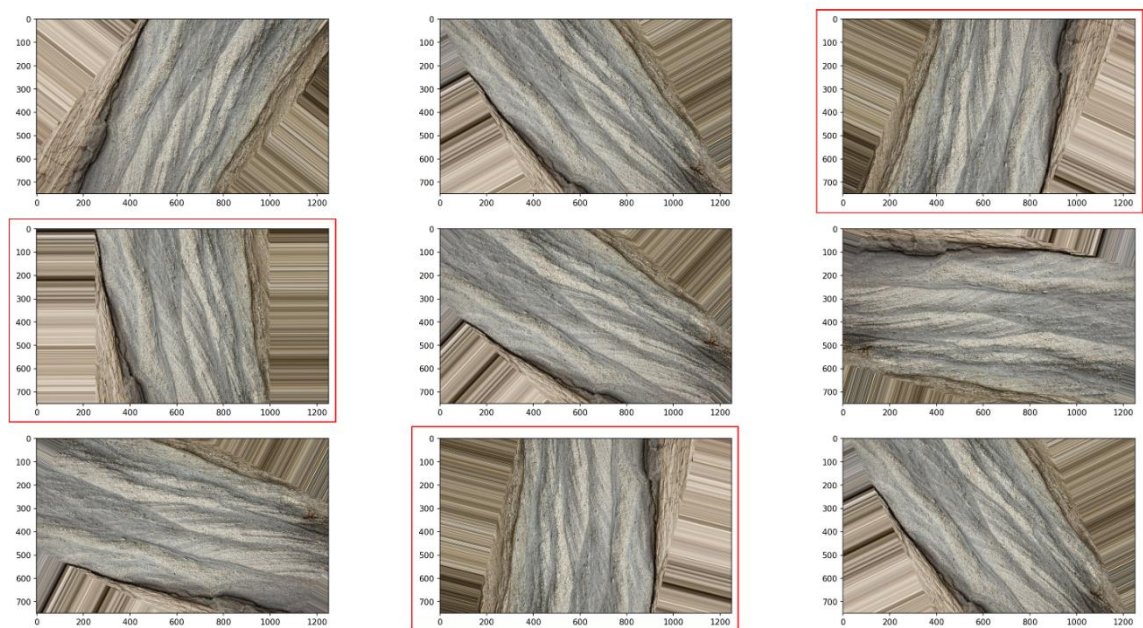


Figure 4-7: Nine augmented images of an outcrop image with a prominent cross-bedding, with applied augmentation techniques such as rotation, flip, random zoom, and contrast/brightness adjustments. The images with the red bounding boxes display some geologically wrong examples, as cross-bedding cannot exist at such an angle.

The impact of data augmentation on geological successions can be positive or negative, depending on the quality of the augmented data and the specific problem being studied. If the augmented data is of high quality and represents the underlying geological features accurately without altering their geological meaning, it can improve the accuracy of the geological succession analysis. On the other hand, if the augmented data is of poor quality or introduces artificial features that do not exist in the real data, it can negatively impact the geological succession analysis.

The augmentation workflow for geologic datasets used in Computer Vision tasks is organized as shown in Figure 4-8. The workflow comprises five steps:

1. The Input data (outcrop images) is imported into the data augmentation pipeline.
2. The data augmentation pipeline is defined by sequential steps of different augmentations, including rotation (± 10 degrees), horizontal or vertical flip, contrast, brightness, and colour adjustments. These five augmentation techniques are defined as viable augmentation techniques, suitable for augmenting geological images without altering the meaning of the depicted geological features and their arrangement within the images.
3. The image goes through the pipeline and is processed for each step with a probability.
4. After the image is processed, the human geologist verifies the augmented results and provides feedback back to the system.
5. After the human verification, all the augmented data is ready to use by the machine learning model's training process.

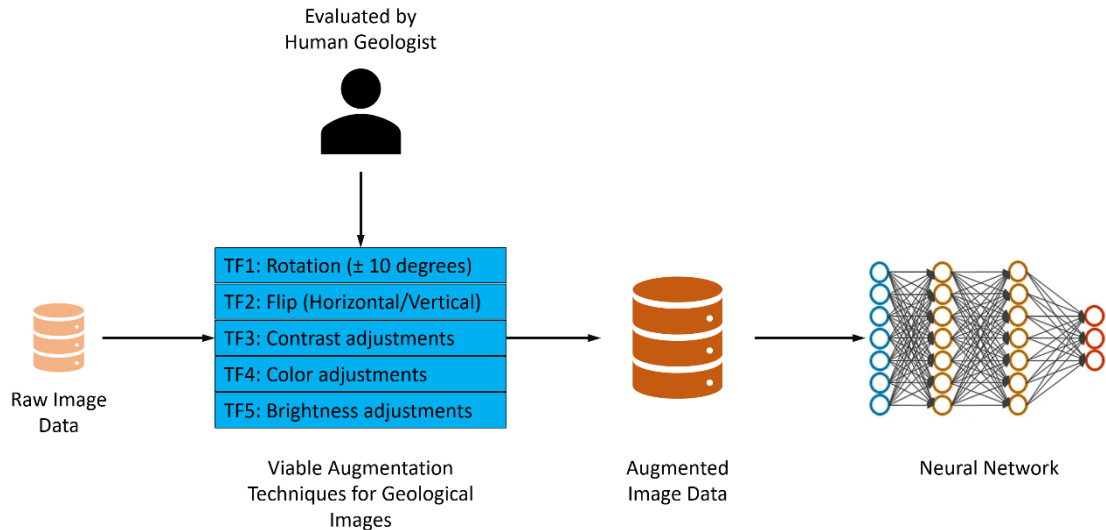


Figure 4-8: The augmentation workflow for geologic datasets used in computer vision tasks.

4.6 Data Annotation

Data annotation refers to the task of labeling or categorizing images, videos, and text through the use of text, annotation tools, or a combination of both. These labels represent the different classes of objects the data belongs to. The purpose of annotating images is to mark the features that the machine learning model is intended to learn to identify these classes of objects (training set) when found in unseen and unlabeled data (test set), allowing the model to be trained using Supervised Learning. The labeling of an image involves adding metadata to a dataset, which is also referred to as tagging, transcribing, or processing. Once the model is deployed, it should be capable of identifying these features in previously unseen images and making decisions or taking action accordingly.

Typically, Image Annotation is used to identify objects and their boundaries, segment images, and comprehend the overall meaning of an image (section 4.6). A significant amount of data is required to train, validate, and test the machine learning model to achieve the desired results.

Image Annotation involves labeling an image with a descriptive phrase called tagging. For complex images, Image Annotation involves detecting, counting, or tracking multiple objects or regions within an image. The complexity of the Image Annotation will depend on the difficulty of the project being undertaken. For instance, in the context of defect inspection, the machine learning algorithm is fed with images displaying features such as rust or cracks. The corresponding annotation comprises polygonal shapes for the localization of cracks or corrosion and tags for identification purposes.

For geology, since we want our object detection and segmentation models to estimate the lithology types and sedimentary structures present in an outcrop image, the annotations are polygons (section 4.6.5), capturing the shape and location of geological features.

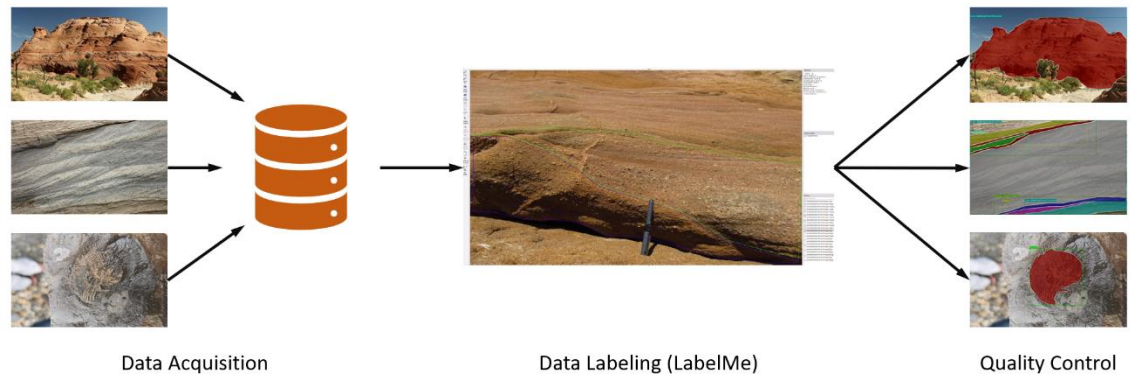


Figure 4-9: Data Annotation Workflow.

The Data Annotation depicted in Figure 4-9 proceeds in the following sequence:

- a. Data Acquisition: The raw data that will be utilised for model training is collected, cleaned, and processed to form a database that can be directly fed into the model.
- b. Data labeling: Such methods are applied to tag the data and provide it with meaningful context that the machine can use as a reference.
- c. Quality Control: The quality of data annotations is commonly evaluated by a geologist based on the level of precision of the tags for a given data point, as well as the accuracy of the coordinate points for bounding box and key point annotations.

Numerous image annotation tools are available today, some with specific labeling capabilities and others with a broad range of features for various use cases. The choice between a specialized tool and one with a more extensive range of features depends on current and future image annotation requirements. However, it is essential to note that no single tool can perform all functions. Therefore, the chosen tool should be able to grow and adapt as the project develops.

Computer Vision annotation tools convert raw image data into labeled images for training the Machine Learning models. These tools, utilising either bounding boxes or polygon annotations, aim to facilitate high-level image detection, segmentation, and ultimately converting raw image data into labeled images (Geiß, et al., 2023).

Computer Vision annotation tools, such as those employing polygon annotation, enable computer vision models to identify and describe objects and their shapes. A typical

application is self-driving cars detecting and avoiding pedestrians, traffic cones, and other vehicles on the road.

In this project, the annotation tool that was used is called LabelMe, created in 2008 by the MIT Computer Science and Artificial Intelligence Laboratory (CSAIL) (Russell, et al., 2008). The user defines a bounding box or a polygon around each object of interest and simultaneously assigns a class for each instance. Each annotation is saved in a separate JSON file corresponding to the annotated image. Then, each pair of image and annotation files are fed into the computer vision model to begin training.

As already described in section 4.4.2, training data refers to data that has already been collected and preprocessed, which the machine learning model uses to learn the patterns in the data, enabling it to extract this information from the outcrop images. If the training data is annotated, the corresponding labels are referred to as ground truth, meaning that information is known beforehand to be true.

Supervised Learning is the most common type of machine learning algorithm that trains on data with corresponding labeled output. Tasks like object detection and segmentation fall under this category. The process involves providing the model with labeled data to learn patterns and testing it on unlabeled data. The testing stage uses labeled data with hidden outputs to evaluate the model's accuracy. Hence, labeled data is necessary for training machine learning models through Supervised Learning.

4.6.1 Annotation Types in Computer Vision

Most Computer Vision models require annotated/labeled data, and these annotations can vary depending on the visual task one has to solve. The annotations types used in this thesis are Image Classification, Object Detection, and Instance Segmentation.

For Image Classification, data labeling entails adding one tag per image and separating the images into different folders according to the class they characterize. The total number of unique tags in the dataset represents the total number of classes the model can classify. There are two types of classification, binary class classification, in which only two tags are to be classified, and multiclass classification containing multiple tags to classify (Mishra, 2021).

For an Object Detection model, the data annotation is a different process from that of Image Classification as each annotated object is captured by bounding boxes, which are small rectangular segments surrounding the objects. For every bounding box, there is also a label assigned to it, representing once more the class the object belongs to.

Instance Segmentation creates masks by assigning polygons of multiple points to capture the shape of each feature, in addition to bounding boxes. A label is assigned to every bounding box, representing the class the object belongs to. The use of a mask assigns the exact pixels in the image that belong to each class.

Both for Object Detection and Segmentation tasks, the coordinates of each bounding box and the respective labels are all stored in a separate JSON file in a dictionary format, with the image number/image ID being the key to the dictionary.

Each type of image annotation is distinct in how it reveals particular features or areas within the image. Determining which type of image annotation is more appropriate to use should be based on the data and algorithms under consideration. Figure 4-10 shows what geological information each annotation and computer vision method can provide.

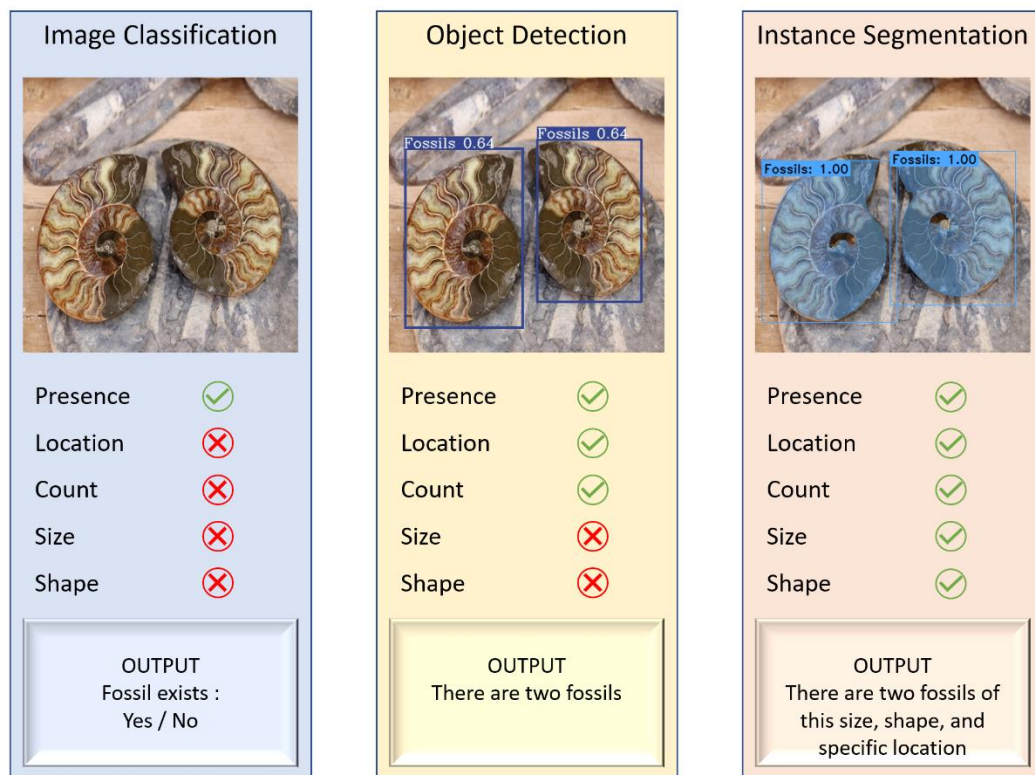


Figure 4-10: This image is an overview of the data types, annotation types, annotation techniques, and workforce types used in image annotation for computer vision.

4.6.2 Data Labeling Approaches

Various labeling strategies are available for annotating data, with some executed by humans and others with the assistance of artificial intelligence.

The most common strategies include in-house data labeling, crowdsourcing, outsourcing, and machine-based annotation. While in-house and crowdsourcing are frequently used, the nomenclature can encompass innovative methods that utilise artificial intelligence and active learning for annotation. In-house labeling can achieve optimal quality, but it is time-consuming. Crowdsourcing involves a large number of freelancers who annotate simple data. Outsourcing serves as an intermediary between crowdsourcing and in-house labeling. Machine-based annotation uses tools and automation to increase annotation speed while maintaining high quality. Recent advancements in automation technology have significantly reduced the workload on human labelers.

In this Ph.D. project, only In-House Data Labeling was used to annotate the images because we needed to ensure the high quality of annotations. The image data is too specific for the machine-based and crowdsourcing annotating techniques, requiring geologists' expertise to annotate correctly.

4.6.3 Boundary Recognition

Image annotations can be utilised to train a machine learning model to recognize various elements in an image, including lines and boundaries of objects. These boundaries can encompass the edges of individual objects, topographical features, or human-made boundaries visible in the image. A machine learning model can learn to recognize patterns in unlabeled images by annotating images correctly.

Boundary recognition is critical to machine learning as it identifies lines and splines, such as traffic lanes, land boundaries, or sidewalks. This recognition is especially crucial for autonomous vehicles to ensure their safe operation. For instance, drones must follow a specific path while avoiding potential obstacles like power lines. Such autonomous systems can operate safely and efficiently by training machine learning models to identify boundaries.

Boundary recognition also plays a role in identifying the foreground from the background in an image and defining exclusion zones. For instance, in a grocery store image, if the

focus is on stocked shelves and not shopping lanes, the lanes can be excluded from the data that algorithms consider. In medical images, annotators can label the boundaries of cells within an image to detect abnormalities.

Boundary recognition is essential in geology because it delineates the exact shape of each geological feature, enabling the geologist and the ML model to segment each feature easily.

4.6.3.1 Image Annotation Techniques

In image annotation, several techniques are commonly employed, and their availability depends on the features of the data annotation tool.

One such technique is using a bounding box, which is utilised for symmetrical objects, like vehicles, pedestrians, or road signs, where shape and occlusion are not a significant concern. Bounding boxes may be 2-D or 3-D and are sometimes called cuboids in the latter case.

The marking of particular areas of an image using masking, accomplished at the pixel level, allows for focusing on specific image regions. Polygon annotation is preferred for marking highly irregular geological objects. Therefore, for this Ph.D., annotators have labeled the boundaries of lithology layers and sedimentary structures with bounding boxes and polygon annotations.

4.6.4 Bounding Boxes Annotations

Bounding box annotations are used for labeling a dataset for an Object Detection task. Specifically, these annotations were used to label sedimentary structures and fossils in outcrop images. There are a few essential points to consider when working with bounding boxes. To achieve precise feature delineation, it is imperative to guarantee an exact fit between the boundaries of bounding boxes and the outermost pixels of the object under consideration. If there are gaps between the pixels of the target object and the corresponding bounding box, several Intersection over Union (IoU) discrepancies may occur (Lin, et al., 2017).

Consistency in box size across training data is crucial because significant variations may adversely impact model performance. Large objects are particularly challenging as their

representation tends to be less affected by changes in Intersection over Union (IoU) when they occupy a greater number of pixels, as opposed to medium or small objects that suffer a more significant impact (Zhao, et al., 2018). As such, models trained on predominantly large objects may demonstrate inferior performance when presented with smaller instances of the same object. If the project at hand has a high number of large objects, it is advised to label all the objects by using polygons rather than bounding boxes, therefore making the task more suitable for Instance Segmentation.

Bounding box detectors that rely on Intersection over Union (IoU) calculations during training must avoid overlapping instances. Overlapping may be expected in cluttered scenes with objects in close proximity, such as items on store shelves or pallets. In the outcrop context, lithology and sedimentary structures will overlap as sedimentary structures are formed on the lithology layers. Therefore, labeling them on top of each other will result in overlapping masks and bounding boxes.

Overlapping bounding boxes on such objects may negatively impact model performance, making associating boxes with the corresponding items challenging. The problem persists as long as the overlap persists.

When defining the object size to label, it is crucial to consider the input size of the model and the degree of network down-sampling. Tiny objects may lose their discriminative information due to the image down-sampling process in the network architecture. Therefore, it is crucial to strike a balance between object size and network resolution during model training (Zhang, et al., 2017).

Objects positioned diagonally, particularly slender ones like a pen or road marker, may occupy a significantly smaller area in a bounding box compared to the surrounding background. While humans can easily discern the object of interest, enclosing it in a bounding box could mislead the model into equally treating each pixel within the box. Consequently, the model may assign a high score to the object by incorrectly identifying the background surrounding it as the object itself.

As with overlapping objects, diagonal objects are best labeled using polygons and instance segmentation to avoid this problem. This explains why object detection is not the best approach for identifying geological objects, as there is a risk of capturing a high amount of unwanted information with rectangular bounding boxes. Nevertheless, with

sufficient training data, bounding box detectors may also be able to identify these objects (Rosebrock, 2016).

To address all these issues, labeling objects using polygon annotations and training an instance segmentation model may be a viable solution, particularly when overlap cannot be entirely avoided due to image nature. This approach is expected to yield a recall improvement of at least 10% (He, et al., 2017).

4.6.5 Polygon Annotations

Polygon annotation is a method of defining the boundaries of an object or region of interest in an image or video by specifying a set of points connected by lines to form a closed shape. These polygons represent the shape of the object or region and are frequently utilised in computer vision to train machine learning models (Pont-Tuset, et al., 2017).

One of the main advantages of polygon annotation is the ability to accurately and precisely label objects or regions in images or videos. This method allows for more complex shapes to be defined than simple bounding boxes or circles, which may not accurately capture the object of interest or could capture areas of the slightest interest. Polygon annotation is especially useful when working with objects or regions with irregular shapes or partially occluded, allowing for a more accurate representation of their boundaries (Keymakr, 2021).

Polygon annotation is a prevalent method used in Image Segmentation, where it is crucial to precisely identify the characteristics and limits of objects or regions in the image. For example, polygon annotation can be used to label objects in images for Instance Segmentation to define the boundaries of cells in microscopy images or radiology for Image Segmentation.

In Machine Learning, high-quality training data is crucial for accurate model development, and Polygon annotation is a necessary process to achieve such high-quality data. The process of polygon annotation can be arduous and lengthy, as it involves manually labeling each point on the object or region boundary. It involves drawing a series of connected straight lines around a region of interest in an image, forming the

closed shape of a polygon. This process is commonly performed using a graphical user interface (GUI), allowing for pixel-perfect labeling of complex and irregular datasets.

In contrast to bounding boxes, which are limited to rectangular and square shapes, polygon annotation allows for more complex shapes to be captured, including irregular shapes common in real-world scenarios. Polygon annotation is performed by clicking on specific points to plot vertices, giving the annotator greater flexibility to capture an object's actual shape.

Once an object is annotated with a polygon, a descriptive label must be assigned to enable Machine Learning image processing. Accurate labeling is crucial, as it ensures the model understands the contents of the polygon. Failure to properly label an image or video can result in inaccurate data.

4.6.6 How to Encapsulate the Scale of the geological features

As mentioned in Chapters 1 and 2, geology is a multiscale problem, meaning that sedimentological features can be present under different scales, and their correct recognition affects the prediction of the depositional environment. While a geologist can easily learn and recognize the significance of scale and identify features accordingly, the same is not valid for a non-human (machine). The Computer Vision models are trained on images including various objects similar to the ones we are trying to predict. For instance, if we want a model to predict soccer balls, we will train it with examples of soccer balls. Now, since a soccer ball is specific and almost always looks the same in terms of shape and patterns, it can be recognized easily at all scales, from a few centimeters to meters. For a human geologist, the scale of the geological features is a differentiating factor. For instance, if a geologist is asked to classify an image of planar bedding and a planar lamination, the distinction between the two is almost impossible without a scale reference. The machine does not automatically understand the concept of scale and its importance as a differentiating factor.

Therefore, adapting the scale to the Computer Vision models is one of the most significant challenges throughout this project. To address this, two steps were followed; Train the CV models with images belonging to various scales and embed the dimensionality of the features with the annotations of the images.

As shown in Chapter 7, an accurate adaptation of scale during the annotations improved the performance of the segmentation model's predictions by allowing more correct label assignments to the geological features of the outcrop.

4.7 Geologic Datasets for Image Classification

This section introduces seven different datasets used for geologic Image Classification. The first part introduces five simple datasets in terms of the number of labels in the dataset as well as the number of images per dataset.

Some datasets consist only of outcrop images, others of geological sketches, while two use a blend of outcrop images and sketches. Adding sketches was initially an idea to enrich the number data by adding more examples of the desired classes. Furthermore, as sketches are often a tool for geologists to communicate their observations with other geologists, they encompass interpretational elements.

Geologists use drawings to explain the subtle features of complex rock textures and sedimentary structures, as sketches contain only essential information. So, I thought that adding sketches to the image classification data for geological structures could improve the CNN model's predictions of the structures as interpretations contain only the relevant information to classify the feature, ignoring irrelevant features such as colour, surface textures, shadows, and vegetation. When training the Image Classification model, using only simple geological sketches might often give little information about the structure of interest as they lack texture and colour. On the other hand, real photographs of geological features, apart from the necessary information (object of interest), usually contain complexity (noise) and irrelevant information (e.g., vegetation and other background elements). Thus, a combination of the two data types is recommended.

According to the results in Chapter 5, for a more successful geological Object Classification with a CNN, both data types, outcrop images, and geological sketches should be included in the input data to encapsulate the shape, scale, texture, and complexity (Nathanail, et al., 2021). Depicting the actual structure on a black-and-white sketch, without any background or complex detail, combined with the real outcrop photos, would help the model learn the entire structure and thus predict it more accurately. Furthermore, due to the lack of large geologic datasets, the use of sketches depicting geological features can provide an additional benefit. Not only it increases the training

data size, but it also enhances its variability and credibility by introducing some idealized examples of certain structures from which the model may improve its learning.

The Data Augmentation techniques suitable for geology, previously discussed in section 4.5, were applied to all datasets described in section 4.6 for both data types and served as an essential part of the dataset's manipulation and pre-processing.

The two types of data we used and blended are data observed directly from outcrop photographs (evidence) and hand-drawn or edited sketches (interpretational data), generated based on established geological interpretations and literature. Section 4.7.1 describes the five different Datasets (D1, D2, D3, D4, and D5, shown in Table 4-3 through Table 4-8, generated for this study, and each one has a unique characteristic in the training and/or the test sets. However, they all consist of the same four sedimentary structures (planar lamination, cross lamination, hummocky cross-stratification (HCS), and cross-bedding), examples of which can be found in Figure 4-11. We decided to use these specific structures as some of them (cross-bedding and cross-lamination) may look similar to the machine, but in reality, they vary in scale and geometry. Any sedimentary structures could have been chosen. The selection of these four structures in the first five datasets was a good starting point to test how image classification performs in a geological setting.



Figure 4-11: The four sedimentary structures present in Datasets 1-5.

Section 4.7.2 describes the addition of two more datasets, Dataset 6 (D6) and Dataset 7 (D7), enriched with more images of outcrops and sketches containing 20 more sedimentary structures and fossil types, reaching a total of 24 classes. Dataset 6, similarly to Dataset 2, includes only real outcrop images, while Dataset 7, similar to Dataset 4, is a blended dataset containing both outcrop images and geological sketches.

4.7.1 Image Classification Datasets (Part 1)

In Chapter 5, a series of experiments were conducted to show the importance of using blended datasets consisting of outcrop images and geological sketches depicting fossils and sedimentary structures. The details of each dataset can be found in the corresponding tables (Table 4-3 through Table 4-8). The first dataset used for this chapter, Dataset 1 (D1), consists only of sketches showing four specific sedimentary structures, cross-bedding, cross-lamination/climbing ripples, planar lamination, and Hummocky cross-stratification. Dataset 2 (D2) contains the aforementioned sedimentary structures but only includes outcrop images instead of sketches. Dataset 3 (D3) includes sketches in the training and validation sets, with the test set consisting only of outcrop images. Dataset 4 (D4) is the blended dataset containing images of outcrops and geological sketches in the training and validation sets. In contrast, the test set consists only of outcrop images, depicting the same four sedimentary structures once more. Finally, Dataset 5 (D5) is like Dataset 2 but includes a greater number of outcrop images in the training set.








Dataset Name		Training Set	Test Predictions
Dataset 1 (D1)		Sketches	
Dataset 2 (D2)		Outcrop Photos	
Dataset 3 (D3)		Sketches	
Dataset 4 (D4)		Sketches + Outcrop Photos	
Dataset 5 (D5)		Outcrop Photos	

Table 4-3: Data type distribution between training, validation, and test data for D1-D5.

These datasets were used to improve sedimentary structure classification by learning from photographs and sketches. Chapter 5 shows how Image classification is used to classify geological features utilising the datasets presented here and how blended datasets can improve the results of such a model.

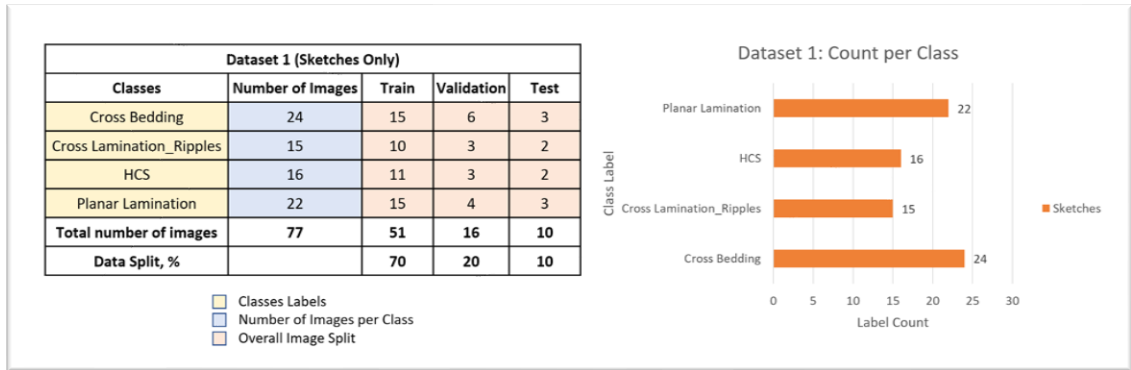


Table 4-4: Detailed breakdown of Dataset 1.

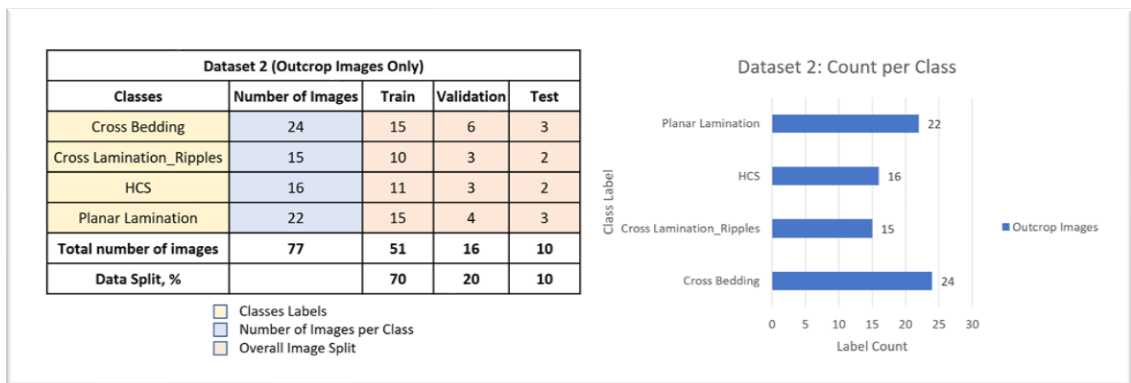


Table 4-5: Detailed breakdown of Dataset 2.

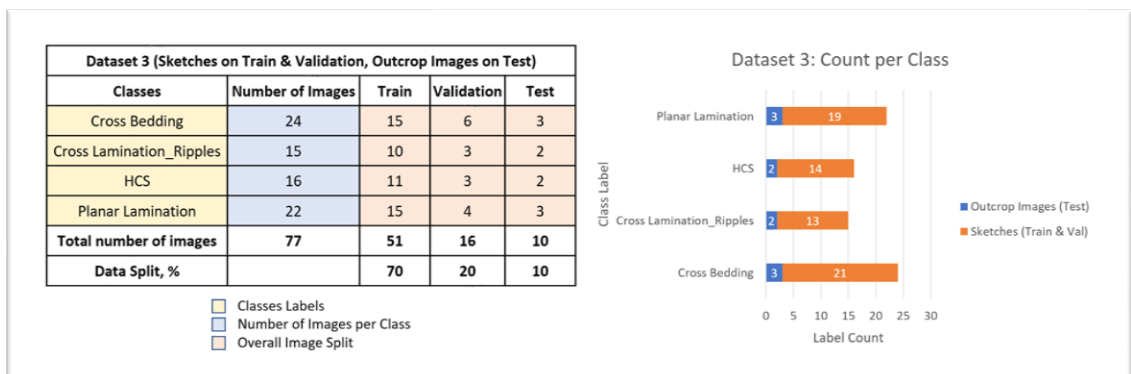


Table 4-6: Detailed breakdown of Dataset 3.

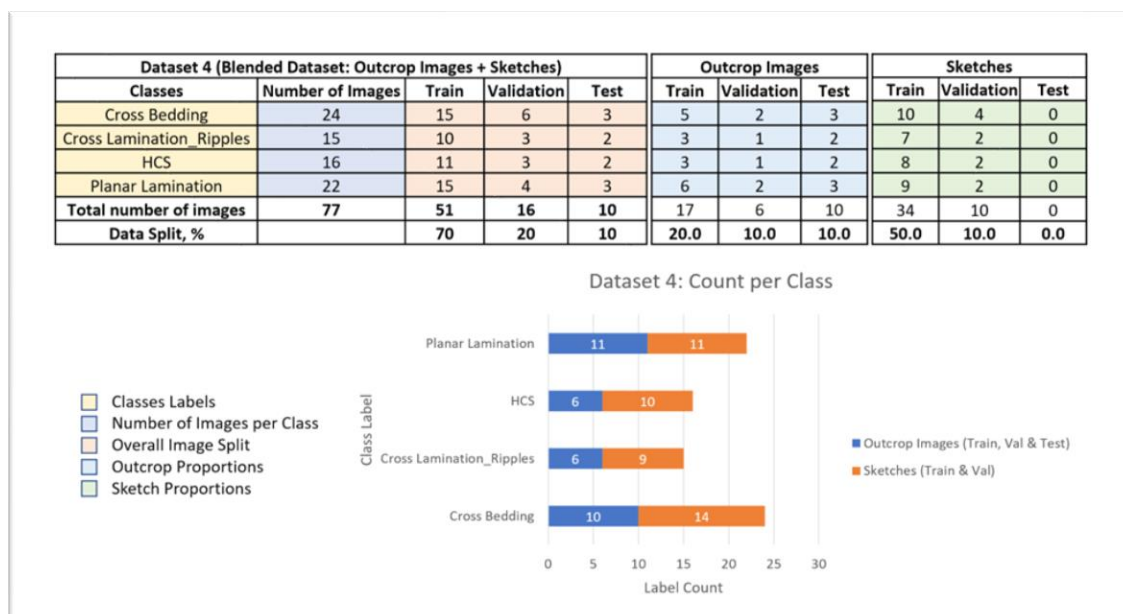


Table 4-7: Detailed breakdown of Dataset 4.

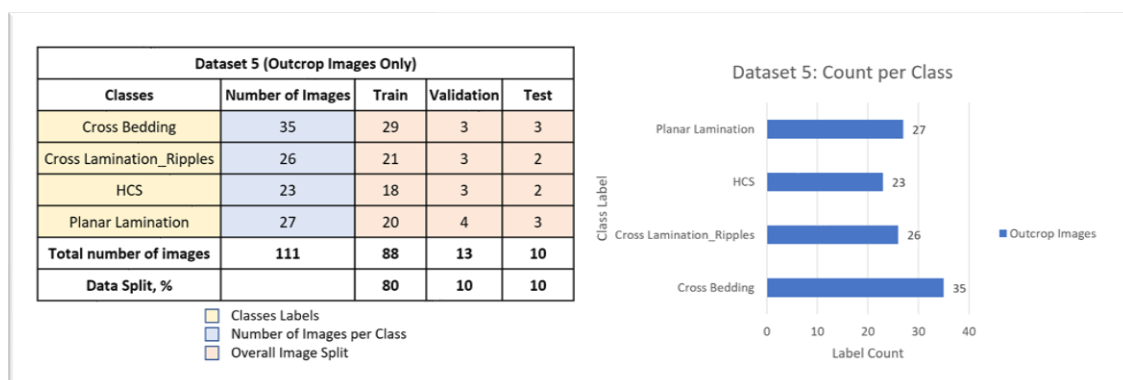


Table 4-8: Detailed breakdown of Dataset 5.

4.7.2 Image Classification Datasets (Part 2)

This part introduces two additional datasets summarized in Table 4-9, one consisting only of outcrop images (Dataset 6) and the other being once more a blended dataset (Dataset 7). Compared to Part 1 (section 4.7.1), the difference lies in the number of classes present in these datasets and, consequently, the number of images within the datasets. Twenty more classes were added, as shown in the pairs of Table 4-10 and Figure 4-12 and Table 4-11 and Figure 4-13, respectively, for each dataset. Therefore, D6 and D7 were built with 24 classes each, containing a range of sedimentary structures and fossils. D7 was generated based on the results of Chapter 5 using the five datasets (D1-D5) and particularly D4, where it was found that the optimal proportion of sketches to improve

the CNN classifier’s learning is between 40-50% in the dataset. D7 has 40% sketches and 60% outcrop images in the proportions shown in Table 4-11. The datasets in section 4.7.1 were created to establish a benchmark only to investigate how image classification performs on outcrop data. The datasets in section 4.7.2 (D6 and D7) were created to train a refined CNN model for geologic image classification tasks.


Dataset Name		Training Set	Test Predictions
Dataset 6 (D6)		Outcrop Photos	
Dataset 7 (D7)		Sketches + Outcrop Photos	

Table 4-9: Data type distribution between training, validation, and test data for D6 and D7.

Dataset 6 (Outcrop Images)				
Classes	Number of Images	Train	Validation	Test
Animal Fossils	19	12	4	3
Planar(Parallel) Lamination	16	8	4	4
Flame Structures	13	9	4	0
Convolute Lamination	12	7	3	2
Dish Structures	7	5	2	0
Flaser Lamination	12	6	4	2
Lenticular Lamination	8	5	2	1
Wavy Lamination	24	15	6	3
Climbing Ripples	13	9	3	1
Cross Lamination	14	8	3	3
Wave Ripples	7	5	2	0
Flute Marks	11	9	2	0
Mud Cracks	7	4	2	1
Ripple Marks	12	9	3	0
Syneresis Cracks	14	11	3	0
Herringbone Cross Stratification	9	6	3	0
Swaley Cross Stratification	1	1	0	0
Hummocky Cross Stratification	6	4	2	0
Ammonites	27	18	9	0
Belemnites	20	13	5	2
Corals	14	9	3	2
Crinoids	17	11	4	2
Plant Fossils	12	8	4	0
Trilobites	15	9	4	2
Total number of images	310	201	81	28
Data Split, %	100	65	26	9

Classes Labels
 Number of Images per Class
 Overall Image Split

Table 4-10: Detailed breakdown of Dataset 6.

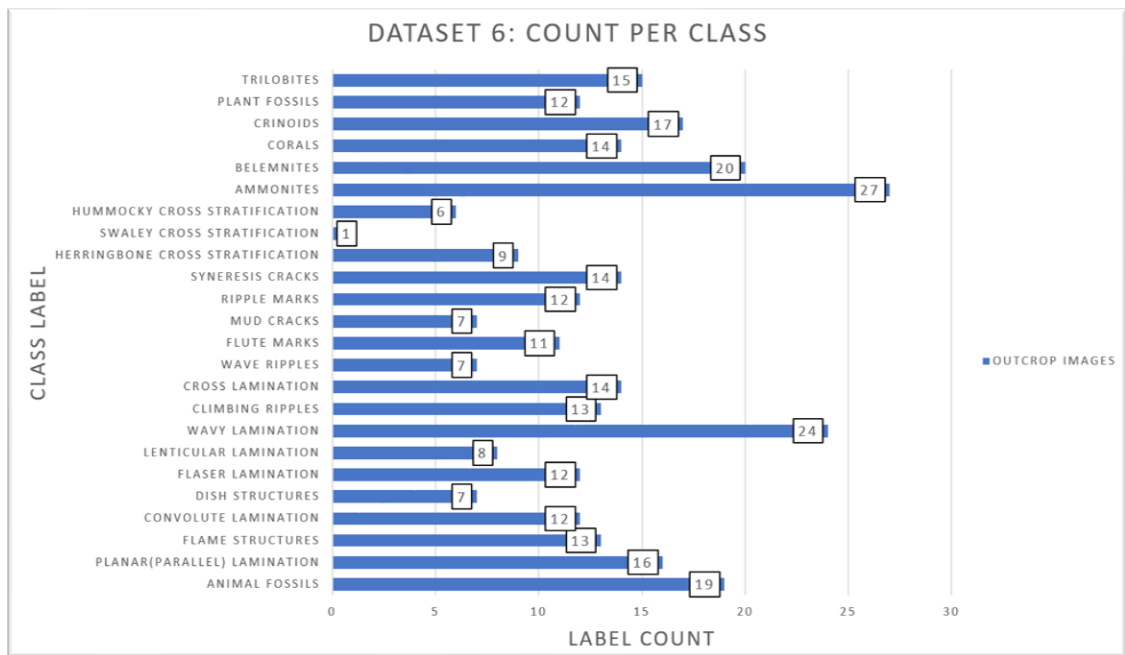


Figure 4-12: Dataset 6 Class labels vs. the label count. The figure shows the number of occurrences of each class in the dataset.

Dataset 7 (Blended Dataset: Outcrop Images + Sketches)					Outcrop Images			Sketches		
Classes	Number of Images	Train	Validation	Test	Train	Validation	Test	Train	Validation	Test
Animal Fossils	35	26	6	3	18	4	3	8	2	0
Planar(Parallel) Lamination	34	24	6	4	15	4	4	9	2	0
Flame Structures	29	21	6	2	12	4	2	9	2	0
Convolute Lamination	24	17	5	2	9	3	2	8	2	0
Dish Structures	20	13	5	2	6	3	2	7	2	0
Flaser Lamination	30	22	6	2	11	3	2	11	3	0
Lenticular Lamination	23	17	4	2	10	2	2	7	2	0
Wavy Lamination	37	23	9	5	12	6	5	11	3	0
Climbing Ripples	26	19	5	2	12	3	2	7	2	0
Cross Lamination	32	23	6	3	13	3	3	10	3	0
Wave Ripples	16	11	3	2	6	2	2	5	1	0
Flute Marks	28	19	6	3	11	3	3	8	3	0
Mud Cracks	19	14	4	1	8	2	1	6	2	0
Ripple Marks	31	23	6	2	12	3	2	11	3	0
Syneresis Cracks	27	20	5	2	12	3	2	8	2	0
Herringbone Cross Stratification	20	15	3	2	9	3	2	6	0	0
Swaley Cross Stratification	2	1	0	1	1	0	1	0	0	0
Hummocky Cross Stratification	17	12	4	1	3	2	1	9	2	0
Ammonites	49	33	12	4	23	9	4	10	3	0
Belemnites	36	27	7	2	19	4	2	8	3	0
Corals	27	20	5	2	15	3	2	5	2	0
Crinoids	32	23	6	3	16	4	3	7	2	0
Plant Fossils	25	17	6	2	11	4	2	6	2	0
Trilobites	33	25	6	2	15	4	2	10	2	0
Total number of images	652	465	131	56	279	81	56	186	50	0
Data Split, %	100	71	20	9	60.0	61.8	100.0	40.0	38.2	0.0

Table 4-11: Detailed breakdown of Dataset 7.

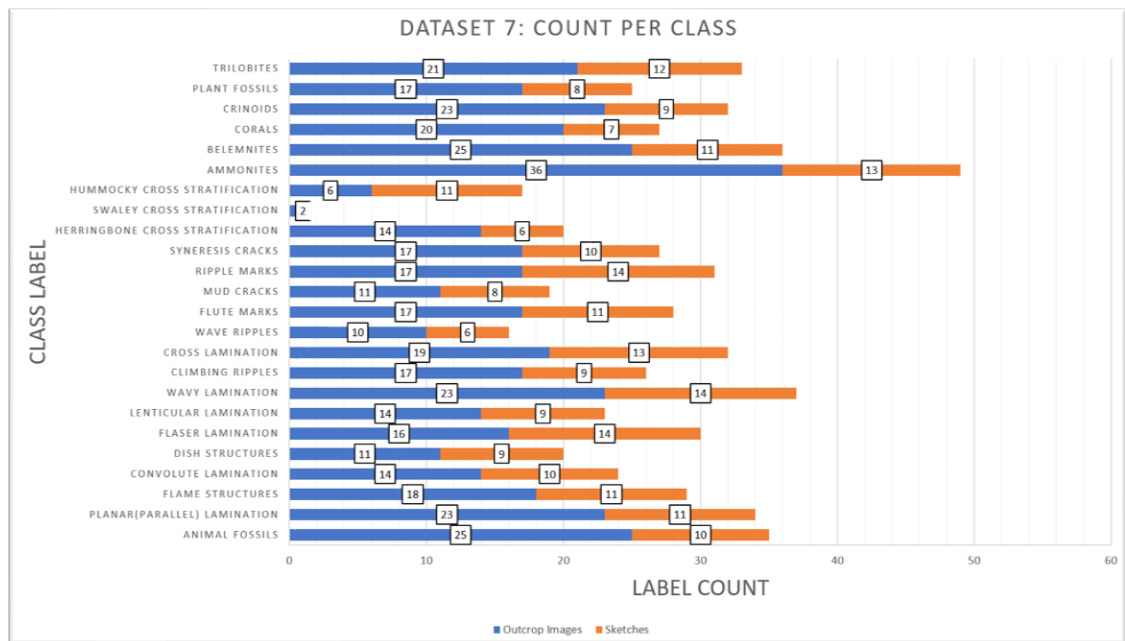


Figure 4-13: Dataset 7 Class labels vs. the label count. The figure shows the number of occurrences of each class in the dataset.

4.8 Geologic Dataset for Object Detection

In this section, a new dataset, D8 (Table 4-12/Figure 4-14) and D11 (Table 4-13/Figure 4-15) is introduced and used in Chapter 6 for geological Object Detection of sedimentary structures and fossils. Compared to section 4.7, the dataset of the current section does not include any sketches, as it was found that sketches were helpful for Image Classification but not for Object Detection and Instance Segmentation tasks. As mentioned in section 4.6, both Object Detection and Instance Segmentation use annotations in the form of bounding boxes and polygons, respectively, to capture the objects of interest, while Image Classification was only using different data folders to separate the classes. In that case, you can think of sketches as an additional Supervised Learning technique to guide the Image Classification model where to look at the entire image; in practice, it acts as an annotation technique for Image Classification. Object Detection is not the best approach for identifying geological objects, as there is a risk of capturing a high amount of unwanted information with rectangular bounding boxes.

However, I assembled Datasets 8 and 11 to train an Object Detection model that, at least, can provide indications about the geologically interesting parts of the outcrop, which can be further investigated with Image Classification and Instance Segmentation. Each

geological object belonging to a class was manually annotated by assigning bounding boxes (section 4.6.4) around each instance using the LabelMe annotation tool (section 4.6.3.1).

Dataset 8 consists of a total of 138 outcrop images and 23 classes of sedimentary structures. The specific geological classes, the number of labels per class, and the number of images per data split are shown in Table 4-12 and Figure 4-14. The label 'Fault' has been incorporated into the data sets used to train the object detection and instance segmentation models. This label was incorporated with the sedimentological features only to test the models' ability to predict structural features from the outcrops. Using these models to identify structural features will require another custom dataset for training and will be part of future work and recommendations for improving this thesis further.

Dataset 11 consists of a total of 142 fossil images and 7 fossil types. The specific geological classes, the number of labels per class, and the number of images per data split are shown in Table 4-13 and Figure 4-15. An important note regarding the 'Animal Fossil' label needs to be mentioned. That particular name for the label was chosen to group multiple fossil types under a single category, including various types of big and smaller animal fossils, other skeletons, and animal remains. The grouping of multiple fossils under a single, more generic label aimed to save a lot of time during the annotation of the data and also during the model's training.

Dataset 8			
Object Detection YOLOv6 Labels for Sedimentary Structures	Label Count in Training and Validation set	Label Count in Training set	Label Count in Validation set
Bioturbation	25	18	8
Clasts	32	22	10
Convolute/Irregular Bedding	2	1	1
Cross Bedding/Stratification	25	18	8
Cross Lamination/Climbing Ripples	13	9	4
Desiccation Cracks	5	4	2
Erosive Features	24	17	7
Fault	7	5	2
Flame Structures	3	2	1
Flaser Lamination	2	1	1
Fossils	4	3	1
Herringbone Cross Stratification	7	5	2
Hummocky Cross Stratification	6	4	2
Lenses	13	9	4
Lenticular Bedding	5	4	2
Lenticular Lamination	4	3	1
Planar/Parallel Bedding	26	18	8
Planar/Parallel Lamination	15	11	5
Structureless	24	17	7
Swaley Cross Stratification	2	1	1
Syneresis Cracks	2	1	1
Wave Ripples/Lamination	5	4	2
Wavy Bedding	2	1	1
Total Number of Labels	253	177	76
Percentage of labels, %	100	70	30

Dataset 8 (Outcrop Images)		
Total Number of Images	138	100%
Training set Images	97	70%
Validation set Images	41	30%

Table 4-12: Detailed breakdown of Dataset 8.

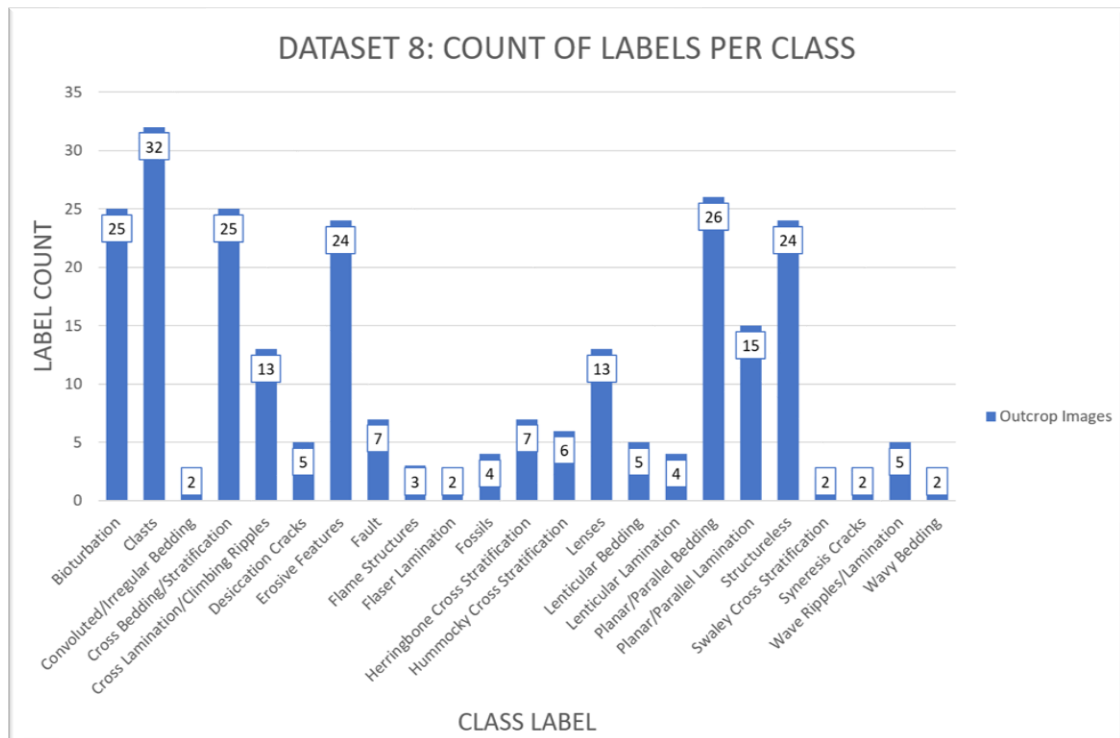


Figure 4-14: Count of labels per class for Dataset 8.

Dataset 11			
Object Detection Yolov6 Labels for Fossils	Label Count in Training and Validation set	Label Count in Training set	Label Count in Validation set
Ammonite	52	36	16
Animal Fossil	17	12	5
Belemnite	43	30	13
Coral	25	18	8
Crinoid	63	44	19
Plant Fossil	27	19	8
Trilobite	21	15	6
Total Number of Labels	248	174	74
Percentage of labels, %	100	70	30

Dataset 8 (Outcrop Images)		
Total Number of Images	142	100%
Training set Images	99	70%
Validation set Images	43	30%

- Classes
- Number of Labels
- Label Split
- Image Split

Table 4-13: Detailed breakdown of Dataset 11.

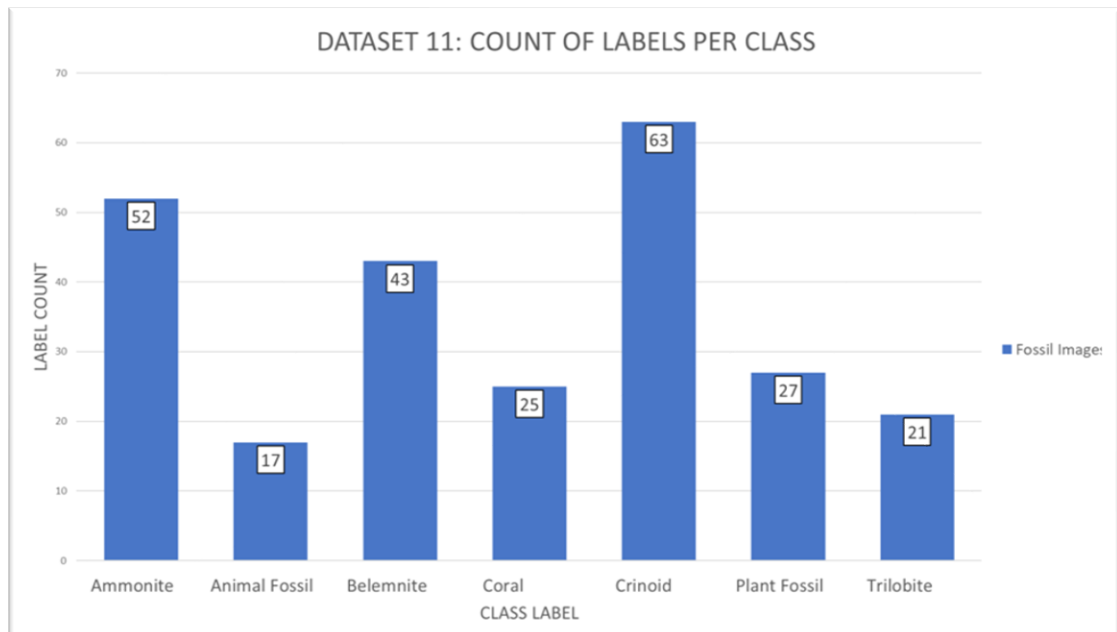


Figure 4-15: Count of labels per class for Dataset 11.

4.9 Geologic Datasets for Instance Segmentation

This section presents the last two image datasets (D9 and D10) assembled for this thesis. Datasets 9 and 10 were created to train the Instance Segmentation model, which, according to the results of Chapter 7, provides the most valuable information out of the three Computer Vision methods described in this thesis. Dataset 9 was used to create a benchmark to evaluate the Instance Segmentation model (similar to sections 4.7.1 and 4.7.2). In contrast, Dataset 10 (A & B) was created to train the Outcrop Segmentation model and improve the segmentation of an outcrop's lithology and sedimentary structures. Each geological object belonging to a class was manually annotated by assigning polygons (section 4.6.5) around each instance using the LabelMe annotation tool (section 4.6.3.1). The use of polygons captures the exact shape and location of each geological feature.

Lithology and sedimentary structures are both important components of sedimentology, but they differ in terms of the data sets they focus on and their relationship to each other. The relationship between lithology and sedimentary structures is intertwined. Lithological characteristics, such as grain size, sorting, and mineralogy, can influence the formation and preservation of sedimentary structures. Conversely, the presence of specific sedimentary structures can provide clues about the lithological properties of the rocks.

Lithology primarily deals with the physical and chemical characteristics of rocks, such as their mineral composition, grain size, texture, colour, and overall appearance. It involves the description and classification of rocks based on these attributes. Lithological data sets typically involve detailed observations and measurements of rock samples collected from outcrops, boreholes, or other geological settings. These data sets provide valuable information about the properties and characteristics of the rocks themselves, aiding in the interpretation of depositional environments and geological processes.

On the other hand, sedimentary structures refer to the features and patterns within sedimentary rocks that are indicative of specific depositional processes and environments. These structures include bedding planes, cross-bedding, ripple marks, mud cracks, and various other sedimentary features. Sedimentary structures are typically observed and documented in the field or within rock exposures. They provide insights into the

dynamics of sediment transport, deposition, and subsequent geological events that have affected the sedimentary sequence.

There are multiple environments and sub-environments of deposition containing a variety of sedimentary structures, lithology and fossil types, but only Aeolian, Fluvial, Alluvial, Shallow Marine, and Deep Marine environments are considered in this research and datasets. These environments are crucial to understand and interpret, as a large number of the earth's hydrocarbon reserves are found in these environments. Understanding the sand bodies' internal characteristics, distribution, geometry, and lateral extent is an essential element in exploring and exploiting hydrocarbons and other natural resources (Siddiqui, et al., 2017).

Deltaic environments are not considered at this stage as "most deltas cannot be seen in their entirety from a single outcrop, only bits of the system represent terrestrial to offshore deposits in increasing water depth," according to (Allen, 2014). He explains that deltaic sedimentation is complex and influenced by many factors, including the shape and size of the delta, the type and volume of sediment being deposited, and the dynamics of the river and marine currents. Coleman et al. (1981) and Dalrymple et al. (1994) emphasize that the complex nature of deltaic sedimentation makes it difficult to interpret the depositional environment based on limited outcrop data (Coleman, 1981), (Dalrymple, et al., 1994). Therefore, a more comprehensive approach that integrates multiple data sources (e.g., well logs, seismic data, and sediment cores) is often required to understand deltaic systems fully.

4.9.1 Instance Segmentation Yolact with Mixed Labels (Dataset 9)

This section introduces Dataset 9 (D9), which includes outcrop images and their corresponding annotations in JSON files. The annotations include a mix of classes for lithology types and sedimentary structures.

Dataset 9 consists of 62 images and a total of 15 distinct geological labels present in the images in different proportions, as shown in Table 4-14/Figure 4-16. The images in this dataset represent different environments from three different outcrops, a Fluvial/Aeolian (Wessex Basin, UK), a Fluvial/Alluvial (Cinca Canal, Spain), and a Deep marine environment (Tourmaline Beach, CA, USA). These three outcrops were chosen as they were not very complex in terms of the sedimentary structure representation and therefore

provided a good starting point for testing the segmentation model’s capabilities in segmenting the geology.

Dataset 9 (Outcrop Images)			
Instance Segmentation Yolact Mixed Labels	Label Count in Training and Validation set	Label Count in Training set	Label Count in Validation set
Planar Bedding	13	9	4
Planar Lamination	6	4	2
Cross Bedding	6	4	2
Cross Lamination	6	4	2
Interbedded Sands	33	23	10
Erosive Feature	34	24	10
Cemented Sands/Eroded Sands	10	7	3
Mudstones	79	55	24
Medium to Fine Sandstone	1	1	0
Coarse to Medium Sandstone	6	4	2
Conglomerate	50	35	15
Siltstone	3	2	1
Unconformity	12	8	4
Rip up clasts/Silty sands	12	8	4
Sandstone	103	72	31
Total Number	374	260	111
Percentage of labels	100	70	30

Dataset 9 (Outcrop Images)		
Total Number of Images	62	100%
Training set Images	44	70%
Validation set Images	18	30%

Table 4-14: Detailed breakdown of Dataset 9.

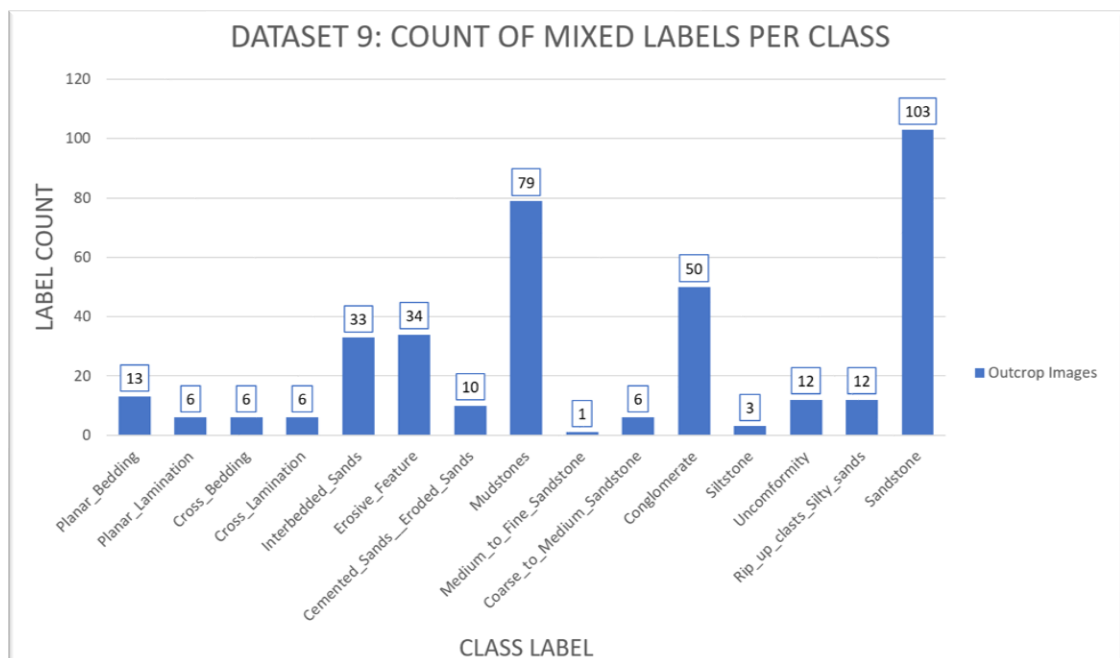


Figure 4-16: Count of labels per class for Dataset 9.

4.9.2 Instance Segmentation (Dataset 10A & 10B)

This section describes Datasets 10A and 10B. Compared to Dataset 9, where the annotations for lithology and sedimentary structures were mixed, Dataset 10 was split into two parts, part A containing only annotations for the segmentation of lithology and part B containing the annotations for the segmentation of sedimentary structures. Both datasets contain the same 70 images, which were differently annotated depending on the segmentation task.

The reasoning behind the formation of these two datasets was to train two different segmentation models, 10A to segment and estimate the lithology and 10B to predict the sedimentary structures. The details of D10A and D10B regarding the classes, number of labels, and images, as well as the data-split proportions, are summarized in Table 4-15/Figure 4-17 and Table 4-16/Figure 4-18, respectively.

4.9.2.1 Instance Segmentation Yolact Labels for Lithology (Dataset 10A)

Dataset 10A (Outcrop Images)			
Instance Segmentation Yolact Labels for Lithology	Label Count in Training and Validation set	Label Count in Training set	Label Count in Validation set
Amalgamated/Cemented Bed	1	1	0
Breccia	6	4	2
Carbonates	7	5	2
Conglomerate	29	20	9
Interbedded mudstone-siltstone	27	19	8
Interbedded sandstone-mudstone	6	4	2
Interbedded sandstone-siltstone	21	15	6
Iron Rich Sediment	7	5	2
Mudstone	53	37	16
Organic Material	37	26	11
Red (Sandstone) Beds	21	15	6
Sandstone	79	55	24
Siltstone	28	20	8
Total Number	322	225	97
Percentage of labels	100	70	30

- Classes
- Number of Labels
- Label Split
- Image Split

Dataset 10A (Outcrop Images)		
Total Number of Images	70	100%
Training set Images	49	70%
Validation set Images	21	30%

Table 4-15: Detailed breakdown of Dataset 10A.

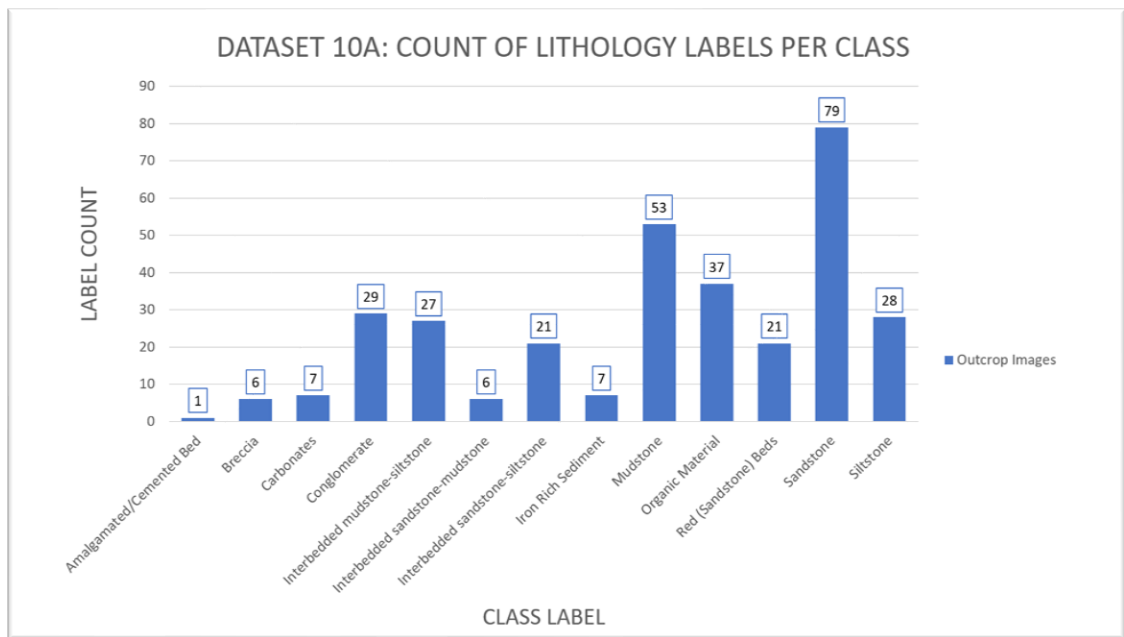


Figure 4-17: Count of labels per class for Dataset 10A.

4.9.2.2 Instance Segmentation Yolact Labels for Sedimentary Structures (Dataset 10B)

Dataset 10B (Outcrop Images)			
Instance Segmentation Yolact Labels for Sedimentary Structures	Label Count in Training and Validation set	Label Count in Training set	Label Count in Validation set
Bioturbation	48	34	14
Clasts	52	36	16
Convoluted/Irregular Lamination	2	1	1
Convoluted/Irregular Bedding	2	1	1
Cross Bedding/Stratification	39	27	12
Cross Lamination/Climbing Ripples	22	15	7
Dessication Cracks	6	4	2
Erosive Contacts/Bases	76	53	23
Erosive Features	31	22	9
Faults	16	11	5
Flame Structures	6	4	2
Flaser Lamination	3	2	1
Flute Marks	42	29	13
Fossils	12	8	4
Herringbone Cross Stratification	9	6	3
Hummocky Cross Stratification	10	7	3
Lenticular Bedding	8	6	2
Lenticular Lamination	6	4	2
Planar/Parallel Bedding	46	32	14
Planar/Parallel Lamination	29	20	9
Scour Marks	6	4	2
Structureless	56	39	17
Swaley Cross Stratification	4	3	1
Syneresis Cracks	2	1	1
Wave Ripples/Lamination	6	4	2
Wavy Bedding	6	4	2
Total Number	545	382	164
Percentage of labels	100	70	30

- Classes
- Number of Labels
- Label Split
- Image Split

Dataset 10B (Outcrop Images)		
Total Number of Images	70	100%
Training set Images	49	70%
Validation set Images	21	30%

Table 4-16: Detailed breakdown of dataset 10B.

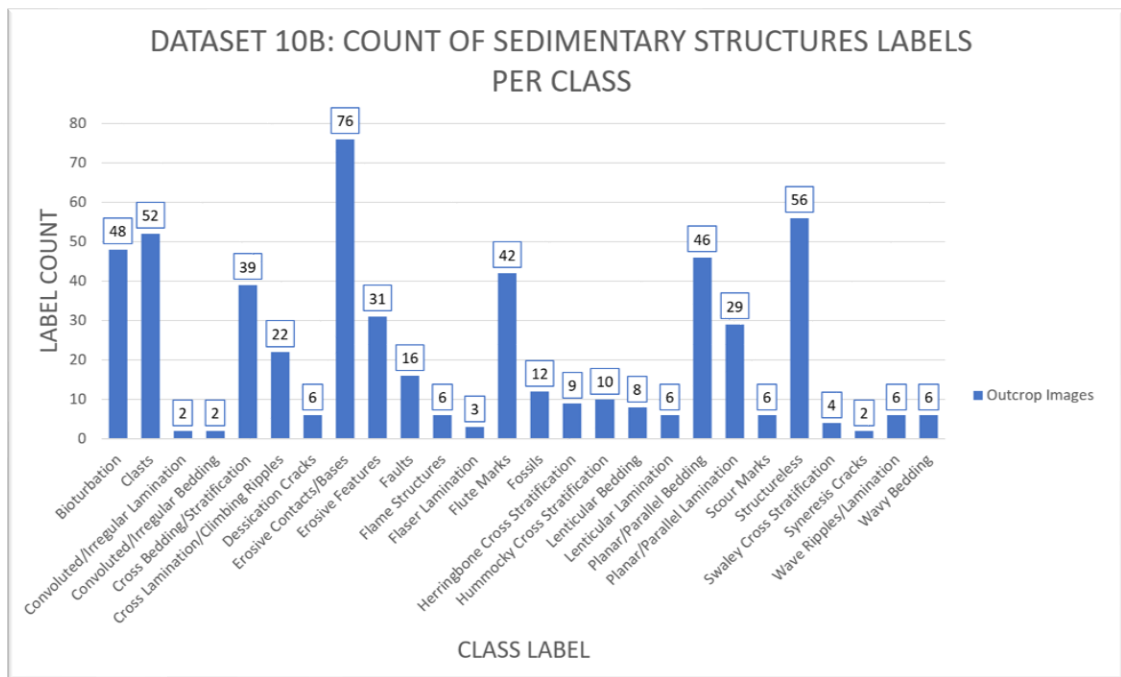


Figure 4-18: Count of labels per class for dataset 10B.

4.10 Chapter's Conclusion

This chapter demonstrates in detail the workflow I used to assemble the eleven geological datasets used in this thesis. Each CV technique served a different purpose in the overall thesis's workflow and proved useful in different ways regarding the extraction of the sedimentological features from the outcrop images.

As shown in Figure 4-10 of this chapter and previously explained in Chapter 3, Image Classification categorizes single fossils or sedimentary structures based on their appearance, Object Detection allows multiple object classification and localization of fossils and sedimentary structures within a single image at different scales, and finally, Instance Segmentation assigns masks around multiple lithology types, in addition to fossils and sedimentary structures, capturing their shape within the bounding boxes.

For each of the three Computer Vision methods, different datasets had to be compiled instead of using one big dataset. Some of the images are the same across the eleven datasets, but the majority is different because, for example, Instance segmentation also focuses on lithology, while Object Detection and Image Classification only deal with sedimentary structures and fossils.

Seven datasets, D1-D7, were used to train my custom Image Classification model, two datasets, D8 and D11, were used to train the YOLOv6-S model for Object Detection, and datasets D9 and D10 were utilised for training the YOLACT model for Instance Segmentation. The summary of all eleven datasets is shown in Table 4-17.

Dataset	Data Type	Image Number	Number of Geological Features	Task
Dataset 1 (D1)	Sketches	77	4	Image Classification
Dataset 2 (D2)	Outcrops	77	4	Image Classification
Dataset 3 (D3)	Sketches	77	4	Image Classification
Dataset 4 (D4)	Sketches + Outcrops	77	4	Image Classification
Dataset 5 (D5)	Outcrops	111	4	Image Classification
Dataset 6 (D6)	Outcrops	310	24	Image Classification
Dataset 7 (D7)	Sketches + Outcrops	652	24	Image Classification
Dataset 8 (D8)	Outcrops	138	23	Object Detection
Dataset 9 (D9)	Outcrops	62	15	Instance Segmentation
Dataset 10 (D10A)	Outcrops	70	13	Instance Segmentation
Dataset 10 (D10B)	Outcrops	70	26	Instance Segmentation
Dataset 11 (D11)	Outcrops	142	7	Object Detection

Table 4-17: Summary table of all the assembled datasets and their corresponding properties.

The following three chapters demonstrate the results for each of the abovementioned Computer Vision models exploiting their respective datasets.

CHAPTER 5 - THE USE OF SKETCHES TO IMPROVE THE IMAGE CLASSIFICATION OF SEDIMENTARY STRUCTURES

This chapter demonstrates how Image Classification can be utilised to classify single geological structures and fossils. This Chapter tackles the first step of the thesis's high-level workflow introduced in Chapter 1. According to the chapter's findings, training an Image Classification model with a blended dataset consisting of 2D outcrop images and simplified sketches of geological structures improves the model's predictions and learning capability to identify complex sedimentary structures and fossils. The model assigns a class for each tested image and displays each prediction next to the ground truth label for a geologist to evaluate.

5.1 Introduction

Geologists use drawings to explain the subtle features of complex rock textures and sedimentary structures, as sketches contain only essential information. On the thought that machine learning aims to imitate human learning and thinking, I assumed that, by adding sketches to the image classification dataset, the CNN model's robustness and prediction accuracy of the sedimentary structures should improve. As sketched interpretations contain only the relevant information to classify the feature, ignoring irrelevant features such as colour, surface textures, shadows, and vegetation, we can think of sketches as realistic templates of the geology, which the model can learn upon. Hence, sketches could assist the model in learning only the desired geological patterns by constraining it from interpreting all sorts of patterns.

Geological sketches encapsulate geological knowledge in a simplified manner. According to the findings of this chapter, supplementing photographic datasets of geological outcrops with sketched interpretation data can enhance the accuracy of sedimentary structure classification, even with smaller volumes of data. CNNs trained on outcrop images can classify the geological features without sketches in their training data but result in lower test accuracy scores. The application of my custom image classification model on Datasets 2, 5, and 6 demonstrates that the classification of geological features is feasible but not as good without the sketches due to the complexity of the geology, as shown in section 5.4.

Using only simple geological sketches in the training dataset gives little information about the structure of interest as they lack texture and colour. On the other hand, natural photographs of geological features, apart from the necessary information (object of interest), usually contain complexity (noise) and irrelevant information (e.g., vegetation and other background elements). Therefore, the use of sketches aims to make the AI pattern recognitions focus on the right kind of features represented by the sketches, ignoring any irrelevant features in the images.

By incorporating sketches of sedimentary structures and fossils with natural outcrop photos, a CNN Image Classification model can predict and categorize specific geological structures more accurately based on the prominent features in each image. This can significantly facilitate the work of geologists in distinguishing between different sedimentary structures and fossils in a matter of seconds.

The results show that the CNN model misclassified various geological features when trained only with one data type (outcrop photos or geological sketches). On the contrary, using a blended dataset for the model's learning results in fewer misclassifications and higher test accuracy of the model predictions of the geological features.

5.2 Previous Use of Sketches in Machine Learning

In recent years, sketch and image recognition have had a plethora of applications in image and 3D shape retrieval, as well as in synthesis and reconstruction. According to Zheng, (2021), one of the main differences between sketch recognition and object recognition is that sketches do not indicate any texture or colour (Zheng, et al., 2021). Sketches are abstractive in general, making sketch recognition an incredible challenge (Zhang, 2021). Zheng et al. (2021) demonstrate how to use sketches as an augmentation technique to enhance sketch recognition with sketch-based datasets (Zheng, et al., 2021).

As this chapter is not dealing with sketch recognition but rather with outcrop image classification, my proposal is not to use sketches as an augmentation technique but rather as a separate data type that will be combined with images to train a CNN model. The lack of colour and texture of sketches is an advantage when geologists draw geological features/outcrops, as they remove the noise and leave important features. Sketches can help simplify the complex textures and details of an object, allowing for easier recognition

and classification by making the actual lines and shape of the object to be more discernible.

Changjian Li et al. (2018) presented an approach for modeling generic free-form 3D surfaces from expressive and sparse 2D sketches by using a Convolutional Neural Network to process the sketches. According to Li et al. (2018), sketching provides an intuitive user interface for communicating free-form shapes (Li, et al., 2018). Likewise, geologists use sketches to explain the observed complex features and their interpretations to other geologists. While humans can easily describe and communicate their observations in the form of a sketch, this process can be very challenging to replicate algorithmically. Therefore, a new method is necessary for a machine to improve the detection of these geological features/structures.

I anticipated that depicting the actual structure on a black-and-white sketch, without any background or intricate detail, combined with the real outcrop photos, would help the model learn the entire structure and thus predict it more accurately. Due to the lack of large geologic datasets, using sketches to depict geological features not only increased the training data size but also enhanced its variability and credibility by introducing some idealized examples of specific structures from which the model may improve its learning.

All the sketches were generated based on established geological sketches from the geological literature, such as publications, and books. The Data Augmentation techniques mentioned in section 4.5.3, Figure 4-8, were applied for both data types and were an essential part of the dataset's manipulation and pre-processing.

5.3 Proposed Workflow for the Classification of Geological Images



Figure 5-1: High-Level Geological Image Classification Workflow.

The proposed workflow presented in Figure 5-1 consists of five simplified steps. This workflow was utilised under different datasets and model backbones, as demonstrated in the following sections of this chapter. A brief explanation of each step of the workflow is shown below.

5.3.1 Dataset Selection

The proposed workflow's first step deals with dataset selection. The selection and formulation of the geologic datasets used in this chapter were previously described in Chapter 4, where all the necessary components comprising a geologic dataset were explained in detail. Data collection and preprocessing take place in this step to compose the geologic dataset. Datasets 1-5 were composed in order to stretch the model's learning ability of four sedimentary structures and, at the same time, examine the effect of a blended dataset on the model's learning and predictions. Datasets 6 and 7 were composed to test the model's learning ability of 17 sedimentary structures and 7 fossil types to examine the effect of a blended dataset on the model's learning and predictions.

5.3.2 Dataset Split

The second step involves splitting the dataset into training, validation, and test sets. The algorithm splits the dataset into 70, 20, and 10% (70-20-10), a typical split for the training, validation, and test sets. Images are randomly assigned to each of the subsets. These split proportions were consistent across datasets D1-D4. For D5, I used an 80-10-10 split because I wanted to test the performance of the model when having more outcrop examples in the training set. For Dataset 6, the proportions were changed to 65-26-9 on purpose to test the performance of the model when having fewer outcrop examples in the training set and slightly more in the validation set, always compared to the standard split 70-20-10. As for D7, the split proportions were 71-20-9.

5.3.3 Model Training

This third step of the workflow is crucial for setting up the model and tuning it accordingly, depending on the task at hand. I created a custom image classification model using the Pytorch framework (PyTorch, 2016) to classify the geological features, specifically sedimentary structures, and fossils, employing five different backbones.

The backbones used for this chapter, previously described in Chapter 3, are from the ResNet (He, et al., 2016) and VGG families (Simonyan & Zisserman, 2014), and more specifically, ResNet18, ResNet50, ResNet101, VGG16, and VGG19. The choice behind these 5 backbones was to make a comparative study and establish the best model for the task. It was important to test the validity of my conclusions across different model architectures to ensure that my results were consistently accurate.

Furthermore, at this step, the training parameters are adjusted to minimize the error and improve the accuracy of the model's predictions. The specific setup of the hyperparameters used for each backbone is shown in Table 5-2 and Table 5-3. The hyperparameters that affect the model's training process, in this case, are the learning rate, batch size, and the number of epochs.

To automatically stop the training, an EarlyStopping handler was used to monitor the training, and if no improvement in the model's accuracy and losses was observed after five consecutive epochs, then the training was terminated.

5.3.4 Feature Extraction

The fourth step of the workflow deals with extracting the features from the images. Technically, feature extraction is part of the training step. However, it can be differentiated because manually examining the feature map can provide additional information on whether the model is learning the desired patterns. For instance, if we consider the images in Figure 5-2, we can observe how the features are extracted after each convolution into the feature map. If the desired objects are not recognized, then changes should be made in the model and/or the training images. It is important to check the feature extraction during the production stage of the model, not every single time, but for this thesis, it was checked to ensure that the model is extracting the appropriate geological patterns from the images.

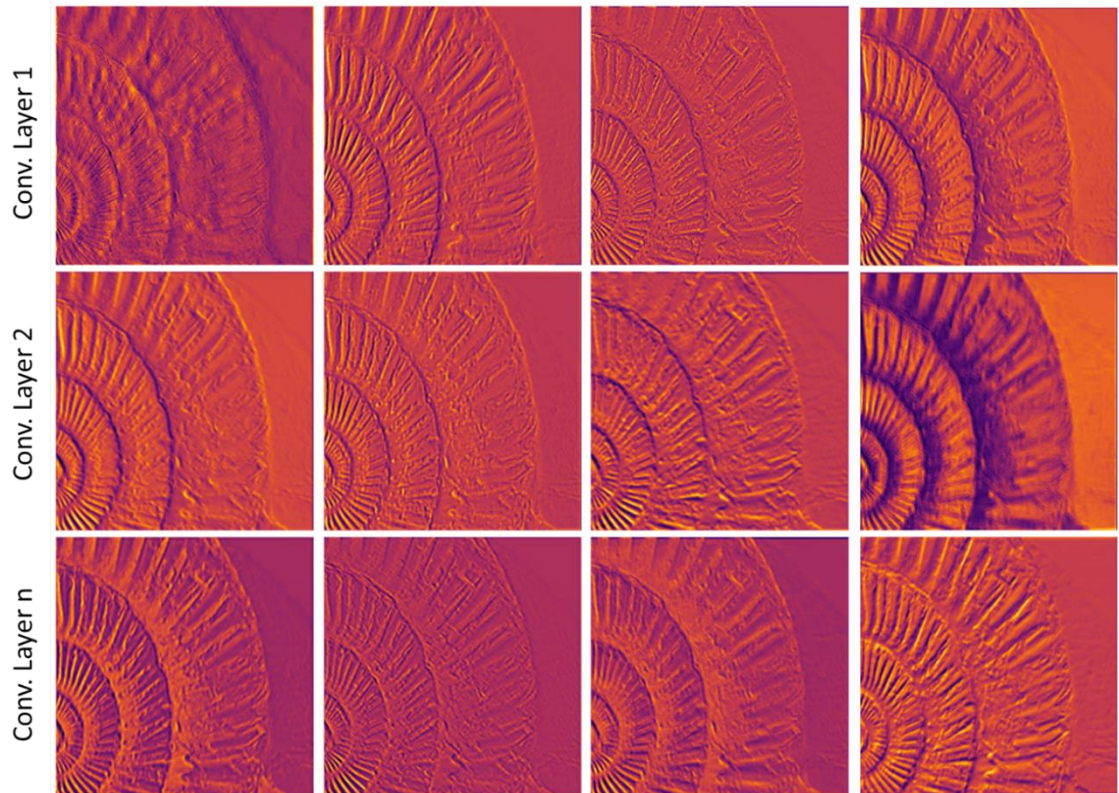


Figure 5-2: Feature extraction after each convolution layer.

In Figure 5-2, the extracted features from an ammonite image are shown progressively after 1, 2, and n number of convolutions. All four sample images in the third row show more detail in the pattern of the ammonite compared to the previous convolutions.

5.3.5 Model Testing and Evaluation

The workflow's last step involves the model's testing stage when the model is tested with unknown images and attempts to predict the class of the geologic feature present in each image. The model's performance is evaluated with the confusion matrix and the expertise of a human geologist.

5.4 The Custom Model for Geological Image Classification

The present section involves the building and testing of a custom image classification model using various datasets and backbones to examine the impact of incorporating sketches on model performance. The aim is to identify the optimal model setup, parameters, and combination of data to achieve the best possible performance. The results of the investigation are presented in two parts.

In the first part, the suitability of image classification as a method for extracting single sedimentary structures or fossils from outcrop/fossil images was initially established. The results of section 5.4.1 provided evidence that image classification is a useful technique, generating a list of potential predictions ranked according to their probabilities. Additionally, this part of the study helped to identify the most appropriate dataset and hyperparameters for the geologic classification task. To achieve this, the custom CNN model was trained using Datasets 1 - 5, assessing whether the inclusion of sketches in the training process impacted the model's predictions. Once this was established, the optimizer and learning rate were adjusted to determine the optimal values for the model configuration, including the ideal proportion of sketches and outcrops.

The second part of the study aimed to develop a robust image classifier for sedimentary structures and fossils based on the findings from Part One. In this context, a robust model refers to one that can accurately classify images under challenging conditions by recognizing more features, achieving higher accuracy in feature recognition, and handling noise more effectively. This is an essential consideration for image classification models since images in real-world scenarios are frequently subject to unpredictable and challenging conditions, such as lighting, noise, and image quality. To achieve this robustness, the custom image classification model was trained with Datasets 6 and 7, using five different backbones each time for each dataset to determine which backbone is the most appropriate. This resulted in the development of an image classifier that can accurately distinguish between seventeen sedimentary structures and seven fossil types.

5.4.1 Part One: Establish Method, Dataset, and Model Parameters

The custom model developed in this chapter was initially trained utilising the four distinct sedimentary structures available in Datasets 1-5, as previously detailed in Chapter 4. To assess the model's capability for image classification of outcrop images, it was imperative to verify that the model could make at least one accurate prediction for the given test data. Indeed, even a single correct prediction (Figure 5-3) is indicative that the model can be utilised for the task and optimized through modifications to its architecture and configuration, as well as through the selection of a more appropriate training and validation dataset.

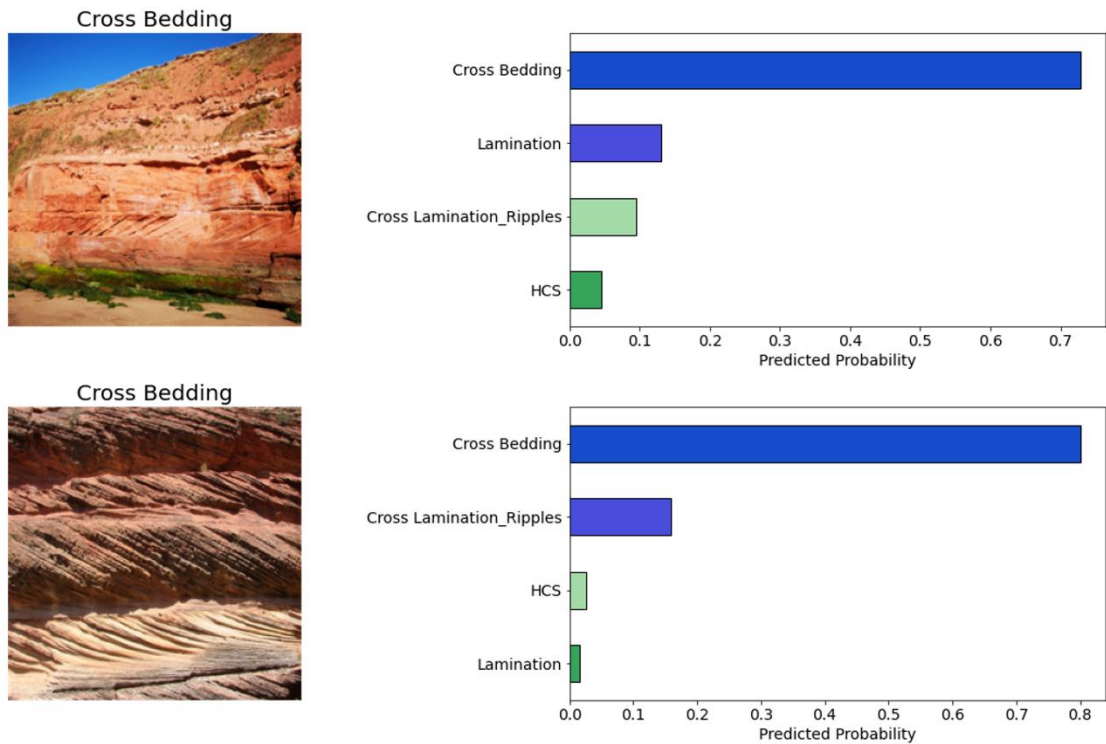


Figure 5-3: Example results of image classification model, trained from scratch, with the ResNet50 backbone.

In Figure 5-3, two individual predictions of the model are shown, both correct according to the ground truth, displayed at the top part of each image. The predictions on the right-hand side of the figure show the cumulative probability, which is equal to 1 and distributed across the predictions for the four classes in this example. According to the results, the number one prediction of the model is cross bedding, which is correct in both cases.

Nevertheless, examining the confusion matrix (Figure 5-4), which is the metric used to evaluate the image classification models of this chapter, as described in Chapters 2 and 3, it is clear that the model's accuracy for the other three classes is rather low, as it fails to correctly identify the other sedimentary structures.

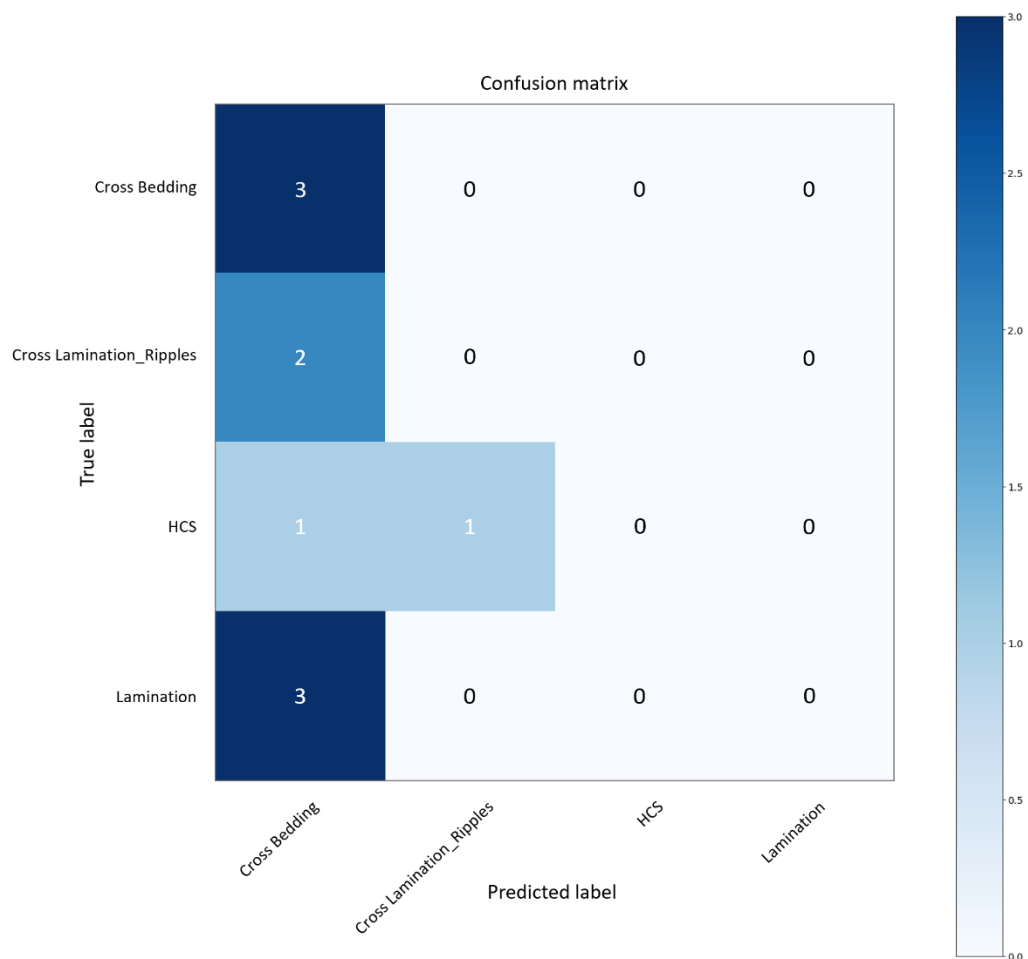


Figure 5-4: The confusion matrix corresponding to Figure 5-3.

The overall accuracy of the specific model, trained from scratch with Dataset 2, was 30%, which is very low. To reach these results, I trained my model with Dataset 2 from scratch by utilising the Resnet50 backbone. Training a deep learning model from scratch can be resource-intensive, prone to overfitting, and require a large dataset and hyperparameter optimization (Kornblith, et al., 2018). However, using a pre-trained model such as ResNet50 can save significant time, data, and computational resources compared to training a model from scratch (Kornblith, et al., 2018).

Therefore, I trained the exact same model with D2, but this time I used the pre-trained version of ResNet50 backbone. Some of the new results are shown in Figure 5-5.

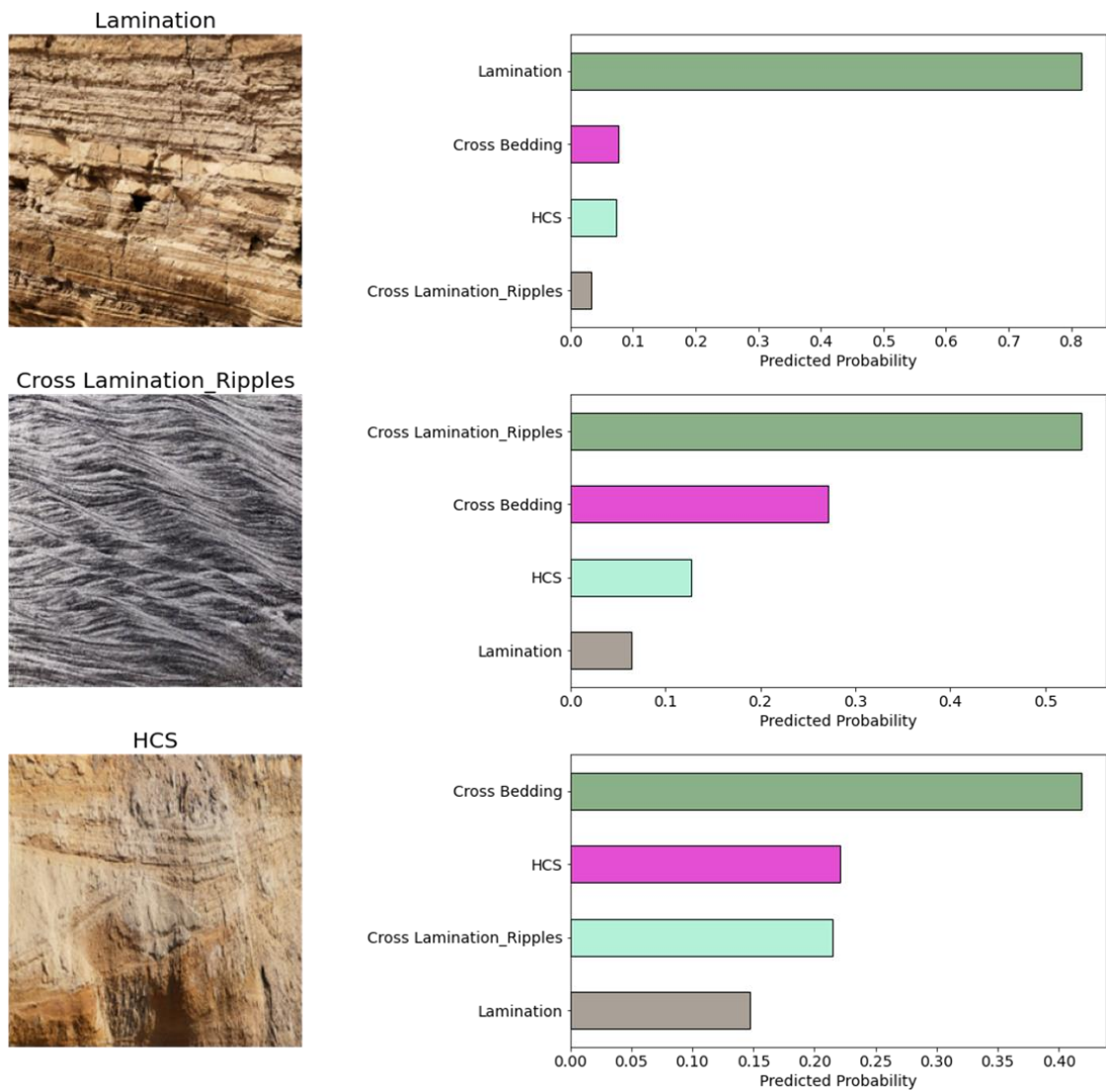


Figure 5-5: Example results of image classification model, trained with the pretrained version of ResNet50 backbone.

In Figure 5-5, three individual predictions of the model are shown, two of them being correct according to the ground truth, while HCS is misclassified as cross bedding. The confusion matrix of this model is shown in Figure 5-6 and demonstrates the predictions of the model for each class.

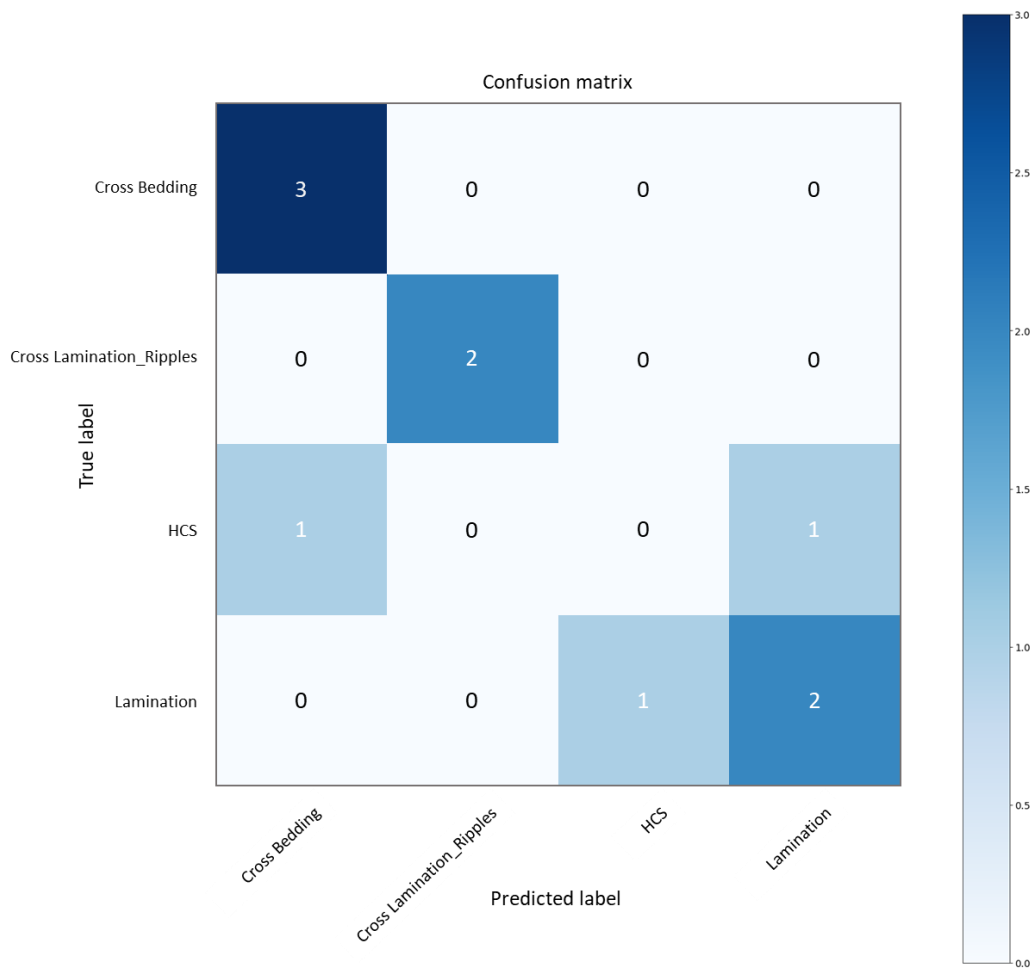


Figure 5-6: The confusion matrix corresponding to Figure 5-5.

The overall accuracy of the specific model, trained with pre-trained weights and with Dataset 2, was 70%, which is a significant improvement compared to the previous model.

Further tuning of the model and training dataset is implemented to improve the model’s predictions of the sedimentological features. To help the model learn the sedimentological features and thus predict them more accurately, black-and-white sketches, without any background or intricate detail, were blended with the real outcrop photos. This led to a series of experiments, testing the efficacy of this addition. I compared image classification with and without sketches. For this purpose, I applied my custom Image Classification model (using the pre-trained ResNet50 backbone) on Datasets D2 and D4 (Table 5-1) to make a clear comparison and demonstrate that when the model is trained with D4, the sedimentary structure classification accuracy improves by learning from the evidence and interpretation data (geological sketches).

	Dataset	Training Data Type	Test Data Type	Image Number	Number of Geological Features	Task
Part 1	Dataset 1 (D1)	Sketches	Sketches	77	4	Image Classification
	Dataset 2 (D2)	Outcrops	Outcrops	77	4	Image Classification
	Dataset 3 (D3)	Sketches	Outcrops	77	4	Image Classification
	Dataset 4 (D4)	Sketches + Outcrops	Outcrops	77	4	Image Classification
	Dataset 5 (D5)	Outcrops	Outcrops	111	4	Image Classification
Part 2	Dataset 6 (D6)	Outcrops	Outcrops	310	24	Image Classification
	Dataset 7 (D7)	Sketches + Outcrops	Outcrops	652	24	Image Classification

Table 5-1: Summary of the properties of Datasets 1-5(Part 1) & Datasets 6-7(Part 2).

To evaluate the performance of our custom image classification model, a comparison was made between its test accuracy for datasets that did and did not include sketches. Specifically, Dataset 3 (D3) presented a challenging task, as it exclusively consisted of sketches in the training data and required the model to predict complex outcrop photographs. When only sketches were utilised for training, the model's test accuracy was notably low, with numerous misclassifications resulting in a mere 30% accuracy rate.

Datasets 2, 3, 4, and 5 were subjected to identical testing conditions, including the same model, hyperparameters, and outcrop photographs for testing. Any observed improvement in the performance of D4, in comparison to the other datasets, can therefore be attributed solely to the inclusion of sketches in the training data.

The custom classification model, for this first part, only employed the ResNet50 backbone, which was utilised for each of the datasets, and the resulting classification accuracies were compared. Figure 5-7 presents the results of the model's performance when trained with Datasets 1-5 (Table 5-1, Part 1).

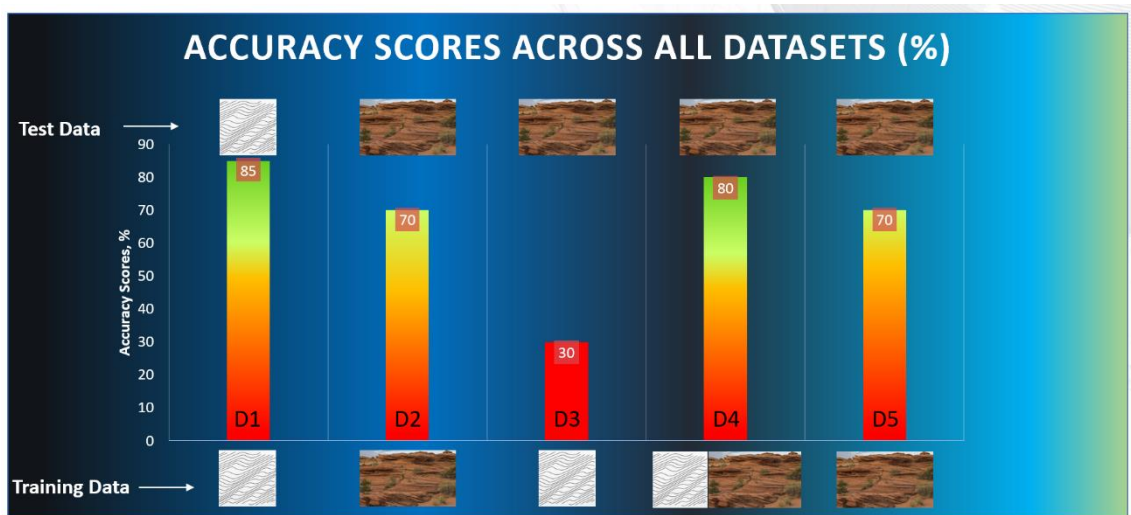


Figure 5-7: Summary of all the prediction accuracies across the five datasets throughout the experiment.

The datasets of primary interest in this study are D2, D4, and D5. D4 contains both sketches and outcrops in the training set and predicts only on outcrops, while D2 and D5 have only outcrops in both the training and test sets. The objective is to demonstrate that D4 consistently produces more accurate predictions compared to D2 and D5. Even when D4, the blended dataset consisting of 77 images and sketches, is compared to D5, which contains 111 images, the model trained with D4 still performs better than D2 and D5, regardless of the number of images.

In fact, D4 has the highest accuracy rate of 80%, while D2 and D5 have an accuracy rate of 70%. This finding shows that using a blend of training data leads to a 10% increase in accuracy. Therefore, the results suggest that using a blended dataset such as D4 can lead to more accurate predictions of sedimentary structures in outcrops compared to using only outcrop images in the training and test sets.

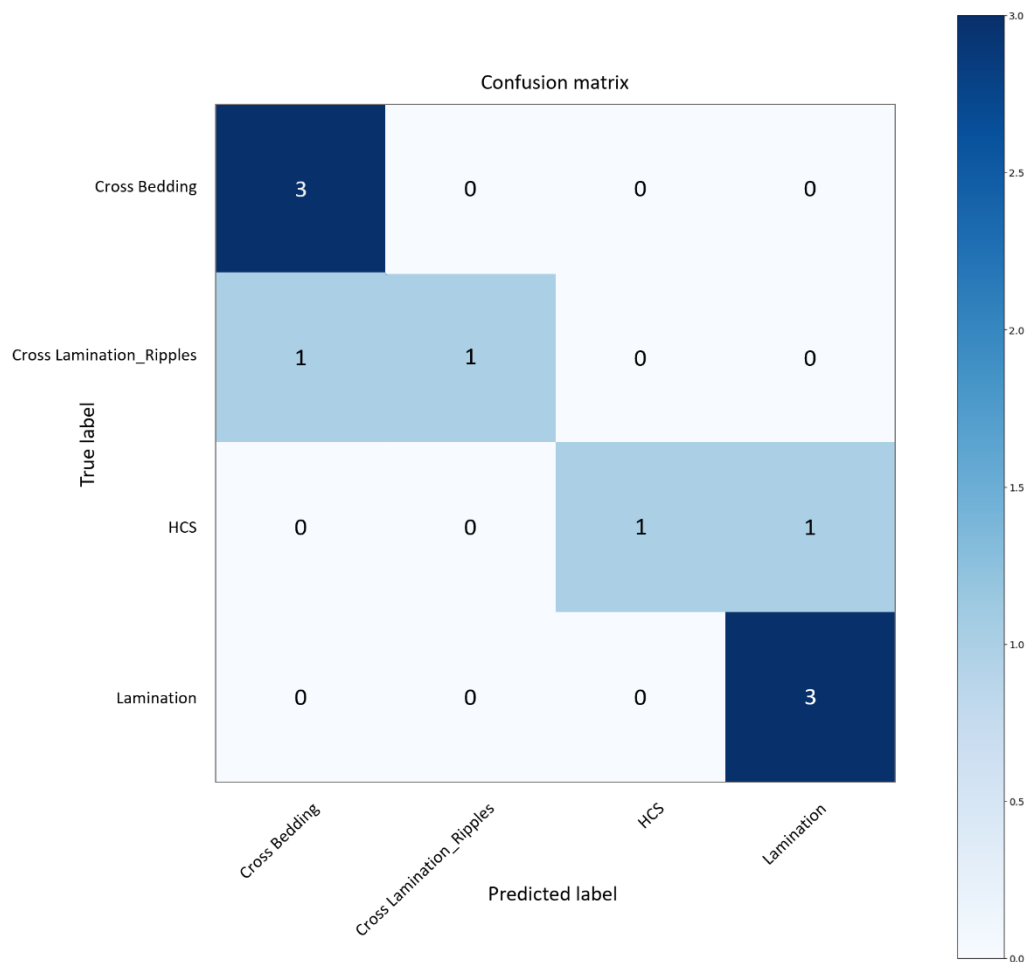


Figure 5-8: Confusion matrix of the model trained with Dataset 4.

Figure 5-8 shows the confusion matrix of the model trained with Dataset 4 and demonstrates the predictions of the model for each class.

Figure 5-9 shows the application of the custom image classification model on an image showing a Hummocky Cross Stratification (ground truth). The model was trained separately on D2 and D4 and was tasked to predict the prominent sedimentary structure present in the test outcrop image. The results of the model for the test image in Figure 5-9, first display the top 1 prediction of the model (HCS) both when the model is trained with D2 (blue) and D4 (red) followed by the next possible predictions, cross-lamination and cross-bedding.

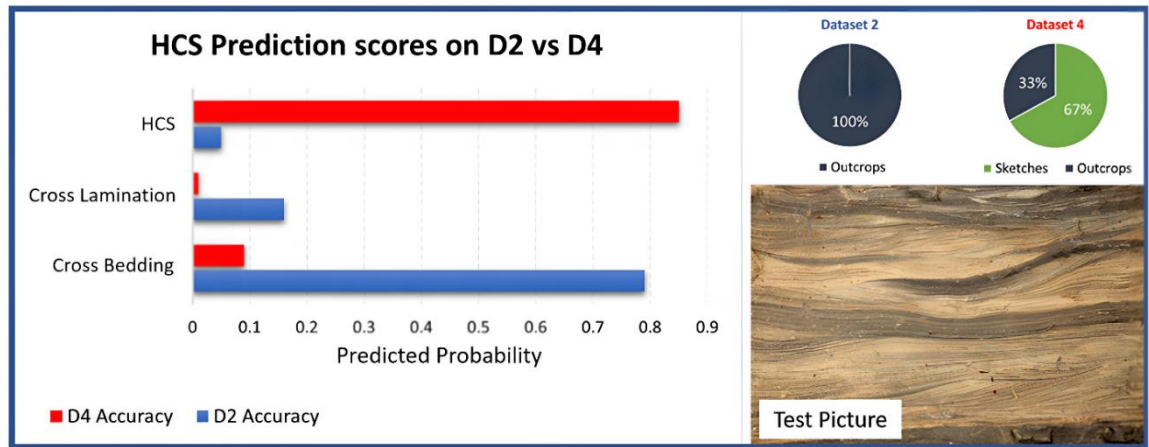


Figure 5-9: Model's HCS Prediction scores on D2 vs. D4 (backbone ResNet50) when trained on outcrops vs. when trained with a blended dataset (outcrops + sketches).

For Dataset 2, when the model was trained only with outcrops, the prediction of Hummocky Cross Stratification (HCS) was accurate only to 5%. However, when sketches were blended with the outcrop pictures on the training set, the accuracy was improved by 80% for the same test image, indicating that sketches helped the model learn the structure better and improved its prediction for the target class. The difference between the two datasets lies in the HCS class, in which Dataset 4 performed better.

In most of the tested images, not displayed here, cross-bedding and cross-lamination showed similar prediction scores and probabilities, as the model cannot distinguish well the difference between the two structures. This happens because, at this stage, the model does not incorporate the scale of the features, which is a significant factor in differentiating between these two structures. The results of Figure 5-9 triggered the additional question of finding the optimal balance of sketches and photographs in the training set.

5.4.1.1 Experimenting with the Dataset's Proportions

This experiment aims to investigate how the distribution of images and sketches in the training data affects classification performance. As evidenced in Part 1, when utilising a small geologic dataset comprising outcrop images and geological sketches, the trained CNN model performs better when exposed to a high-quality dataset, incorporating geological sketches into the training image data. Various combinations of photos and sketches were randomly selected for the training sets, resulting in greater variability in

the sketches-to-outcrops ratio. This led to further experimentation with the number of sketches versus the number of existing outcrops in Dataset 4. The first element that required further investigation was the optimal proportion of sketches and outcrops within the dataset. In the initial training set of D4, the proportion of the sketches (67%) and outcrops (33%) was considered the benchmark case for the rest of this experiment. Further experimentation and testing were necessary to support and validate the findings of Part One (5.4.1). Thus, 11 subsets of Dataset 4 were generated, containing sketches and outcrops in the training set, starting from 0% sketches up to 100%, varying it every time by 10% (Figure 5-10). The classification model presented in Figure 5-9 was applied to each case and compared the accuracies.

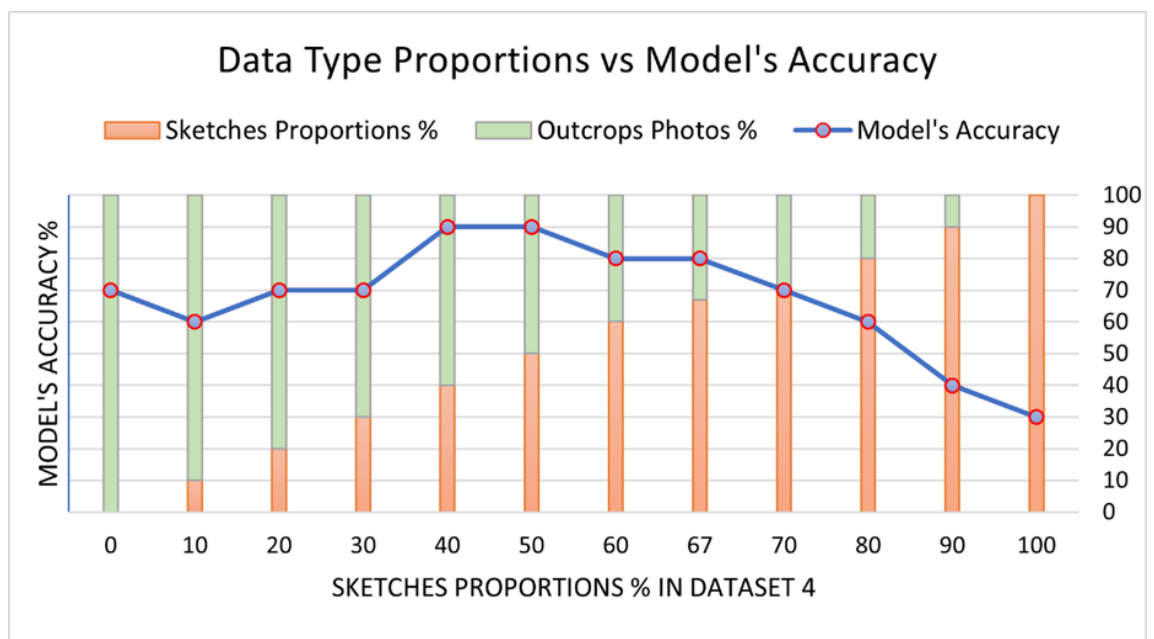


Figure 5-10: Data type proportions versus the model's accuracy when trained with Dataset 4.

Figure 5-10 shows how the model's accuracy fluctuated as the blended data types' proportions varied in the training data. The plot clearly shows how the model's accuracy on the test data deteriorates dramatically when only sketches are used for training. On the contrary, the test accuracy drops slightly, with only images being used for training. The plot highlights the optimal balance between sketches and images around 40-50%, in which range the model achieved its highest test accuracy of 90%. All the scores indicate that including sketches in the training data set enhanced the dataset's quality when blended in the right proportions with the natural structure pictures/outcrop photos (ranging from 40-67% sketches). In addition, the proportions of the images and sketches

were changed for every class and amid the training, validation, and test sets. After repeating the first cycle on the randomly selected data, the results remained almost constant (marginal difference of 1%), showing that the model's accuracy and performance do not heavily depend on selected 'good' images as it performed well overall, regardless of the use of ideal pictures or not. Incorporating a degree of semi-controlled randomness in the data shuffling and the variation within the training, validation, and test sets solidifies the model's robustness. The degree of randomness might need to be constrained, as adding any random and potentially too complicated or very simple picture might not increase the accuracy of the predictions. Even though Dataset 5 (D5) was an enriched version of Dataset 2 (D2) with more outcrop images in the training and validation data, running the classification model on both D2 and D5 yielded identical test accuracy of 70%. This supported the hypothesis that adding outcrops requires more sketches to maintain the optimal proportions for the lowest classification error and vice versa.

The best accuracy (90%) is achieved when the training set consists of 40% sketches and 60% outcrops. All the scores indicate that the training set enhances its quality only when sketches are blended in the right proportions with the natural structure pictures/outcrops (ranging from 40-67%). The breakdown per sedimentological feature is shown in Figure 5-11 showcasing the confusion matrix for this particular blend proportions. According to this confusion matrix, for the same test set, consisting of 10 images, and used for all the results up to this point, there is only one misclassification of the HCS sedimentary structure.

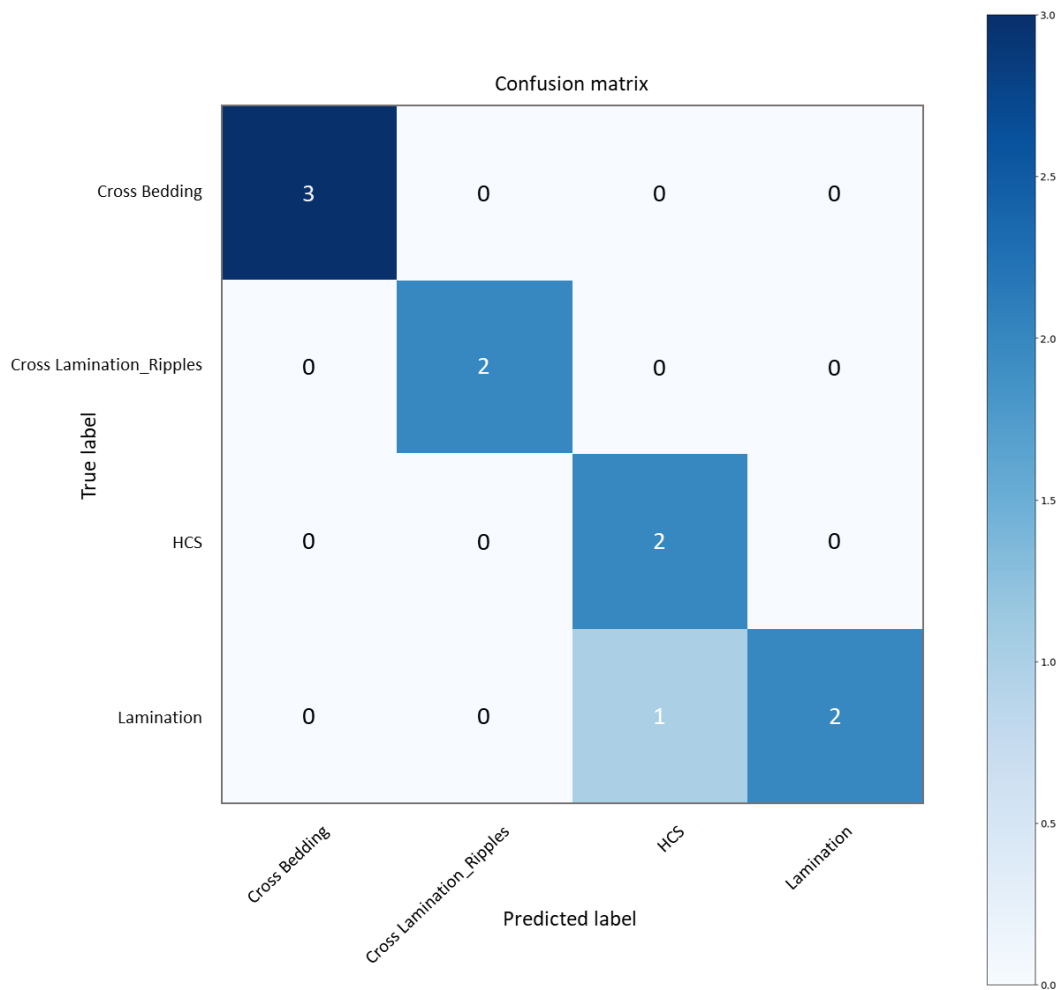


Figure 5-11: Confusion matrix of the model trained with Dataset 4 (40% sketches and 60% outcrops).

5.4.1.2 Experimenting with the Model's hyperparameters

Fine-tuning a pre-trained model can be an effective strategy for adapting a model to our specific task while still leveraging the knowledge gained from a large-scale training dataset. Once the optimal proportions of sketches and outcrops within the datasets were established, the model's performance and accuracy were examined when two of the most critical parameters, the optimizer and learning rate, were modified. Optimizers update the weight parameters to minimize the loss function. The loss function guides the terrain, telling the optimizer if it is moving in the right direction to reach the bottom of the valley (the global minimum). The learning rate is a hyperparameter that controls the adjustment of the network weights with respect to the loss gradient. The learning rate affects how quickly a model can converge to a local minimum (Goodfellow, et al., 2016). Thus,

finding the sweet spot (usually through trial and error) from the start would lead to faster training times for the model.

According to (Lowndes, 2015), a good learning rate can be identified from the trend of the model's loss versus the number of epochs, as shown in Figure 5-12.

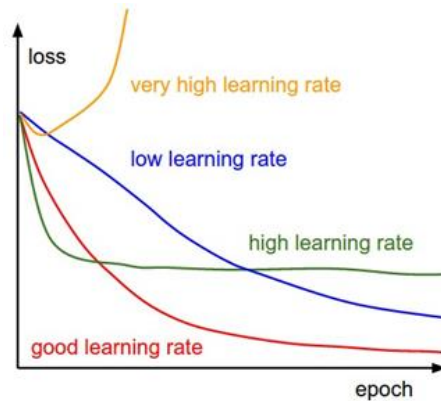


Figure 5-12: Learning rate variations based on the trends of the model's loss vs. the number of epochs (Lowndes, 2015).

Two different optimizers, Adam (Adaptive Moment Estimation) and Adabound (Luo, et al., 2019), were tested along with a short range of different learning rates (0.0001, 0.001, 0.01). Typically, the faster the model converges (very high learning rate), the poorer its accuracy and predictions. Figure 5-13 shows six optimizers and how the model's prediction accuracy evolves over the number of epochs according to the chosen optimizer. Of course, as discussed previously, besides the model and its hyperparameter choices, the dataset the model is trained with is the primary factor determining its performance.

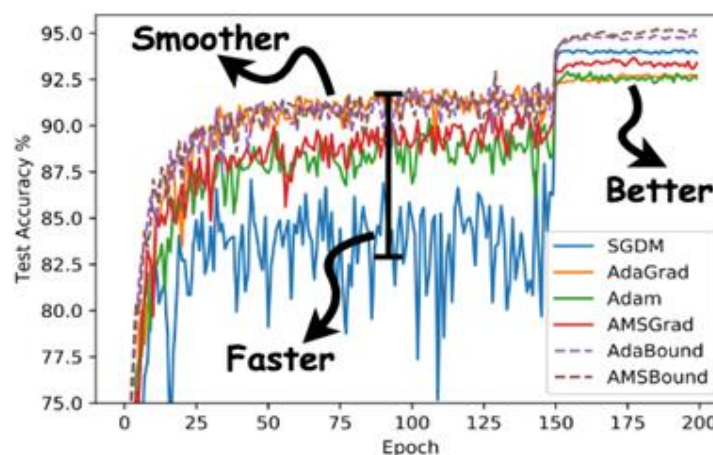


Figure 5-13: Optimiser Scenarios vs. Test Accuracy (Luo, et al., 2019).

Adam is an optimization algorithm commonly used in machine learning to optimize the parameters of a neural network (Kingma & Ba, 2014). It is an extension of the stochastic gradient descent (SGD) algorithm that incorporates elements of both momentum and adaptive learning rates (Luo, et al., 2019). This optimizer computes the adaptive learning rate for each parameter by computing the weighted moving average of the squared gradients and the moving average of the gradients themselves. These moving averages calculate an update for each parameter, taking into account the gradient's magnitude, direction, and variance. The main advantages of the Adam optimizer are that it requires little memory, is computationally efficient, and is effective for a wide range of deep-learning tasks.

AdaBound is a variant of the Adam optimizer that introduces dynamic bounds on the learning rates (Luo, et al., 2019). The basic idea behind AdaBound is to introduce two hyperparameters, lower-bound and upper-bound, that constrain the learning rate during the optimization process. The learning rate is dynamically adjusted between these bounds based on the progress of the optimization.

During the early stages of optimization, AdaBound behaves like the Adam optimizer, smoothly adjusting the learning rate to match the scale of the gradients. However, as the optimization progresses, the learning rate gradually decreases and becomes more tightly bound to the lower-bound and upper-bound hyperparameters. This helps prevent the learning rate from becoming too large and causing the optimization to diverge (Luo, et al., 2019).

Through trial and error, exercising different combinations between the two optimizers and the three learning rates, it was found that the best optimizer was Adam, with a learning rate of 0.001, which happens to be the typical (default) learning rate used with the Adam optimizer for classification tasks.

5.4.2 Part Two: Build a robust geological image classifier

The experimentations with the model's hyperparameters and optimal proportion of sketches and outcrop images provided enough information to enable the compilation of a richer dataset and a fine-tuned model suitable for classifying various sedimentary structures and fossils. While Part One demonstrated how the model predicts four sedimentary structures, part two shows the successful application of the custom, fine-

tuned model over 24 classes, 17 sedimentary structures, and 7 fossil types. In Part One, a comparison was made between the blended and outcrop datasets using five different datasets. In Part Two, only two datasets, D6 and D7, are utilised and are fully described in Chapter 4. These enriched datasets are tested against five different backbones to solidify the hypothesis that blended datasets consistently improve the quality of the model's learning, as shown in Chapter 5, section 5.4.1, and train a more robust geological classifier.

5.4.2.1 Application of the Fine-tuned model trained on Dataset 6 (outcrops)

In this experiment, we will compare how different CNN architectures (backbones) affect the classification of the geological features. We will consider different CNN training architectures with the same framework of experiments, first applying the different backbones on Dataset 6 and then on Dataset 7 to compare the results. For consistency in the experiment flow and comparative purposes, the models were first applied to D6, including only outcrops, and then to D7, including the blended data.

In this sub-section, the results of the fine-tuned model are presented when the model is trained purely on outcrop examples with Dataset 6. As described in Chapter 3, five different backbones were used in increasing order of complexity, meaning that each backbone has more parameters (model depth) compared to its predecessors. Specifically, the backbones used were from the ResNet family, ResNet 18, 50, and 101, and the VGG family, the VGG16 and 19. All the training hyperparameters used for the models' training are found in Table 5-2.

Training Hyperparameters	Value	Value	Value	Value	Value
Pretrained weights	ResNet18.pth	ResNet50.pth	ResNet101.pth	VGG16.pth	VGG19.pth
Image size	512	512	512	512	512
Batch size	16	16	16	16	13
Epochs	70	28	16	6	12
workers	4	4	4	4	4
Evaluation interval	1	1	1	1	1
Gpu count	1	1	1	1	1
Optimizer	ADAM	ADAM	ADAM	ADAM	ADAM
Learning rate	0.001	0.001	0.001	0.001	0.001
Total Parameters	11,438,487	24,038,744	43,030,872	135,315,544	140,625,240
Training Parameters	530,455	530,712	530,712	1,055,000	1,055,000
Position Augmentation	Value	Value	Value	Value	Value
Rotation	± 10 degrees	± 10 degrees	± 10 degrees	± 10 degrees	± 10 degrees
Horizontal Flip	Yes	Yes	Yes	Yes	Yes

Table 5-2: Model's training hyperparameters when trained on Dataset 6.

The image size, batch size, and number of workers were adjusted depending on the available computing resources. The number of epochs was set to 100 for all the models. However, in the code, an additional snippet was added to stop the training when the model could not learn more information. Regardless of the chosen backbone, all five training sessions ended their training before 100 epochs. The optimizer and learning rates were chosen and kept constant of the values established in Part 1. Lastly, to enhance the size and variability of the dataset without distorting the geological features and their meaning, two position augmentation techniques were used for each image: rotation by plus or minus 10 degrees and horizontal flip.

Figure 5-14 illustrates the model's accuracy and loss during the training and validation steps. These scores were calculated for each model separately, and they were grouped into two rows, with the top row showing the losses and the bottom row showing the accuracies of each model.

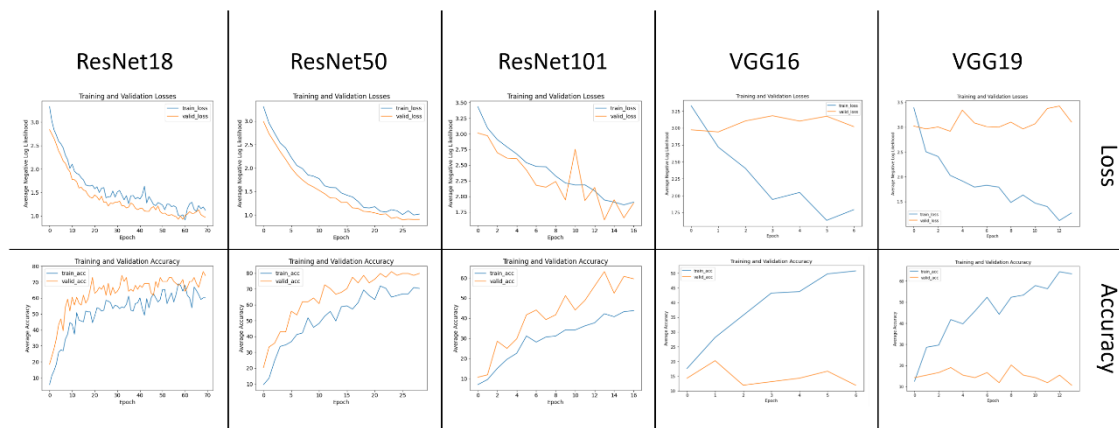


Figure 5-14: Loss and Accuracy versus the number of epochs for each model.

The training loss is used to gauge how well a deep learning model fits the training data. It evaluates the error of the model on the training set. Computationally, the training and validation losses are measured after each epoch and calculated by taking the sum of errors for each instance in the training and validation sets, respectively. It provides information on whether the model needs further tuning or adjustments.

A high loss value usually indicates the model is producing incorrect results, while a low loss score implies fewer errors in the model. The cost function, used to calculate the loss, is chosen according to the problem being solved and the data being used. In this case, categorical cross-entropy (Zhang & Sabuncu, 2018) was used as it is one of the most common loss functions for multi-label classification problems.

For the ResNet 18, 50, and 101 models, the couples training and validation losses and train and validation accuracy follow the same trend; they are close together, indicating a good fit of the model to the data, with no over-fitting or under-fitting occurring. On the contrary, VGG 16 and 19 show a great separation of the curve couples, both for the loss and accuracy, implying underfitting. The high values of the validation losses and the low values of the validation accuracies support this observation.

The training accuracy measures how well the model fits the training data and can be calculated by comparing the predicted outputs with the actual outputs of the training data. The validation accuracy measures how accurately the model can predict the outputs of the validation data. Both training and validation accuracies are essential metrics to evaluate the performance of a machine learning model. A high training accuracy with a low validation accuracy may indicate overfitting, where the model has memorized the

training data instead of learning to generalize to new data. Conversely, a low training accuracy with a low validation accuracy may indicate underfitting, where the model has yet to learn the underlying patterns in the data.

To make the results more readable, please refer to Figure 5-15, which shows an example of what each element in the figure means. An important note here is that the model was advised to display only the top six predictions for each test image by assigning a probability to each one. The model is trained with 24 labels, meaning for a test image, the model predicts how possible it is for the image to belong to every class it is trained with; in other words, the cumulative probability is equal to 1 and distributed across the predictions for 24 classes.

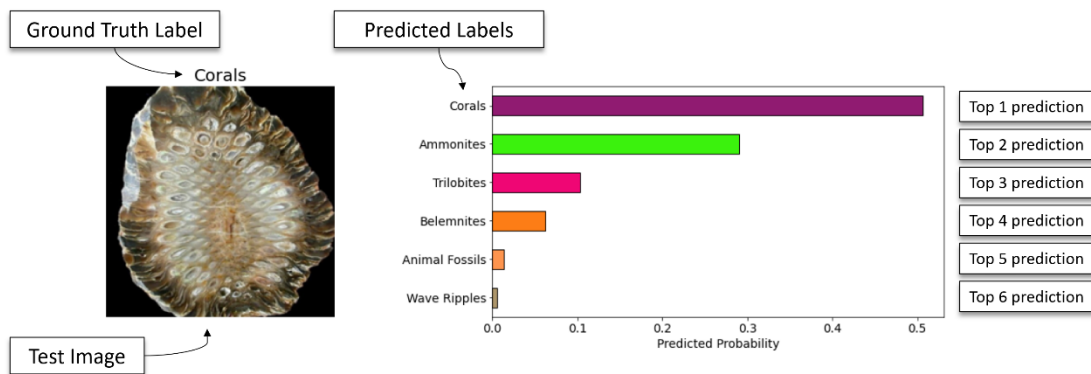
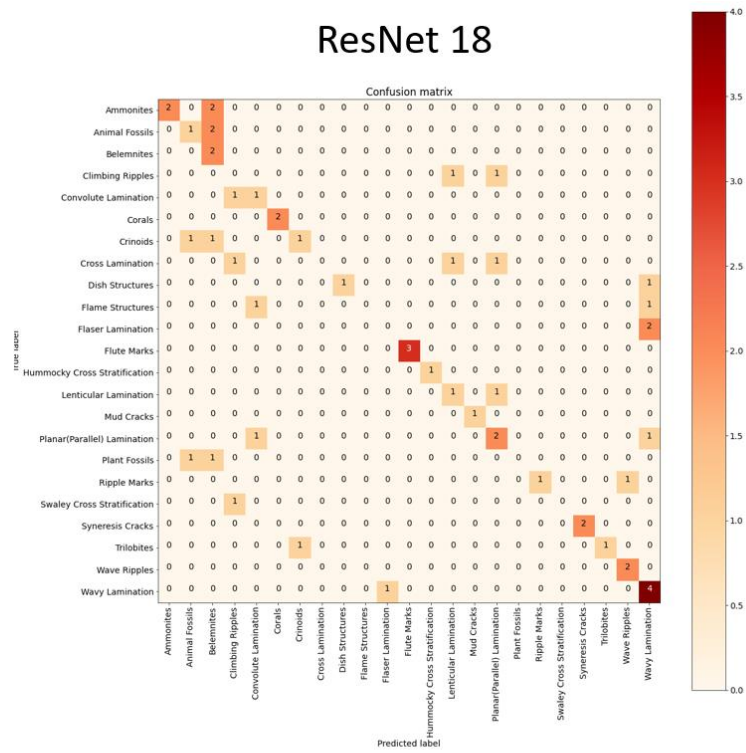


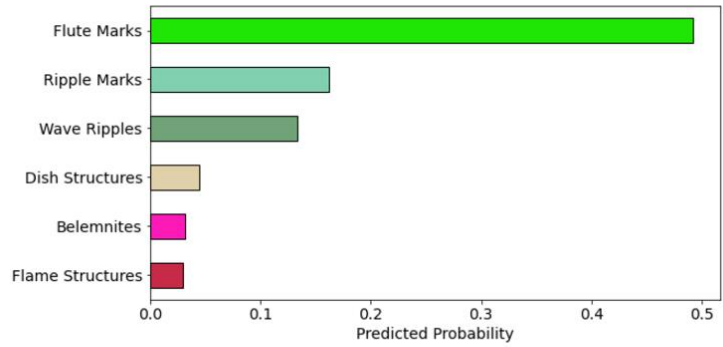
Figure 5-15: Example figure explaining how to interpret the results.

Figure 5-16 through Figure 5-20 show the results of each model trained with Dataset 6 on two random examples from the test set. In addition to the two test images and their corresponding predictions by the model, the confusion matrix for each model is included to provide an idea of the model's overall accuracy across the entire test set.

ResNet 18



Flute Marks



Lenticular Lamination

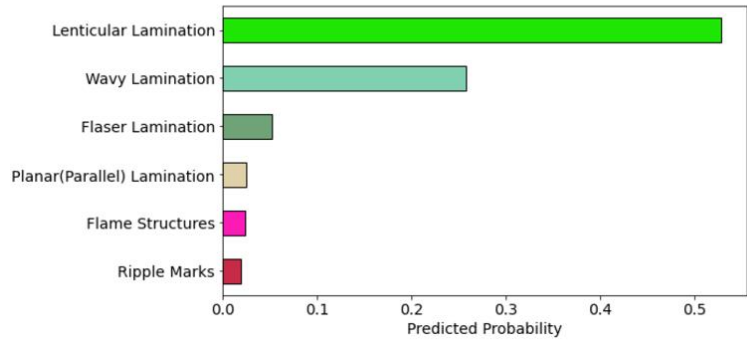
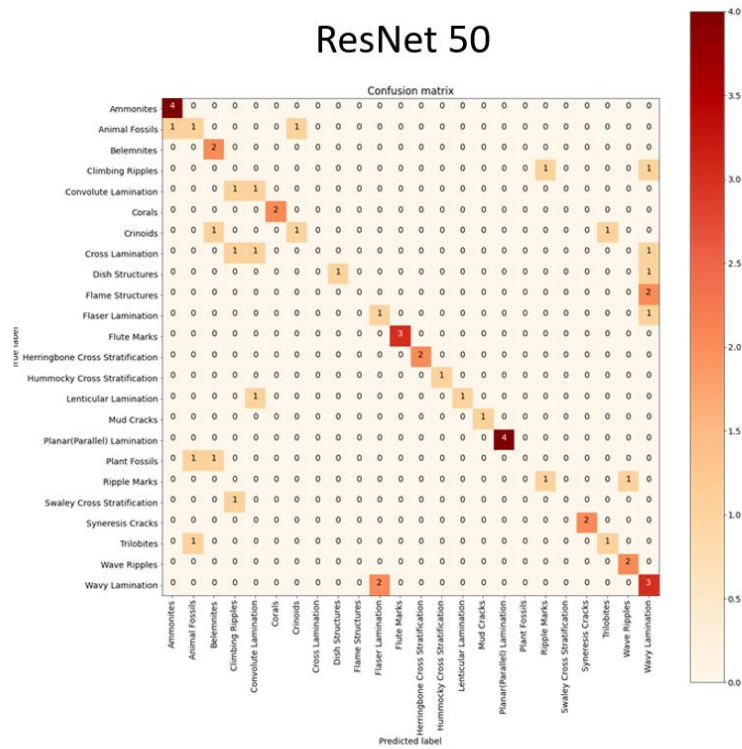
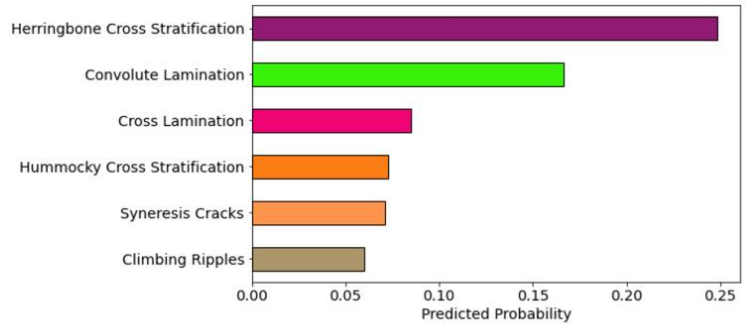


Figure 5-16: ResNet 18 (trained with D6) image classification predictions.

ResNet 50



Herringbone Cross Stratification



Hummocky Cross Stratification

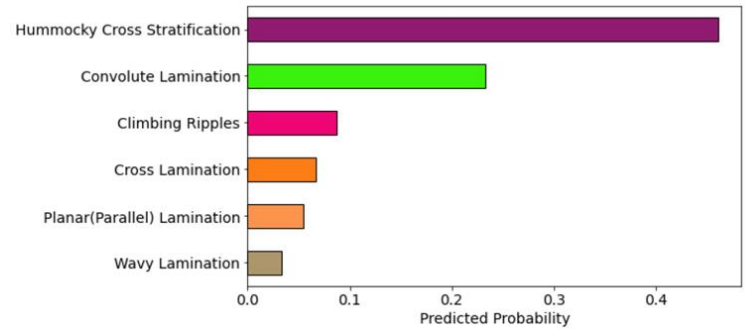
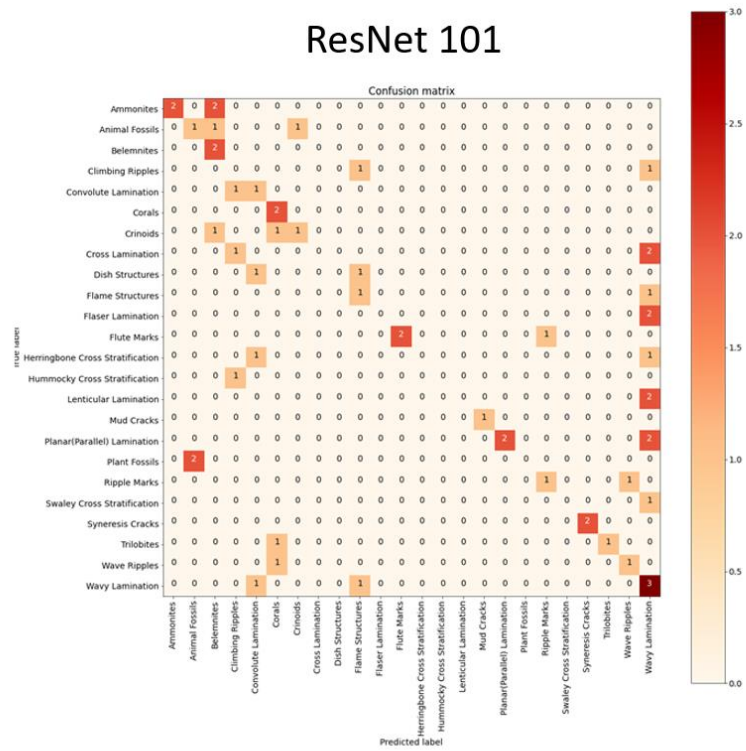
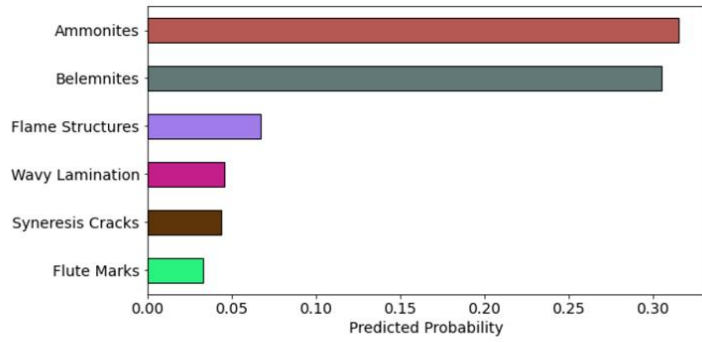


Figure 5-17: ResNet 50 (trained with D6) image classification predictions.

ResNet 101



Ammonites



Convolute Lamination

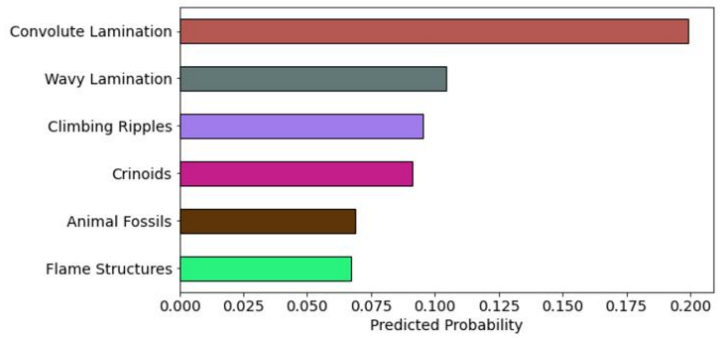


Figure 5-18: ResNet 101 (trained with D6) image classification predictions.

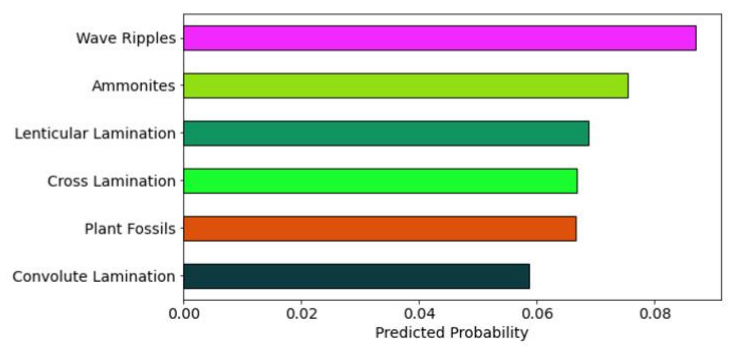
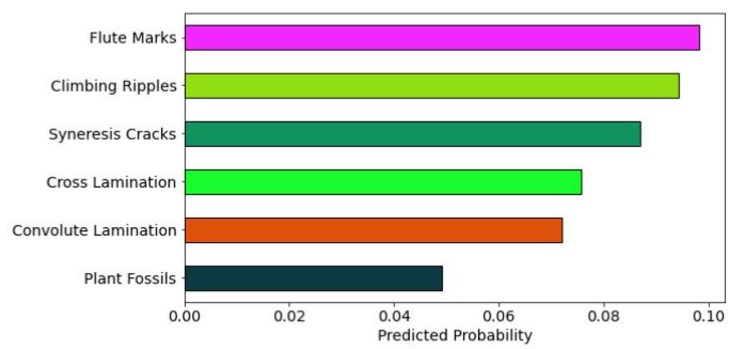
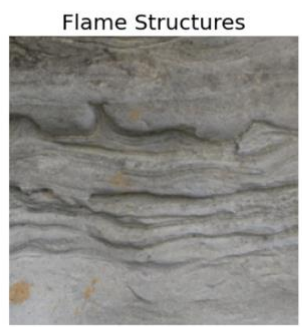
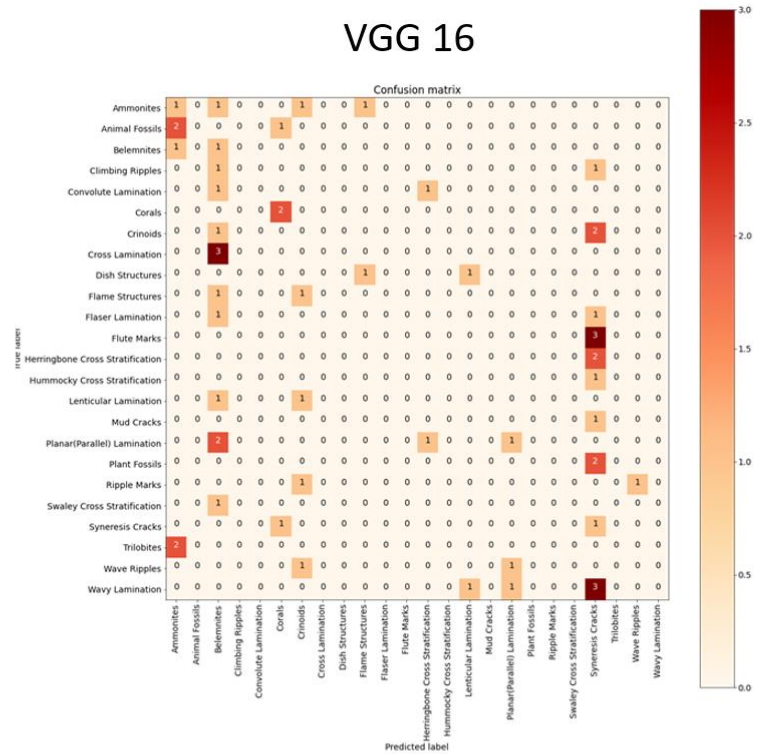


Figure 5-19: VGG 16 (trained with D6) image classification predictions.

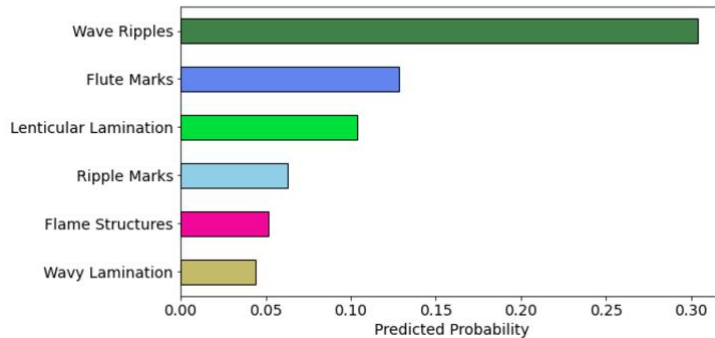
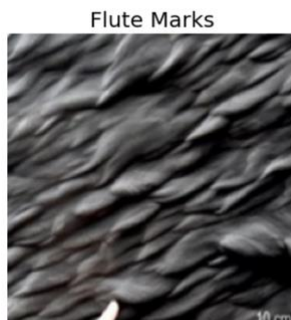
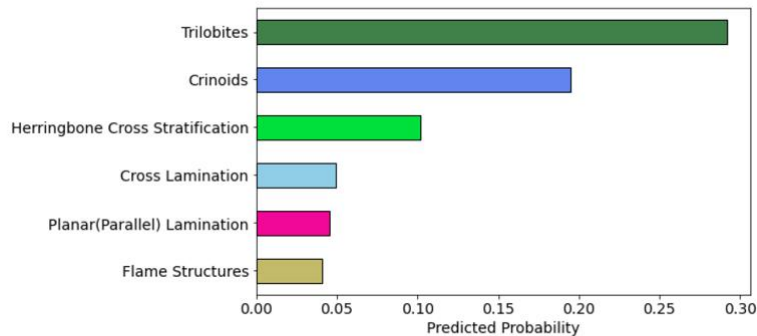
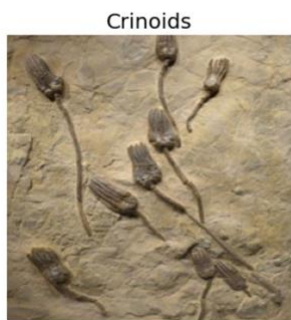
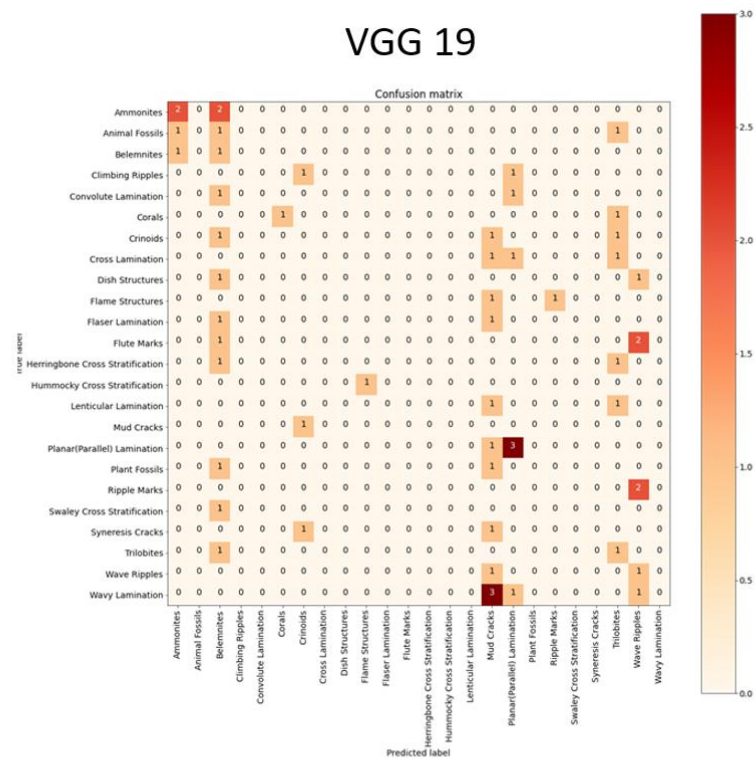


Figure 5-20: VGG 19 (trained with D6) image classification predictions.

The first observation made from Figure 5-16 through Figure 5-20 is that as the number of parameters and depth of the model's architecture increases, the worse the predictions become. For instance, looking only at the confusion matrices for each model, it is evident that as the number of parameters increases, the class predictions become sparser and move

away from the diagonal of the confusion matrix, which is considered the ideal (most accurate predictions). The confusion matrices in Figure 5-16 and Figure 5-17 are closer to that ideal scenario, as most of the predictions are either on top of the diagonal of the confusion matrix or close to it. As for the other three models, their predictions progressively become sparser. A reasonable explanation could be that there is a big difference between the total number of parameters and the trainable parameters per model (shown in Table 5-2). In Table 5-2, across all five models, the trainable parameters are about 530,712 for the ResNet and 1,055,000 for the VGG. These numbers depend on the number of features extracted from the dataset with which the model is trained. Since the trainable parameters are constant for each model family, and the number of the total parameters increases, the gap between the trainable and total parameters continuously widens, resulting in worse model performance.

The second conclusion derived from these results is that for the majority of results, the model's top 1 to top 6 predictions display almost equal probabilities, indicating that, in most cases, the model is not quite sure about its predictions. For instance, considering Figure 5-19 (top test image), the top one prediction is the Flute Mark class, with about 0.1 probability, which does not match the ground truth label. On top of that, the following top 2-5 predictions all show the wrong labels and a probability between 0.05-0.08, which is close to the top prediction, indicating the model thinks all these scenarios are almost equally possible.

This is precisely where the addition of sketches can emerge and improve the quality of the model's predictions by making them more accurate and certain. The results supporting that statement are found in the next section, 5.4.2.2.

Figure 5-21 shows a comparison of the overall test accuracy across the five different models. The winner is the ResNet 50 model with a 61% accuracy against the particular test set and when trained with Dataset 6. The second-best model proved to be ResNet 18, with a test accuracy of 52%.

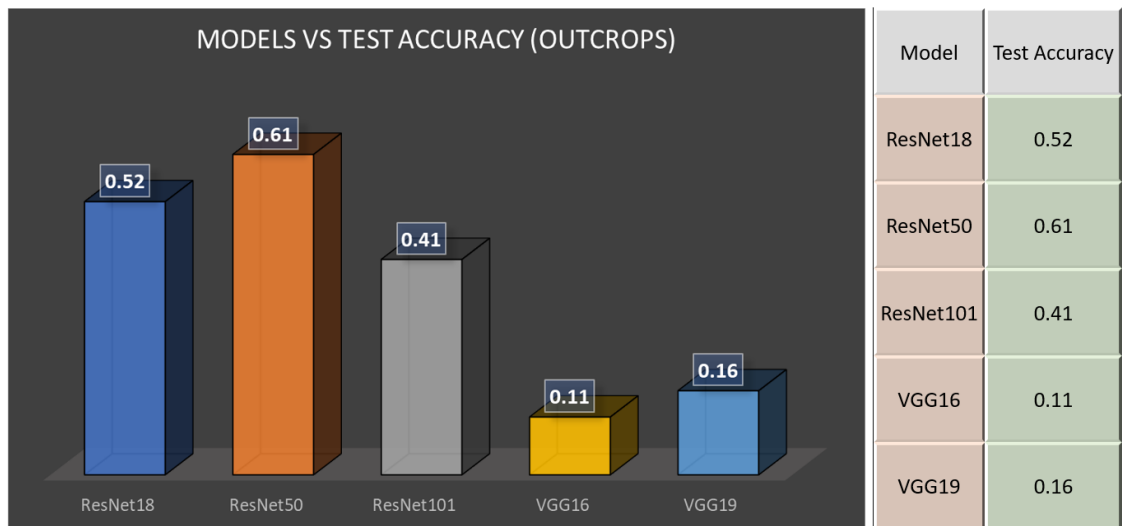


Figure 5-21: Overall test accuracy across the five different models.

The comparative study between the five backbones concluded that when using only outcrops, the highest accuracy was 61% which was achieved by ResNet50.

Finally, Figure 5-22 and Figure 5-23 visually demonstrate the model’s accuracy and loss per class for the test set for each of the five models. The custom image classification model, with all the different backbones, was tested on the same test images.

For Figure 5-22, in the case of flaser lamination, flame, and dish structures, the displayed backbones were the only successful ones for making a prediction. In the case of Swaley cross stratification, cross-lamination, plant fossils, and climbing ripples, all models failed to make a prediction. Generally, where certain backbones do not show accuracy scores, it means that no prediction was made.

For Figure 5-23, it is obvious that VGG 16 and VGG 19 show the highest loss, meaning that they have the worst accuracy, while ResNet18 and ResNet50 demonstrate lower loss values and therefore have the best accuracy.

Model Accuracy Per Class (Outcrops)

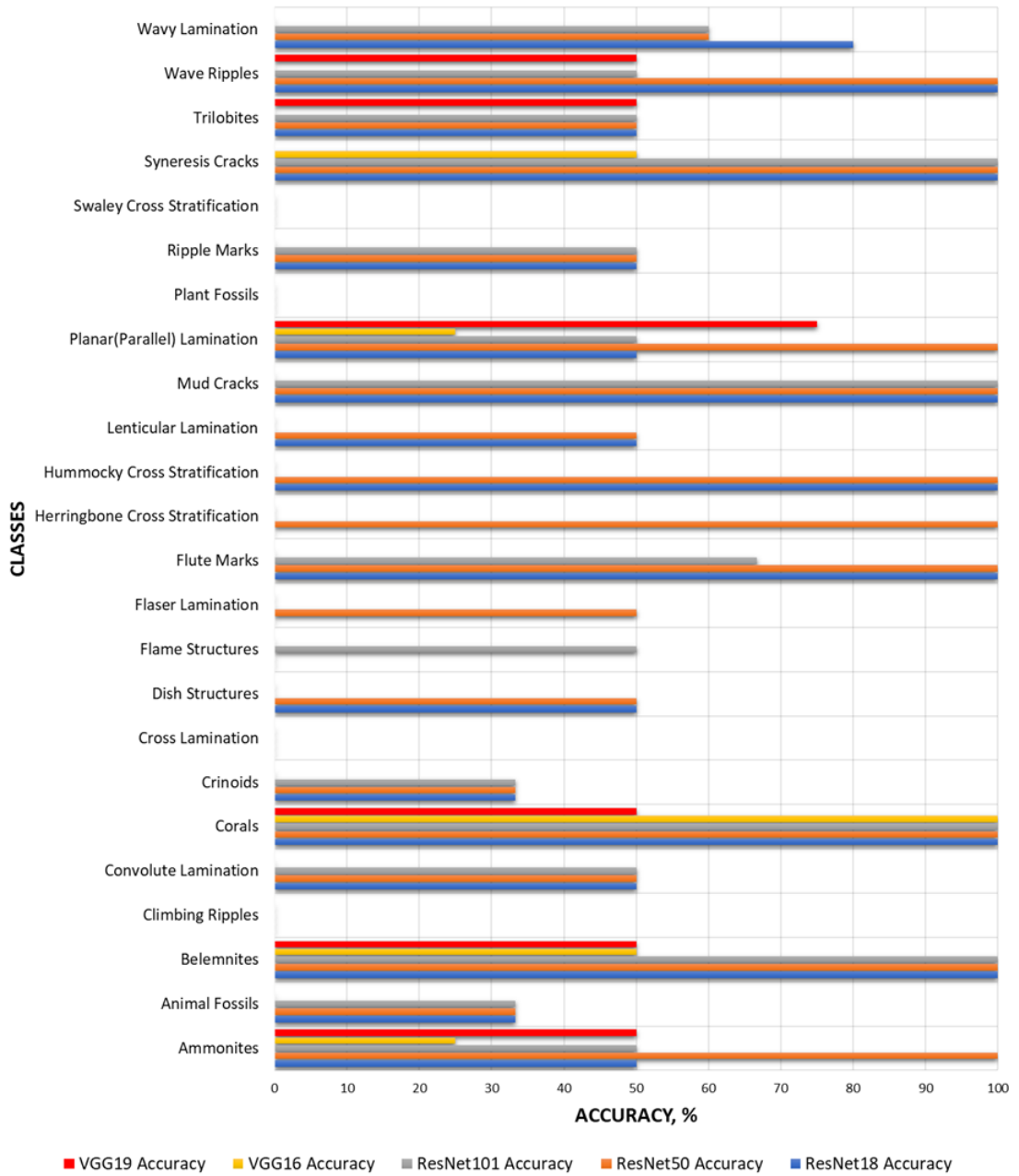


Figure 5-22: Models Accuracy per class across the five models.

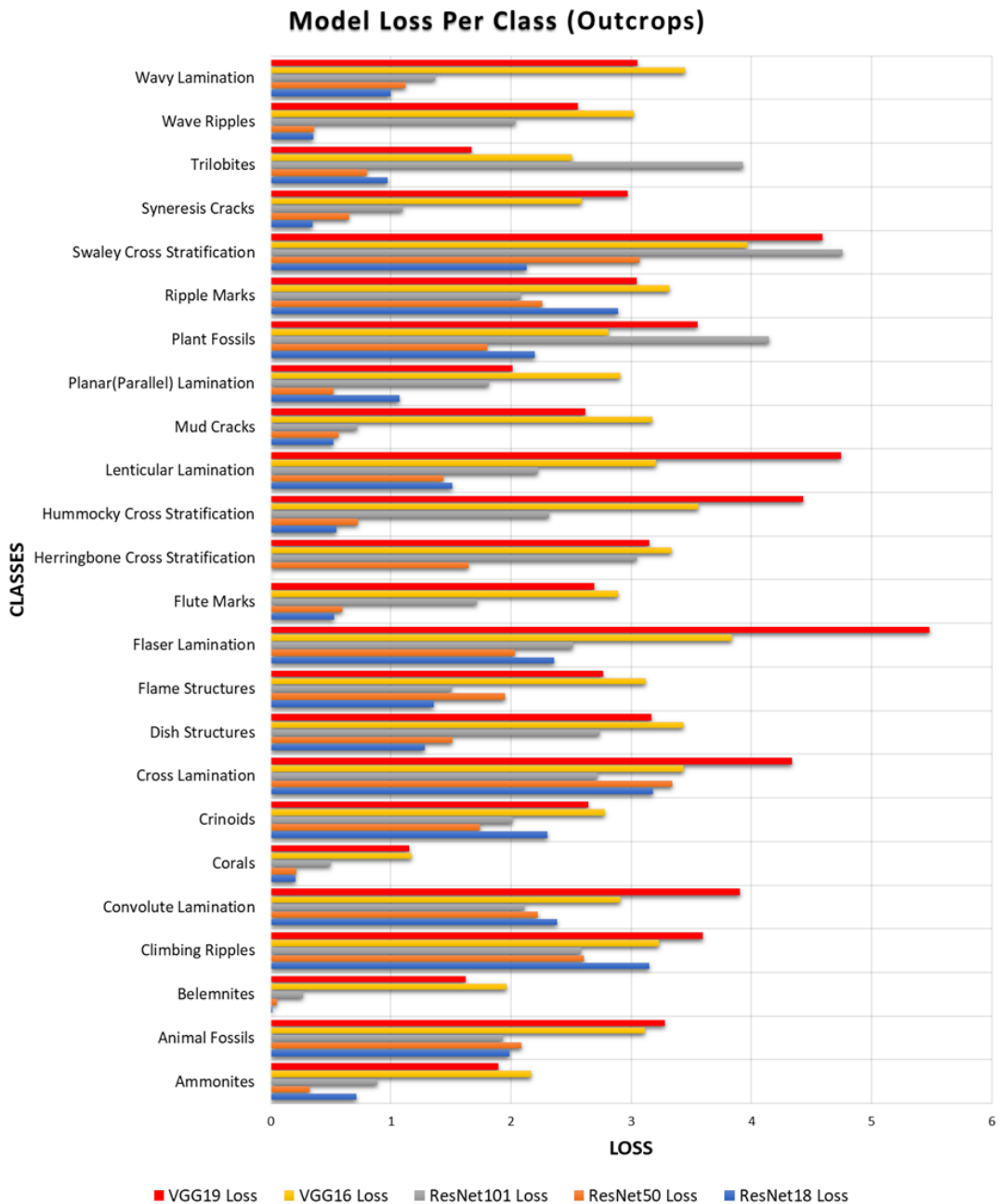


Figure 5-23: Models Loss per class across the five models.

The test accuracy of the ResNet50 is lower when trained on D6 compared to the accuracy the model shows when trained with D2. Although only outcrop images are used in both datasets, D6 has 20 more geological features and 233 more images, as shown in table x. The reason behind the accuracy drop is the class imbalance. If the custom dataset is imbalanced, with some classes having fewer examples than others, it can be challenging to train the network effectively. For Dataset 2, we have 4 classes with approximately 20 images per class, making D2 a balanced dataset. D6 has 24 classes with 310 total images

but displays a much higher imbalance between the number of images in each class. This imbalance is previously shown in Figure 4-12 in Chapter 4.

The performance of the VGG models is the worst among the five backbones. VGG was trained on a large dataset of millions of images for object recognition tasks. When working with custom datasets, there may not be enough data to train VGG effectively, leading to overfitting or poor generalization. Additionally, if the custom dataset contains images with different characteristics, which is true in our case, due to the complexity and variability of the geological features VGG may not have learned relevant features during its original training, resulting in poor performance.

Overall, for this experiment, ResNet50 performed better across all classified features, while no particular features were improved with the other CNN configurations.

5.4.2.2 Application of the Fine-tuned model trained on Dataset 7 (outcrops + sketches)

In this sub-section, the results of the fine-tuned model are presented when the model is trained on outcrop and sketches examples with Dataset 7, while maintaining the optimal proportions of sketches (40%) to outcrops (60%) established in part one. As described in Chapter 3, five different backbones were used in increasing order of complexity, meaning that each backbone has more parameters (model depth) compared to its predecessors. Specifically, the backbones used were from the ResNet family, ResNet 18, 50, and 101, and the VGG family, the VGG16 and 19. All the training hyperparameters used for the models' training are found in Table 5-3.

Training Hyperparameters	Value	Value	Value	Value	Value
Pretrained weights	ResNet18.pth	ResNet50.pth	ResNet101.pth	VGG16.pth	VGG19.pth
Image size	512	512	512	512	512
Batch size	16	16	16	16	16
Epochs	72	77	42	7	13
workers	4	4	4	4	4
Evaluation interval	1	1	1	1	1
Gpu count	1	1	1	1	1
Optimizer	ADAM	ADAM	ADAM	ADAM	ADAM
Learning rate	0.001	0.001	0.001	0.001	0.001
Total Parameters	11,438,487	24,038,744	43,030,872	135,315,544	140,625,240
Training Parameters	530,455	530,712	530,712	1,055,000	1,055,000
Position Augmentation	Value	Value	Value	Value	Value
Rotation	± 10 degrees	± 10 degrees	± 10 degrees	± 10 degrees	± 10 degrees
Horizontal Flip	Yes	Yes	Yes	Yes	Yes

Table 5-3: Model's training hyperparameters when trained on Dataset 7.

The adjustments for image size, batch size, and number of workers were made based on the available computing resources. In accordance with section 5.4.2.1, the number of epochs was uniformly set to 100 for all models. However, an additional code snippet was introduced to halt the training when the model reached its learning limit. Irrespective of the chosen backbone, all five training sessions concluded prior to completing 100 epochs. The optimizer and learning rates remained consistent with the values established in Part One. Finally, to augment the dataset's size and variability while preserving the geological features and their significance, two position augmentation techniques were implemented for each image: rotation by plus or minus 10 degrees and horizontal flipping.

In Figure 5-24, the training and validation steps depict the accuracy and loss of the model. The scores for each individual model were computed and organized into two rows. The top row displays the losses, while the bottom row displays the accuracies of each model.

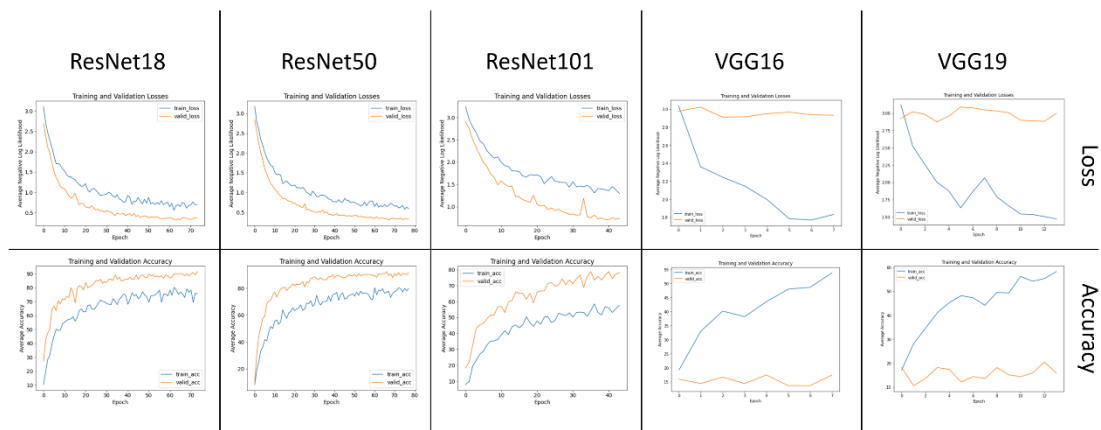
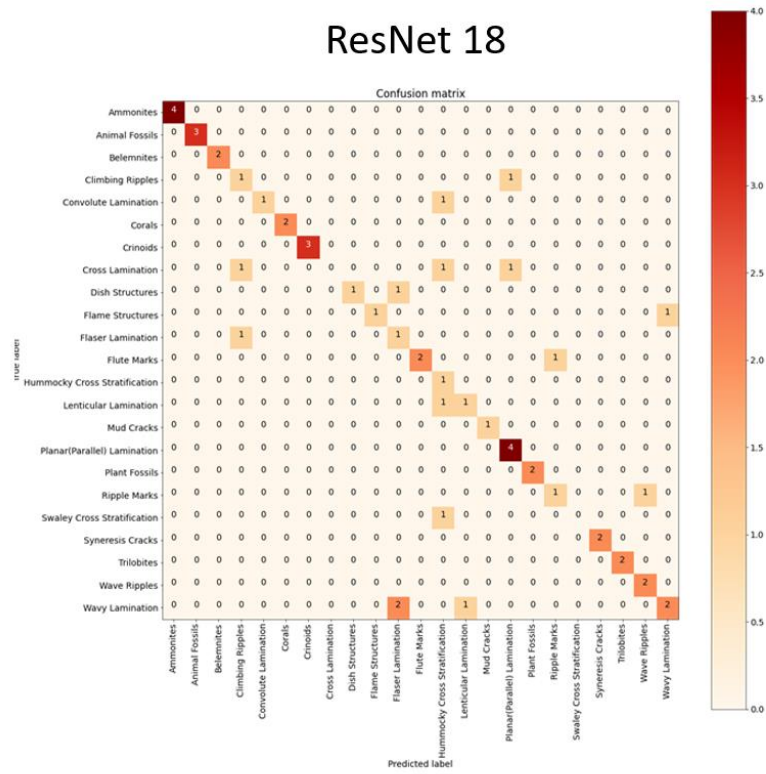


Figure 5-24: Accuracy and Loss versus the number of epochs for each model.

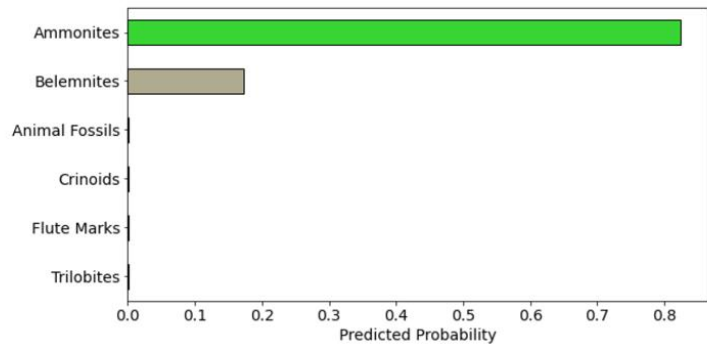
For the ResNet 18, 50, and 101 models, the couples training and validation losses and train and validation accuracies follow the same trend; they are close together, indicating a good fit of the model to the data, with no over-fitting or under-fitting occurring. On the contrary, VGG 16 and 19 show a great separation of the curve couples, both for the loss and accuracy, implying underfitting. The high values of the validation losses and the low values of the validation accuracies support this observation. In Figure 5-24, the ResNet 18, 50, and 101 models, the behavior of the loss and accuracy trend couple seems to be smoother when these models are trained with the blended dataset (D7) compared to Figure 5-14, where the same models were trained only with the outcrop images (D6). The smooth trend in the graphs for the ResNet 18, 50, and 101 models in Figure 5-24 indicates that the data exhibits a consistent and predictable pattern over time. This means there is little or no volatility or fluctuation in the data and the values are changing gradually and steadily. Such a trend can be helpful in identifying patterns and making predictions based on the data.

Figures 5-25, 5-26, 5-27, 5-28 and 5-29 show the results of each model trained with Dataset 7 on two random examples from the test set. In addition to the two test images and their corresponding predictions by the model, the confusion matrix for each model is included to provide an idea of the model's overall accuracy across the entire test set.

ResNet 18



Ammonites



Convolute Lamination

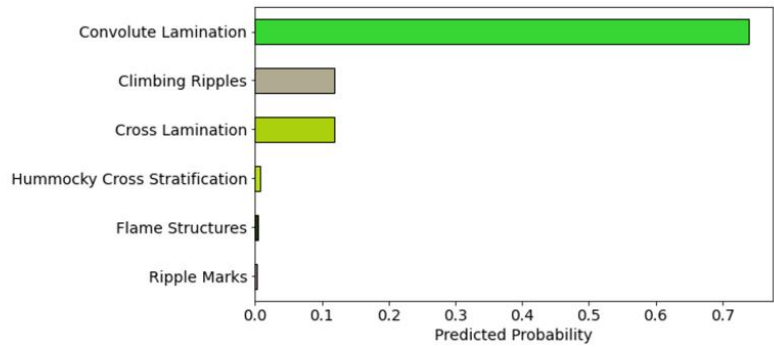
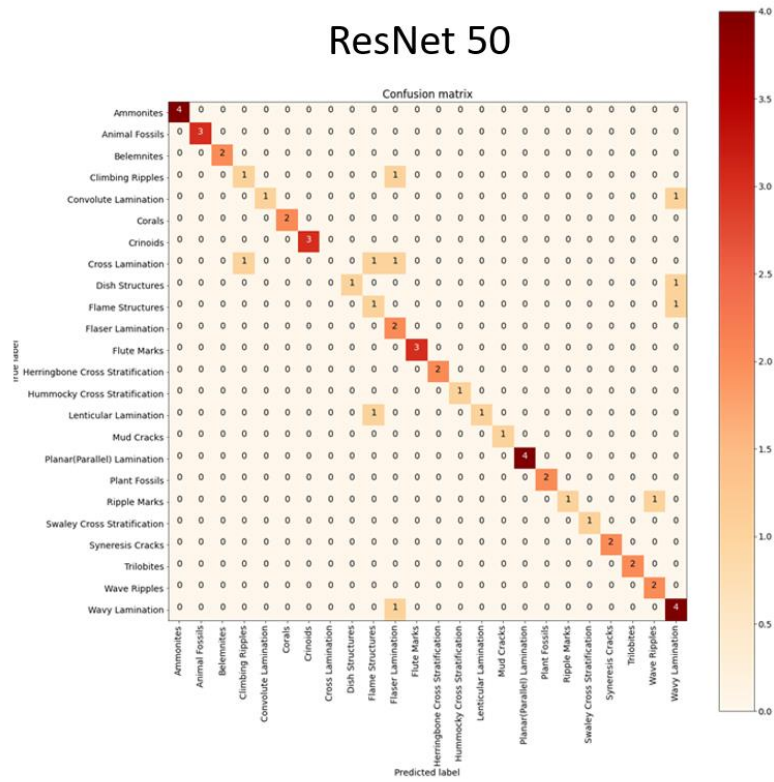
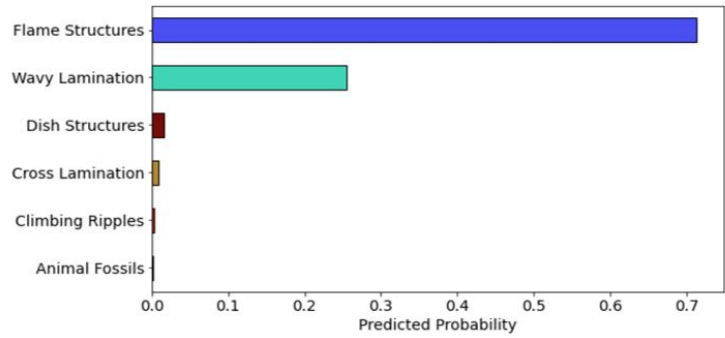


Figure 5-25: ResNet 18 (trained with D7) image classification predictions.

ResNet 50



Flame Structures



Flaser Lamination

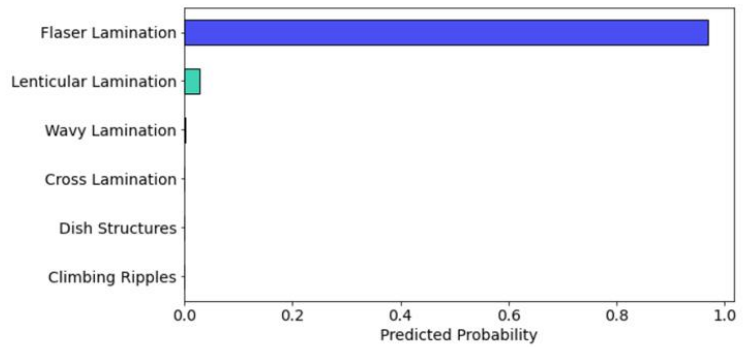
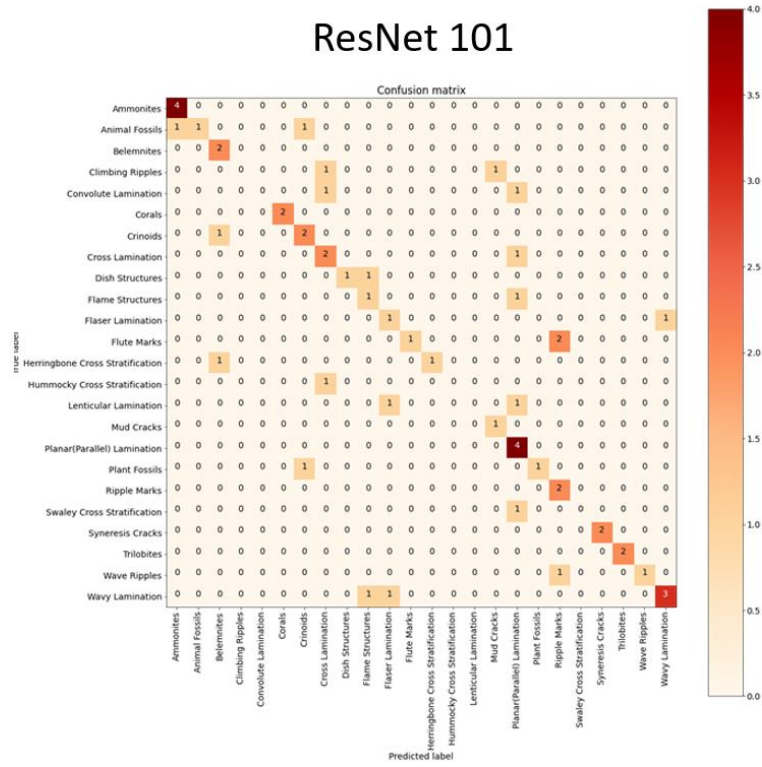
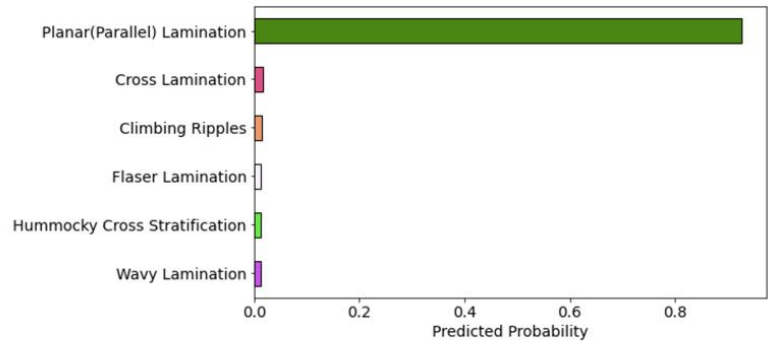
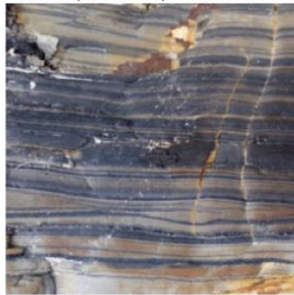


Figure 5-26: ResNet 50 (trained with D7) image classification predictions.

ResNet 101



Planar(Parallel) Lamination



Plant Fossils

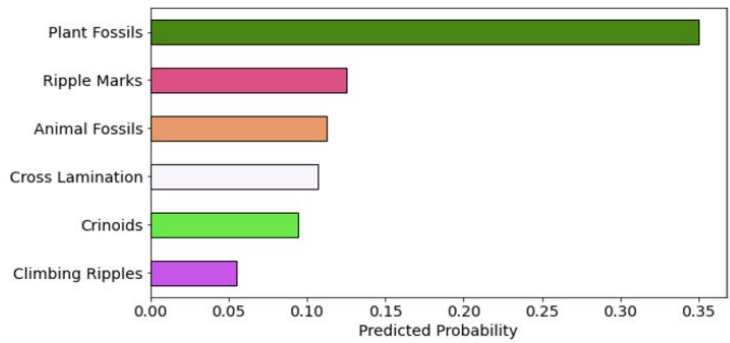
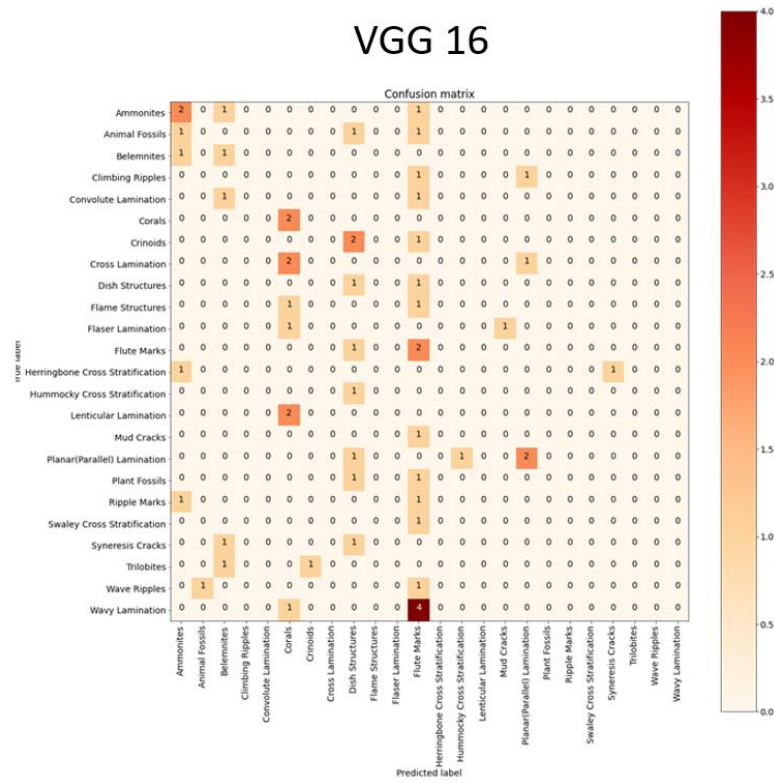
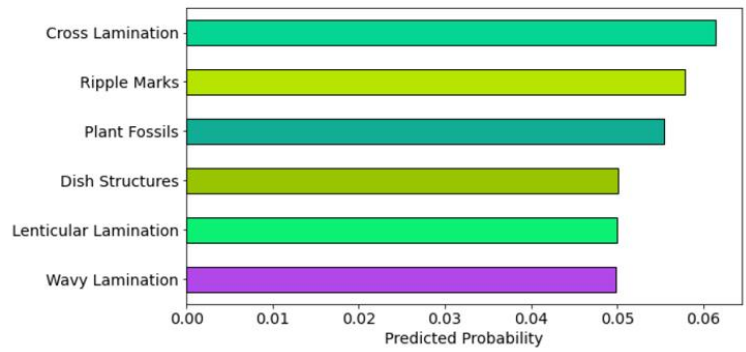


Figure 5-27: ResNet 101 (trained with D7) image classification predictions.

VGG 16



Hummocky Cross Stratification



Planar(Parallel) Lamination

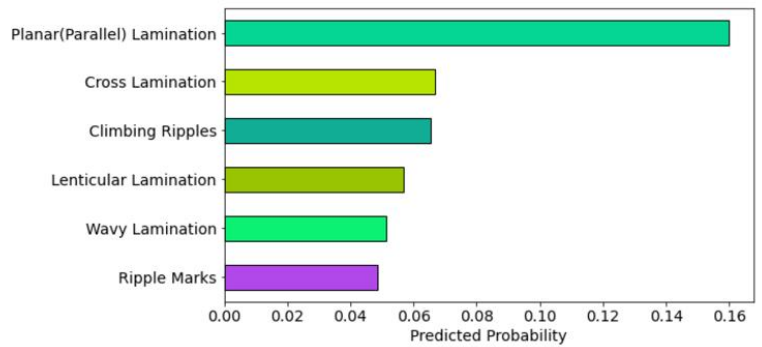
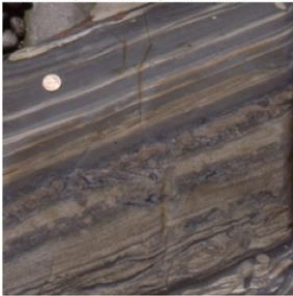


Figure 5-28: VGG 16 (trained with D7) image classification predictions.

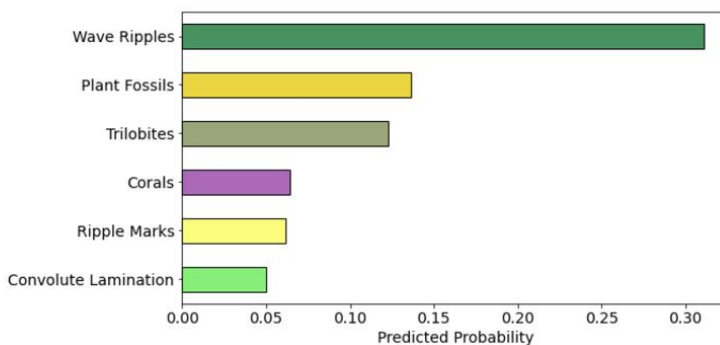
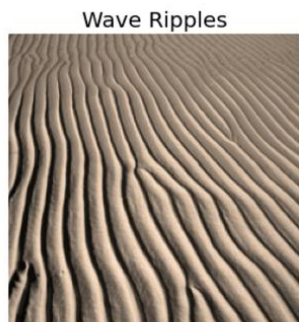
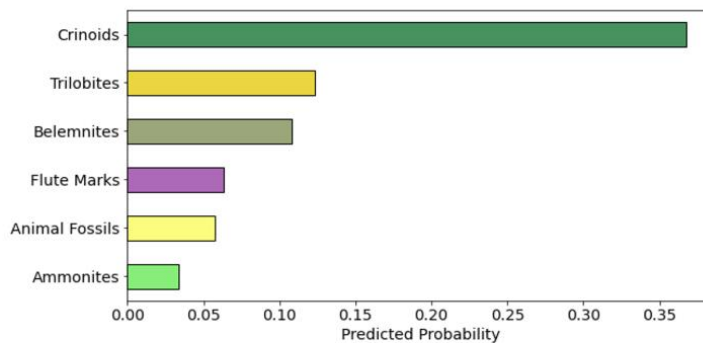
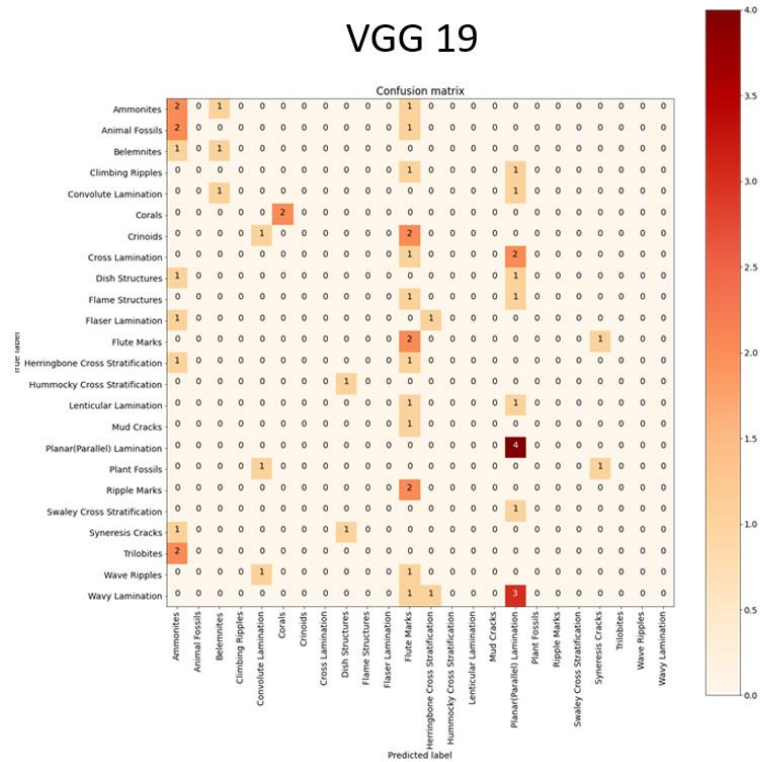


Figure 5-29: VGG 19 (trained with D7) image classification predictions.

A similar observation was made from Figure 5-16 to Figure 5-20: As the number of parameters and depth of the model's architecture increases, the worse the predictions become. Focusing on the confusion matrices for each model, it is evident that as the number of parameters increases, the class predictions become sparser and move away

from the diagonal of the confusion matrix. The diagonal of the confusion matrix is considered the ideal scheme representing the most accurate predictions. The confusion matrices in Figure 5-25 and Figure 5-26 (corresponding to ResNet18 and 50) and Figure 5-27 (corresponding to ResNet101) are closer to that ideal scenario, as most of the predictions are on top of the diagonal of the confusion matrix or close to it. As for the two VGG models, their predictions progressively become sparser. The big difference between the total number of parameters and the trainable parameters per model causes that issue. In Table 5-3, across all five models, the trainable parameters are about 530,712 for the ResNet and 1,055,000 for the VGG. These numbers depend on the number of features extracted from the dataset with which the model is trained. Since the trainable parameters are constant for each model family, and the number of the total parameters increases, the gap between the trainable and total parameters continuously widens, resulting in worse model performance.

The second conclusion derived from these results is that for the majority of the results, the model's top 1 to top 6 predictions do not display almost equal probabilities anymore, indicating that, in most cases, the addition of sketches improved the model's learning of the geological features, allowing better accuracy and certainty in its predictions. This statement is valid only for the ResNet models. For instance, considering Figure 5-26 (top test image), the top 1 prediction is the Flame Structures class, with about 0.7 probability, matching the ground truth label this time. The top 2 prediction shows the label Wavy Lamination with a probability of 0.25, while the rest of the 22 classes amass only a 0.05 probability. When tested with the ResNet50 and trained with D7, the same image yielded different top 1 to top 6 predictions. The addition of sketches not only assists the model in achieving the correct prediction of the geologic feature but also makes it more certain for its predictions by helping it to distinguish better between the top 1-6 predictions.

For the VGG models, the behavior is the same regardless of the addition of sketches, indicating that the VGG family provides such a high number of parameters that skew the model's results, therefore making VGG not a good candidate for the particular task.

Figure 5-30 shows a comparison of the overall test accuracy across the five different models. The winner is the ResNet 50 model with an 82% accuracy against the particular test set and when trained with dataset 7. The second-best model proved to be ResNet 18 a test accuracy of 72%.

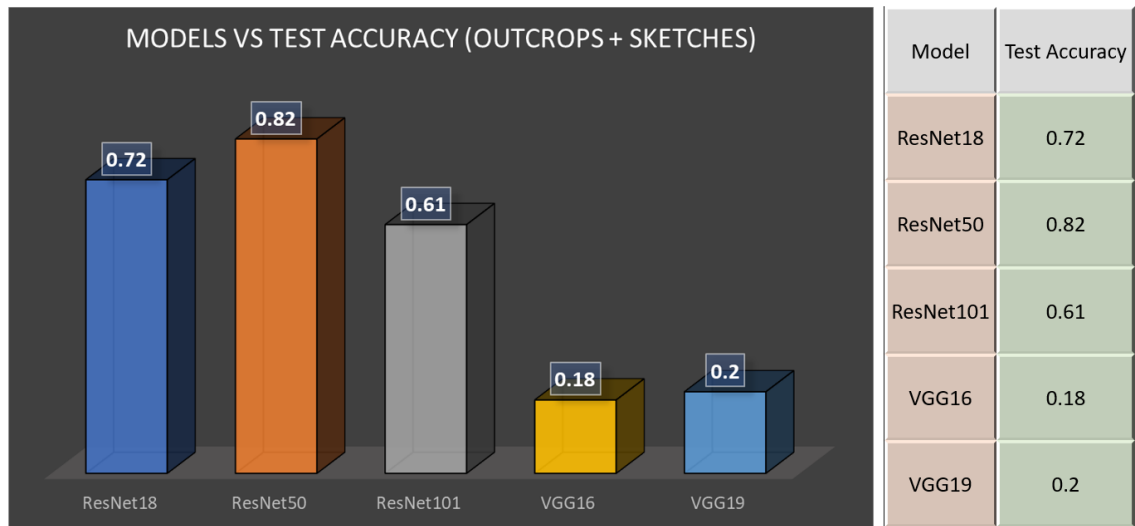


Figure 5-30: Overall test accuracy across the five different models.

The comparative study between the five backbones concluded that when using sketches and outcrops images in the training set, the highest accuracy was 82% which was achieved by ResNet50 once more.

Finally, Figure 5-31 and Figure 5-32 visually demonstrate the model's accuracy and loss per class for the test set for each of the five models. The custom image classification model, with all the different backbones, was tested on the same test images.

For Figure 5-31, in the case of Swaley cross stratification and cross-lamination, the displayed backbones were the only successful ones for making a prediction. This time, there were not any classes that remained unpredicted as no model failed to make a prediction.

For Figure 5-32, again, it is obvious that VGG 16 and VGG 19 show the highest loss, meaning that they have the worst accuracy, while ResNet18 and ResNet50 demonstrate lower loss values and therefore have the best accuracy.

Compared to Figure 5-22 and Figure 5-23, the overall performance (higher accuracies and reduced losses) is significantly improved for each backbone, suggesting that the incorporation of sketches in the training dataset, using the optimal proportions, enhanced the models' prediction accuracy for all the classes.

Model Accuracy Per Class (Outcrops + Sketches)

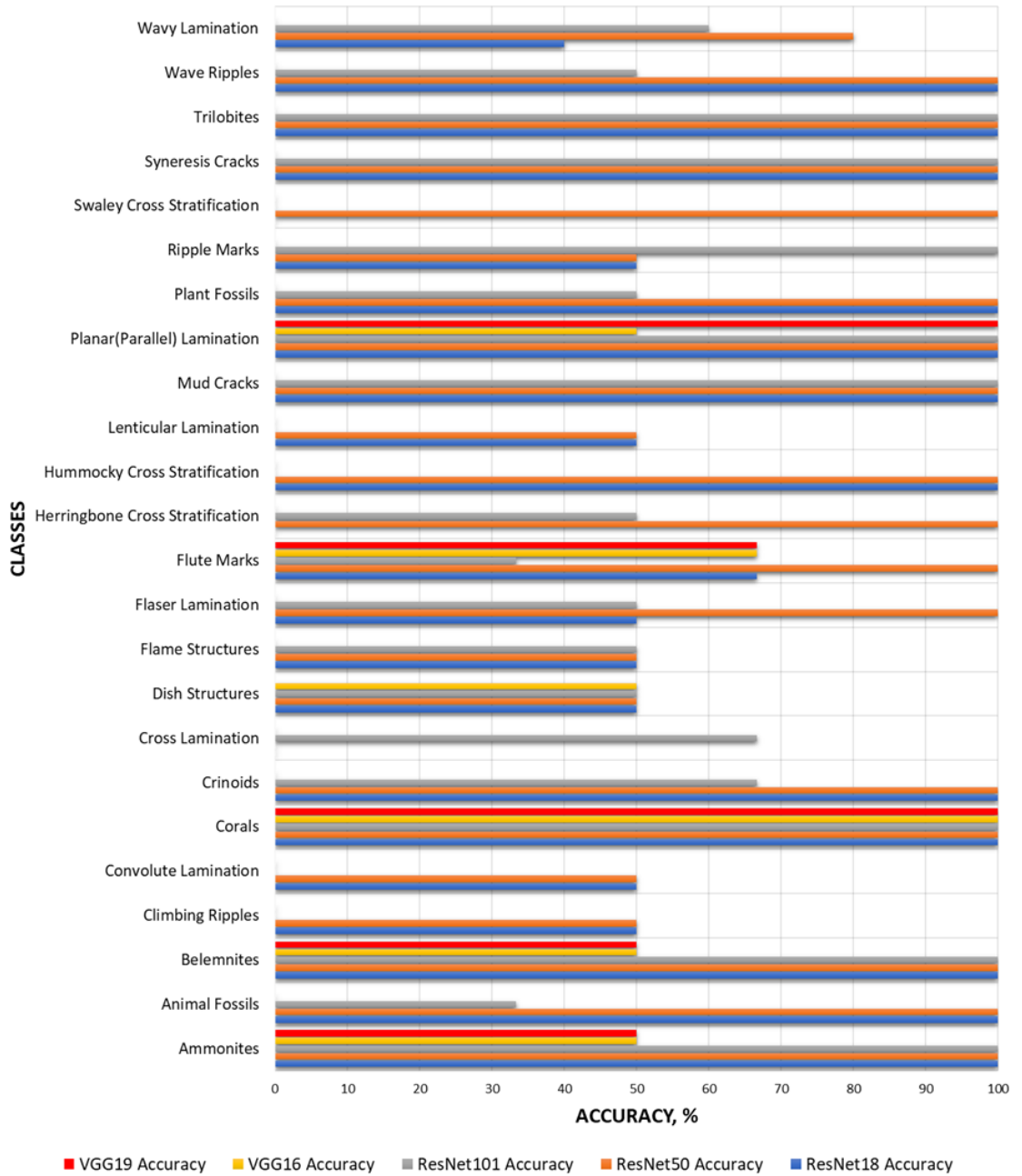


Figure 5-31: Models Accuracy per class across the five models.

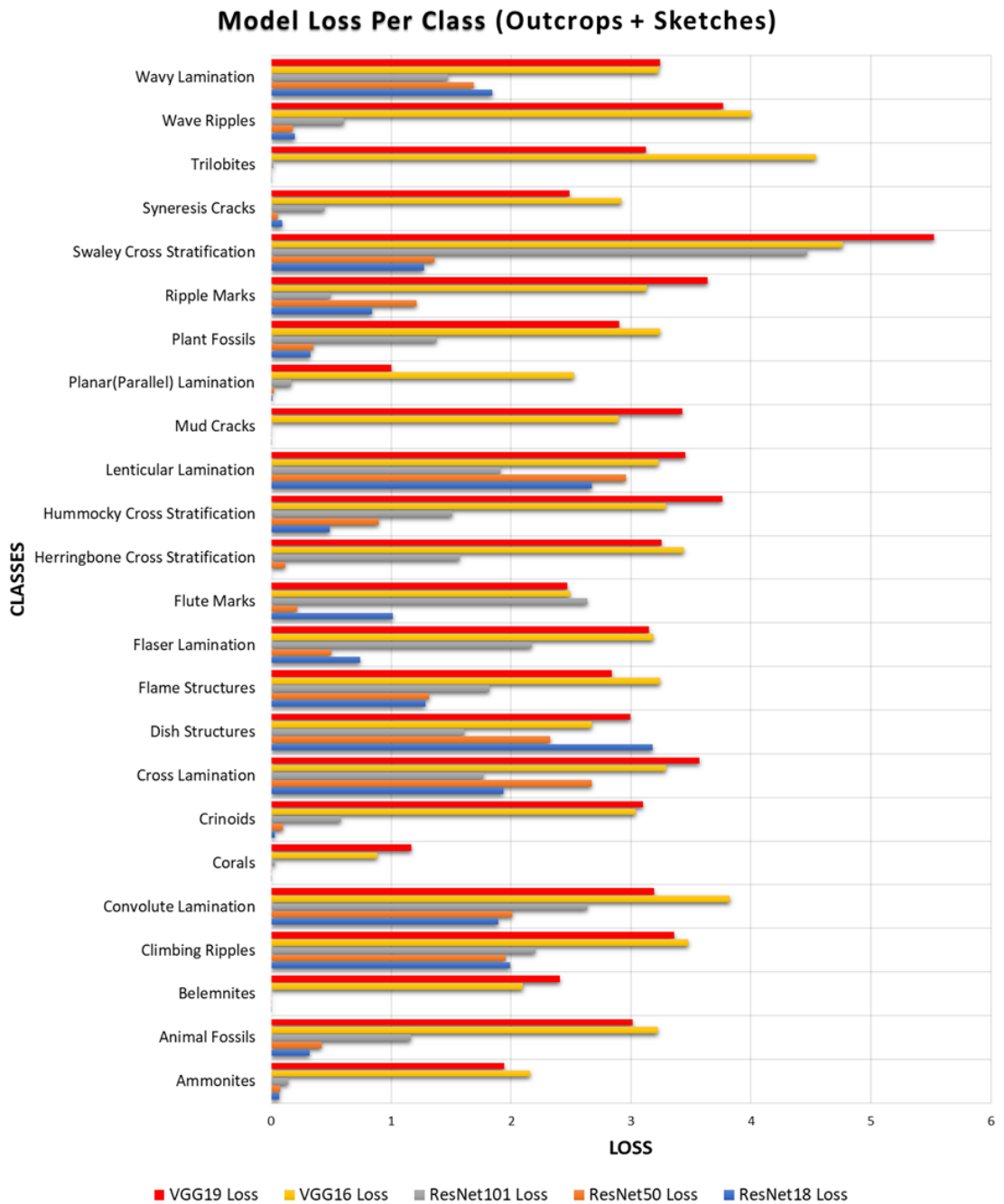


Figure 5-32: Models Loss per class across the five models.

5.4.2.3 Comparison of the two training methods

The two previous sections established that the ResNet 50 was the best model for this geology image classification task in all cases, with ResNet 18 being the runner-up. To provide additional results and comparisons between the models trained with Dataset 6 versus when trained with Dataset 7, Figure 5-33 through Figure 5-36 were generated.

Figure 5-33 and Figure 5-34 show the same picture of a crinoid stem tested with the previously described models, trained first with D6 and then separately with D7. In Figure 5-33, the ground truth label is predicted correctly only twice with the VGG models. ResNet 18 and 50 misclassified the crinoid stem as a belemnite which could be easily mistaken due to its characteristic shape. In Figure 5-34, the ground truth label is predicted correctly 3/5 times, with the two misclassifications belonging again to the belemnite class.

Figure 5-35 and Figure 5-36 show the same picture of dish structures being tested with the previously described models, trained first with D6 and then separately with D7. In Figure 5-35, the ground truth label is predicted correctly only twice with ResNet 18 and 50 models. All models display a low degree of certainty in their predictions, assigning almost equal probabilities for the top 1-6 predictions. The predictions of the dish structure label from ResNet 18 and 50 do not exceed the probability of 0.44. In Figure 5-36, the ground truth label is predicted correctly 3/5 times with the ResNet 18, 50, and 101 models. The predictions of the dish structure label from ResNet 18 and 50 almost reach a probability of 1 in their predictions, indicating a very high degree of certainty. Figure 5-35 shows that the ResNet 101 and two VGG models have a lower level of prediction certainty than the ResNet 18 and 50 models. However, even with this decreased certainty, their predictions are still better than those derived from the model trained on D6.

Outcrops Models

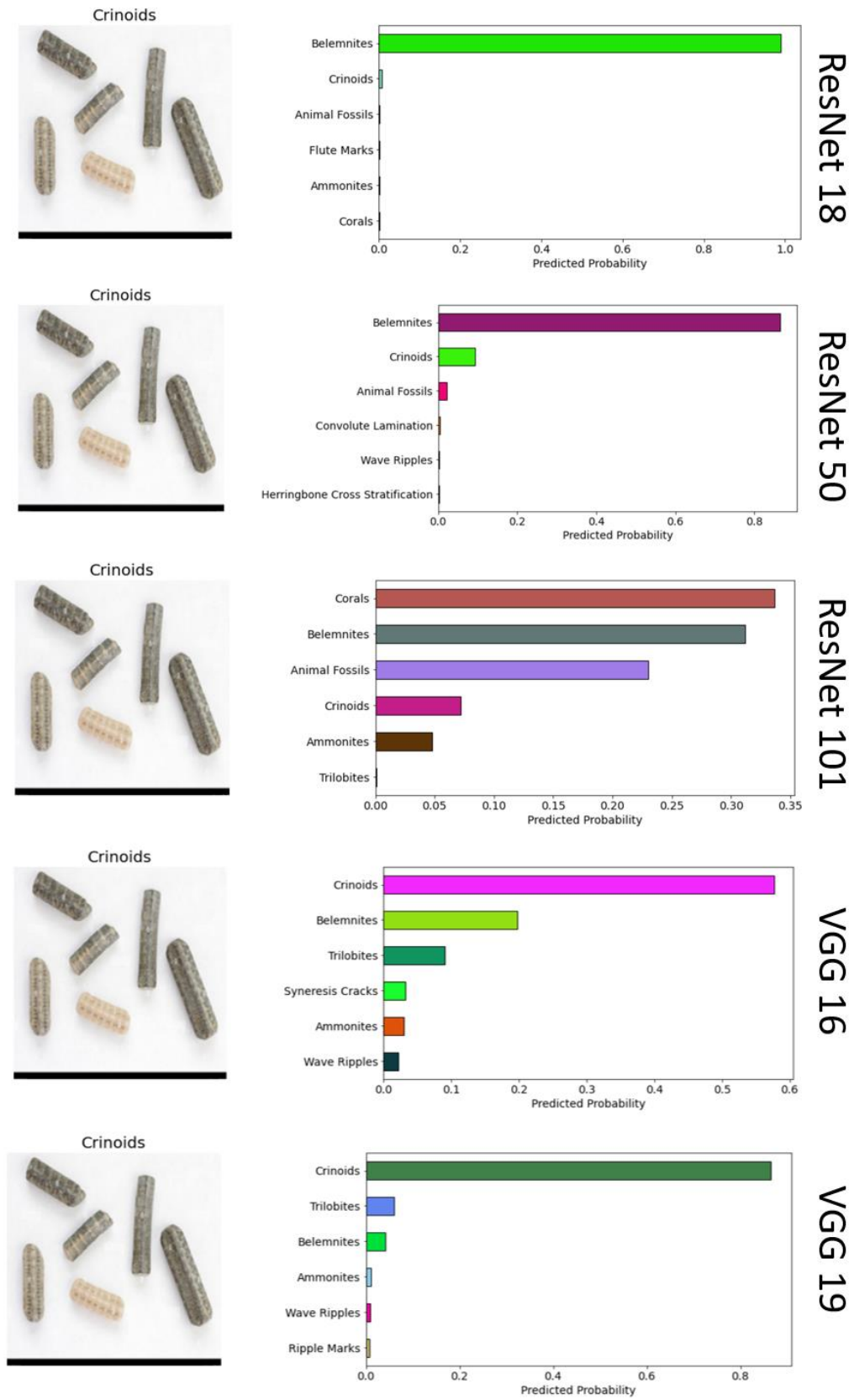


Figure 5-33: A test image of a crinoid stem tested against the five models trained with D6.

Outcrops + Sketches Models

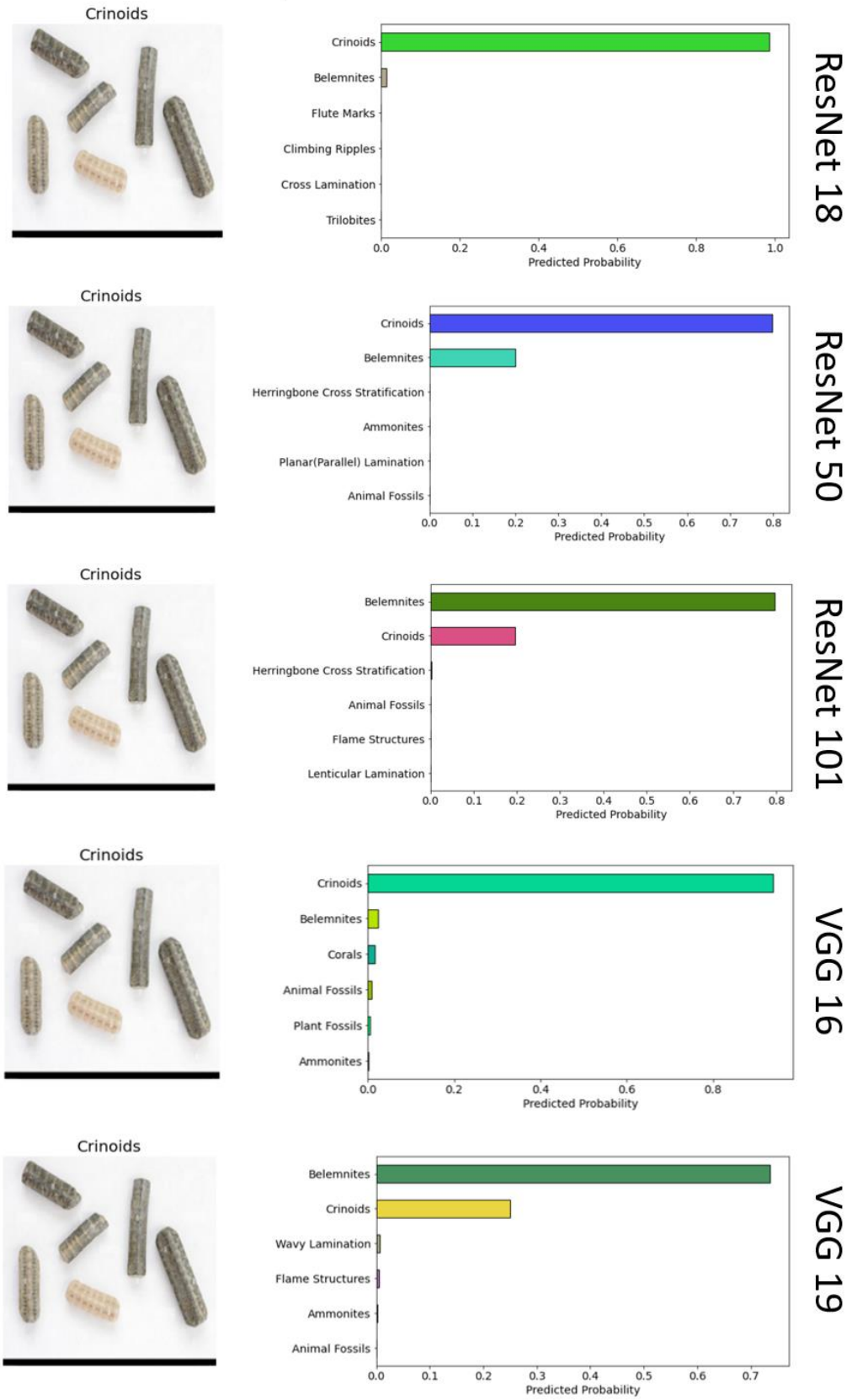
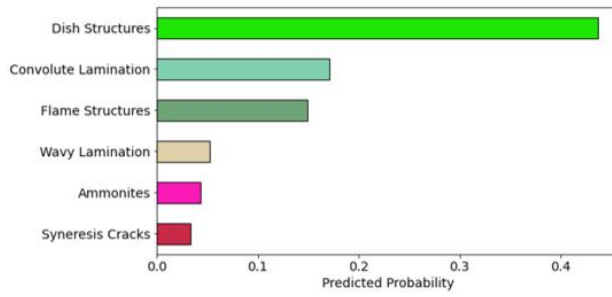
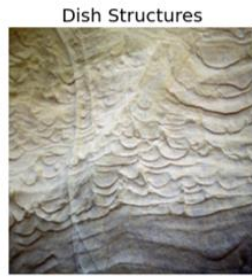
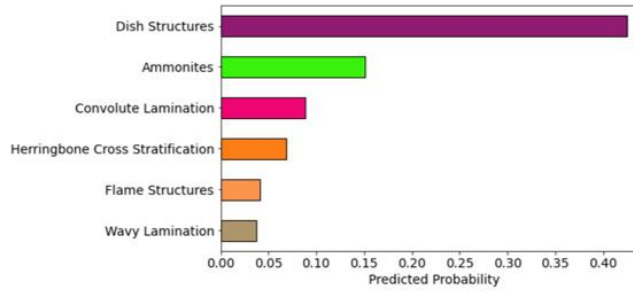
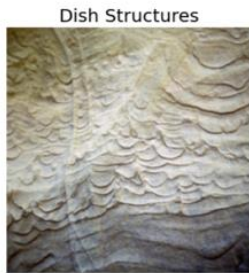


Figure 5-34: A test image of a crinoid stem tested against the five models trained with D7.

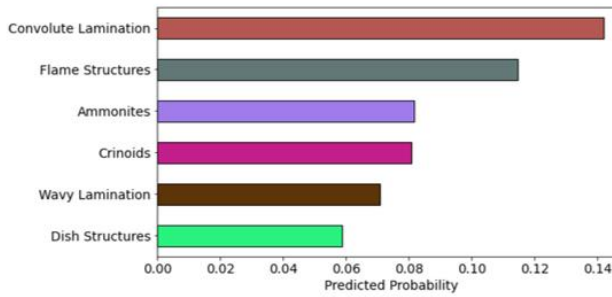
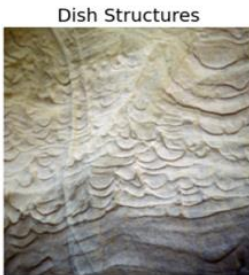
Outcrops Models



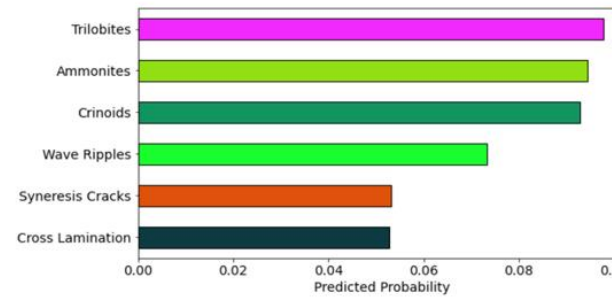
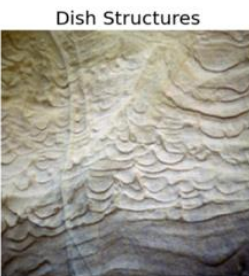
ResNet 18



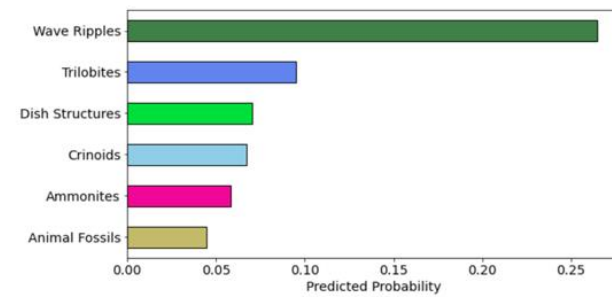
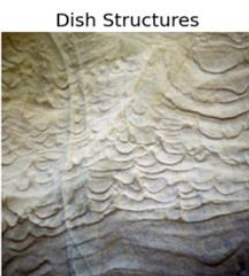
ResNet 50



ResNet 101



VGG 16



VGG 19

Figure 5-35: A test image of dish structures tested against the five models trained with D6.

Outcrops + Sketches Models

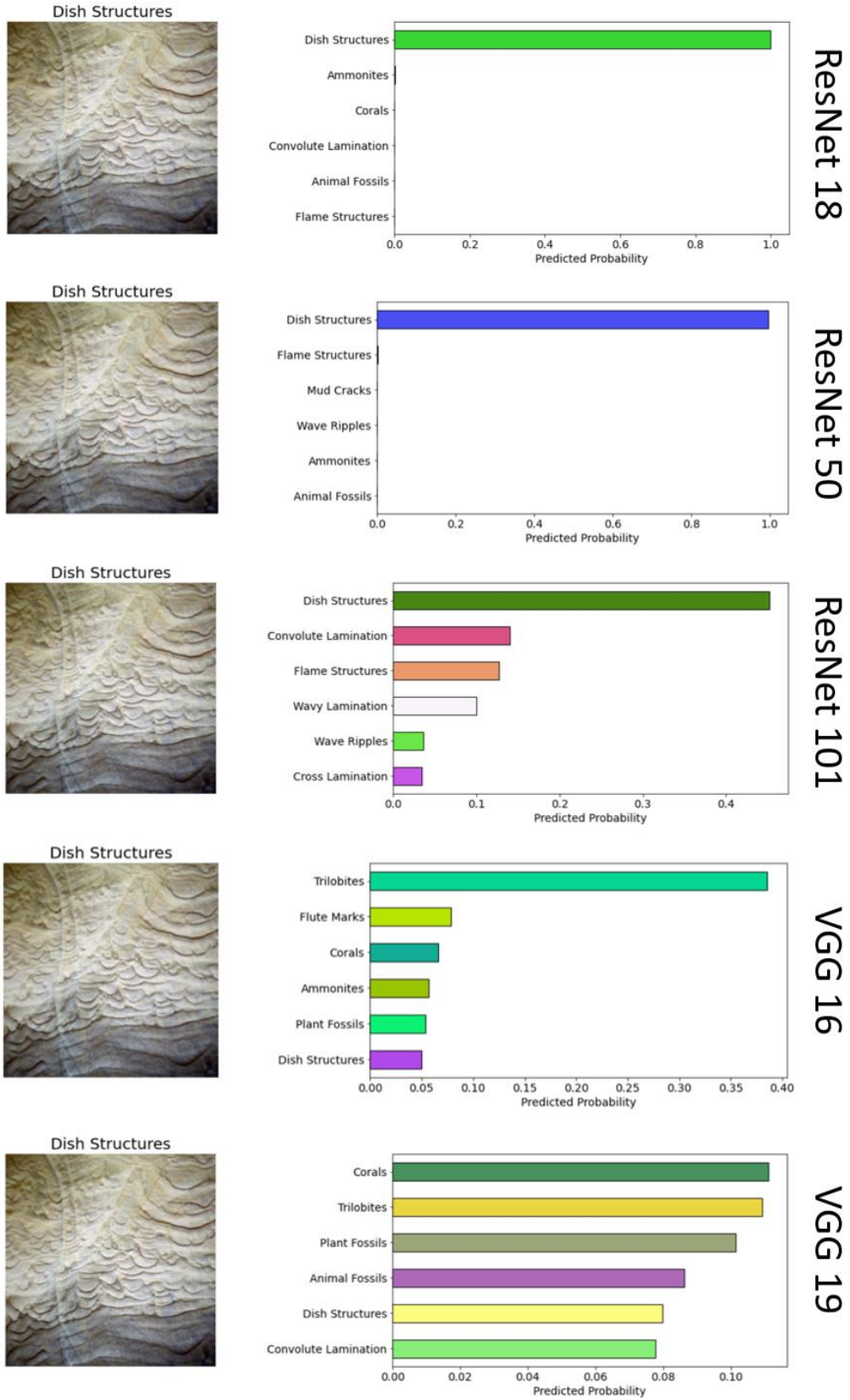


Figure 5-36: A test image of dish structures tested against the five models trained with D7.



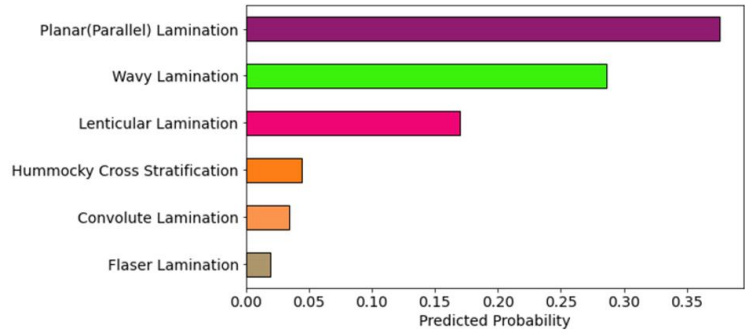
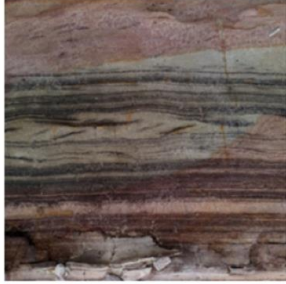
Figure 5-37: Ground truth example.

The image shown in Figure 5-37 has three unique sedimentary structures and I am trying to show how Image Classification will classify this image only according to the most prominent feature. According to the human image segmentation and the assigned ground truth labels, the majority of the pixels in the image belong to the Planar/Parallel Lamination label. However, since three classes are present in the image and the classification model will label the image once, this image was put in the test set three times, each carrying one of the three possible labels. This way, the model is given a chance to predict, at least once, one or more of the ground truth labels. Since ResNet 50 was the best model for the geological image classification task, as presented in this chapter, that particular version of the model was tested to classify the above image. The predictions of the ResNet50 trained with outcrops and the ResNet50 trained with D6 and D7 can be found in Figure 5-38 and Figure 5-39, respectively.

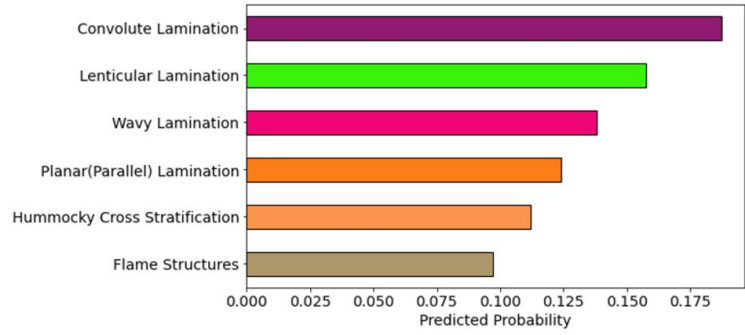
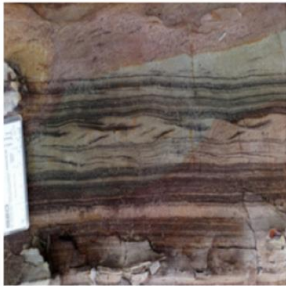
In both figures, only 1/3 of the labels are predicted correctly, the label of Planar Lamination. This example of a test image is evidence that, in such examples, there is a need for a classifier to classify all the features present in the image simultaneously. Therefore, additional methods such as Object Detection and Segmentation are needed for such images in order to capture all the features.

ResNet50 (Outcrop Training)

Planar(Parallel) Lamination



Lenticular Lamination



Cross Lamination

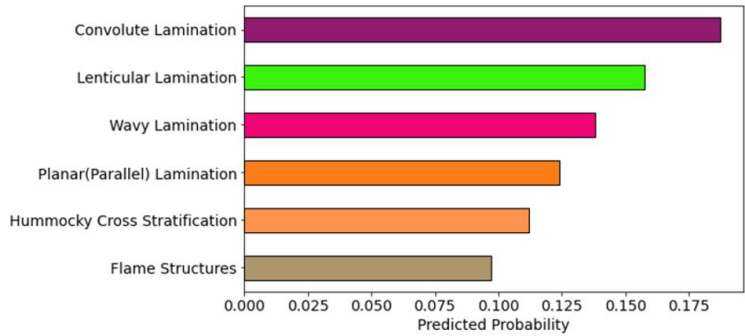
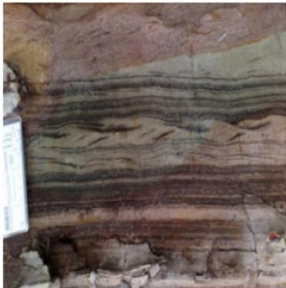
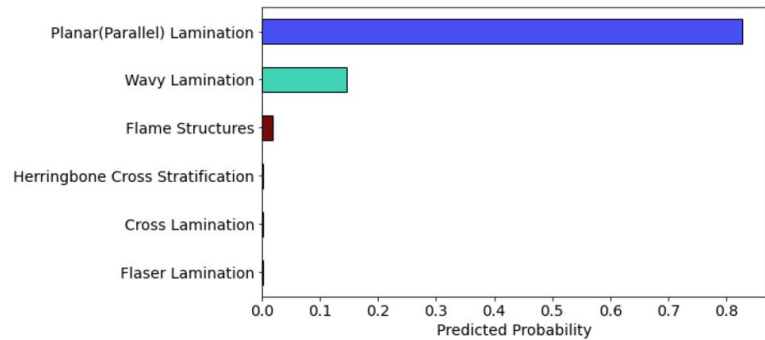


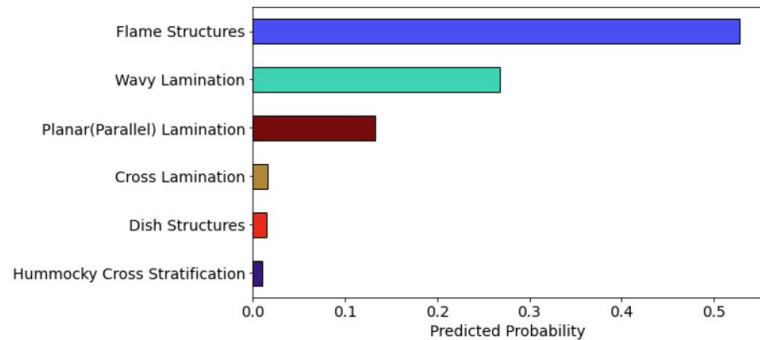
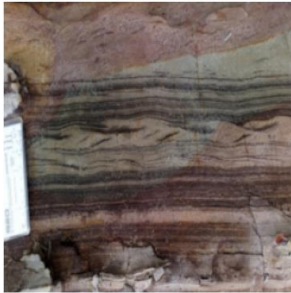
Figure 5-38: ResNet 50 predictions (trained with D6).

ResNet50 (Outcrop + Sketches Training)

Planar(Parallel) Lamination



Lenticular Lamination



Cross Lamination

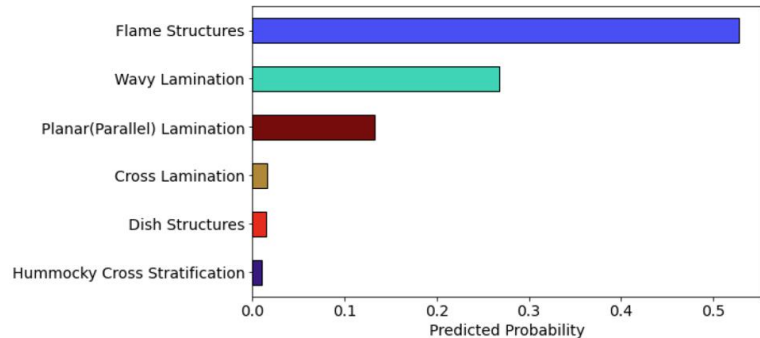
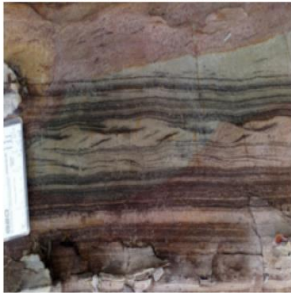


Figure 5-39: ResNet 50 predictions (trained with D7).

A possible explanation about the model's misclassifications in Figure 5-39 (middle) is that cross-lamination can be confused with flame structures and wavy lamination due to some similarities in their appearance. Cross-lamination refers to the alternating layers of sediment that are inclined at an angle to the main bedding plane. Flame structures are similar, but the inclined layers are curved or flamelike in shape, with a V-shaped point at the top of the curve. Wavy lamination is characterized by gentle, undulating layers that are parallel to the main bedding plane. All three structures can be formed by the action of flowing water, such as in rivers or ocean currents. Cross-lamination, flame structures, and wavy lamination can also occur together in the same sedimentary rock, further adding to the confusion between the structures. To differentiate between these structures,

geologists need to carefully examine the shape and orientation of the layers, as well as the sedimentary environment and conditions under which they were deposited.

5.5 Chapter's Conclusions

This chapter demonstrated how building an image classification model can help us distinguish between individual sedimentary structures by classifying the image based on the predominant sedimentary feature represented in each image.

Part One (5.4.1) established that training a model with a dataset of sketches and outcrop photos (D4) led to better test classification accuracies than a dataset of only outcrop photos (D2, D5), reducing potential misclassifications.

An optimal balance between sketches and outcrop photos is suggested for the training data to improve the model's learning. When the sketch/outcrop images proportion ranges from 40% to 67%, the model's accuracy significantly improves, compared to the model's accuracy when trained only with outcrop images.

In addition, different hyperparameters and optimizers were tested with the model. Through trial and error, and according to the model's predictions, it was found that the Adam optimizer and a learning rate of 0.001 yielded the best results.

Part Two (5.4.2) shows the results of a robust image classifier trained with a blended dataset (D7) for a broader range of geological classes of sedimentary structures and fossils (almost five times more variety). A comparison between the model trained with Dataset 6 and the same model trained with Dataset 7 showed that adding sketches to the training dataset consistently improves the model's accuracy and certainty in its predictions (Table 5-4).

Dataset	Data Type	Image Number	Number of Geological Features	ResNet50 Accuracy, %	Task
Dataset 6 (D6)	Outcrops	310	24	61	Image Classification
Dataset 7 (D7)	Sketches + Outcrops	652	24	82	Image Classification

Table 5-4: Summary of Datasets 6 and 7 along with their corresponding test accuracies, results of the ResNet50 model architecture.

From 61% test accuracy, the ResNet50 model demonstrated that when trained on dataset 6, there was a significant improvement to the test accuracy, up to 82% when trained on Dataset 7. Although some features, such as cross lamination, flame structures, convolute and lenticular lamination, are more difficult to classify, even for those, the accuracy improves from 20 to 50%.

This blending not only adds quality to the interpretation by utilising the knowledge and simplicity incorporated in the sketches but also considers the real-world complexity depicted in the photos, helping our model to learn from the data and recognize the structures more efficiently.

The accuracy of the models described in this chapter can be further improved if a more diverse and balanced dataset was used to train these models. Increasing the number of classes significantly, by default, required a lot of additional data, which, unfortunately, were unavailable at the time.

The next step will be to improve this model by making it capable of identifying multiple features, their proportions, and spatial distribution in the observed area (outcrop), which is helpful to add context to the interpretation. According to the discussion around Figure 5-37, there is a need for a model to detect and localize multiple features at once. Since image classification cannot help with such a task, YOLOv6 (Li, et al., 2022), an object detection model, will be employed.

CHAPTER 6 - IDENTIFYING MULTIPLE GEOLOGICAL FEATURES USING OBJECT DETECTION ON OUTCROP AND FOSSIL IMAGES

6.1 Introduction

This chapter tackles the problem of identifying and localizing multiple sedimentary structures and fossils from 2D images with Object Detection, which refers to the second step of the thesis's high-level workflow introduced in Chapter 1.

The Object Detection model utilised in this Chapter is YOLOv6 (Li, et al., 2022) and uses annotated image datasets as an input, consisting of outcrop and fossil images to assign a bounding box around each object detected in each image of the test data. A bounding box is a rectangular frame or box used in object detection that encloses an object of interest in an image or video. The bounding box aids in identifying the limits of the object within the image and serves as a spatial reference for the object's location.

The model is trained and tested using two different datasets (Datasets 8 and 11) to monitor its performance in detecting sedimentary structures and fossils in both outcrop and core images. To evaluate the model's performance, we set up three similar experiments, each consisting of two sub-sections. The first sub-section provides a quantitative analysis of the model's predictions and misclassifications, presented in tables. The second sub-section translates the quantitative results into a qualitative analysis by showing visual examples of the geology and the model's predictions. The model outputs the test images with bounding boxes and corresponding labels around the various geological features, along with a confidence score for each prediction.

In Experiment 1, we use the YOLOv6-S model to detect sedimentary structures from 2D outcrop images. The model is trained and validated with Dataset 8 and tested on a test set consisting of unseen outcrop images. In Experiment 2, we use the YOLOv6-S model to detect seven different types of fossils from images. The model is trained and validated with Dataset 11 and tested on a test set consisting of unseen fossil images. In Experiment 3, we apply the previously trained YOLOv6-S model from Experiment 1 to evaluate its capability of transferring geological knowledge from outcrops to core data. The model, which was trained and validated exclusively on outcrop images, is challenged to detect

sedimentary structures on core data. Core images represent fragmental geological evidence from the subsurface, which is at a much smaller scale than outcrops. The application of the previously trained YOLOv6-S model shows how the model can apply geological knowledge from outcrops to core samples and make good predictions.

According to the chapter's findings, the presented geological Object Detector is successful at predicting the geology at different scales compared to the Image classification model, which was useful for different scales, from close apps to zoomed-out ones, as well as on fossil images. The chapter concludes with a discussion of how this method can help us extract visual evidence (collection of features) from an outcrop, which is helpful for a geologist to form an interpretation. The main drawbacks of this method are also discussed, leading to the next Chapter 7, on Instance Segmentation.

6.2 The YOLOv6 Model

The YOLO family of models is an anchor-based object detection method, which means it uses predefined anchor boxes of different scales and aspect ratios to detect objects. In contrast, anchor-free methods directly predict the bounding boxes of objects without using predefined anchor boxes (Li, et al., 2022). In computer vision, fixed-shape bounding boxes are called anchors. Anchor-free methods belong to a class of object detection approaches that do not rely on predefined anchor boxes for object localization. Instead, these methods directly predict objects' bounding boxes and associated prediction scores. YOLOv6 incorporates some anchor-free elements in its design, such as the use of a focal loss function and a center prediction module. These elements help YOLOv6 to improve its performance, providing better generalizability, costing less time in post-processing, and making it more robust to object occlusion and scale variation. YOLOv6 has three times fewer predefined anchors, making it 51% faster than most anchor-based object detectors (Li, et al., 2022).

The YOLOv6 model encompasses multiple variations, each characterized by distinct levels of computational complexity and accuracy (Li, et al., 2022). Among these variants, YOLOv6s/YOLOv6-S, with 's' denoting the 'small' variant, emerges as the most compact and fast backbone version within the YOLOv6 model. Possessing a reduced number of layers and demanding minimal computational resources, YOLOv6s accommodates devices featuring constrained computational capabilities, such as mobile

phones and embedded systems, without compromising the attainment of high-quality Object Detection outcomes.

In this thesis, the interest is shifted toward the YOLOv6-S, as mentioned in Chapter 3, section 3.4. The decision to prioritize the small version (s) of YOLOv6 over its larger counterpart (L), YOLOv6-L, despite the latter exhibiting a higher mean Average Precision (mAP) score, according to Li et al. (2022), stems from a key consideration: parameter count. YOLOv6s encompasses 17.2 million parameters, while YOLOv6L entails a significantly larger parameter count of 58.5 million. Consequently, the substantial size difference renders YOLOv6L computationally demanding, surpassing the computational capabilities of the available hardware. Given this constraint, YOLOv6s emerges as the optimal choice, aligning with the hardware specifications at hand.

Figure 6-1 and Figure 6-2 show a benchmark comparison of the YOLOv6 models with previous state-of-the-art YOLO versions both for the mAP scores and the latency.

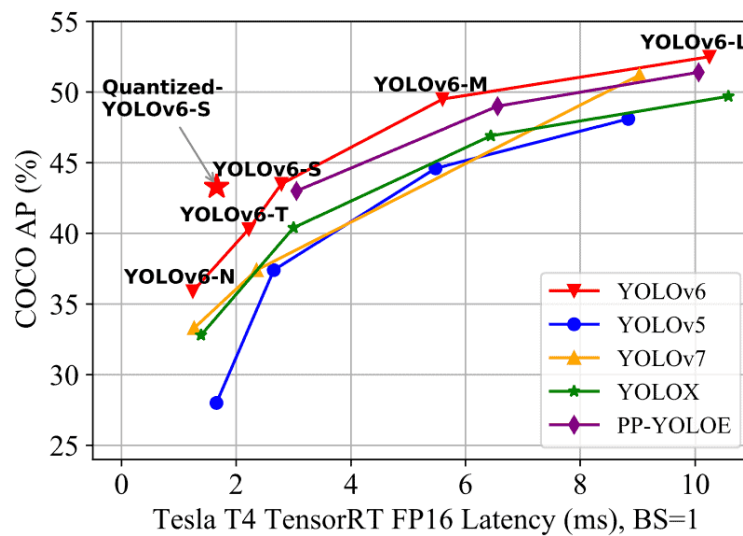


Figure 6-1: Latency comparison of YOLOv6 with other YOLO models (Li, et al., 2022).

One important observation derived from Figure 1 is that the Quantized YOLOv6-S model is faster and has higher mAP than its counterparts in other YOLO versions. In terms of mAP, all the YOLOv6 models seem to perform better than the other YOLO versions. Furthermore, Figure 6-2 supports the above statement by comparing the mAP and FPS of the same YOLO versions as Figure 6-1.

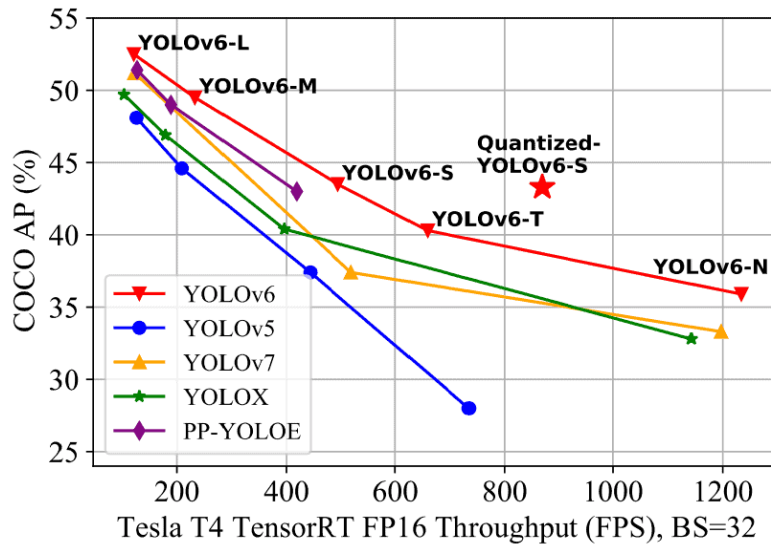


Figure 6-2: YOLOv6 FPS comparison with other models (Li, et al., 2022).

Again, the YOLOv6 models are the most accurate while maintaining at least the same FPS, demonstrating that this version of the model has a good trade-off between accuracy and speed, making it a great candidate for real-time applications as in this thesis, I am detecting geological features for both static images and videos.

6.2.1 The YOLOv6 Backbone Architecture

The architecture of a one-stage object detector like YOLOv6 comprises the following parts: a backbone, a neck, and a head. The backbone network is responsible for extracting features from the input image, while the neck network is used to fuse and refine these features. Finally, the head network predicts the bounding boxes and class probabilities of the objects in the image.

The *backbone* in YOLOv6 is designed to be parameterizable, which means that it can be easily adjusted to different input sizes without significant changes to the architecture. This allows for efficient scaling of the model to different sizes, which is particularly useful in scenarios with limited computational resources. Generally, the backbone mainly determines the feature representation ability, while its design critically influences the inference efficiency.

The *neck* is responsible for integrating and refining information from different layers of the backbone network to enhance the model's object detection performance. This integration involves combining high-level semantic features, which encode more abstract

and context-rich information, with low-level physical features that capture fine-grained visual attributes and spatial details. Low-level physical features typically encompass local textures, edges, corners, colour gradients, and other elemental visual cues that contribute to object localization and recognition. They are considered as the building blocks for higher-level feature representations. High-level semantic features represent abstract and context-rich information extracted from deeper layers of the neural network. They capture global characteristics, object classes, and contextual relationships, enhancing the model's ability to recognize and localize objects accurately (Lin, et al., 2017). The combination of low-level physical features with high-level semantic features results in a feature pyramid map at multiple levels. A feature pyramid map refers to the resulting output of the FPN architecture. The feature maps within the pyramid retain spatial information and capture features at multiple scales, facilitating object detection and recognition tasks. Integrating features at multiple scales has been shown to be a crucial and effective part of object detection (Lin, et al., 2017; Ghiasi, et al., 2019).

The *head* consists of several convolutional layers, predicting final detection results according to multi-level features assembled by the neck. From the structure's perspective, it can be categorized as anchor-based and anchor-free. Anchor-free detection methods are favored due to their superior generalization ability and simplified decoding of prediction results (Li, et al., 2022). Generalization and decoding are important in computer vision models because they enable the models to accurately classify and detect objects in new/unseen images and interpret the output of the model in a meaningful way.

6.2.2 Metrics for YOLOv6 Evaluation

The *mAP* (mean Average Precision) is a widely used performance metric for object detection models in computer vision (Everingham, et al., 2010). It is defined as the average precision (AP) across different object categories. The mAP score is calculated based on the validation set of an object detection or segmentation dataset. The purpose of the validation set is to evaluate the performance of the object detection model on data that it has not seen during the training in order to assess its ability to generalize to new data. Therefore, the validation set should be carefully selected to ensure that it covers a wide range of object types, backgrounds, lighting conditions, and other factors that may affect the performance of the model in practice.

The IoU (Intersection over Union) measures the overlap between the predicted and the ground truth bounding boxes. The value of IoU can be used to evaluate the accuracy of object detection algorithms and to determine the threshold for true positive predictions.

The mAP@0.50:0.95 metric is an extension of mAP that measures the AP at different IoU thresholds between 0.50 and 0.95, with a step size of 0.05. It calculates the AP for each object category separately and then computes the mean AP across all categories. The thresholds between 0.50 and 0.95 are used in this project because they are commonly used in object detection benchmarks such as COCO (Common Objects in Context) and PASCAL VOC (Visual Object Classes). It is considered to be a more comprehensive metric than mAP@0.50, which only measures the AP at a single IoU threshold of 0.50.

6.3 Workflow for YOLOv6 applied to Geology

All the models of the YOLOv6 family operate under the same principles and processes. Figure 6-3 illustrates how YOLOv6 can be used for geologic object detection on outcrop and fossil images regardless of the model's versions.

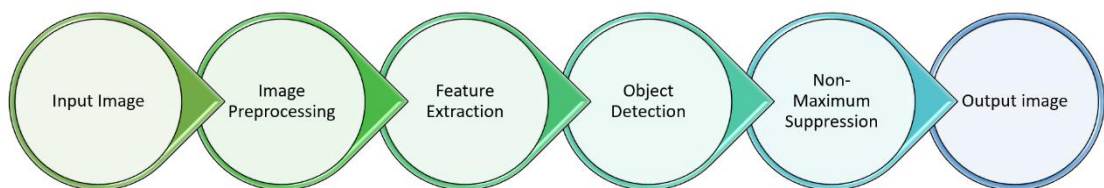


Figure 6-3: How Yolov6 works for geologic object detection.

6.3.1 Step 1: Input Images and Preprocessing

The input to YOLOv6 for training is 1:1 aspect ratio images that need to be analysed for object detection. The test images can be of any size and resolution, and the algorithm can handle predictions on images of different aspect ratios. Before the training images are fed into the model, they undergo some preprocessing steps to make them suitable for the algorithm. Such preprocessing might be the adjustment of the image's aspect ratio to 1:1. YOLOv6 allows an automatic resize feature, allowing the user to specify the image dimensions prior to training. The image is resized to a fixed size, and the 1:1 aspect ratio is maintained by adding padding if necessary. That way, the features in the images are

not distorted. The scale of the geological features, as explained in earlier chapters, is taken into account by the model with the help of the annotations.

6.3.2 Step 2: Dataset annotation

The next and one of the most critical parts of this workflow is the annotation of all the images that will serve as input into the model along with the images. Datasets 8 and 11 were used in this chapter and were annotated with various sedimentary structures and fossil types, respectively. The details of each dataset can be found in Chapter 4, section 4.8. During the annotation step, the geologist/geoscientist must draw a bounding box around the object of interest and manually label each object for every image in the dataset. An example of bounding box annotations is illustrated in Figure 6-4.

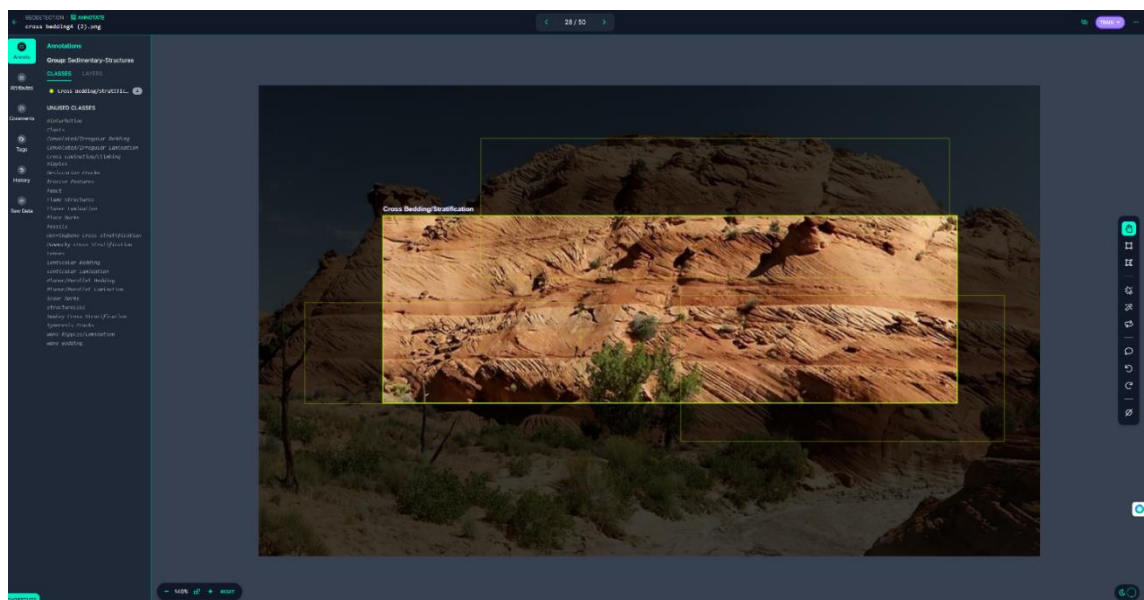


Figure 6-4: Bounding Box outcrop image annotation example.

These annotations encapsulate the geological knowledge and understanding of the geological object scale and act as the learning criteria the model will be trained upon to detect and accurately predict the desired geology from the 2D outcrop images. Distinguishing geological features from different scales (laminae vs. bedding) is essential to interpreting the depositional environment. Therefore, carefully annotating each object is essential to capture the correct scale per annotated feature. The training and validation datasets predominantly consist of images containing reference objects that facilitate scale determination. In cases where a reference object is absent, the image source provides

valuable information regarding the size of the depicted features or objects, serving as a guide for the annotator.

To minimize the bias associated with the interpretation, the annotation step should ideally be completed by a team of geoscientists. If the labeler is unsure of the label, there is always the option of a label 'unknown.' The label unknown can be used in cases the model encounters unseen features that are not part of the data set or the annotations. Also, it can represent several possible features that look identical, and the model does not know how to classify them. It is preferable if the model outputs a prediction with the label 'unknown' than a wrong one. If the model makes such a prediction, at the very least, it points out a specific part of an outcrop to investigate further, either manually by the geologist or by using additional computer vision methods, such as image classification and segmentation.

In my annotations, such a label was not incorporated as it was a recent idea, and time did not allow for its incorporation. Nevertheless, it will be included in my models as part of further improvements and future work.

6.3.3 Step 3: Feature Extraction

The next step is to extract features from the input images. YOLOv6 uses the architecture described in section 6.2.1 to extract sedimentary structures and fossil types from the input images. During this step, the model automatically learns and recognizes the visual patterns and characteristics associated with geological features, allowing for efficient and accurate detection in geological imagery. The input image is passed through the layers of the backbone network. Each layer performs convolutional operations, pooling, and non-linear transformations to extract increasingly abstract and informative features from the image. These features are representative of different scales and levels of detail.

6.3.4 Step 4: Anchor Box Selection & Object Detection

Anchor boxes are pre-defined bounding boxes that are used to detect objects in an image. YOLOv6 uses anchor boxes of different shapes and sizes to detect objects of different sizes and aspect ratios. The anchor boxes are selected based on the statistics of the ground-truth bounding boxes in the training dataset (annotations).

Once the anchor boxes are selected, YOLOv6 uses a prediction head to predict the probability of each anchor box containing an object and the corresponding bounding box coordinates of the object. The prediction head consists of a series of convolutional layers that predict the mAP score, the class probabilities, and the bounding box coordinates for each anchor box. YOLOv6 also uses a multi-scale prediction strategy to improve object detection accuracy (Wang, et al., 2021). This is ideal for the detection of the geological features from the outcrop and core images, which represent the geology at different scales.

6.3.5 Step 5: Non-Maximum Suppression

After object detection, YOLOv6 performs non-maximum suppression (NMS) to remove duplicate detections (multiple bounding boxes for a single object) and select the best bounding boxes for each object; in other words, it filters redundant or overlapping predictions. NMS is a post-processing step that selects the highest-scoring (highest probability of prediction) bounding boxes for each object based on their overlap with other bounding boxes (Neubeck & Van Gool, 2006).

6.3.6 Step 6: Output Image

The final output of YOLOv6 is a set of bounding boxes that enclose the objects detected in the input image, along with the corresponding class labels and confidence scores. The confidence scores are calculated based on the output of the algorithm's prediction for a given region of an image, which is computed by the mAP scores. This output consists of a set of scores, one for each class of object that the algorithm has been trained to detect. The confidence score for a specific class is calculated as the probability that the object in the region of interest belongs to that class, as output by the algorithm and is represented as a value between 0 and 1, with higher values indicating greater confidence that the object is present in the region of interest and belongs to the predicted class. In most object detection systems and for YOLOv6, the IoU threshold value is adjusted by the user during inference. Any predictions below that value are discarded as false positives. In this Chapter, all the threshold value, for all model runs were set to 0.35. The predicted bounding boxes are visualized on the test image to show the location and size of the detected objects.

6.4 Training and Testing of the YOLOv6-S Model on Geology

Geological structures and fossils exhibit various patterns and shapes, yet they share a common characteristic—they are embedded in rocks, resulting in a degree of similarity in texture. Conversely, everyday objects, such as a ball, a dog, or a table, possess distinct patterns, textures, and shapes. In the context of geology, sedimentary structures have similar fabric within the same outcrop, but they remain distinct entities. Thus, to accurately identify and locate sedimentary structures and fossils from outcrop images, it is essential to train YOLOv6s using a custom dataset.

Two custom datasets were used to train the YOLOv6s model, Datasets 8 and 11, which were both fully described in section 4.8. A brief recap of the datasets characteristics is provided in Table 6-1 below.

Dataset	Data Type	Image Number	Number of Geological Features	Total Number of Annotations	Task
Dataset 8 (D8)	Outcrops	138	23	253	Object Detection
Dataset 11 (D11)	Outcrops	142	7	248	Object Detection

Table 6-1: Characteristics of Datasets 8 and 11.

6.4.1 Experiment 1: Object Detection of Sedimentary Structures on Outcrop Images (Dataset 8)

In this section, the YOLOv6-S model was used to detect sedimentary structures from 2D outcrop images. The model was trained and validated with Dataset 8, and tested on an unseen test set consisting of outcrop images. The model outputs outcrop images, including multiple bounding boxes around the geological objects, with a confidence score (0 to 1) assigned to each prediction.

In evaluating an Object Detection model for a geological task, it is crucial to seek feedback from a human geologist rather than relying solely on metrics like mean Average Precision (mAP) scores and confidence in the final predictions. This is because even if some labels or annotations are incorrect, the model can still produce predictions with high numerical confidence scores that may be inaccurate in terms of geological interpretation. Thus, the evaluation by a geologist becomes vital to assess the model's practical

understanding of geology. To ensure the effectiveness and reliability of my model, I conducted evaluations at both the annotation stage and the test stage.

6.4.1.1 Training hyperparameters

The hyperparameters used for the custom training and validation of the YOLOv6-S model on outcrop data are shown in Table 6-2. The weight file “yolov6s.pt” comprises pre-trained weights on the ImageNet dataset. The chosen number of epochs was 400 in order for the mAP scores of the custom training to be comparable with the mAP scores of the COCO val scores presented by Li et al. (2022). The image size was set to 640 x 640 pixels for all the input images used for training and a batch size of 8, which depends heavily on the hardware available to train the model. Colour augmentation was also used to increase the ability to learn and extract patterns based on texture rather than colour. HSV stands for Hue, Saturation, and Value and is a colour space developed by A. R. Smith in 1978 (Smith, 1978). It was based on intuitive colour properties, often known as the Hexcone Model. This model's colour parameters are hue (H), saturation (S), and lightness (V). The values for hsv_h, hsv_s, and hsv_v were randomly assigned. All the hyperparameters in Table 6-2 were chosen based on the available computing resources and through trial and error. This combination of parameters was found to yield the best results.

Training Hyperparameters	Value
Pretrained weights	yolov6s.pt
Image size	640
Batch size	8
Epochs	400
workers	8
Evaluation interval	20
Gpu count	1
Optimizer	SGD
Learning rate scheduler	Cosine
Learning rate (initial)	0.0032
Learning rate (final)	0.12
Color Augmentation	Value
hsv_h	0.0138
hsv_s	0.664
hsv_v	0.464

Table 6-2: Training hyperparameters used for the custom training of the YOLOv6 model for geologic object detection (Dataset 8).

Table 6-3 and Figure 6-5 show the results of the model’s training and demonstrate the evolution of mAP scores progressively as the number of epochs increases. The mAP was monitored both for IoU 0.5 and IoU 0.5-0.95 and as mentioned earlier, the mean average precision score we focused on is only the mAP@0.5-0.95. It is evident that as the number of epochs increases, the mAP score increases, meaning that the bounding box predictions and assignment of geological classes improve over time. The best mAP@0.5-0.95 is achieved at epoch 360, with a value of 55.5. This score is very satisfying when compared to the performance of YOLOv6-S on the benchmark COCO dataset. Such a comparison can be found in Table 6-4.

Epoch Number	mAP@0.5	mAP@0.5, %	mAP@0.50:0.95	mAP@0.50:0.95, %
20	0.01	0.7	0.00	0.2
40	0.06	6.0	0.03	3.2
60	0.17	16.7	0.09	8.8
80	0.17	16.9	0.10	10.5
100	0.35	35.0	0.25	24.7
120	0.38	37.8	0.24	23.5
140	0.44	43.6	0.30	29.6
160	0.47	47.3	0.33	32.9
180	0.57	56.7	0.41	41.4
200	0.59	59.3	0.41	41.1
220	0.65	65.2	0.46	45.8
240	0.66	65.9	0.48	48.3
260	0.69	69.3	0.51	50.7
280	0.73	73.3	0.52	51.5
300	0.73	73.0	0.54	53.6
320	0.75	74.8	0.54	53.9
340	0.76	76.0	0.55	54.9
360	0.76	75.7	0.56	55.5
380	0.74	74.3	0.54	53.7
400	0.75	75.3	0.53	53.3

Table 6-3: Summary of mAP scores for IoU 0.5. and for IoU from 0.5-0.95 over 400 epochs (Dataset 8).

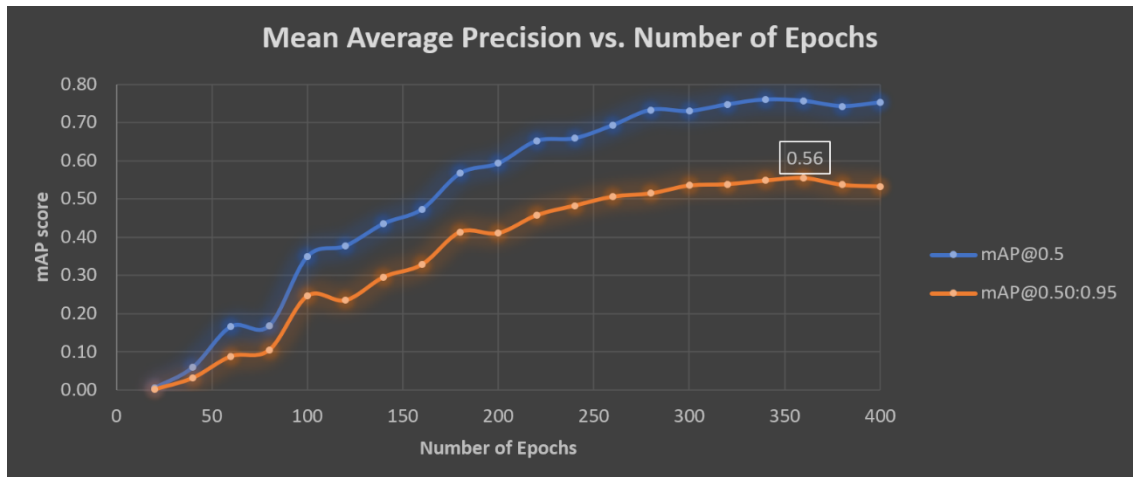


Figure 6-5: Mean Average Precision Scores versus the number of epochs (Dataset 8).

From Figure 6-5 it is obvious that from epoch 200 onwards, the mAP score growth rate starts to decrease and to flatten toward forming a plateau. This is an indication of the model reaching a point where additional epochs do not benefit its learning of the geology any further; therefore, 400 epochs is a good point to stop training the model.

Model	Number of Epochs	mAP 0.5:0.95 (COCO val), %	FPS on Tesla T4 TRT (Batch Size 1)	Parameters ,millions
YOLOv6-S	400	43.8	495	17.2

Model	Number of Epochs	mAP 0.5:0.95 (Dataset 8 val), %	FPS on Tesla T4 TRT (Batch Size 1)	Parameters ,millions
YOLOv6-S for Geology	400	55.5	495	18.51

Table 6-4: Comparison of custom training validation mAP@0.5-0.95 (Dataset 8) with that of the benchmark COCO val dataset.

6.4.1.2 Quantitative Results for the Detection of Sedimentary Structures on Outcrop Data

Object Detection Yolov6 for Sedimentary Structures on Outcrop Data				
Labels/Classes	Label Count in Training and Validation Sets	Predicted Label Appearances in test data	Misclassifications per Label (Class)	Percentage of misclassifications per class, %
Bioturbation	25	59	0	0
Clasts	32	47	7	15
Convoluted/Irregular Bedding	2	4	0	0
Cross Bedding/Stratification	25	41	14	34
Cross Lamination/Climbing Ripples	13	14	3	21
Desiccation Cracks	5	6	0	0
Erosive Features	24	17	1	6
Fault	7	9	0	0
Flame Structures	3	2	0	0
Flaser Lamination	2	0	0	0
Fossils	4	27	1	4
Herringbone Cross Stratification	7	15	11	73
Hummocky Cross Stratification	6	4	1	25
Lenses	13	12	0	0
Lenticular Bedding	5	6	0	0
Lenticular Lamination	4	3	0	0
Planar/Parallel Bedding	26	39	1	3
Planar/Parallel Lamination	15	25	4	16
Structureless	24	34	1	3
Swaley Cross Stratification	2	2	1	50
Syneresis Cracks	2	2	1	50
Wave Ripples/Lamination	5	7	0	0
Wavy Bedding	2	5	0	0
Total	253	380	46	
Total Percentage of misclassifications for Test set, %				12

Table 6-5: Quantitative Results of YOLOv6-S on Outcrop Images (Dataset 8).

Table 6-5 provides a detailed breakdown of the model’s performance per class/label in Dataset 8. This table shows the number of labels present in the training and validation sets, the number of label appearances in the test data, the number of misclassifications per label, and the percentage of misclassifications in the entire test set. The highlighted values, in yellow colour, in Table 6-5 represent the higher percentages (>50%) of the misclassifications per class. The last row (orange colour) of the table shows the total percentage of misclassifications for the entire Test set.

According to Figure 6-6, the top two misclassified classes are Cross Bedding/Stratification, with 14 misclassifications, and Herringbone Cross Stratification, with 11 wrong predictions. The results showed that some of the predictions were completely wrong, as the bounding boxes were out of place, and the predicted labels did not make geological sense in some cases. There were examples of predictions, as discussed in Figure 6-9 a & b, that would be wrong according to the ground truth but accurate on their localization and bounding boxes. Nevertheless, if these misclassifications are examined more closely, one can understand why the model made such mistakes. Herringbone cross stratification and cross-bending are geological features that share certain visual similarities, like patterns of inclined layers and curvature, leading

to misclassifications and incorrect interpretations by the model. To mitigate this confusion, it is important to train machine learning models with a comprehensive and well-annotated dataset that includes a wide range of variations in both herringbone cross stratification and cross-bending.

An important note here is that a single image can contain multiple predictions of different or the same class, meaning that the number of predicted labels is expected to be higher than the total number of images in the training and validation sets; of course, this depends on the size of the test set. In this case, the test dataset included 70 images randomly chosen during the dataset split into training, validation, and test sets. Thus, in 70 outcrop images, there are 380 appearances of geological objects, sedimentary structures in this case. The total percentage of misclassifications across the entire test set was calculated as the ratio of the sum of misclassifications per class (46) over the sum of the predicted label appearances in the test set (380), as shown in Table 6-5. The resulting percentage of misclassifications in the test set adds up to 12%, giving YOLOv6-S an 88% accuracy for the particular test set. A visual illustration of the results shown in Table 6-5 can be found in Figure 6-7, Figure 6-8, and Figure 6-9.

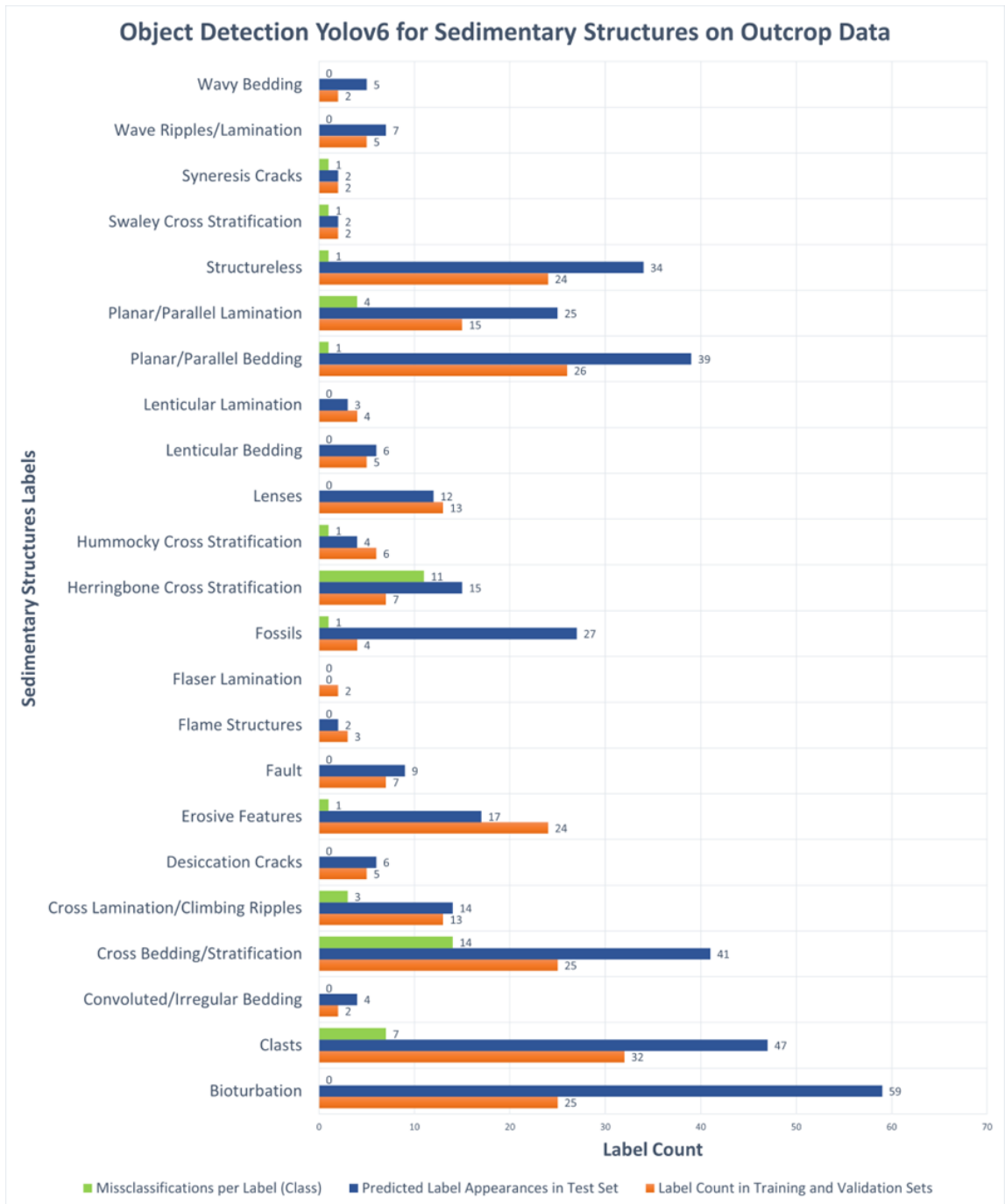


Figure 6-6: Quantitative Results of YOLOv6-S on Outcrop Images (Dataset 8).

6.4.1.3 Qualitative Results for the Detection of Sedimentary Structures on Outcrop data

The quantitative results previously explained in Table 6-5 and Figure 6-6 are illustrated in Figure 6-7 and Figure 6-8, which showcase multiple examples of right and wrong predictions correspondingly. In the abovementioned test set, 22 labels are predicted in 20 images, as shown in Figure 6-7. All these examples were accurately predicted by the model, and to validate the results, each image was cross-referenced with the literature.

To enhance readability and aid in the identification of the model's predictions, coloured labels have been introduced at the bottom section of all figures displaying small text. These labels correspond to the colours assigned to each bounding box label, essentially acting as a legend for the figures.

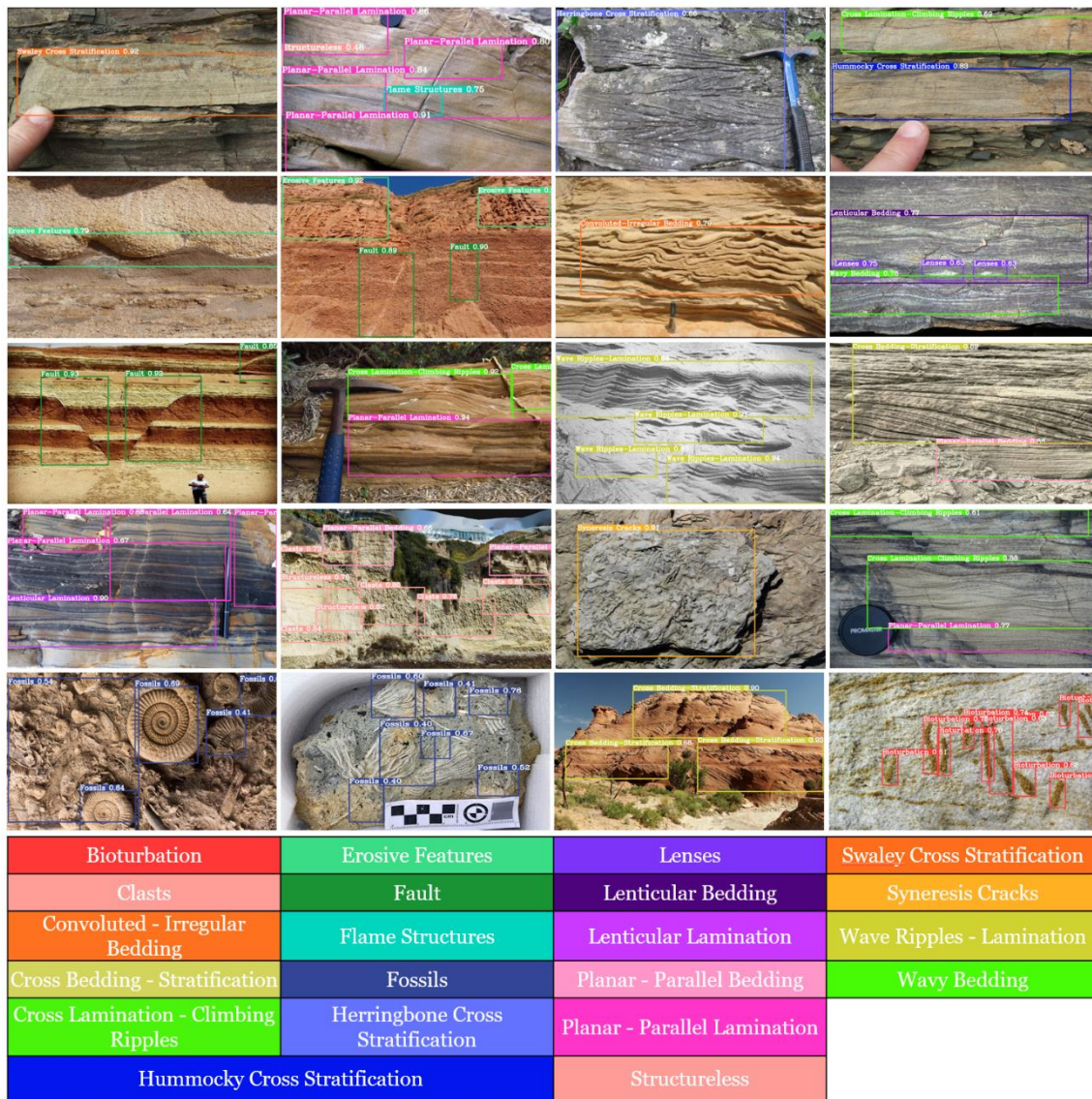


Figure 6-7: Correct Predictions of Sedimentary Structures on Outcrop Images with YOLOv6-S model.

Figure 6-8, on the other hand, portrays images in which the object detection model not only failed to assign the bounding box around the geological objects appropriately but also to assign the correct label and classify the sedimentary structures. This happens for two reasons a) because the patterns of the test images are more complex compared to what the model is trained with, and b) the test images are much different compared to the training sets. Both of these reasons result in misclassifications of the geology, signifying

because the pattern of cross-bedding is widespread in nature as a full feature or part of other entities. For instance, if we look at parts a and b of Figure 6-9, it is evident that the patterns in part b) are rightfully misclassified as the ML model lacks a holistic understanding of the image. The pattern of cross-bedding appears as a series of thin, angled layers or laminations, which are inclined relative to the main bedding plane of the larger deposit. These inclined layers are called cross-strata, and they often show a distinctive curved or S-shaped pattern, with the steeper face of the layer pointing in the direction of current flow at the time the sediment was deposited. The patterns present in all four images in Figure 6-9b are similar to the cross-bedding pattern, resulting in misclassifications. The model will never overcome this challenge unless provided with hundreds or thousands of examples of each class.

This challenge requires an immediate solution, as gathering and annotating such a large amount of authentic geological images for each class is not feasible. Through much experimentation with the input images and annotations and evaluation of the model's performance, it was found that when we lack rich geological datasets, it is better to train separate models depending on the features we desire to predict. For instance, in Dataset 8, 22 labels represent sedimentary structures, with the 23rd label called Fossils. The term fossils can be distilled down to another hundred or more labels, making the task of object detection even more complex. Thus, in this case, the model should be trained separately with two distinct datasets, one covering only sedimentary structures and the other only consisting of fossil images.

6.4.2 Experiment 2: Object Detection of Fossils (Dataset 11)

In this section, the YOLOv6-S model was used to detect seven different fossil types from images. The model was trained and validated with Dataset 11, and tested on an unseen test set containing fossil images. The model outputs the tested fossil images, including multiple bounding boxes around the geological objects, with a confidence score (0 to 1) assigned to each prediction.

Again, the best way to evaluate an Object Detection model for a geological task is with feedback from a human geologist, not the mAP scores nor the probability of the predictions. The geologist's evaluation is essential only during the annotation step to ensure the model's practical understanding of the geology. However, I performed the

evaluation both during the annotation step and the test stage to ensure that the model I build is working properly and to improve the model’s robustness.

6.4.2.1 Training hyperparameters

The hyperparameters used for the custom training and validation of the YOLOv6-S model on the fossil data are shown in Table 6-6. The weight file “yolov6s.pt” comprises pre-trained weights on the ImageNet dataset described briefly in Chapter 4. The chosen number of epochs was 100, and the mAP scores of the custom training were comparable with the mAP scores of the COCO val scores shown in Table 6-6. The image size was set to 640 x 640 pixels for all the input images used for training and a batch size of 8, which depends heavily on the hardware available to train the model. The same *Colour Augmentation* techniques described in Table 6-2 were used, and the values for hsv_h, hsv_s, and hsv_v were randomly assigned. All the hyperparameters in Table 6-6 were chosen based on the available computing resources and through trial and error. This combination of parameters was found to yield the best results.

Training Hyperparameters	Value
Pretrained weights	yolov6s.pt
Image size	640
Batch size	8
Epochs	100
workers	8
Evaluation interval	5
Gpu count	1
Optimizer	SGD
Learning rate scheduler	Cosine
Learning rate (initial)	0.0032
Learning rate (final)	0.12
Color Augmentation	Value
hsv_h	0.0138
hsv_s	0.664
hsv_v	0.464

Table 6-6: Training hyperparameters used for the custom training of the YOLOv6 model for object detection on fossil images (Dataset 11).

Table 6-7 and Figure 6-10 demonstrate the evolution of mAP scores progressively as the number of epochs increases. The mAP was monitored both for IoU 0.5 and IoU 0.5-0.95, and as mentioned earlier, the mean average precision score we focused on is only the mAP@0.5-0.95. It is evident that as the number of epochs increases, the mAP score increases, meaning that the bounding box predictions and assignment of geological classes improve over time. The best mAP@0.5-0.95 is achieved at epoch 75, with a value of 58.1. This score is very satisfying when compared to the performance of YOLOv6-S on the benchmark COCO dataset. Such a comparison can be found in Table 6-8.

Epoch Number	mAP@0.5	mAP@0.5, %	mAP@0.50:0.95	mAP@0.50:0.95, %
5	0.119	11.9	0.053	5.3
10	0.385	38.5	0.238	23.8
15	0.547	54.7	0.284	28.4
20	0.625	62.5	0.350	35.0
25	0.761	76.1	0.454	45.4
30	0.775	77.5	0.446	44.6
35	0.788	78.8	0.461	46.1
40	0.841	84.1	0.506	50.6
45	0.836	83.6	0.523	52.3
50	0.866	86.6	0.541	54.1
55	0.891	89.1	0.558	55.8
60	0.803	80.3	0.491	49.1
65	0.751	75.1	0.453	45.3
70	0.901	90.1	0.569	56.9
75	0.899	89.9	0.581	58.1
80	0.861	86.1	0.546	54.6
85	0.875	87.5	0.547	54.7
90	0.901	90.1	0.559	55.9
95	0.865	86.5	0.563	56.3
100	0.744	74.4	0.450	45.0

Table 6-7: Summary of mAP scores for IoU 0.5. and for IoU from 0.5-0.95 over 100 epochs (Dataset 11).

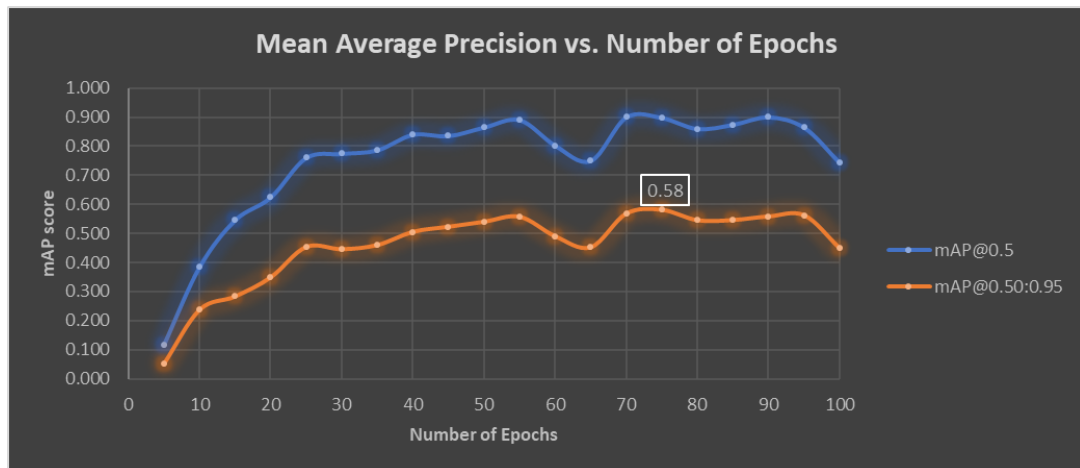


Figure 6-10: Mean Average Precision Scores versus the number of epochs (Dataset 11).

From Figure 6-10, it is evident that this model started to converge much quicker compared to the one in the previous section. While it took 200 epochs for the previous model to start entering a plateau for the mAP, in this case, it only took 25 epochs to observe that phenomenon. Between epochs 55 and 70, a sudden drop in the mAP score occurs relating to the batch size number. The batch size of 8 means that the model randomly chooses eight images from the training set for every epoch to iterate through. If all these images are complex examples, the model does not learn sufficiently during that particular epoch. According to the graph, the most significant drop in mAP occurred at epoch 65, while its highest value occurred at epoch 75 with a mAP@0.5-0.95 of 0.58. Another mAP drop, similar to the one of epoch 65, happened at epoch 95 where the mAP score reached the value of 0.45. The model's training was completed after 100 epochs. Due to its early mAP plateau, training for a higher number of epochs was unnecessary as it would not improve the mean average precision scores further due to that early convergence. An ML model reaches convergence when it achieves a state during training where loss settles to within an error range around the final value.

Model	Number of Epochs	mAP 0.5:0.95 (COCO val), %	FPS on Tesla T4 TRT (Batch Size 1)	Parameters ,millions
YOLOv6-S	400	43.8	495	17.2

Model	Number of Epochs	mAP 0.5:0.95 (Dataset 11 val), %	FPS on Tesla T4 TRT (Batch Size 1)	Parameters ,millions
YOLOv6-S for Geology	100	58.1	495	18.5

Table 6-8: Comparison of custom training validation mAP@0.5-0.95 (Dataset 11) with that of the benchmark COCO val dataset.

6.4.2.2 Quantitative Results for the Detection of Fossils

Table 6-9 provides a detailed breakdown of the model’s performance per class/label present in Dataset 11. This table shows the number of labels present in the training and validation sets, the number of label appearances in the test data, the number of misclassifications per label, and the percentage of misclassifications in the entire test set. The last row (orange colour) of the table shows the total percentage of misclassifications for the entire Test set.

Object Detection Yolov6 for Fossils				
Labels/Classes	Label Count in Training and Validation Sets	Predicted Label Appearances in test data	Misclassifications per Label (Class)	Percentage of misclassifications per class, %
Ammonite	52	40	2	5
Animal Fossil	17	11	4	36
Belemnite	43	31	0	0
Coral	25	11	2	18
Crinoid	63	7	2	29
Plant Fossil	27	12	0	0
Trilobite	21	11	0	0
Total	248	123	10	
Total Percentage of misclassifications for Test set, %				8

Table 6-9: Quantitative Results of YOLOv6-S on Fossil Images (Dataset 11).

According to Figure 6-11, the top class that is misclassified is the Animal Fossil Class, with four misclassifications, while the classes Crinoid, Coral, and Ammonite are only misclassified twice each. The results showed that some predictions were wrong and did not make geological sense both in terms of the predicted bounding boxes and assigned labels. Like before, an important note here is that a single image can contain multiple predictions of different or the same class, meaning that the number of predicted labels is expected to be higher than the total number of images in the training and validation sets; of course, this depends on the size of the test set. In this case, the test dataset included 70 images of fossils randomly chosen during the dataset split into training, validation, and

test sets. Thus, in 70 outcrop images, there are 123 appearances of geological objects, fossils in this case. The total percentage of misclassifications across the entire test set was calculated as the ratio of the sum of misclassifications per class (10) over the sum of the predicted label appearances in the test set (123), as shown in Table 12. The resulting percentage of misclassifications in the test set adds up to 8%, giving YOLOv6-S a 92% accuracy for the particular test set. A visual illustration of the results shown in Table 6-9 can be found in Figure 6-12, and Figure 6-13.

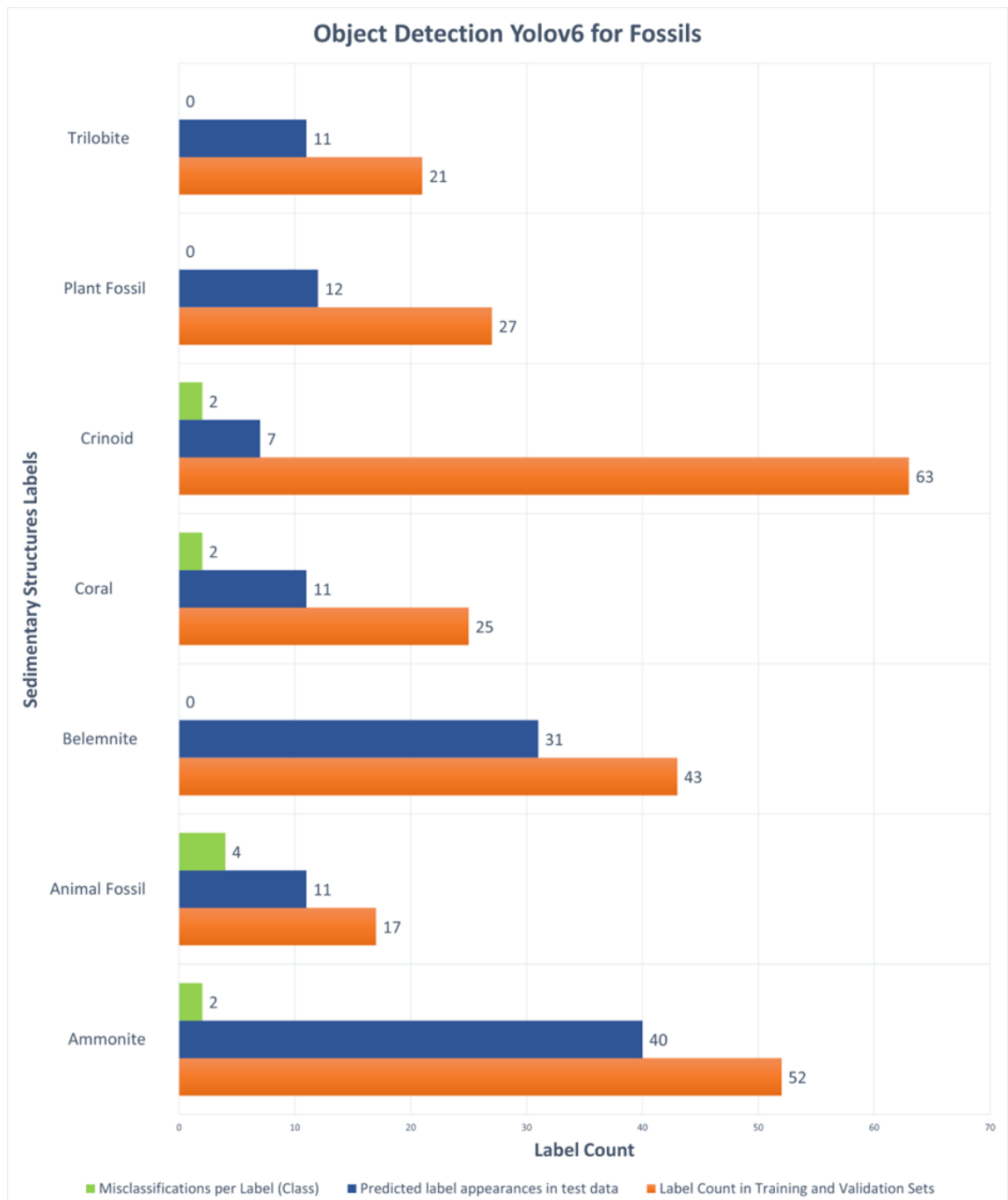


Figure 6-11: Quantitative Results of YOLOv6-S on Outcrop Images (Dataset 11).

6.4.2.3 Qualitative Results for the Detection of Sedimentary Structures on Outcrop Data

The quantitative results explained in Table 6-9 and Figure 6-11 are illustrated in Figure 6-12 and Figure 6-13, which showcase multiple examples of right and wrong predictions correspondingly. In the abovementioned test set, seven labels are predicted in 24 images, as shown in Figure 6-12. The model accurately predicted all these examples, and for validation reasons, each image was cross-referenced with the literature.



Figure 6-12: Correct Predictions of fossils with the YOLOv6-S model.

In Figure 6-12, all the predictions with the label ‘Animal Fossils’ were correctly identified and classified in the particular images. This was achieved with YOLOv6-S model when trained on Dataset 11, which consists only of fossil images across training and validation sets. On the contrary, when the same model is trained with a dataset including various

fossil types under a generic label ‘Fossils,’ it does not yield correct results for detecting fossils as previously shown in Figure 6-9b.

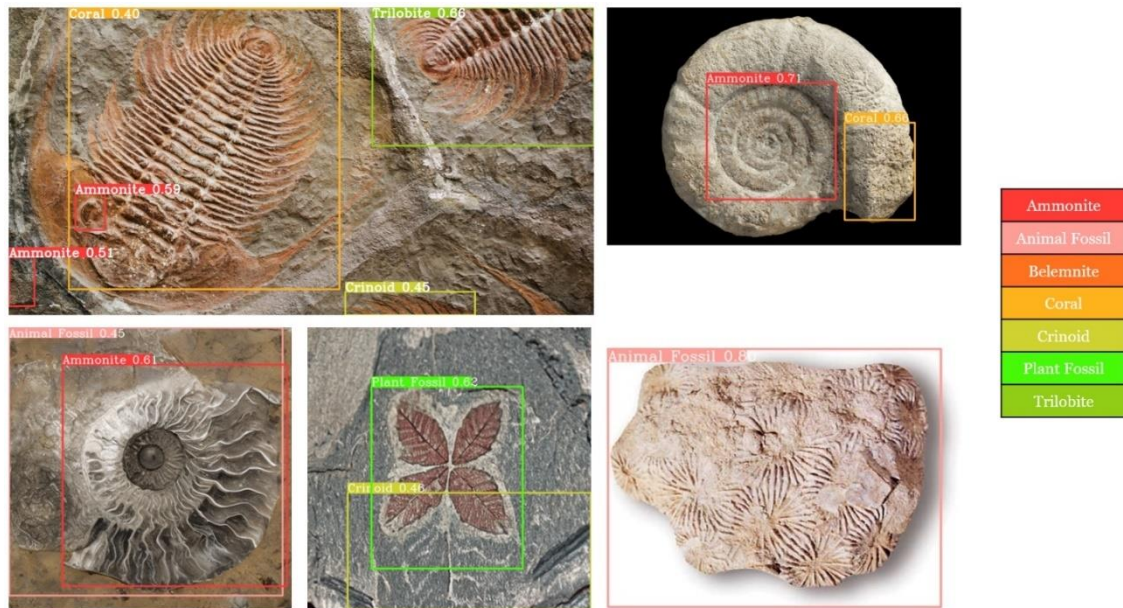


Figure 6-13: Inaccurate Predictions of fossils with the YOLOv6-S model.

Figure 6-13, on the other hand, portrays images in which the object detection model failed to assign the bounding box around the geological objects appropriately and to assign the correct label and classify the sedimentary structures.

6.4.3 Experiment 3: Object Detection of Sedimentary Structures on Core Images

The final section of this chapter showcases an application of the previously trained YOLOv6-S model described in section 6.4.1, aiming to evaluate the model’s capability of transferring its geological knowledge from the outcrops to core data. While this model was trained and validated with Dataset 8, which consists exclusively of outcrop images, this time, the model is challenged to detect sedimentary structures but on core data. In other words, the test set of this section consists only of core images, selected from a collection of core data from SEPM (Society for Sedimentary Geology). The particular test set was assembled from images belonging to the Blaze Canyon 1 dataset from the SEPM webpage. The key message of this section is that core images represent the fragmental geological evidence from the subsurface. Such fragmental evidence is at a different, much smaller scale than that of the outcrops, which are laterally extensive. The application of the previously trained YOLOv6-S model shows how the model can apply

the geological knowledge from the outcrops to the much smaller core samples and still make good predictions.

6.4.3.1 Quantitative Results for the Detection of Sedimentary Structures on Core Images

Object Detection Yolov6 for Sedimentary Structures on Core Data				
Labels/Classes	Label Count in Training and Validation Sets	Predicted Label Appearances in test data	Misclassifications per Label (Class)	Percentage of misclassifications per class, %
Bioturbation	25	5	5	100
Clasts	32	20	0	0
Convoluted/Irregular Bedding	2	0	0	0
Cross Bedding/Stratification	25	57	14	25
Cross Lamination/Climbing Ripples	13	6	0	0
Desiccation Cracks	5	0	0	0
Erosive Features	24	28	1	4
Fault	7	0	0	0
Flame Structures	3	0	0	0
Flaser Lamination	2	0	0	0
Fossils	4	0	0	0
Herringbone Cross Stratification	7	4	4	100
Hummocky Cross Stratification	6	1	1	100
Lenses	13	0	0	0
Lenticular Bedding	5	3	0	0
Lenticular Lamination	4	5	1	20
Planar/Parallel Bedding	26	10	1	10
Planar/Parallel Lamination	15	14	4	29
Structureless	24	12	4	33
Swaley Cross Stratification	2	0	0	0
Syneresis Cracks	2	0	0	0
Wave Ripples/Lamination	5	0	0	0
Wavy Bedding	2	3	1	33
Total	253	168	36	
Total Percentage of misclassifications for Test set, %				21

Table 6-10: Quantitative Results for the Detection of Sedimentary Structures on Core Images.

Table 6-10 provides a detailed breakdown of the model's performance per class/label present in Dataset 8. This table shows the number of labels present in the training and validation sets, the number of label appearances in the test data, the number of misclassifications per label, and the percentage of misclassifications in the entire test set. The highlighted values, in yellow colour, in Table 6-10 represent the higher percentages (>50%) of the misclassifications per class. The last row (orange colour) of the table shows the total percentage of misclassifications for the entire Test set.

According to Figure 6-14, the most misclassified class is the Cross Bedding/Stratification Class, with 14 misclassifications, with the Bioturbation class being the second in the list with 5 misclassifications, while the classes Structureless, Herringbone Cross Stratification, and Planar/Parallel Lamination exhibit 4 or fewer misclassifications each. The results showed that some predictions were wrong, dislocated, or did not make geological sense. Like before, an important note here is that a single image can contain

multiple predictions of different or the same class. In this case, the test dataset included 40 images of core samples selected from a collection of core data from SEPM (Society for Sedimentary Geology). In these 40 core images, there are 168 appearances of geological objects, sedimentary structures in this case. The total percentage of misclassifications across the test set was calculated as the ratio of the sum of misclassifications per class (36) over the sum of the predicted label appearances in the test set (168), as shown in Table 6-10. The resulting percentage of misclassifications in the entire test set adds up to 21%, giving YOLOv6-S a 79% accuracy for the particular test set. A visual illustration of the results shown in Table 6-10 can be found in Figure 6-15 through Figure 6-19.

The performance and accuracy of the model are impressive, considering that it was trained on outcrop data and managed to make several correct predictions on core data, which is a significantly different data type, meaning that the sedimentary structures in the outcrop are depicted differently compared to the core, in terms of the fabric. All this proves that YOLOv6-S has the ability to generalize surprisingly well between the two tested geological data types. These results open the horizons for applying YOLOv6 in geological tasks. A part of future work related to this chapter will be to train this model on a larger and more variable dataset, with many more images and labels of fossils and sedimentary structures, leading to a more robust model for detecting geological sedimentary structures and fossil types.

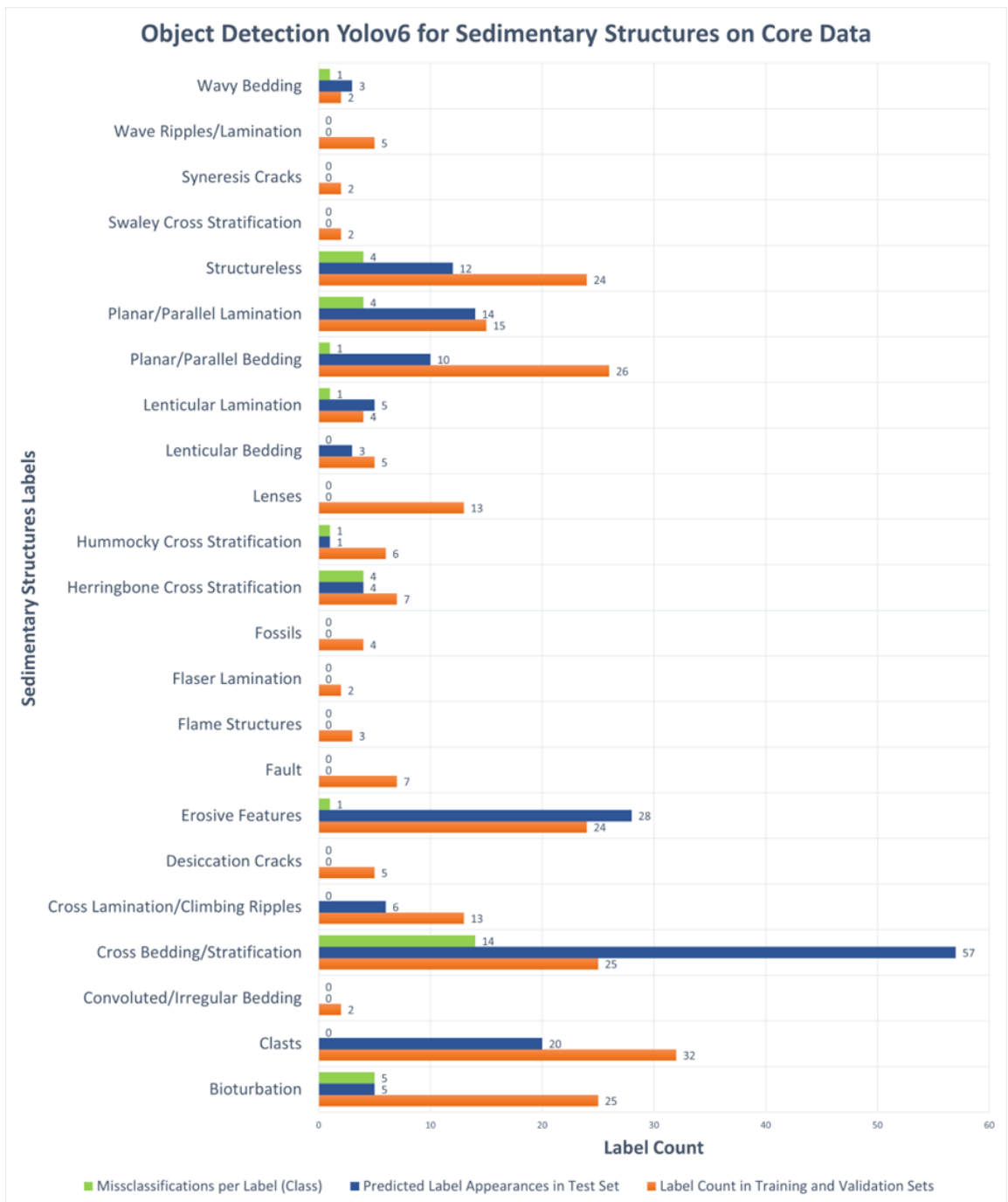


Figure 6-14: Quantitative Results for the Detection of Sedimentary Structures on Core Images.

6.4.3.2 Qualitative Results for the Detection of Sedimentary Structures on Core Images

This subsection presents the qualitative results of Table 6-10. Figure 6-14 demonstrates the performance of the YOLOv6-S model on core data while it has only been trained and validated with Dataset 8, including solely outcrop images. There are several misclassifications of the Cross Bedding/Stratification Herringbone Cross Stratification

labels. In a few instances, both labels are predicted where they do not exist, and in other occasions, the Cross Bedding label is misclassified. For instance, in this entire test set, the Herringbone Cross Stratification class should not have been predicted a single time because it does not exist in the images. The model was tested against nine different core samples, as shown in Figure 6-15.

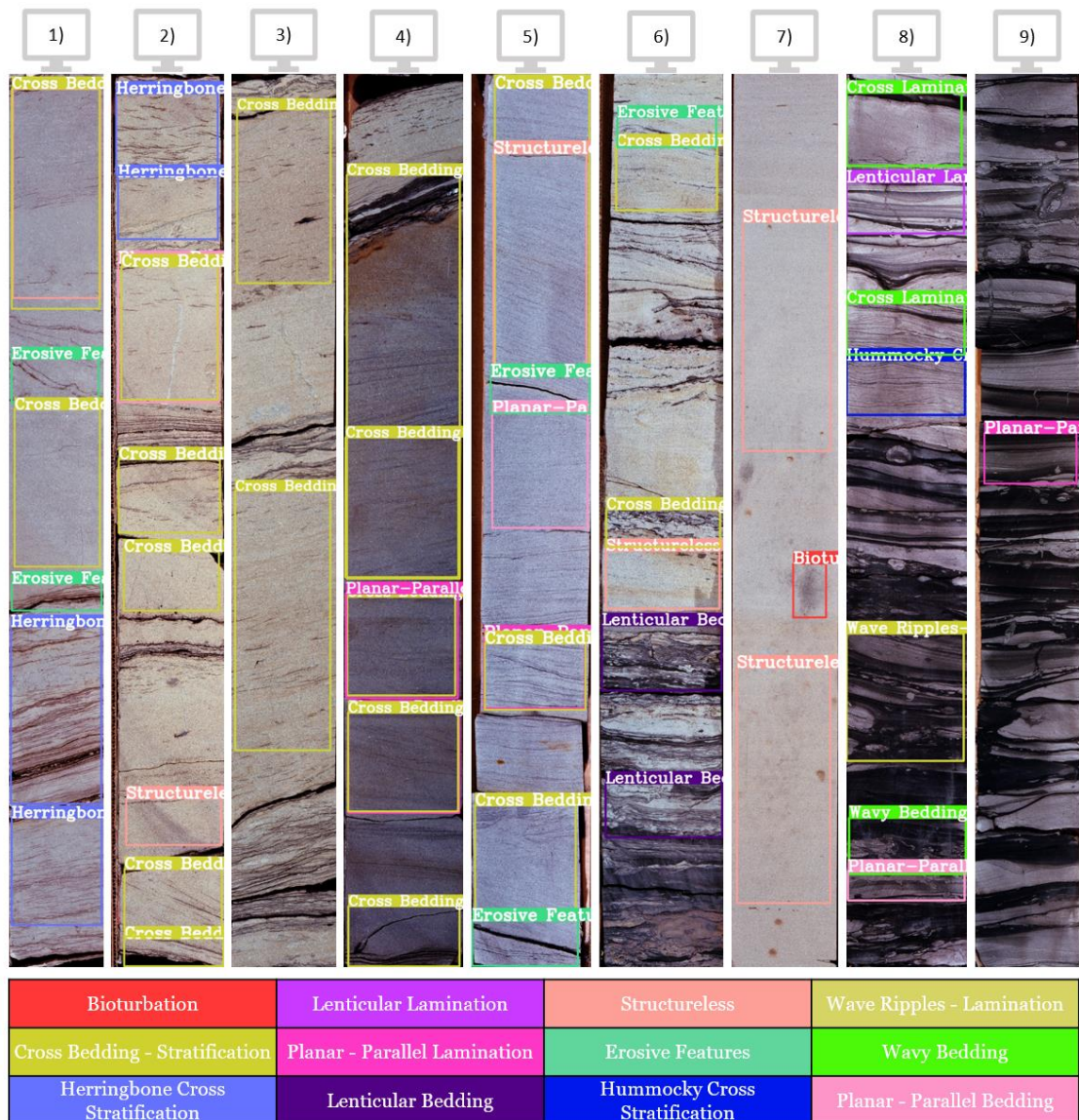


Figure 6-15: Application of YOLOv6-S on nine different core samples.

There is a mixture of correct and inaccurate predictions. Something important to consider here is the aspect ratio and image size of all these 9 test images. While YOLOv6-S has been trained and validated with images of 640 x 640 pixels (aspect ratio 1:1), all these core images have an average size of 500 x 3500 pixels (aspect ratio 1:7). This difference can severely hinder the model's performance and affect its predictions. To account for

this additional challenge, each of the nine images was split into two equals, a top, and a bottom part, to decrease the image size down to 500 x 1750 pixels (aspect ratio 2:7) and reapply the object detection model to each image respectively and compare the results. The nine images were split into three sets of three images for each set, as shown in Figure 6-16, Figure 6-17, and Figure 6-18.

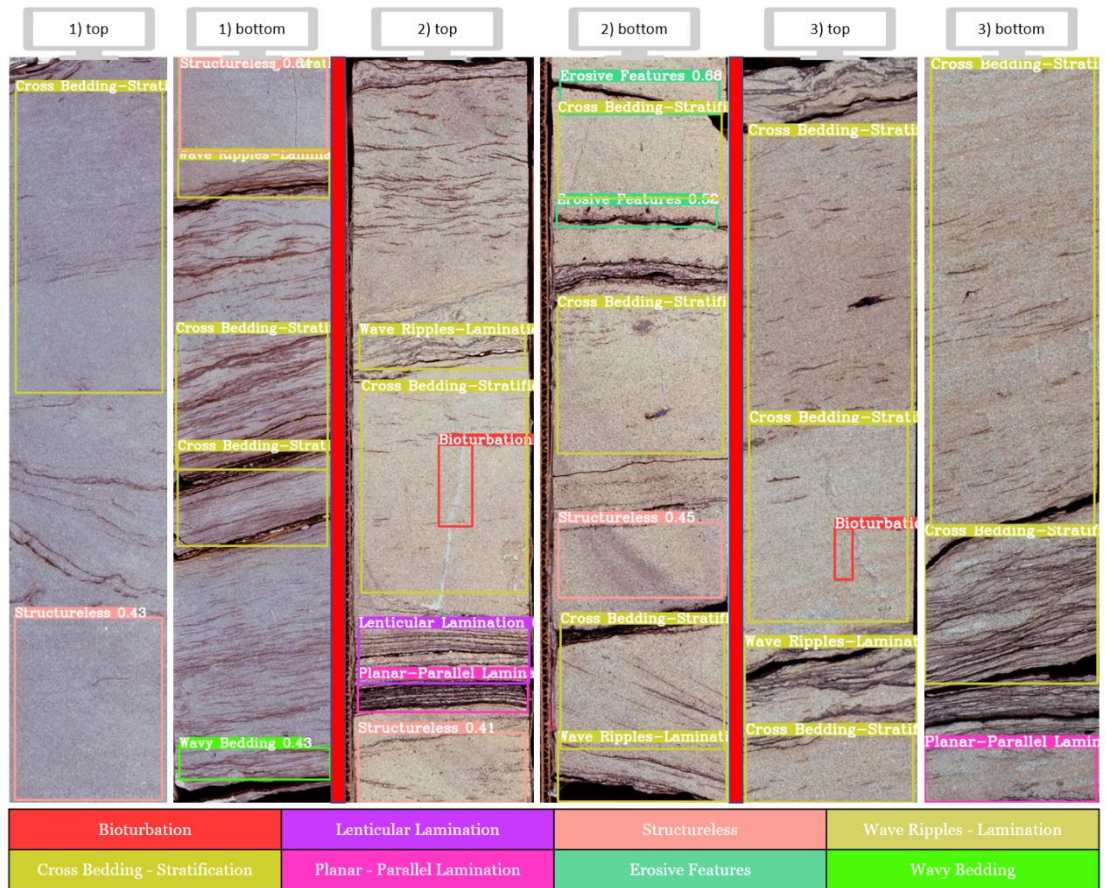


Figure 6-16: Results of the application of the YOLOv6-S model on the top and bottom parts of core samples 1, 2 and 3.

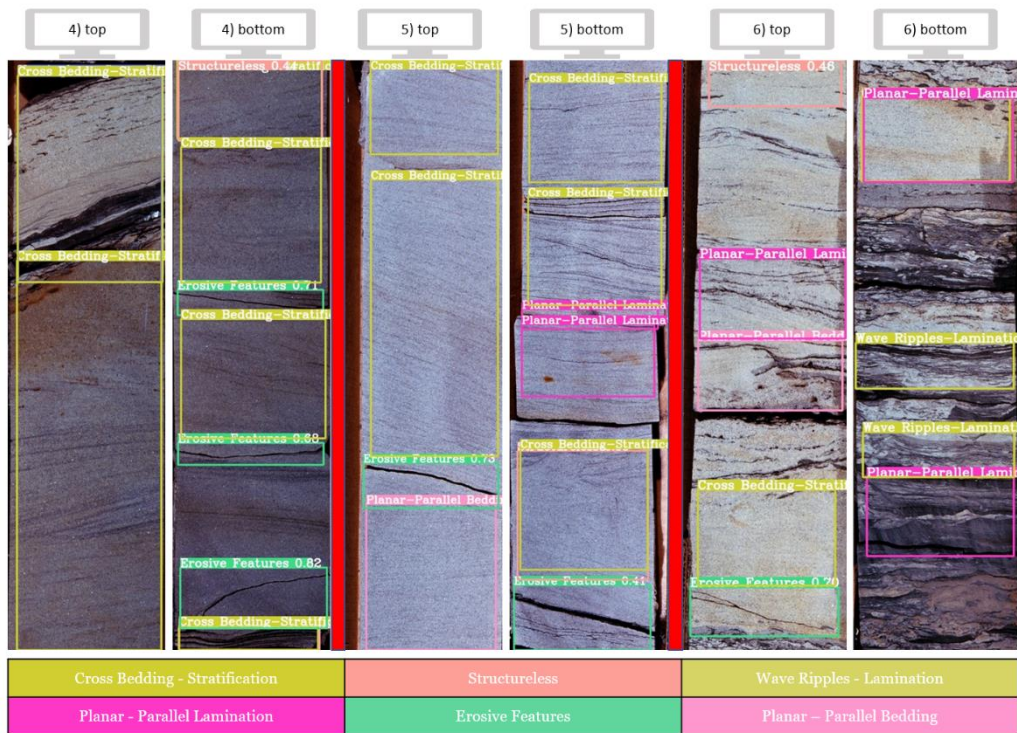


Figure 6-17: Results of the application of the YOLOv6-S model on the top and bottom parts of core samples 4, 5 and 6.

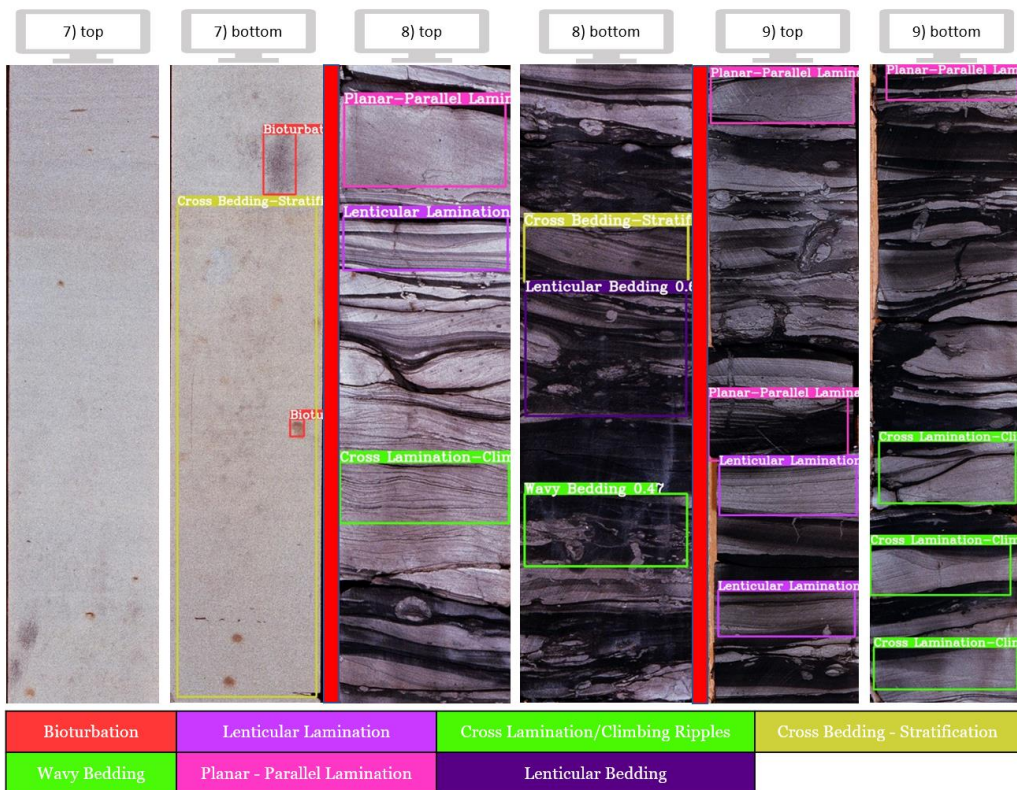


Figure 6-18: Results of the application of the YOLOv6-S model on the top and bottom parts of core samples 7, 8 and 9.

An important observation derived from Figure 6-16, Figure 6-17, and Figure 6-18 is that the split and reapplication of the model individually to each part of the original images yielded more accurate predictions and reduced the number of misclassifications. A crucial achievement from this experiment is that the model no longer predicts labels that do not exist, such as the Herringbone Cross Stratification described earlier. Therefore, the aspect ratio affects the model's predictions, and it is another parameter to consider when compiling the test set for an Object Detection model.

To test the model's detection capabilities even further, several versions of a new core sample were generated by applying different image filters (such as embossing and sharpening) and effects on the image to alter its texture. Then YOLOv6-S was applied to all of the instances of this new core example, along with three additional core images, as shown in Figure 6-19. These additional core images are part of the Salt Wash 1 dataset, again found in the SEPM website. The results are consistent across the image variations, with a minor misclassification occurring on the embossed images, where the Clasts label was predicted instead of Cross bedding.



Figure 6-19: Results of the application of the YOLOv6-S model on the distorted images of core samples.

Despite the applied filters on the images of Figure 6-19, for the first 5 images of the top row, it is obvious that the model predicts exactly the same geological features correctly, and it only misses one cross-bedding label in the middle (more embossed) image, meaning that the model has learned quite well the geological patterns that was trained on with Dataset 8. This led to two additional conclusions for the section: a) the model's performance is not severely affected by the resolution of the test image, and b) the model can generalize well among outcrops and core images because its predictions are based on the learning of the patterns and not the scale due to its feature pyramid network (FPN) architecture. This explains why the model performs well on the core data, although it has never seen a core image in the training and validation sets.

6.5 Conclusions

This chapter discussed using the YOLOv6-S model to detect geological structures in different images, including images of outcrops, cores, and fossils. The model was trained and tested with different datasets (Datasets 8 and 11) to monitor the model's predictions of sedimentary structures and fossils across the two data sources (outcrops and core images).

To do so, three similar experiments were set up, in terms of experiment structure and result presentation. Each experiment consisted of two sub-sections, one showing the quantitative analysis of the model's predictions and misclassifications, summarized in tables, while the other was translating the numbers of the quantitative results into a qualitative analysis by showing visual examples of the geology and the model's predictions.

In Experiment 1, the YOLOv6-S model was used to detect sedimentary structures from 2D outcrop images. The model was trained and validated with Dataset 8 and tested on an unseen test set consisting again of outcrop images. The resulting percentage of misclassifications in the test set adds up to 12%, giving YOLOv6-S an 88% accuracy for the particular test set.

In Experiment 2, the YOLOv6-S model was used to detect seven different fossil types from images. The model was trained and validated with Dataset 11 and was tested on an unseen test set containing fossil images. The resulting percentage of misclassifications in the test set adds up to 8%, giving YOLOv6-S a 92% accuracy for the particular test set.

Experiment 3 showcases the application of the previously trained YOLOv6-S model described in experiment 1, aiming to evaluate the model's capability of transferring its geological knowledge from the outcrops to core data. While this model was trained and validated exclusively on outcrop images, this time, the model is challenged to detect sedimentary structures but on core data. The key message of this section is that core images represent the fragmental geological evidence from the subsurface. Such fragmental evidence is at a different, much smaller scale than that of the outcrops, which are laterally extensive. The application of the previously trained YOLOv6-S model shows how the model can apply the geological knowledge from the outcrops to the much smaller core samples and still make good predictions. The resulting percentage of misclassifications in the entire test set adds up to 21%, giving YOLOv6-S a 79% accuracy for the particular test set.

The challenges the model faced regarding its performance and accuracy of its predictions were associated with the detection of particular sedimentary structures, such as cross-bedding and herringbone cross-stratification, which were the two most misclassified classes among all classes. The richness and variability of the datasets used to train and validate the model are directly related to the percentage of misclassifications. The greater the variability and size of the training set, the better the model's predictions and generalization. Finally, the aspect ratio of the images in the test set affects the model's detection capabilities if it differs significantly from the model's default aspect ratio (1:1).

The findings of sections 6.4.1 and 6.4.2 suggest using separately trained models with different datasets (D8 and D11) to detect different features to improve the accuracy of geological object detection. Sedimentary structures and fossils are two broad categories with numerous subclasses. Combining all these subclasses under a single dataset would result in an imbalanced dataset of multiple classes, requiring hundreds of examples for each class, resulting in a huge dataset, which due to the limited data and computational resources during my Ph.D., was not feasible to assemble nor use. Section 6.6.1 provides a solution to the aspect ratio challenge, which altered the aspect ratio by splitting the elongated core images into two equal parts, reducing the disproportionate aspect ratio. The application of YOLOv6-S on each separate image yielded improved results.

This chapter accentuates the importance of human evaluation during the annotation stage to ensure the model's practical understanding of geology. The study concludes that YOLOv6-S has the potential to transfer geological knowledge from outcrop to core data

and generalize well across different geological data types. Although YOLOv6-S's performance is impressive according to its prediction scores for each experiment (88%, 92%, and 79%, respectively), one thing that it is not able to do is predict the lithology in the outcrop images. It can undoubtedly assign bounding boxes and labels around the lithology types and layers; however, this does not provide the geologists sufficient information as they also need to know and be able to define clear boundaries and distinctions between the various lithologies, an essential element of the outcrop interpretation and, consequently, that of the depositional environment. On an outcrop image, the bounding boxes unavoidably capture more information and additional objects that are not important. Therefore, there is a need for another Computer Vision method, able to take Object Detection a step further by counting and capturing the detailed shape of each object in the image and estimating the lithology types in addition to its localization and labeling that Object detection offers. Such a method is Instance Segmentation which will be explained in the following Chapter 7.

CHAPTER 7 - LEARNING COMPLEX GEOLOGICAL PATTERNS FROM OUTCROP DATA (2D IMAGES) BY USING IMAGE ANALYSIS WITH INSTANCE SEGMENTATION

7.1 Introduction

This chapter demonstrates, in a series of experiments, how the proposed Instance Segmentation model, called YOLACT (Bolya, et al., 2019), assists geologists in interpreting 2-dimensional (2D) images of outcrops by accurately delineating the boundaries of various sedimentary structures and lithology types in 2D outcrop images, alongside their recognition and localization. As mentioned in Chapter 2, interpreting outcrops is a segmentation problem, where geologists aim to identify diagnostic features to form a comprehensive interpretation. The combination, arrangement, and scale of these diagnostic features are important for the understanding of the depositional environment. The developed workflow in section 7.3, delineates the key geological features of the outcrop automatically from an outcrop image/video.

The suitability of YOLACT (DarkNet53) for outcrop geology segmentation is assessed in Experiment 1. Building on the findings, Experiment 2 focuses on refining and improving the segmentation outputs by modifying the YOLACT model. Experiment 3 takes a comparative approach, conducting a study that evaluates the performance of various YOLACT models with different backbones. Finally, Experiment 4 examines the generalization ability of the trained YOLACT (cDarkNet53) model by subjecting it to testing on core images. Through these progressive experiments, the effectiveness and adaptability of YOLACT models in geology segmentation are explored and enhanced. The datasets used to train and validate the model for the above experiments were Datasets 9 for Experiment 1, and Datasets 10a, and 10b for Experiments 2 and 3. Experiment 4 used the trained and improved model described in Experiment 2.

According to the chapter's findings, the geological Instance Segmentation model outperforms the Object Detection model, previously described in Chapter 6, in accurately predicting the geology across varying scales, in terms of mAP score. Instance Segmentation also provides more detailed information regarding the shape and location of each geological object and enables the estimation of lithology by using the masks. Furthermore, applying this model to real-time data makes it a novel approach and a valuable tool used in the field for outcrop segmentation on the fly.

In addition, a final comparison of the three computer vision algorithms used in this thesis is presented, highlighting how each tool complements the other and why it is necessary to use all three tools to move to the final chapter.

7.2 The Yolact model

YOLOACT, created by Bolya et al. 2019, is a real-time and single-shot Object Instance Segmentation model that can offer a comprehensive solution for analysing geological imagery. By leveraging its prediction head, the model can accurately detect, classify, and segment various geological elements such as sedimentary structures, lithology, and fossil types. Through the learned mask coefficients, YOLACT generates precise instance masks that outline the boundaries of each feature, allowing for pixel-level segmentation. The mask prototypes provided a priori, serve as representative templates, aiding in the initial shape estimation of specific objects or classes. This integration enables efficient and automated geological feature extraction, facilitating tasks such as lithology mapping, facies analysis, geological structure identification, and fossil recognition. The structure of the YOLACT model adapted to geology is shown in Figure 7-1.

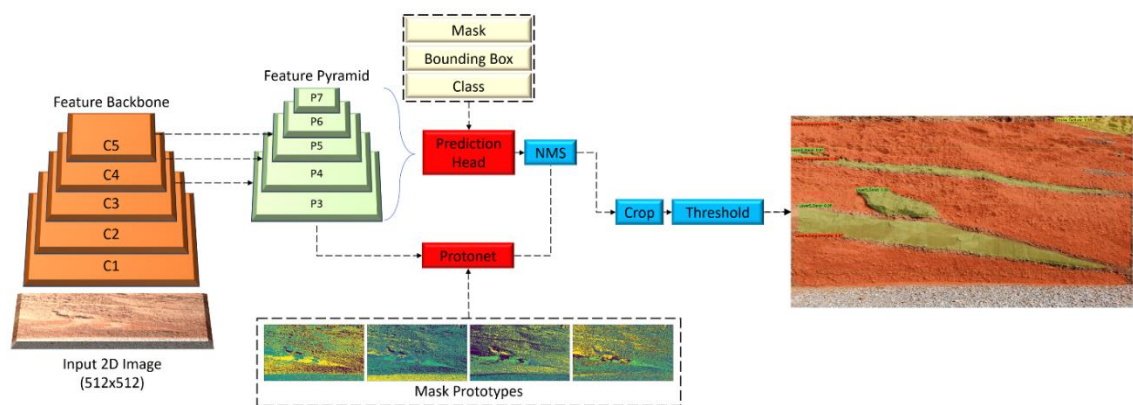


Figure 7-1: custom YOLACT model functionality inspired from Bolya et al. 2019.

The essential components of YOLACT and their functionality are briefly described below:

Similar to the YOLOv6 model described in Chapter 6, the YOLACT model also uses a *Feature Pyramid Network* (FPN) to address the problem of scale variation in images by creating a feature pyramid containing multi-scale representations of the input image (Lin, et al., 2017). At each level of the pyramid, the network combines the feature maps from the previous level with feature maps from the same level that have been unsampled to

match the spatial resolution of the previous level. This process creates a pyramid of feature maps that can be used to detect objects at different scales in the input image.

The *Mask Branch* is a crucial component of the model's architecture responsible for generating pixel-level masks for each individual object instance within an image (He, et al., 2017). The main objective of the mask branch is to accurately delineate the boundaries of objects and assign a binary mask to each instance. During the inference process, the mask branch takes feature maps extracted by the backbone network and performs spatial upsampling to match the original image resolution (Bolya, et al., 2019).

The *Mask Coefficients* represent a set of learned parameters within the YOLACT model that determine the shape and extent of the masks. They are trained through the optimization process to align the generated masks with the ground truth masks or annotations in the training data. In the case of outcrop images with sedimentary structures, lithology, and fossil types, the mask coefficients would aim to capture the outlines or boundaries of these features. The mask coefficients control the spatial transformation and scaling of the mask prototypes to fit each detected instance accurately (Bolya, et al., 2019).

The *Mask Prototypes* refer to a set of representative masks or templates that are learned during the training of a model. These prototypes serve as initial shapes or patterns from which instance-specific masks can be generated. By providing the initial mask prototypes through the K-means clustering process, the model learns to refine and adapt these prototypes during the training process. During inference, the learned mask prototypes are combined with the mask coefficients, which are learned on a per-instance basis, to generate the final instance masks. The use of mask prototypes allows the model to have a prior knowledge of the expected shapes and structures of different objects or classes. This helps improve the accuracy and efficiency of instance segmentation by providing a starting point for generating instance-specific masks.

The *Protonet*, or ProtoNet, refers to the Prototypical Network, which is a type of deep learning architecture commonly used for few-shot learning tasks (Snell, et al., 2017). Few-shot learning aims to enable models to recognize and classify new instances or classes with limited labeled training examples. Applying ProtoNet in a geological context would allow for the recognition and classification of sedimentological features where data availability is limited.

The *Prediction Head* refers to the final layer or set of layers in the neural network that produces the predictions for various tasks, such as object classification, bounding box regression, and mask generation (Bolya, et al., 2019). Regarding sedimentary structures, lithology, and fossil types, the prediction head plays a crucial role in classifying and localizing these features in an image. The prediction head would output the probabilities or scores for each class of sedimentary structure and provide bounding box coordinates for their localization.

The *NMS*, *Crop*, and *Threshold* steps are functioning as follows:

After the YOLACT model generates object detections and segmentation masks, the *NMS* step is applied to remove redundant or overlapping detections. NMS compares the confidence scores of different detections and suppresses those that exceed a specified threshold, keeping only the highest-scoring detection for each object instance.

Once the NMS step is completed, the *crop* operation is performed. This step involves extracting the regions of interest (ROIs) corresponding to the remaining object detections. The cropped regions are then resized and passed to subsequent processing steps, such as mask refinement or classification.

A *Threshold* value is applied to the mask predictions to obtain a binary mask, where each pixel is either considered part of the object or background. This thresholding operation helps refine the masks and remove unwanted noise or low-confidence predictions.

7.3 Methodology (YOLACT)

The methodology consists of a series of experiments and comparisons conducted to incrementally improve the adaptation of YOLACT [Bolya et al., 2019] for outcrop characterization. YOLACT combines accuracy and speed, enabling real-time outcrop Segmentation in areas not accessible by geologists.

YOLACT was adapted to identify, predict and label the parts of the outcrops by applying colourful masks around the objects, providing useful information and evidence for the depositional environment interpretation. The efficiency of this study will be quantified by the time of the completion of the presented Machine Learning workflow and the quality of the geological insights obtained by fine-tuning the Segmentation model

compared to the ground truth by a geologist. The entire geological Segmentation process (workflow) is illustrated in Figure 7-2.

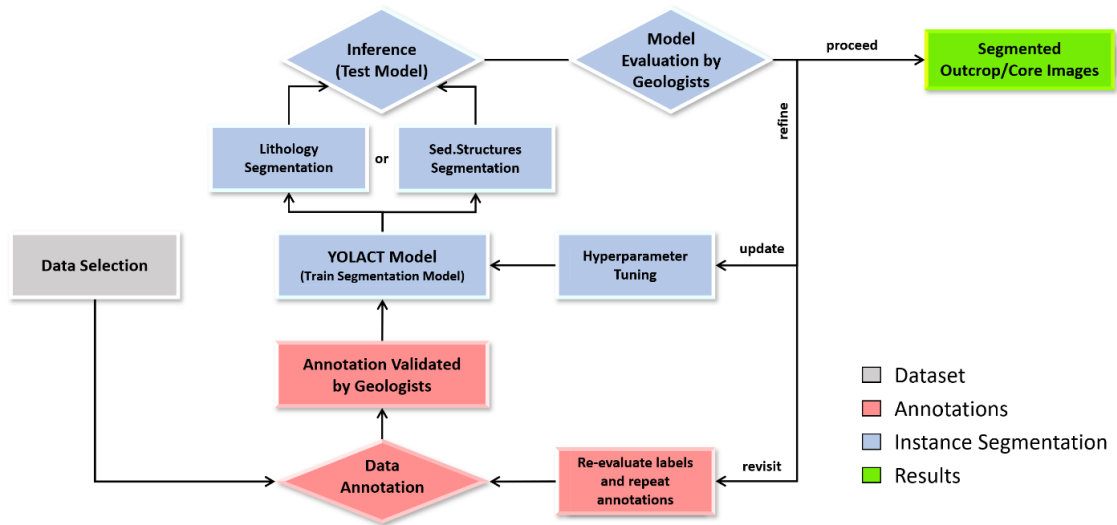


Figure 7-2: High-Level geological Segmentation Workflow.

The initial step is to assemble a dataset according to the workflow in section 4.2. The datasets used in this chapter were 9 and 10 (10a and 10b), consisting of 2-dimensional outcrop photographs capturing various sedimentological features. Three custom datasets were used to train the YOLACT model, Datasets 9 and 10 (10a and 10b), which were fully described in section 4.9. A brief recap of these datasets characteristics is provided in Table 7-1 below.

Dataset	Data Type	Image Number	Number of Geological Features	Total Number of Annotations	Task
Dataset 9 (D9)	Outcrops	62	15	374	Instance Segmentation
Dataset 10a (D10a)	Outcrops	70	13	322	Instance Segmentation
Dataset 10b (D10b)	Outcrops	70	26	545	Instance Segmentation

Table 7-1: Characteristics of Datasets 9, 10a and 10b.

The workflow comprises four main parts: dataset selection, dataset annotation, the Segmentation model training, and the results and evaluation of the Segmentation model.

7.3.1 Step 1: Dataset Selection

Pre-processing techniques like resizing and adjusting the image size are used before splitting the dataset to prepare it for the annotation step. Maintaining the aspect ratios of

the images is important to reduce the loss of information. Rectangular images were divided into two parts to maintain the original image's aspect ratio to ensure there is no distortion of the geological features depicted in the images. An example of some images used to train and test the model is shown in Figure 7-3. According to the results in Experiment 3, if it is desired to enhance the accuracy of the model slightly, the input image size should be set to 550x550 pixels. However, if speed is the priority, images of 512x512 pixels should be used.



Figure 7-3: Sample of training and test images.

7.3.2 Step 2: Dataset annotation

The next and one of the most critical parts of this workflow is the annotation of all the training images that will serve as input into the model along with the images. During the annotation step, the geologist/geoscientist must draw a closed polygon line around the edges of the object of interest and manually label each object for every image in the dataset. An example of such an annotation for geological features is shown in Figure 7-4.

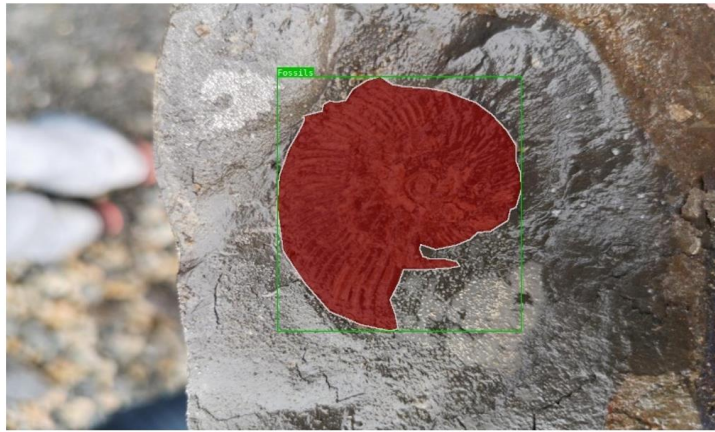


Figure 7-4: Example of polygon annotation for a fossil.

These annotations encapsulate the geological knowledge and understanding of the geological object scale and act as the learning criteria the model will be trained upon to segment and accurately predict the desired geology from the 2D outcrop images.

Distinguishing geological features from different scales (laminae vs. bedding) is essential to interpreting the depositional environment. Therefore, a careful and consistent annotation of each object is necessary to ensure the capturing of the correct scale per annotated feature. Consistent annotation, in this case, refers to how the data is annotated as well as the choice of labels.

In order to minimize bias and ensure accurate interpretation, it is recommended that a team of geoscientists complete the annotation step. When unsure of a label, the 'unknown' label should be used. This is particularly useful when the model encounters features that are not part of the data set or annotations, or when there are multiple features that look identical and the model cannot classify them. It is better for the model to output a prediction with the 'unknown' label than to provide an incorrect one. Such predictions can highlight specific areas of an outcrop that require further investigation by geologists, either manually or with additional computer vision methods such as image classification and segmentation. Although this label was not included in previous annotations due to its recent introduction, it will be incorporated into future models to improve accuracy and ensure comprehensive coverage.

7.3.3 Step 3: Instance Segmentation model training

Moving forward, YOLACT (Bolya, et al., 2019) is trained by adjusting the model's depth (backbone) and hyperparameters. For this study, the DarkKNet53 (Redmon, et al., 2016).

with FPN (Lin, et al., 2017) was used as the default feature backbone, along with an image size of 550×550 pixels, which are the image dimensions Bolya et al. (2019) used in their work as the default values.

A second model training was conducted, but this time the backbone was a custom-modified version of Darknet53 architecture (cDarknet53) with an input image size of 512×512 pixels to develop a comparative study of the results using the two different backbones (*see results section 7.4, experiment 3*).

These two specific backbones were used because, as mentioned in Chapter 3, they are state-of-the-art backbones for tasks such as Segmentation and Object Detection. Both of these architectures were used to build two YOLACT models, one targeting the lithology and the other the sedimentary structures of the outcrops, resulting in 4 models, 2 for lithology and 2 for sedimentary structures prediction and a comparison of the backbone performance.

To enhance the performance of the model's accuracy and/or speed, the same adjustments were made to both backbones and hyperparameters, including batch size, number of workers, validation images used per epoch, number of epochs, learning rate, and learning steps based on the number of iterations before training (section 7.4.2).

7.3.3.1 Instance Segmentation Model (split model)

Attempting to predict both lithology and sedimentary structures simultaneously through a Segmentation model is a tempting prospect. However, such a method results in a severe overlap of masks, rendering the output of the model difficult to discern and thus to interpret, as the results of Experiment 1 show. Deciphering and interpreting these results becomes an arduous and frustrating exercise.

There is no possibility of including both predictions for sedimentary structures and lithology under one label per prediction. Such a combination would result in an extraordinary number of labels and annotations, as it would be needed to cover all possible combinations of all the lithology, sedimentary structures, and fossils. Such an approach would require a vast dataset which would require an excessive amount of time to annotate the data and train this model, making the entire process costly and computationally expensive. Therefore, the Segmentation model must approach each task separately, ensuring accuracy and clarity in the results obtained.

Consequently, as shown in Figure 7-5, I split up the lithology and sedimentary structures tasks. I used two similar models, one to segment the lithology and the other to segment the sedimentary structures of an outcrop. I optimized the workflow by testing various network configurations taking into account both their speed and accuracy.

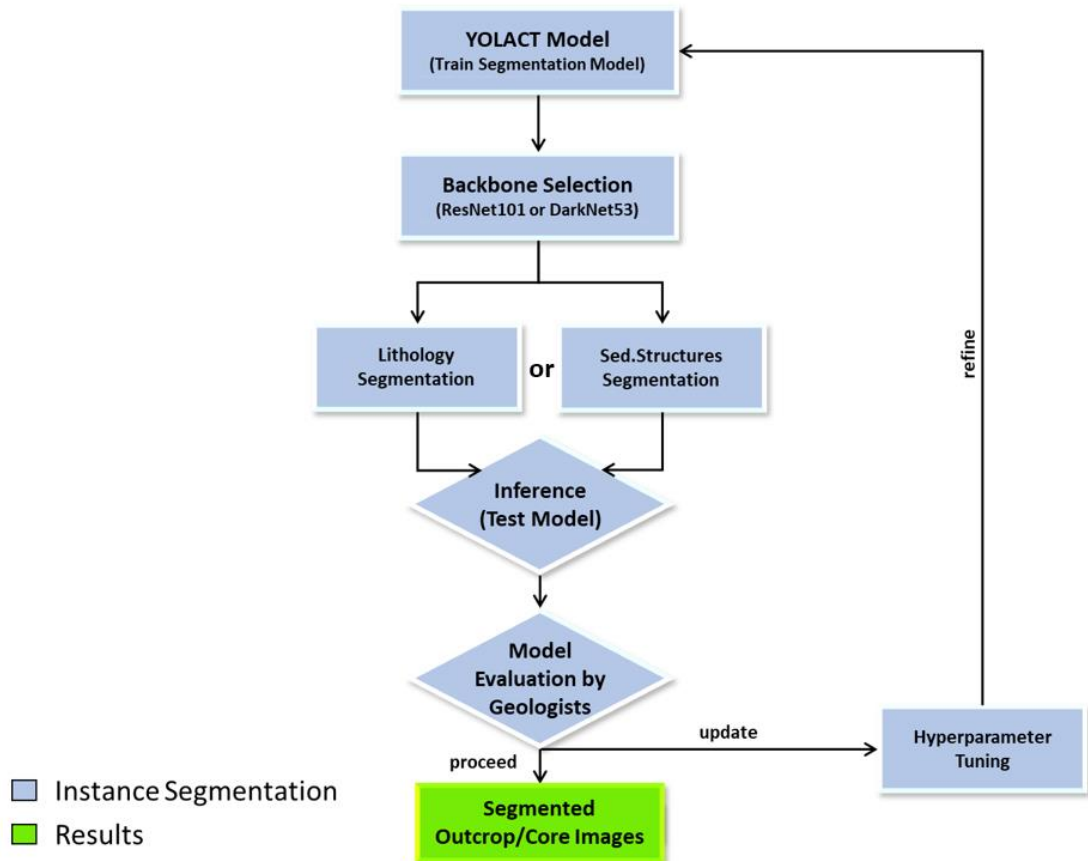


Figure 7-5: A more detailed visual breakdown of the training and inference steps of the workflow in Figure 7-2.

7.3.4 Step 4: Results (evaluation & inference)

The testing stage follows, in which the user provides the model with a set of images and/or videos for the model to make predictions on. The lithology and sedimentary structures are segmented separately for the given test data. The output, generated in a few seconds, is segmented outcrop images and/or videos with multiple colourful masks and bounding boxes around the geological objects in addition to a confidence score (0 to 1) assigned to each prediction. The confidence scores are calculated based on the output of the algorithm's prediction for a given region of an image, which is computed by the mAP scores. As in Chapter 6, the IoU threshold values, for all model runs, were set to 0.35, discarding any predictions below that value as false positives.

The best way to evaluate an Instance Segmentation model for a geological task is by obtaining feedback from a human geologist rather than relying solely on the mAP scores and confidence in the final predictions. This is because even if some labels or annotations are incorrect, the model may still predict the wrong label with high accuracy according to the numerical prediction confidence scores. However, in reality, the predicted label may be geologically incorrect. The geologist's evaluation is essential only once, during the annotation step, to ensure the model's practical understanding of the geology. Nonetheless, to ensure the proper functionality and robustness of my model, I conducted evaluations during both the annotation step and the test stage.

An example of the Segmentation results for lithology and sedimentary structures is shown in Figure 7-6.

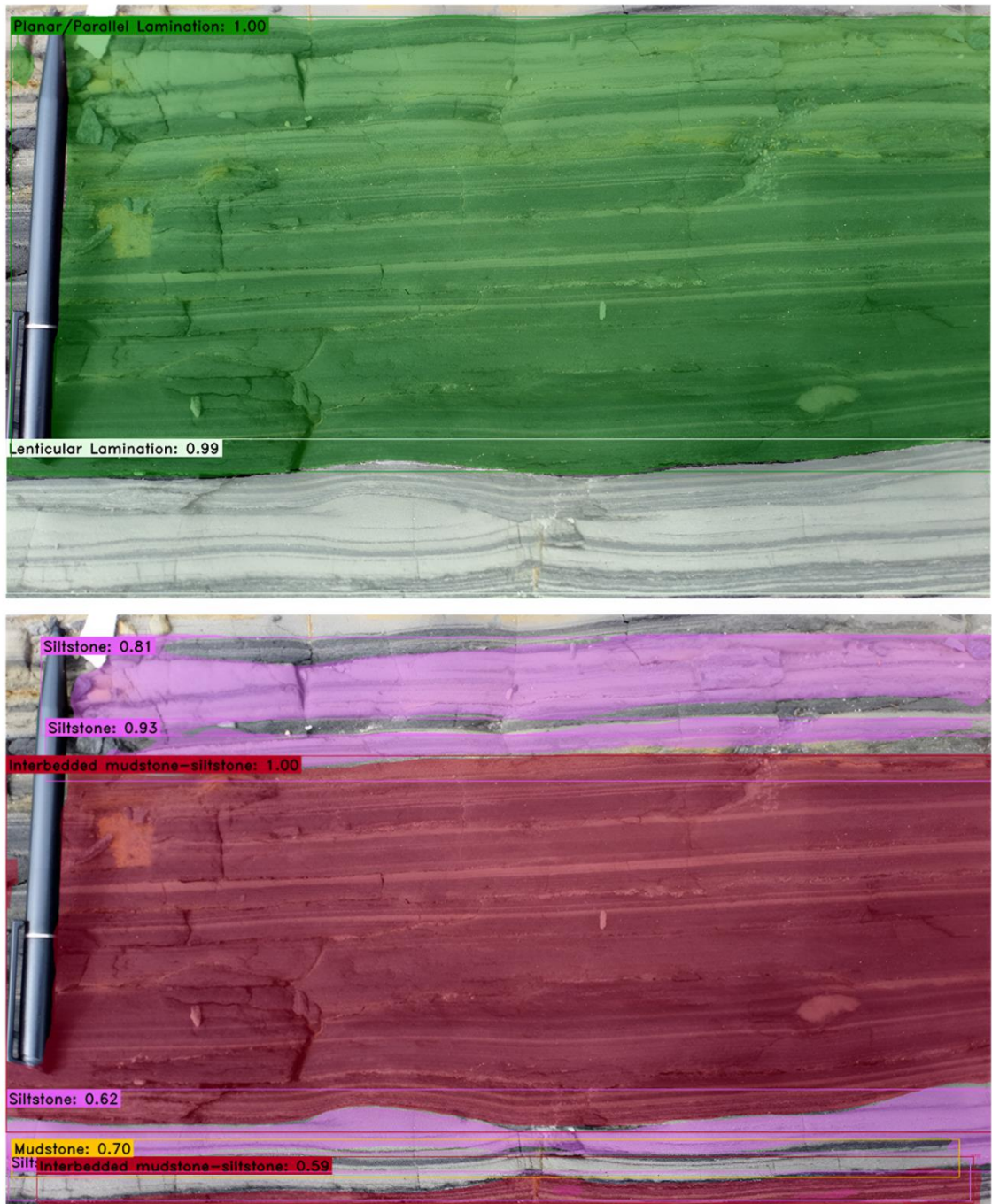


Figure 7-6: Example results for the segmentation of sedimentary structures (top part) and lithology (bottom part).

7.4 Chapter's Findings & Discussion

The results section comprises four experiments, which were performed to test how the default YOLACT model presented by Bolya et al. 2019 performs for the Segmentation of an outcrop, assess its capabilities, improve its performance for the Segmentation of geological features, and finally, test its ability to generalize between different data types.

For each experiment, the objective, key findings, training dataset/data type, test data type, backbone and hyperparameter setup are shown in Table 7-2.

In order to maintain consistency, each experiment follows the same structure as outlined in Chapter 6. This ensures a standardized approach to training and testing the model, allowing for meaningful comparisons and analysis of the results.

Throughout the experiments, we present the findings in both quantitative and qualitative formats, combining numerical data with visual representations.

However, in Experiment 4, there is no training phase involved. Instead, we utilise the trained model from Experiment 2, which was specifically trained on outcrop images. The purpose of Experiment 4 is to examine the application and generalizability of this trained model on core images and videos. By subjecting the model to these different datasets, we can evaluate its performance and assess its adaptability in varying contexts.

Experiments	Objective	Key Findings	Training Dataset/Data Type	Test Data Type	Backbone	Hyperparameter Setup
Experiment 1	Assess the suitability of YOLACT (DarkNet53) for outcrop geology segmentation	YOLACT is a suitable model for this task, but it often misclassifies image features and generates masks with significant overlap	Dataset 9 / Outcrop Images	Outcrop Images	DarkNet53	Table 7.
Experiment 2	Refine and improve segmentation outputs by modifying the YOLACT model	1) Using a shallower version of the Darknet53 backbone (cDarkNet53) improved the model's predictions. 2) Training the model separately on datasets 10a (for lithology) and 10b (for sedimentary structures) yields more interpretable results. 3) Higher dataset variability leads to better and more generalized results on unseen data.	Dataset 10a, 10b / Outcrop Images	Outcrop Images/Video	cDarkNet53	Table 7.
Experiment 3	Conduct a comparative study between YOLACT models with different backbones (cDarkNet53 and ResNet101)	YOLACT (ResNet101) offered slightly better accuracy and mask fit, while YOLACT (cDarkNet53) provided real-time predictions, faster inference, and FPS performance.	Dataset 10a, 10b / Outcrop Images	Outcrop Images	cDarkNet53, ResNet101	Table 7.
Experiment 4	Test the trained YOLACT (cDarkNet53) model on core images to assess its generalization ability	The YOLACT (cDarkNet53) model generalized well on core images without using any core images in training. The Instance Segmentation model demonstrated adaptability and good performance on diverse geological datasets.	-	Core Images/Video	cDarkNet53	

Table 7-2: Objective, key findings, training dataset/data type, test data type, backbone, and hyperparameter for each experiment.

7.4.1 Experiment 1: Application of the default YOLACT model on outcrop data

This first section of the results demonstrates the application of the default YOLACT model on 2D outcrop images. The aim of Experiment 1 is to establish whether YOLACT is a useful tool for segmenting the geological features of an outcrop. In this experiment,

the model simultaneously segments the lithology and sedimentary structures in the test images. The workflow of this subsection, shown in Figure 7-7, is a slightly modified version of the high-level workflow presented in Figure 7-2.

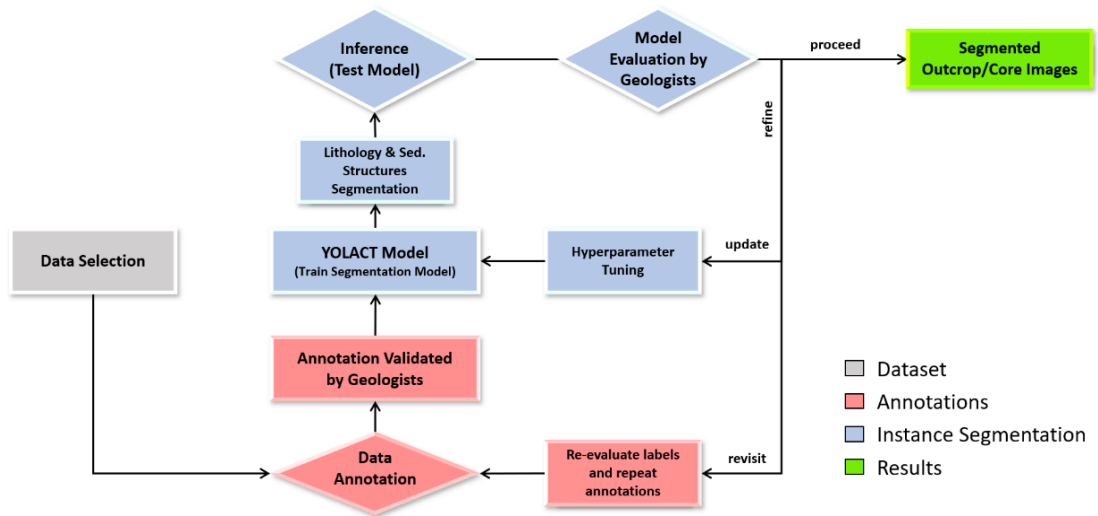


Figure 7-7: Modified version of the Segmentation workflow for the default YOLACT model.

This section’s model is trained on Dataset 9, consisting of 15 labels in total (Table 7-4). During the testing phase, a variety of outcrop images are tested. Some of these images are similar to the ones used for training, originating from the same outcrop. However, they are intentionally kept unseen by the model during the training process. This allows for the evaluation of the model's ability to generalize and accurately classify similar but unseen images from the same outcrop (referred to as partially seen). Additionally, the testing phase incorporates entirely unseen outcrop images. These images are sourced from different outcrops, providing a more challenging assessment of the model's performance. These unseen images assess the model's capability to classify and identify geological features in unfamiliar contexts.

The results, presented later in this section, show that the model makes correct predictions on certain occasions, demonstrating its suitability for the geological segmentation task. However, mask overlapping and misclassifications occur in most cases, making the model predictions difficult to interpret.

The model's accuracy is evaluated by the formation and fit of masks and bounding boxes and the label assignment per instance/geological object within every image. The model

will predict every geological instance in every image, assign a confidence score for each prediction calculated by the mAP scores, and localize each feature by drawing the masks and polygons around them. The model's prediction confidence is calculated with the mAP scores as shown in Figure 7-8, but for the particular Segmentation task, the model's predictions were also evaluated by geologists.

7.4.1.1 Training YOLACT (DarkNet53) for the Segmentation of Geology

The first experiment tested the default YOLACT model using the Darknet53 as a backbone and trained on Dataset 9, where the annotations for lithology types and sedimentary structures were grouped.

It was important first to understand how the model performs with a simple geological dataset consisting of three outcrops and three depositional environments, described in Chapter 4. The choice behind the three outcrops can be justified because, in these outcrops, there was a total of 15 labels (Table 7-4), with easy to medium complexity, serving as a good starting point to test if the model is applicable to geology and that the results produced are satisfying.

The training hyperparameters used for training the YOLACT (DarkNet53) model can be found in Table 7-3. The choice of most of the hyperparameters depended on the computational power available. As for the learning rate and steps, the optimal number of learning steps was determined through trial and error. The learning steps define the number of epochs at which the model changes the learning rate in order to adapt its learning. The initial learning rate was 0.001, which is a good learning rate to start with, as chapters 5 and 6 demonstrated, but also, it is a standard starting point across the machine learning research community. At the designated steps, chosen based on the total number of epochs, the model changed the learning from 0.001 to 0.01 incrementally in four steps. The learning steps help the model learn better by allowing it to adjust its internal parameters based on the training data gradually and can also help prevent overfitting, which occurs when a model becomes too specialized to the training data and does not generalize well to new data.

Training Hyperparameters	Value
Pretrained weights	darknet53.pth
Image size	512
Batch size	8
Epochs	8000
workers	4
Evaluation interval	20
Gpu count	1
Optimizer	SGD
Learning rate	0.001
Learning Steps	2800, 6000, 7000, 7500

Table 7-3: Training hyperparameters used for training the YOLACT (DarkNe53) model.

Figure 7.8 shows the mAP scores of masks and bounding boxes on validation data versus the number of iterations/epochs. As the number of iterations increases, the Mean Average Precision scores also increase, which is the expected behavior. But it is obvious from the Figure that the mAP score reaches a dashing score of 70%, which is very high and almost unrealistic compared to the mAP score of Bolya et al. 2019. As they stated, YOLACT is the first real-time (above 30 FPS) approach with a mask mAP score of 28.89 on the COCO test-dev dataset.

The trained model (Figure 7-8) seems to overfit the data over the first 1000 iterations. After around epoch 1500 onwards, when reaching a plateau, the model shows no further improvement in the mAP scores. This overfitting results probably from the limited dataset used to train the model. Overfitting can occur when a model is trained on a limited dataset that does not sufficiently represent the full range of variability present in the target population or test data. With a limited dataset, the model may memorize specific examples rather than learn the underlying patterns or features that are representative of the broader dataset. As a result, when tested on new or unseen data, the model's performance may deteriorate significantly.

Dataset 9, as explained in Chapter 4, has only 62 images containing 15 different classes of lithology types and sedimentary structures combined. Ideally, a training dataset should have hundreds, if not thousands, of images per class for such a Segmentation model. Due

to the aforementioned, poor performance of the model is expected when tested on data beyond the three outcrops the model is trained with.

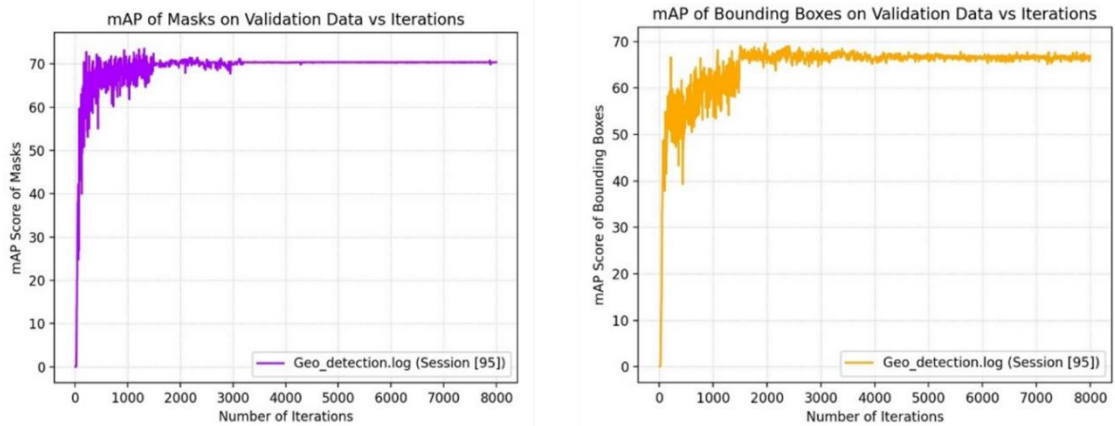


Figure 7-8: The mAP scores of masks and bounding boxes on validation data versus the number of iterations/epochs.

Figure 7-9 shows the overall Loss scores of classes confidence, masks, and bounding boxes on validation data versus the number of iterations/epochs. It is obvious that all three loss scores show a similar trend for the reduction of losses over the number of iterations. The lowest losses score for the class confidence and the box localization is about 0.25, and about 0.5 for the mask loss. Both the loss scores and mAP scores indicate that the Segmentation model has learned the geology well from the three outcrops in Dataset 9; however, its ability to generalize to unseen data is not expected to be satisfying.

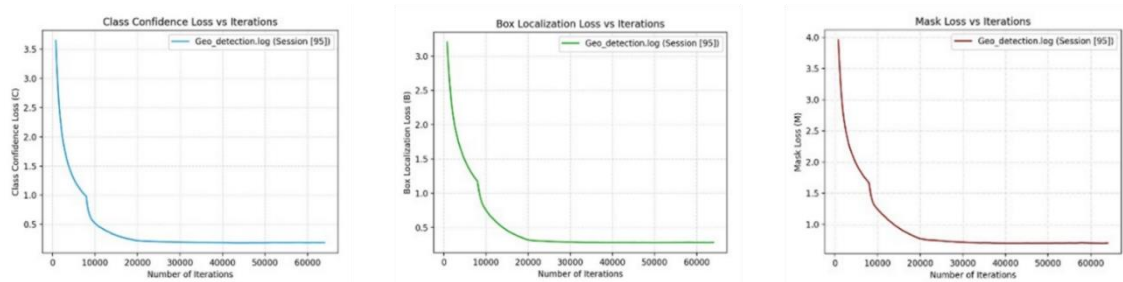


Figure 7-9: Overall Loss scores of classes confidence, masks, and bounding boxes on validation data versus the number of iterations/epochs.

7.4.1.2 Testing YOLACT (DarkNet53) for the Segmentation of Geology

This subsection shows a quantitative and qualitative analysis of the abovementioned Segmentation model’s results when tested on a set of images. This particular test set includes images that are part of the outcrops with which the model has been trained and

validated but belong to unseen sections of the outcrops. Moreover, certain test images are entirely novel to the model as they stem from different outcrops chosen at random.

Table 7-4 and Table 7-5 provide a detailed breakdown of the model’s performance per class/label present in Dataset 9. These Tables show the number of labels present in the training and validation sets, the number of label appearances in the test data, the number of misclassifications per label, and the percentage of misclassifications in the entire test set.

Instance Segmentation Yolact for Lithology & Sedimentary Structures on Partially Seen Data				
Labels/Classes	Label Count in Training and Validation Sets	Predicted Label Appearances in test data	Misclassifications per Label (Class)	Percentage of misclassifications per class
Planar_Bedding	13	22	2	9
Planar_Lamination	6	9	2	22
Cross_Bedding	5	7	3	43
Cross_Lamination	5	4	1	25
Interbedded_Sands	33	29	4	14
Erosive_Feature	34	25	5	20
Cemented_Sands_Eroded_Sands	10	18	7	39
Mudstones	79	52	4	8
Medium_to_Fine_Sandstone	1	6	2	33
Coarse_to_Medium_Sandstone	5	4	1	25
Conglomerate	50	33	4	12
Siltstone	3	3	0	0
Unconformity	12	9	6	67
Rip_up_clasts_Silty_sands	12	5	1	20
Sandstone	103	78	5	6
Total	371	304	47	
Total Percentage of misclassifications for Test set, %				15.5

Table 7-4: Quantitative Results of the default YOLACT model on partially seen outcrop images.

Instance Segmentation Yolact for Lithology & Sedimentary Structures on Unknown				
Labels/Classes	Label Count in Training and Validation Sets	Predicted Label Appearances in test data	Misclassifications per Label (Class)	Percentage of misclassifications per class
Planar_Bedding	13	10	4	40
Planar_Lamination	6	2	2	100
Cross_Bedding	5	4	0	0
Cross_Lamination	5	0	0	0
Interbedded_Sands	33	0	0	0
Erosive_Feature	34	0	0	0
Cemented_Sands_Eroded_Sands	10	10	9	90
Mudstones	79	0	0	0
Medium_to_Fine_Sandstone	1	0	0	0
Coarse_to_Medium_Sandstone	5	0	0	0
Conglomerate	50	0	0	0
Siltstone	3	0	0	0
Unconformity	12	0	0	0
Rip_up_clasts_Silty_sands	12	3	2	67
Sandstone	103	0	0	0
Total	371	29	17	
Total Percentage of misclassifications for Test set, %				58.6

Table 7-5: Quantitative Results of the default YOLACT model on unknown outcrop images.

Table 7-4 and Table 7-5 show the number of labels present in the training and validation sets, the number of label appearances in the test data, the number of misclassifications per label, and the percentage of misclassifications in the entire test set. The highlighted values, in yellow colour, in the Tables represent the higher percentages (>50%) of the misclassifications per class. The last row (orange colour) of the Table shows the total percentage of misclassifications for the entire Test set.

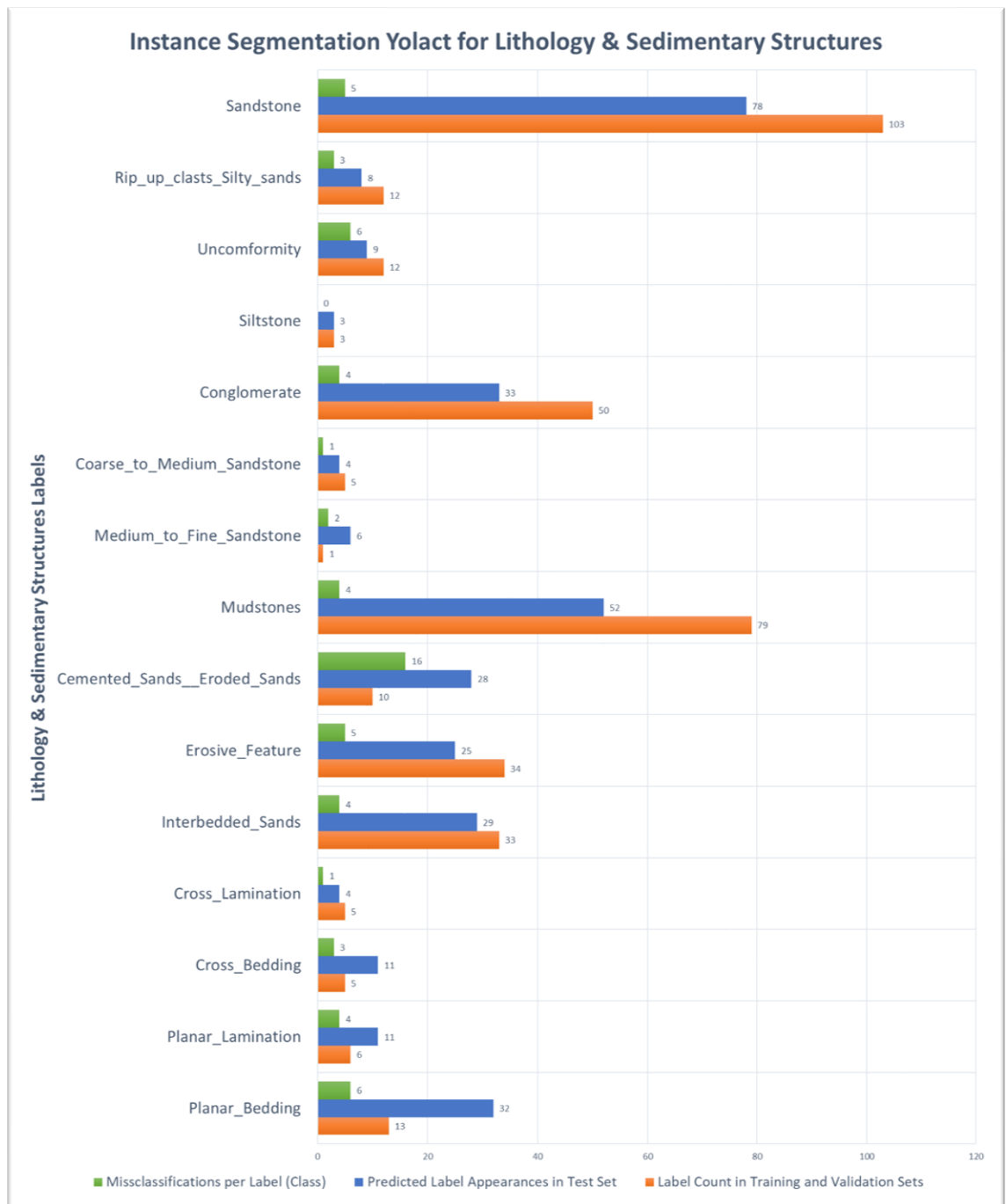


Figure 7-10: Quantitative Results of the default YOLACT model on outcrop images.

According to Figure 7-10, the top class that is misclassified is the Cemented/Eroded Sands Class, with 16 misclassifications, while the classes Planar Bedding and Unconformity are misclassified six times each.

The results showed that some predictions were wrong and did not make geological sense. As in Chapter 6, an important note here is that a single image can contain multiple predictions of the same class or a different one, meaning that the number of predicted labels is expected to be higher than the total number of images in the training and validation sets; of course, this depends on the size of the test set. In this case, the test dataset included 60 images of outcrops, 45 partially seen and 15 unseen images, randomly chosen during the dataset split into training, validation, and test sets. Thus, in 60 outcrop images, there are 333 total appearances of geological objects, sedimentary structures, and lithology types in this case. The total percentage of misclassifications across the test set was calculated as the ratio of the sum of misclassifications per class over the sum of the predicted label appearances in the test set for each Table respectively. The resulting percentage of misclassifications in the test set adds up to 15.5% (Table 7-4) and 58.6% (Table 7-5), giving YOLACT an 84.5% accuracy for the partially seen test data and 41% accuracy for the unknown test data. These results support the statement made by observing the graphs of the model's training that the default YOLACT model trained on dataset 9 overfits the training data and performs poorly on unseen outcrops images.

The qualitative results of the default Yolact model are shown in Figure 7-11 and Figure 7-12, in which the original test images are shown alongside the same segmented image. Figure 7-11a shows a test image of a big conglomeratic bed, a sandstone channel-looking feature, and a few erosional features in the top right corner.

The model's successful predictions are indicated firstly by the fit of the masks and bounding boxes around the geological features and secondly by the decimal number, showing the model's confidence associated with each prediction.

The Instance Segmentation is quite successful in this example as the model can accurately identify and segment the three main features in the image. In Figure 7-11b, the conglomerate and sandstone beds' shape and contacts are more irregular than in the previous example. Therefore, this example is more difficult for the Segmentation task as the model is tasked to generate and fit a mask on the boundaries of the two prominent classes. For the human eye, it is easy to distinguish the sandstone from the conglomerate

beds in this image. The shape of the bed contacts is very irregular, and for Segmentation models, fitting the mask for such irregular shapes is a challenge. In this example, the mask fits nicely around the objects' periphery and can capture the pinching out of the beds. Despite this challenge, the model successfully segments the outcrop into distinct segments while displaying high-quality masks and predictions for the sandstone and conglomerate classes.

Training the model on one half of the outcrop extent and testing it on the other half is comparable to the process conducted by geologists in the field using analogues. Therefore, training the model with similar outcrops to the examined one increases the model's prediction accuracy.

All the test data used to produce the results of Figure 7-11 (a and b) are referred to as partially seen data and are from the same outcrop.

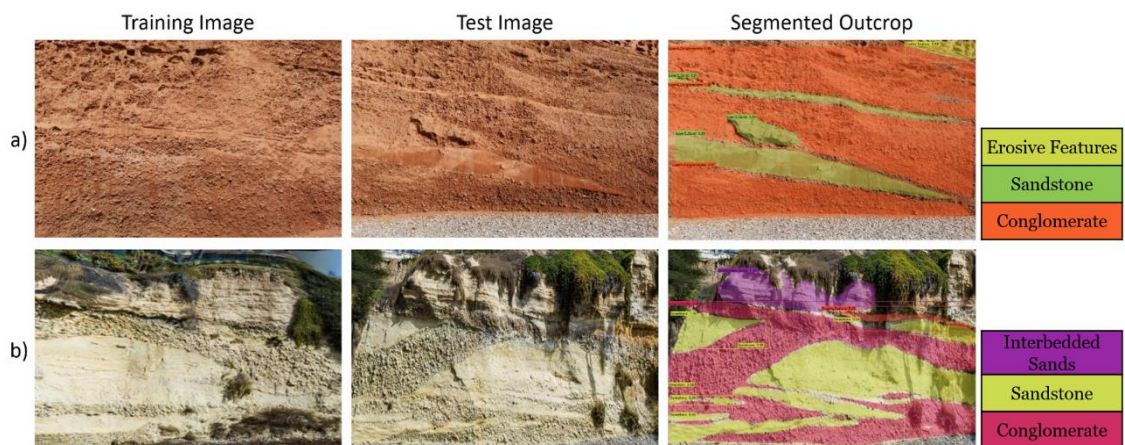


Figure 7-11: a) Instance Segmentation predictions, including a mask, bounding box, label, and the associated probability of the prediction on an Aeolian/Fluvial depositional environment. b) Instance Segmentation predictions, including a mask, bounding box, label, and the associated probability of the prediction on a Deep Marine depositional environment.

Next, the previously described model was tested on unseen data, meaning random images from different outcrops unknown to the model. This unseen portion of the test dataset includes some features and structures included in the training set. It is important to note that the common characteristics are represented differently in terms of scale, angle, colour, or texture within Dataset 9.

A few additional labels of sedimentary structures were present in this unseen test set, including flame structures, fossils, flute, and desiccation cracks, among others. Since these features were not present as labels in the original training set, the model failed to make predictions for them.

The features shared between the train and test sets were common characteristics and sedimentary structures, including cross-bedding, parallel and planar lamination, and others. This presents a challenge for the Segmentation model because even for the labels/objects in common, if they are represented much differently, the model perceives them as unknown features, hence misclassifying them.

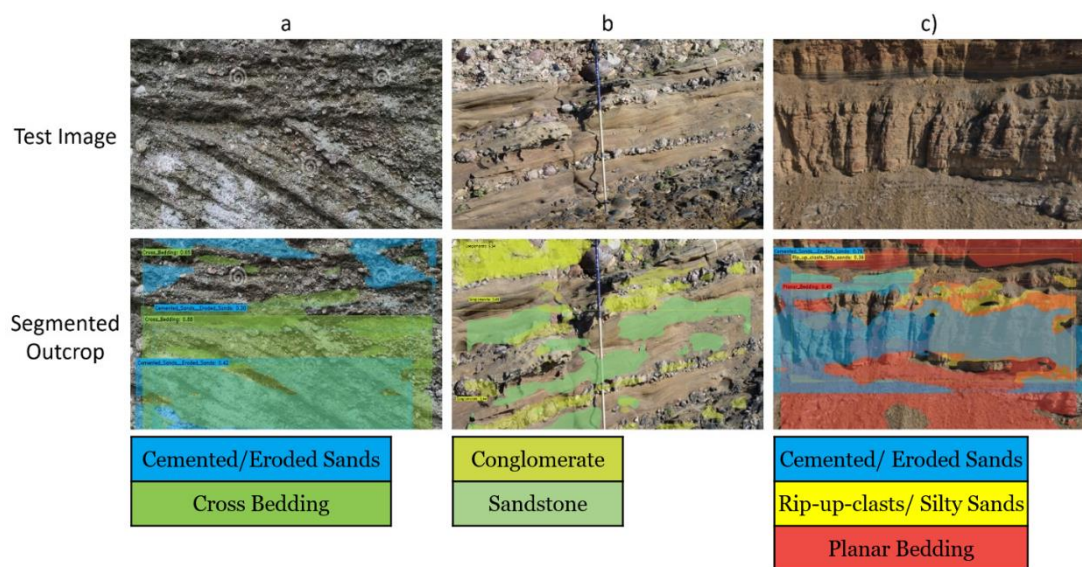


Figure 7-12: Instance Segmentation on unseen data demonstrating the model's performance getting worse as the test pictures are getting progressively different from the train set (from a to c).

In Figure 7-12a, the prominent feature is a large cross-bedding. The model can still predict the cross-bedding feature; it is less accurate than the previous examples but produces few predictions.

Figure 7-12b captures a beach with fine sand and larger rocks or conglomerates. The model makes some predictions, but the confidence scores decrease, and the masks are now sparser without defining clear boundaries between the geological objects.

Figure 7-12c could be separated into four segments; the planar bedding on top, some fractured interbedded sandstone with siltstone in the middle part, and neighboring up and

down with finer sediment. In this example, the model fails to understand and segment the geology of this outcrop. This might happen for a couple of reasons: a) this outcrop has a high number of fractures and shadows, making it difficult for the model to segment the image, producing three overlapping masks and labels around the middle part of the outcrop because lithology and sedimentary structures are combined in this model, and the model tries to display both elements in its predictions, b) the second reason is that the geology depicted in that particular image is very different from the images the model was trained on.

The images get progressively more challenging as their features are either less distinguishable or more complicated to identify in terms of texture. In the examples in Figure 7-12, the instance Segmentation model fails to segment the geology sufficiently, and sometimes it even fails to form a mask. In some of the predictions, only bounding boxes and labels are predicted. The training dataset's limited representation of diverse patterns ultimately impacts the model's ability to generalize effectively to unseen data. To address these misclassifications, it becomes imperative to incorporate a more comprehensive and diverse training dataset. By including a wider range of patterns and variations, the model can learn to recognize and classify features more accurately, accounting for the inherent complexity within different geological contexts.

Unsegmented areas of the images occur because the model does not understand the area and cannot make a prediction, as, during the annotation step, the majority of the images were not annotated entirely but only the essential parts of them, or because the prediction threshold set by the user is higher than the prediction confidence (score) of the model. For example, if the prediction threshold is set to 0.5, the model will only display predictions with confidence ≥ 0.5 . Therefore, the model's performance decreases in accuracy as the test pictures progressively differ from the original training set.

For completeness and to demonstrate YOLACT's potential and applicability of instance segmentation on an outcrop, I applied the model discussed in this experiment on a video captured from a 3D outcrop model. A snippet of the segmented video is shown in Figure 7-13, while the media file will be provided separately from the thesis document.

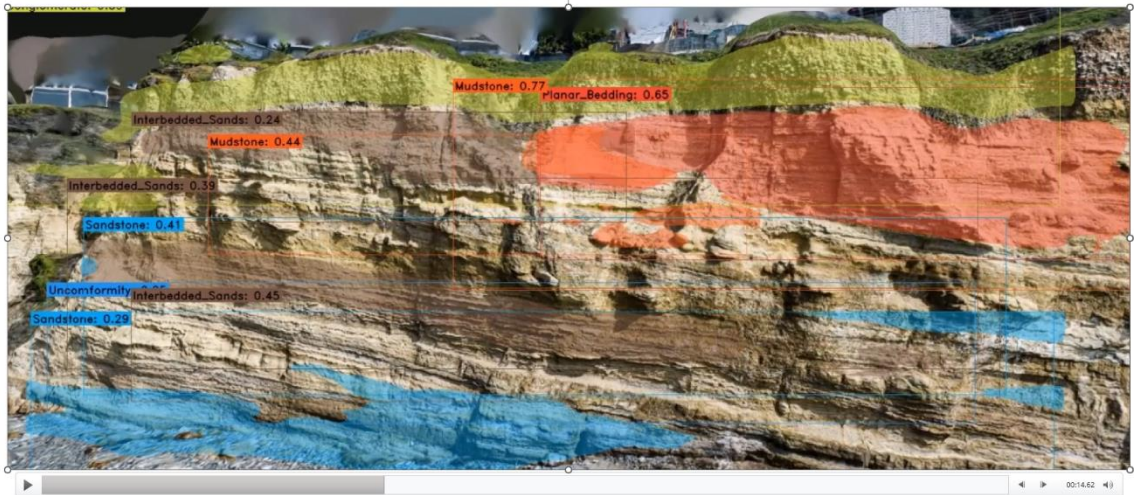


Figure 7-13: Real-time application of the default YOLACT (DarkNet53) model on an outcrop video.

The results of the model are not perfect, but they are very promising. Such an application will be improved and investigated further in future work.

The next section will examine a few ways to improve the Segmentation model, significantly improving its performance and making it a pillar of the overall workflow.

7.4.2 Experiment 2: Recommended Improvements for the default YOLACT ((DarkNet53) model

The previous experiment showed that YOLACT is suitable for the purpose we need it and that it has the potential to become a vital step of the overall workflow of this thesis. In section 7.4.1, it was established that YOLACT is suitable for the segmentation of the outcrop geology. The results show that the model can predict sedimentary structures and lithology types. The purpose of this section and experiment is to improve the model to yield better predictions by using a modified version of the Darknet53 backbone called cDarkNet53, along with modifications to hyperparameters and the dataset used for training.

In order to enhance the overall performance, three key improvements were taken into consideration, encompassing modifications to both the model and the data:

- a) The first improvement focused on enhancing the dataset itself. This involved augmenting the dataset by adding more images, ensuring a larger and more diverse collection. The goal was to improve the quality of the images and enhance the variability,

ensuring that the desired geological features were depicted clearly within the images. By incorporating a broader range of geology representations, the dataset became more comprehensive and representative of the target domain.

b) The second improvement involved refining the annotations used to train the segmentation model. This was accomplished by expanding the number of labels and increasing the complexity within the dataset. By adding more detailed and nuanced annotations, the model could learn to discern finer distinctions and accurately segment different geological features. The enriched annotations contributed to a more robust and precise training process.

c) The final improvement centered around the model's training and architecture, specifically the backbone. The training process was fine-tuned to optimize the model's performance, taking into account factors such as learning rate, batch size, and training duration. Additionally, the architecture of the model, particularly the backbone, was evaluated and modified as necessary. This involved exploring different architectures and selecting the one that yielded the best results for geology segmentation. In section 7.4.1, Dataset 9 was used to train the model with 50 images containing 15 unique labels. In this section, the improved model was trained with Datasets 10a and 10b, each one containing 70 images for training, with 13 and 26 distinct labels, respectively. Therefore, the number of images and labels in the new dataset is increased, enhancing the variability and quality of Dataset 10 (10a and 10b).

The annotations were also improved by ensuring all the labels assigned to each image were cross-referenced with published literature to ensure their validity. The labels themselves are now clearer on what they are describing, encapsulating the scale of each individual feature.

The hyperparameters of the model were modified according to the computational resources available, and through trial and error, I established the optimal configuration of the hyperparameters for Dataset 10. The model's backbone was customized and transformed into a slightly shallower configuration in order to reduce the total number of parameters in the model.

The default YOLACT model with a Darknet53 backbone described in the previous section has a specific configuration that includes 52 convolutional layers that have batch normalization and leaky ReLU activations. Additionally, there is a feature pyramid

network with five more convolutional layers with batch normalization and ReLU activations. The prediction heads consist of 4 separate heads containing fully connected and convolutional layers. Overall, this model has approximately 64.6 million parameters.

One of the experimentations toward improving the model's overall performance was slightly reducing the number of parameters in its backbone to make its training time more efficient. One way to reduce the number of the model's parameters is by removing one or more layers from the backbone without altering the functionality of YOLACT. The total number of parameters decreased by approximately 700,000 by removing the last convolutional layer from the Darknet53 backbone while leaving the rest of the model the same. This modified version of Darknet53 is referred to as "cDarkNet53" in this thesis, with the 'c' standing for custom. Therefore, the model's new total number of parameters would be approximately 63.9 million. However, it is essential to note that this is an estimation, and the actual number of parameters may vary based on how it's implemented.

In order to determine the total number of parameters for a YOLACT model with a ResNet101 backbone (*section 7.4.3*), the model's specific configuration must be considered. However, the number of parameters for this model can be estimated based on a few general guidelines. Typically, the ResNet101 backbone alone has around 44 million parameters. The YOLACT feature pyramid network adds additional layers on top of the backbone, which can contribute around 4 million more parameters. The YOLACT prediction heads, which include fully connected and convolutional layers, can add another 7-10 million parameters per head, depending on the specific configuration.

With these estimates in mind, the YOLACT model with a ResNet101 backbone and four prediction heads would have a total number of parameters in the 75-80 million range. If we compare the two versions of the YOLACT model with the two different backbones, we can conclude that YOLACT (cDarkNet53) has about 63.9 million parameters while YOLACT (ResNet101) has about 75-80 million parameters. The difference is about 10+ million parameters, resulting in different model performances (speed versus accuracy) and training times. The number of parameters can vary based on factors such as the specific architecture configuration, the number of classes being detected, and other hyperparameters.

All the aforementioned improvements were combined together to produce the results of the following sections, leading to a more robust Instance Segmentation model for the segmentation of the outcrop geology.

7.4.2.1 Training the YOLACT (cDarkNet53) model for Lithology Segmentation

This section explains the training of the improved YOLACT model, using the cDarkNet53 as a backbone, with dataset 10a containing 70 images for the training and validation sets and annotations only for lithology types, with 13 different classes.

The training hyperparameters used for training the YOLACT (cDarkNet53) model on Dataset 10a for the Segmentation of the lithology can be found in Table 7-6. The choice of most of the hyperparameters depended on the computational power available. As for the learning rate and steps, the optimal number of learning steps was determined through trial and error and from analyzing the results of the training graphs with the mAP scores and losses. The initial learning rate was 0.001, which is a good learning rate to start with, as chapters 5, 6, and *section 7.4.1* demonstrated. At the designated steps, chosen based on the total number of epochs, the model changed the learning from 0.001 to 0.01 incrementally in eight steps. The learning steps help the model learn better by allowing it to adjust its internal parameters based on the training data gradually and can also help prevent overfitting, occurring when the model becomes too specialized to the training data and does not generalize well to new data.

Training Hyperparameters	Value
Pretrained weights	darknet53.pth
Image size	512
Batch size	8
Epochs	20000
workers	4
Evaluation interval	20
Gpu count	1
Optimizer	SGD
Learning rate	0.001
Initial Learning Steps (epochs 0-10k)	2800, 6000, 7000, 7500
Secondary Learning Steps (epochs 10k-20k)	11000, 14500, 17000, 17500

Table 7-6: Training hyperparameters used for training the YOLACT (cDarkNe53) model for lithology Segmentation.

Figure 7-14 shows the mAP scores of masks and bounding boxes on validation data versus the number of iterations/epochs. As the number of iterations increases, the Mean Average Precision scores also increase, which is the expected behavior. The mAP score reaches its highest value of 0.25 at about 8000 iterations. The trained model shows a normal trend, as the mAP scores increase smoothly with the number of iterations.

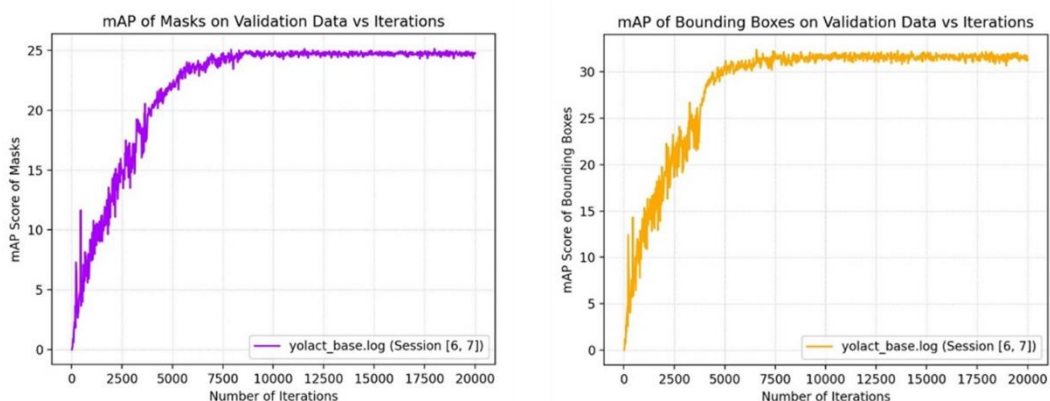


Figure 7-14: The mAP scores of masks and bounding boxes on validation data versus the number of iterations/epochs.

If we recall the quick overfitting of the model to the data in Figure 7-8, in Figure 7-14, the model trains and learns progressively over the first 10k iterations until it reaches the 20k iterations, where it enters a plateau, despite the adaptation of the learning rate at the designated learning steps in Table 7-6. Thus, the model has reached its maximum learning capacity for the particular dataset, showing no further improvement in the mAP scores. Figure 7-15 shows the overall Loss scores of classes confidence, masks, and bounding boxes on validation data versus the number of iterations/epochs. The fact that all three losses enter a plateau and do not reduce further also supports the statement that the model cannot learn new information from the data.

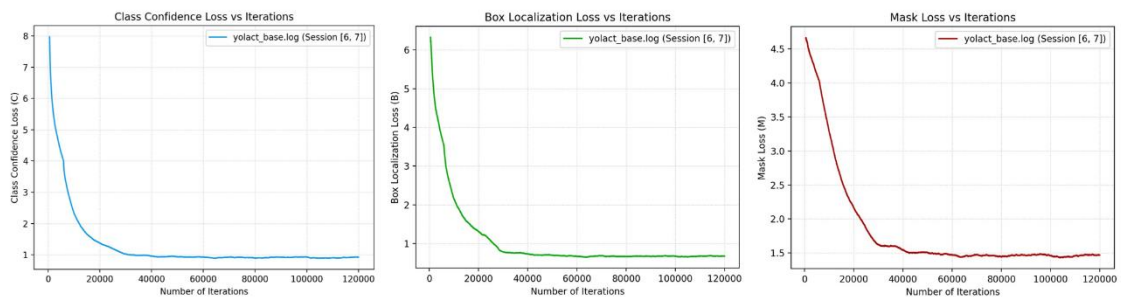


Figure 7-15: Overall Loss scores of classes confidence, masks, and bounding boxes on validation data versus the number of iterations/epochs.

According to this analysis, compared to section 7.4.1.2, the results are expected to have improved significantly, showing improved predictions and generalization over unseen outcrop test images.

7.4.2.2 Testing the YOLACT (cDarkNet53) model for Lithology Segmentation

This subsection shows a quantitative and qualitative analysis of the abovementioned Segmentation model's results when tested on a set of images. This particular test set includes 50 images selected randomly from a pool of outcrop images. Table 7-7 and Figure 7-16 show the quantitative results of the YOLACT (cDarkNet53) model when trained on dataset 10a and tested on outcrop images to segment the various lithology types present.

Table 7-7 provides a detailed breakdown of the model's performance per class/label present in Dataset 10a. The Table shows the number of labels present in the training and validation sets, the number of label appearances in the test data, the number of misclassifications per label, and the percentage of misclassifications in the entire test set.

Instance Segmentation Yolact for Lithology (cDarkNet53)				
Labels/Classes	Label Count in Training and Validation Sets	Predicted Label Appearances in test data	Misclassifications per Label (Class)	Percentage of misclassifications per class
Amalgamated/Cemented Bed	1	0	0	0
Breccia	5	5	0	0
Carbonates	7	6	0	0
Conglomerate	29	20	0	0
Interbedded mudstone-siltstone	27	20	1	5
Interbedded sandstone-mudstone	6	4	0	0
Interbedded sandstone-siltstone	21	21	1	5
Iron Rich Sediment	7	5	2	40
Mudstone	53	43	2	5
Organic Material	37	30	1	3
Red (Sandstone) Beds	21	21	0	0
Sandstone	79	71	1	1
Siltstone	28	23	1	4
Total	321	269	9	
Total Percentage of misclassifications for Test set, %				3.35

Table 7-7: Quantitative Results of the YOLACT (cDarkNet53) model. The model was trained on dataset 10a and tested on outcrop images to segment the various lithology types present.

Table 7-7 shows the number of labels present in the training and validation sets, the number of label appearances in the test data, the number of misclassifications per label, and the percentage of misclassifications in the entire test set. The last row (orange colour) of the Table shows the total percentage of misclassifications for the entire Test set.

According to Figure 7-16, the top misclassified classes are the Iron-Rich Sediments and Mudstone Classes, with two misclassifications. The classes Siltstone, Sandstone, Organic Material, Interbedded sandstone-siltstone, and Interbedded mudstone-siltstone follow with one misclassification each. The results showed that some of these predictions were wrong but not far from the ground truth. In other words, an example was misclassified as siltstone, while the ground truth was mudstone. In geological terms, the difference between the two classes is the grain size, which poses a significant challenge to tackle with the current dataset. Ideally, a separate model would be trained solely to distinguish the lithology based on the grain size. To achieve such a distinction by using state-of-the-art Segmentation models, including YOLACT, would require a highly curated image dataset consisting of images with very high resolution. This brings us back to one of this thesis's significant challenges: the need for more and for better quality data.

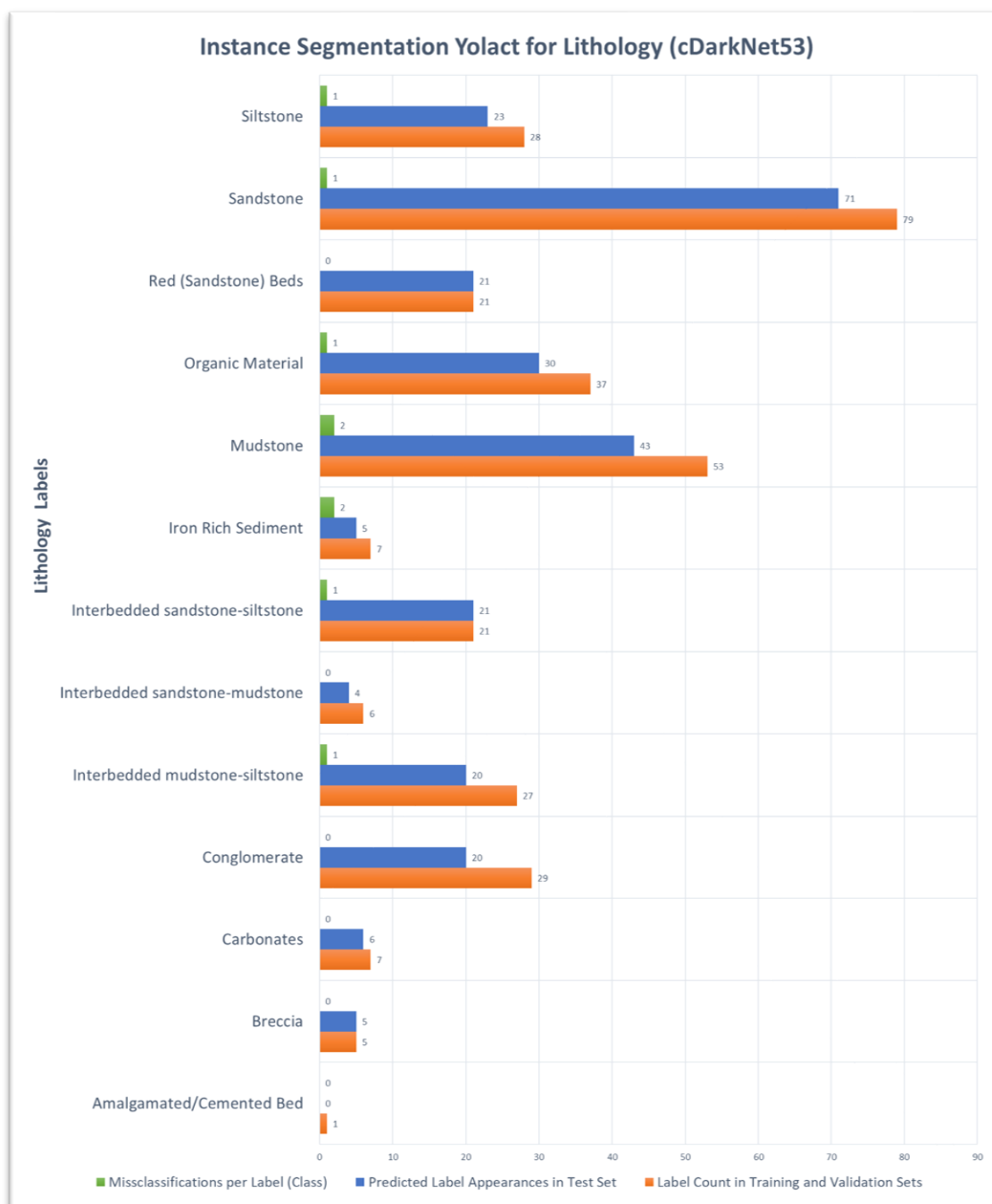


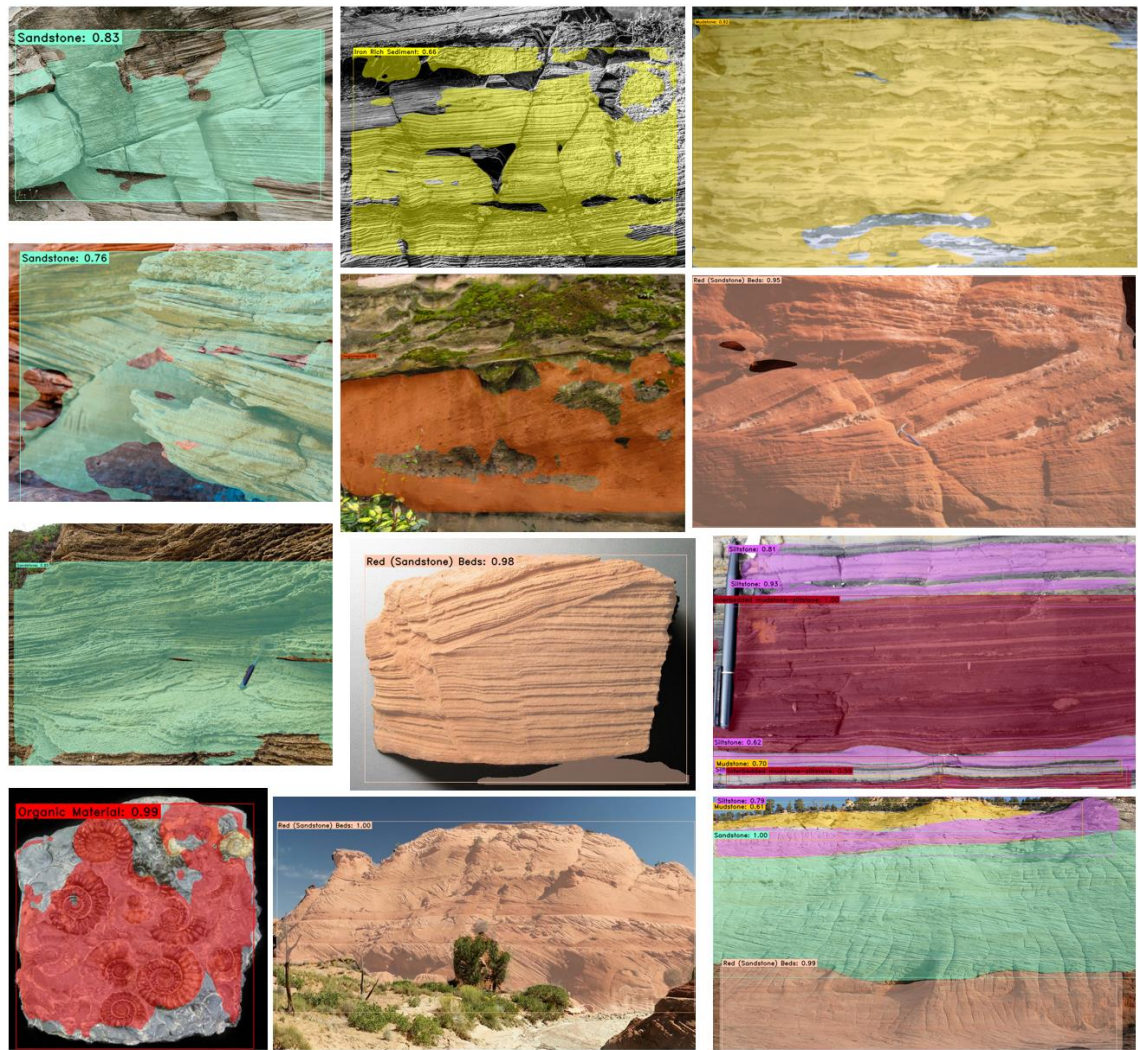
Figure 7-16: Quantitative Results of the YOLACT (cDarkNet53) model. The model was trained on dataset 10a and tested on outcrop images to segment the various lithology types present.

A single image can contain multiple predictions of different or the same class, meaning that the number of predicted labels is expected to be higher than the total number of images in the training and validation sets, depending on the size of the test set. In this case, the test dataset included 50 images of outcrops. In these 50 images, there are 269 total appearances of geological objects, lithology types in this case. The total percentage

of misclassifications across the test set was calculated as the ratio of the sum of misclassifications per class over the sum of the predicted label appearances in the test set for each Table respectively. The resulting percentage of misclassifications in the test set adds up to 3.35% (Table 7-7), giving YOLACT a 96.65% accuracy for the particular test set.

A sample of the qualitative results of this section can be found in Figure 7-17. The YOLACT (cDarkNet53) was tested with twelve different outcrop images to estimate the lithology types in each image. All predictions, but one, are correct when compared with the ground truth labels provided by their sources. The middle image from the 1st row demonstrates an instance in which the model assigned the wrong label (iron-rich sediment) to the image. However, the masks and bounding boxes were still assigned to the correct place, showing improved mask fitting.

Furthermore, in some images, the masks fit perfectly around the objects, accurately contouring the different lithology layers. Finally, in other examples, the masks are sparser, indicating that the model was challenged to assign the masks. Overall, this model shows significant improvement compared to the model and results described in *section 7.4.1.2*.



Sandstone	Red (Sandstone) Beds	Siltstone	Mudstone
Conglomerate	Organic Material	Iron-Rich Sediments	Interbedded mudstone-siltstone

Figure 7-17: Qualitative Results of the YOLACT (cDarkNet53) model on twelve outcrop images.

7.4.2.3 Training the YOLACT (cDarkNet53) model for Sedimentary Structures Segmentation

This section explains the training of the improved YOLACT model, using the cDarkNet53 as a backbone, with dataset 10b containing 70 images for the training and validation sets and annotations only for sedimentary structures, with 26 different classes.

The training hyperparameters used for training the YOLACT (cDarkNet53) model on Dataset 10b for the segmentation of sedimentary structures can be found in Table 7-8. The choice of most of the hyperparameters depended on the computational power

available. As for the learning rate and steps, the optimal number of learning steps was determined through trial and error and from analyzing the results of the training graphs with the mAP scores and losses. The initial learning rate was 0.001, which is a good learning rate to start with, as Chapters 5, 6, and 7 have demonstrated so far. At the designated steps, chosen based on the total number of epochs, the model changed the learning from 0.001 to 0.01 incrementally in twelve steps.

Training Hyperparameters	Value
Pretrained weights	darknet53.pth
Image size	512
Batch size	8
Epochs	30000
workers	4
Evaluation interval	20
Gpu count	1
Optimizer	SGD
Learning rate	0.001
Initial Learning Steps (epochs 0-10k)	2800, 6000, 7000, 7500
Secondary Learning Steps (epochs 10k-20k)	11000, 12000, 15000, 17500
Tertiary Learning Steps (epochs 20k-30k)	21000, 24500, 27000, 27500

Table 7-8: Training hyperparameters used for training the YOLACT (cDarkNe53) model for sedimentary structure segmentation.

Figure 7-18 and Figure 7-19 are used as a guide to fine-tune the hyperparameters by monitoring the behavior of the mAP scores and that of the three loss scores during training. In Figure 7-18, as the number of iterations increased, it was found that the understanding of annotations and the images by the model increased, resulting in progressively better mAP scores. On the other hand, the loss scores (Figure 7-19) for the classes, bounding boxes, and masks decreased, which is the expected behavior in terms of the accuracy/loss couple. At about 5000 iterations, for Figure 7-18, in both graphs, the mAP score enters a plateau, meaning that the model cannot extract and learn any new information from the data. This is a good indication to interrupt the model's training, adjust the hyperparameters, and resume until there is no further improvement. Sessions 9 and 10 refer to the number of times the model was interrupted and then resumed training. Figure 7-18 implies that the model was trained from 0 to 10000 iterations, where a significant increase in the mAP scores occurred, and hence the majority of the learning

took place. Then, the model was interrupted and fine-tuned (session 9), resumed training from 10000 up to 20000 iterations, a part where there was another increase in the mean average precision scores, and where there was another training interruption (session 10). Finally, the training was resumed one last time from 20000 to 30000 iterations, where no further mAP increase was observed. There was another run on the training sessions from 30k up to 40k iterations to ensure the model was still at the plateau, and it was confirmed that the model's mAP scores remained exactly the same. The training's highest recorded mask mAP score was 52.09 at epoch 22356. This accuracy was repeated while on the plateau but was never exceeded.

Continuing the training while on the plateau will only reduce the model's performance as it will start overfitting, losing the ability to generalize to unseen data. Comparing Figure 7-14 and Figure 7-18, it is evident that the model for the lithology converged much earlier compared to that of the sedimentary structures. Though, the mAP scores do not always indicate good and bad training behavior as they heavily depend on the data availability and number of labels. Even if the model is trained for thousands of iterations, in some cases, the mAP scores will continuously increase for the given dataset only without improving the model's performance. At some point, the model will converge (stop learning). If the model keeps training after converging, the mAP will still increase but will be overfitting the data, losing the ability to generalize.

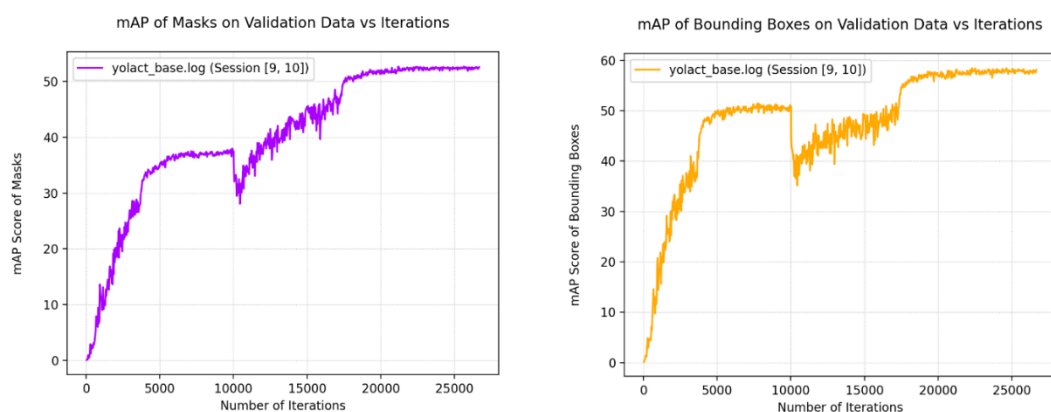


Figure 7-18: The mAP scores of masks and bounding boxes on validation data versus the number of iterations/epochs.

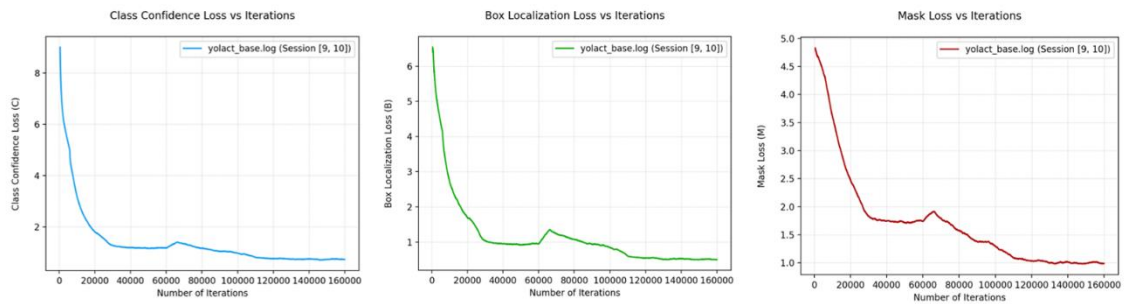


Figure 7-19: Overall Loss scores of classes confidence, masks, and bounding boxes on validation data versus the number of iterations/epochs.

Figure 7-19 shows the overall Loss scores of classes confidence, masks, and bounding boxes on validation data versus the number of iterations/epochs. The fact that all three losses enter a plateau and do not reduce further also supports the statement that the model cannot learn new information from the data.

The time required for the model to complete the training was approximately 22 hours for the ‘Lithology’ model, trained with a total of 70 images and 13 unique labels, while the model for the ‘Sedimentary Structures’ took roughly 31 hours of training time with a total of 70 images and 26 unique labels as the complexity and identification of sedimentary structures are more challenging compared to that of the lithology estimation.

The higher the number of images, image size, and distinct labels, the more computationally expensive the model training becomes. The workstation specs used to train this model include an Intel(R) Core (TM) i7-10700K CPU @ 3.80GHz 3.79 GHz processor, with Installed RAM of 32.0 GB and an NVIDIA RTX 2080 GPU.

According to this analysis, compared to section 7.4.1.2, the results are expected to have improved significantly, showing improved predictions and generalization over unseen outcrop test images for the segmentation of sedimentary structures.

7.4.2.4 Testing the YOLACT (cDarkNet53) model for Sedimentary Structures Segmentation

This subsection shows a quantitative and qualitative analysis of the abovementioned segmentation model’s results when tested on a set of images. This particular test set includes 50 images selected randomly from a pool of outcrop images. Table 7-9 and Figure 7-20 show the quantitative results of the YOLACT (cDarkNet53) model when

trained on Dataset 10b and tested on outcrop images to segment the various sedimentary structures present.

Instance Segmentation Yolact for Sedimentary Structures (cDarkNet53)				
Labels/Classes	Label Count in Training and Validation Sets	Predicted Label Appearances in test data	Misclassifications per Label (Class)	Percentage of misclassifications per class, %
Bioturbation	48	32	0	0
Clasts	52	17	0	0
Convoluted/Irregular Lamination	2	1	0	0
Convoluted/Irregular Bedding	2	2	0	0
Cross Bedding/Stratification	39	34	8	24
Cross Lamination/Climbing Ripples	22	13	1	8
Dessication Cracks	5	4	0	0
Erosive Contacts/Bases	75	26	0	0
Erosive Features	31	10	1	10
Faults	16	7	0	0
Flame Structures	6	3	0	0
Flaser Lamination	3	2	2	100
Flute Marks	42	0	0	0
Fossils	12	12	0	0
Herringbone Cross Stratification	9	3	2	67
Hummocky Cross Stratification	10	2	0	0
Lenticular Bedding	8	4	0	0
Lenticular Lamination	5	3	0	0
Planar/Parallel Bedding	45	20	1	5
Planar/Parallel Lamination	29	18	2	11
Scour Marks	6	3	1	33
Structureless	56	33	1	3
Swaley Cross Stratification	4	1	0	0
Syneresis Cracks	2	4	3	75
Wave Ripples/Lamination	6	6	1	17
Wavy Bedding	5	3	0	0
Total	540	263	23	
Total Percentage of misclassifications for Test set, %				8.75

Table 7-9: Quantitative Results of the YOLACT (cDarkNet53) model. The model was trained on dataset 10b and tested on outcrop images to segment the various sedimentary structures present.

Table 7-9 provides a detailed breakdown of the model's performance per class/label present in Dataset 10b. The Table shows the number of labels present in the training and validation sets, the number of label appearances in the test data, the number of misclassifications per label. The last row (orange colour) of the Table shows the total percentage of misclassifications for the entire Test set. The highlighted values, in yellow colour, in the Tables represent the higher percentages (>50%) of the misclassifications per class.

According to Figure 7-20, the top misclassified class is the Cross Bedding/Stratification class, with eight misclassifications. Although cross-bedding is easy for a human geologist to identify, it seems to be a challenge for the Machine Learning model. As discussed in Chapter 6, cross-bedding proved to be difficult to detect and segment in this case, as the pattern of cross-bedding is widespread in nature as a full feature or part of other entities.

The results showed that some of these predictions were wrong but not far from the ground truth, while others were predicted wrong. This is due to the lack of data available to train

the segmentation model, but also the inability of machine learning in general to obtain a holistic understanding of the image.

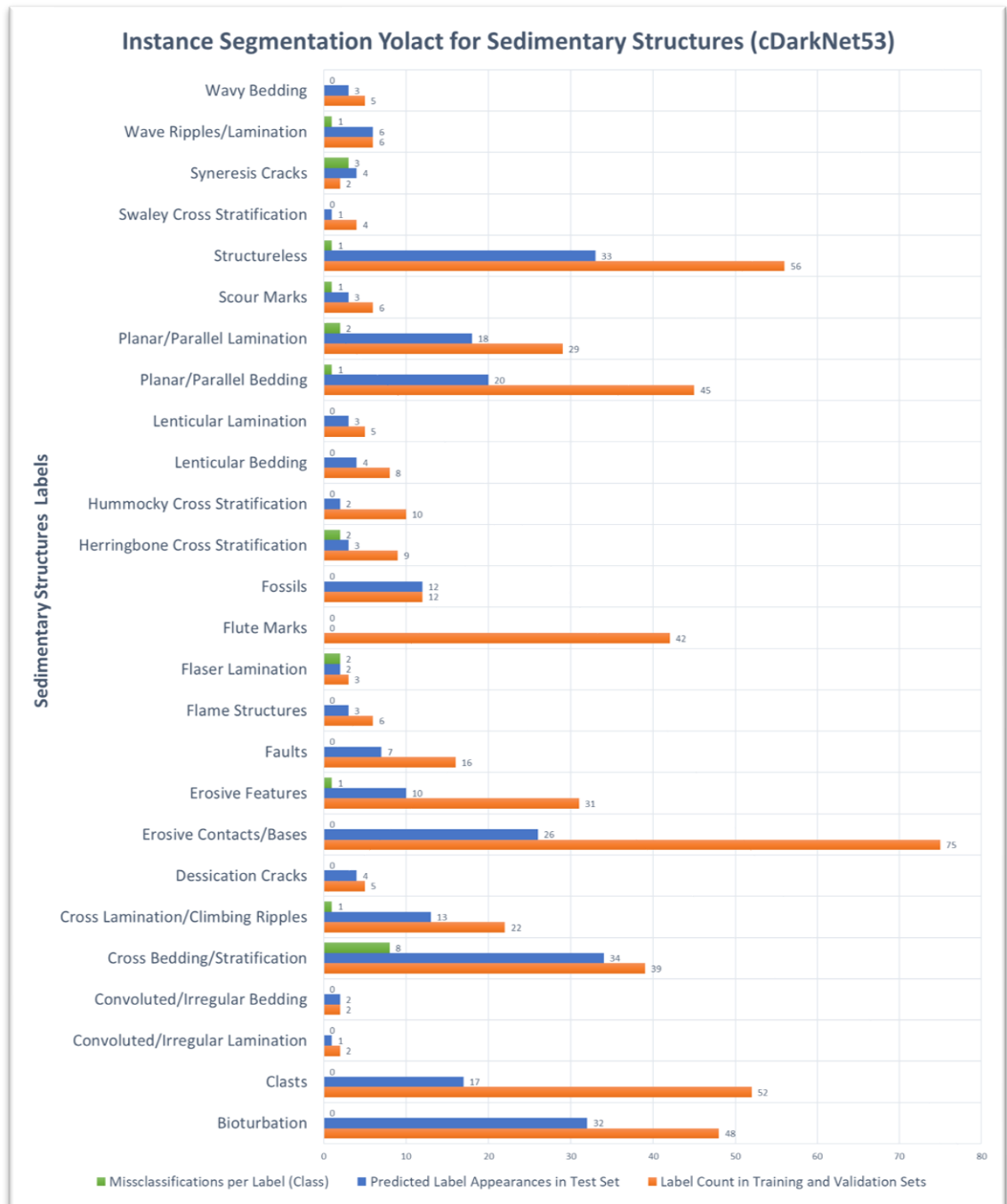


Figure 7-20: Quantitative Results of the YOLACT (cDarkNet53) model. The model was trained on dataset 10b and tested on outcrop images to segment the various sedimentary structures present.

Once again, a single image can contain multiple predictions of different or the same class, meaning that the number of predicted labels is expected to be higher than the total number

of images in the training and validation sets, depending on the size of the test set. In this case, the test dataset included 50 images of outcrops. In these 50 images, there are 263 total appearances of geological objects, sedimentary structures in this case. The total percentage of misclassifications across the test set was calculated as the ratio of the sum of misclassifications per class over the sum of the predicted label appearances in the test set for each Table, respectively. The resulting percentage of misclassifications in the test set adds up to 8.75% (Table 7-9), giving YOLACT a 91.25% accuracy for the particular test set.

A sample of the qualitative results of this section can be found in Figure 7-21. The YOLACT (cDarkNet53) was tested with the same twelve different outcrop images used in Figure 7-17 to segment the sedimentary structures in each image. Most predictions are correct, but two when compared with the ground truth labels. The first row of images does not display any prediction on them. That is either because the confidence threshold of the predictions was set to predict with certainty higher than 0.4 or because the model was unable to identify the structures present in these particular images. The middle images of the second and third rows are the two misclassifications. The one in the second row is predicted as structureless, while in reality, there is a cross-bedding in that place, according to the ground truth. On the other hand, the image in the middle of the third row shows a phenomenon of scale mismatch, meaning that the model mistakenly mixed up a structure at the laminae scale (Planar Lamination) with a structure found in the bedding scale (Cros Bedding/Stratification). Such misclassifications could result in the wrong interpretation of the depositional environment from the machine learning model. Therefore, extra care should be taken during the image annotation. Although I paid close attention during the annotation of the features, the model produced a scale mismatch. That supports my statement that mAP scores are not enough to evaluate the model's performance. However, the geologist's expertise is necessary during the evaluation of the annotations and the model's results to ensure that no rules of the geology are violated.

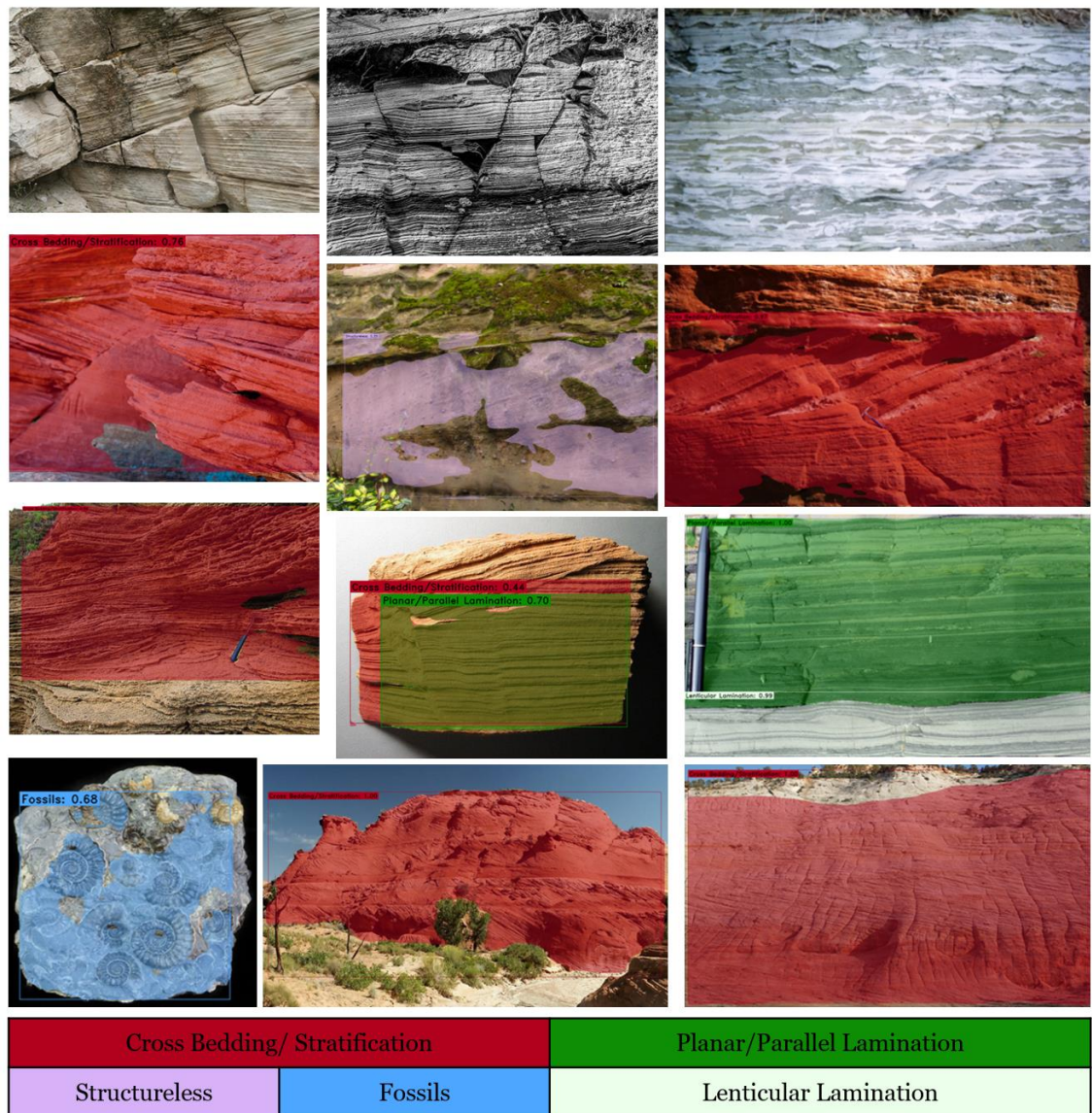


Figure 7-21: Qualitative Results of the YOLACT (cDarkNet53) model on twelve outcrop images.

Furthermore, in most other images in Figure 7-21, the masks fit perfectly around the objects, accurately contouring the different structures, except in the fossil example (bottom left corner), in which the mask is sparser, indicating that the model was challenged to assign the masks. Overall, this model shows significant improvement compared to the model and results described in section 7.4.1.2.

7.4.2.5 Comparative Study: YOLACT (DarkNet53) versus YOLACT (cDarkNet53)

This sub-section compares the default YOLACT (DarkNet53) model and the YOLACT (cDarkNet53) model discussed in section 7.4.2. Figure 7-22 shows a comparison of the

two models on an unseen image. For Figure 7-22, the ground truth is sandstone/red sandstone for the lithology, and cross-bedding for the sedimentary structures.

The default YOLACT (DarkNet53) backbone (*section 7.4.1*) fails to make accurate predictions on the given image, by wrongly predicting cemented sand for the lithology label and in terms of sedimentary structures, it predicts planar lamination. In addition, the mask fit and bounding boxes are not placed correctly and severe mask overlap occurs.

On the contrary, the YOLACT (cDarkNet53) model (*section 7.4.2*) yields more interpretable and accurate results as the lithology and sedimentary structures are predicted separately. All the geological labels are predicted correctly according to the ground truth, and the mask fit is improved.

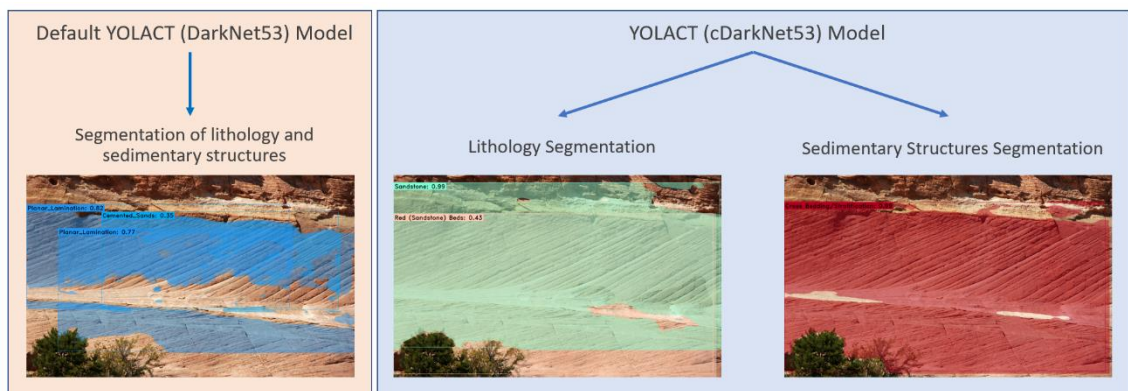


Figure 7-22: A comparison of the Default YOLACT (DarkNet53) Model vs the YOLACT (cDarkNet53) on an unseen image.

The modified backbone (cDarkNet53) was designed to be shallower by reducing the number of convolutional layers, aiming to decrease the number of the model's parameters, increasing inference speed, and decreasing computational costs, with a small loss tradeoff in the model's accuracy. The extraction of parameters from the feature map is now smaller due to the shallower architecture of the model's backbone. The combination of the specific hyperparameters described in Table 7-6 and Table 7-8 enables the model to achieve improved performance for the outcrop segmentation in terms of accuracy and speed of inference. The significant improvements in the model performance originated from combining all the modifications described in section 7.4.2.

To further investigate the benefits of tuning the model and dataset as well as the splitting of the model for the lithology and sedimentary structure segmentation tasks, additional

images were tested, as shown in Figure 7-24, and were compared with Figure 7-23, tested with the default YOLACT model from Experiment 1. Figure 7-23 shows the application of the default model on nine different outcrop images, while Figure 7-24 shows the application of the improved model (Experiment 2) on the same nine images for a clear comparison.

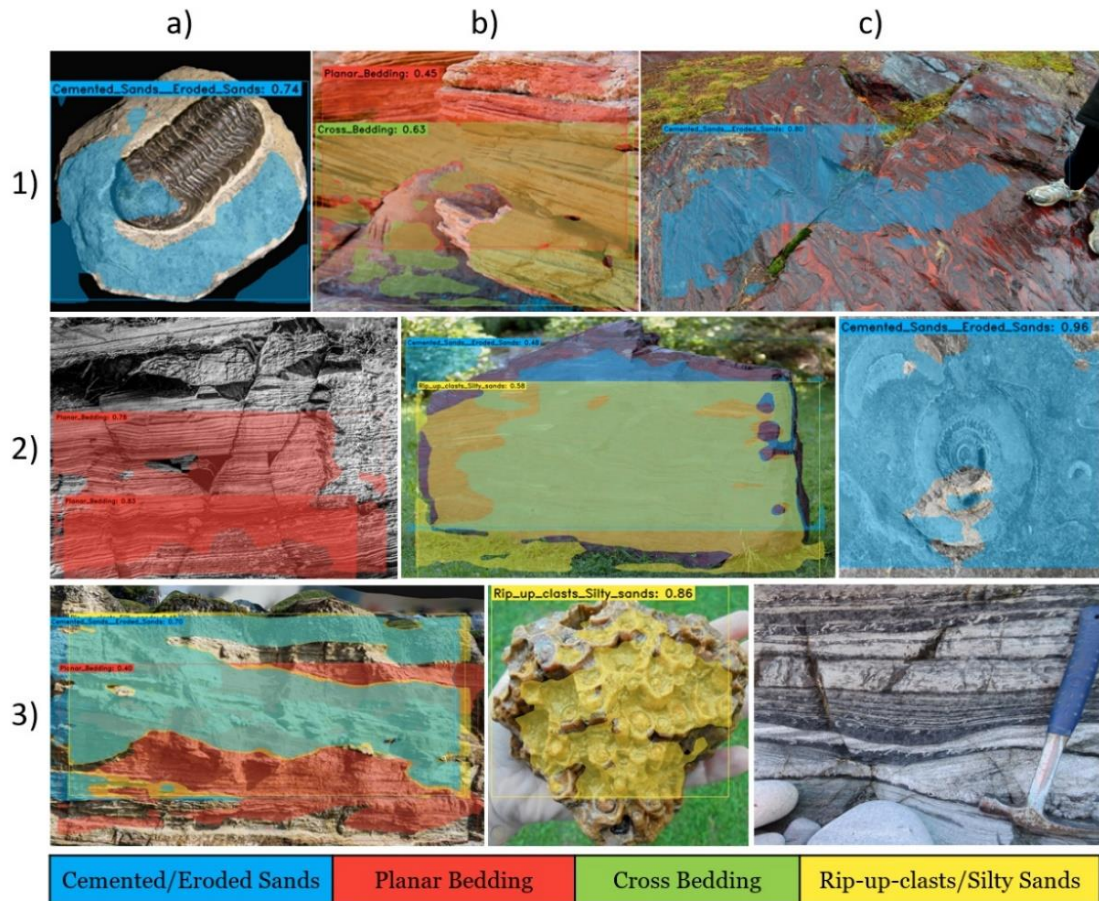


Figure 7-23: Application of the default YOLACT (DarkNet53) model on nine different outcrop images.

In Figure 7-23, the default YOLACT model from Experiment 1 was applied. Images 1b, 2b, and 3a display mask overlapping, making the results very hard to interpret. In addition, all images display wrong predictions, with one of them being partially correct (1b) while the rest are completely wrong in geological terms. Specifically, for image 1b, two labels are predicted, cross-bedding and planar bedding, while the ground truth is only cross-bedding. In general, severe label misclassifications, mask overlapping, and poor mask fit occurred.

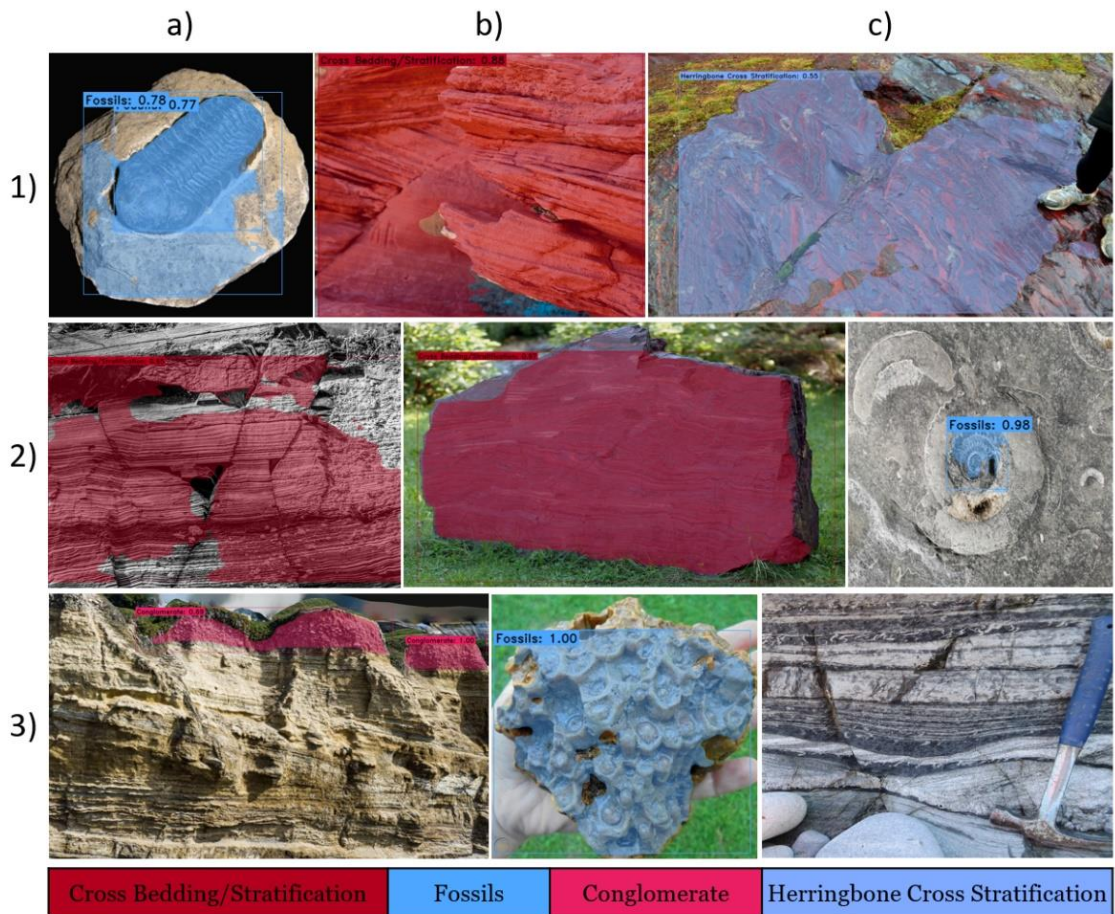


Figure 7-24: Application of the YOLACT (cDarkNet53) model on the same nine outcrop images shown in Figure 7-23.

In Figure 7-24, on the same images, the updated model was applied, and the results were significantly improved. For images 1a, 1b, 2c, and 3b, the Segmentation is successful both in terms of the predicted class and the mask/bounding box fit. It is evident that in some cases, like 1c, 2a, and 2b the model misclassifies the labels of the predictions while the masks and bounding boxes are fitted correctly around the geological objects. Examining closer image 1c, it is predicted as a herringbone cross-stratification, while in reality, we are looking at iron-rich sediment and deformed rocks. The deformation on this particular image looks like a herringbone cross-stratification pattern, which is indeed a label in Dataset 10. The same holds true for images 2a and 2b, where the model predicts the wrong class as there are elements in the images that mislead the model, such as a slight inclination of the bedding planes tricks the model into thinking that we are looking into cross-bedding. Image 3a has only a partial prediction and mask, while 3c has no prediction at all, related mostly to the small data variability.

Both Figure 7-23 and Figure 7-24 contain ‘bad’ examples of the model’s results. It is evident that in the case of the improved model, these bad examples are not as wrong as the poor results of the default model. The misclassified examples in both Figures are part of the misclassifications presented earlier in Table 7-4, Table 7-5, Table 7-7 and Table 7-9.

Splitting the segmentation task into two separate tasks yields improved results, as also shown in Figure 7-25. The Figure shows nine images with 1b and 1c displaying lithology predictions while the rest show predictions of sedimentary structures. The accuracy scores and predicted labels, along with the mask fits and bounding boxes, are very good, based on the ground truth.

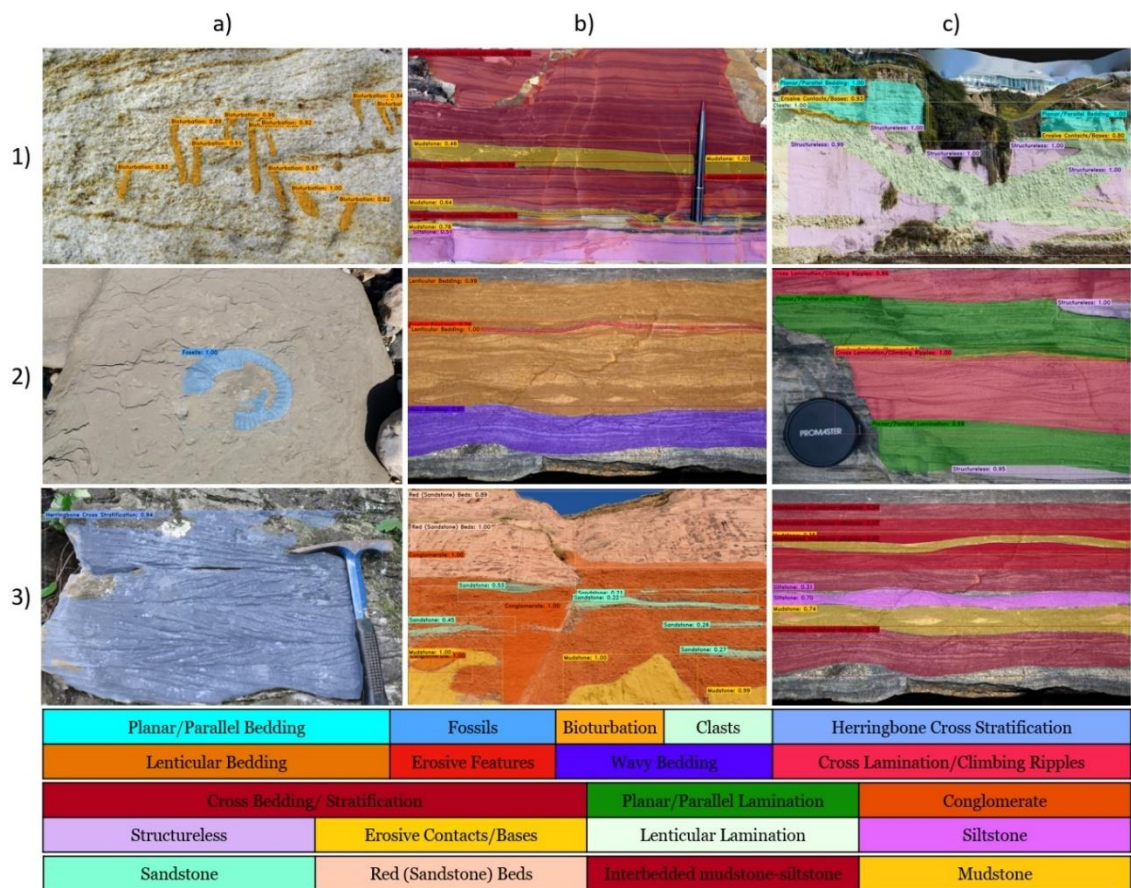


Figure 7-25: Results from the application of the YOLACT (cDarkNet53) model on nine different outcrops, displaying a larger variety of sedimentary features achieving higher accuracy and prediction scores.

The labels used to train the model are the most important criteria for the model’s accurate predictions. An example of mislabeling the dataset can be defined by getting some

erroneous predictions that, in reality, cannot coexist next to each other in the same outcrop, for instance, a Hummocky Cross Stratification right next to a cross-bedding as there is a scale mismatch in this case. The quality of the Segmentation predictions in experiment 2 has greatly improved in comparison to the initial results produced in experiment 1. The mAP scores, label prediction, along with the accuracy and fit of masks and bounding boxes, are shown to be enhanced.

The mAP scores indicate the model's accuracy with respect to the mask and bounding box fitting around the geological objects. These scores are calculated based on the model's predictions compared to the geological ground truth embedded in the annotations of the dataset. Certain misclassifications could be linked to misinterpreted or imprecise training images and annotations, directing the geoscientist to re-train the model by editing the annotations and by fine-tuning the model's hyperparameters or even the backbone's architecture to improve the Instance Segmentation results.

7.4.3 Experiment 3: Comparative study: YOLACT (ResNet101) vs (cDarkNet53) models

The third experiment, presented in this section, introduces the YOLACT (ResNet101) model and makes a comparison of the YOLACT model using two different backbones, the established ResNet101 and the cDarkNet53. A comparative study has been produced by training the lithology and sedimentary structures models on Datasets 10a and 10b, utilising the two different backbones while keeping all the hyperparameters the same.

Subsections 7.4.3.1 and 7.4.3.2 show the results of the YOLACT (ResNet101) model when trained once on Dataset 10a and once on Dataset 10b and tested to segment the lithology and sedimentary structures, respectively, as done previously in section 7.4.2 for the YOLACT (cDarkNet53).

Subsection 7.4.3.3 makes a direct and brief quantitative and qualitative comparison of the YOLACT (ResNet101) and the YOLACT (cDarkNet53) models. Finally, a visual comparison between all three segmentation models explained so far is provided.

7.4.3.1 Testing the YOLACT (ResNet101) model for Lithology Segmentation

This subsection shows a quantitative and qualitative analysis of the YOLACT (ResNet101) Segmentation model's results when tested on a set of images that includes

the same 50 images used for testing in sections 7.4.2.2 and 7.4.2.4. Table 7-10 and Figure 7-26 show the quantitative results of the YOLACT (cDarkNet53) model when trained on Dataset 10a and tested on outcrop images to segment the various lithology types present.

Table 7-10 provides a detailed breakdown of the model's performance per class/label present in Dataset 10a. The Table shows the number of labels present in the training and validation sets, the number of label appearances in the test data, the number of misclassifications per label, and the percentage of misclassifications in the entire test set.

Instance Segmentation Yolact for Lithology (ResNet101)				
Labels/Classes	Label Count in Training and Validation Sets	Predicted Label Appearances in test data	Misclassifications per Label (Class)	Percentage of misclassifications per class
Amalgamated/Cemented Bed	1	0	0	0
Breccia	5	5	0	0
Carbonates	7	8	0	0
Conglomerate	29	20	0	0
Interbedded mudstone-siltstone	27	31	2	6
Interbedded sandstone-mudstone	6	6	0	0
Interbedded sandstone-siltstone	21	27	2	7
Iron Rich Sediment	7	5	1	20
Mudstone	53	54	5	9
Organic Material	37	52	1	2
Red (Sandstone) Beds	21	23	3	13
Sandstone	79	82	3	4
Siltstone	28	35	4	11
Total	321	348	21	
Total Percentage of misclassifications for Test set, %				6

Table 7-10: Quantitative Results of the YOLACT (ResNet101) model. The model was trained on dataset 10b and tested on outcrop images to segment the various sedimentary structures present.

Table 7-10 shows the number of labels present in the training and validation sets, the number of label appearances in the test data, the number of misclassifications per label, and the percentage of misclassifications in the entire test set. The last row (orange colour) of the Table shows the total percentage of misclassifications for the entire Test set.

According to Figure 7-26, the top misclassified classes are the Siltstone and Mudstone Classes, with four and five misclassifications, respectively. The results in Figure 7-27 showed that some of these predictions were wrong but not far from the ground truth as the masks and bounding boxes were fitted correctly around the objects but the predicted class was wrong. In other words, an example was misclassified as siltstone, while the ground truth was mudstone.

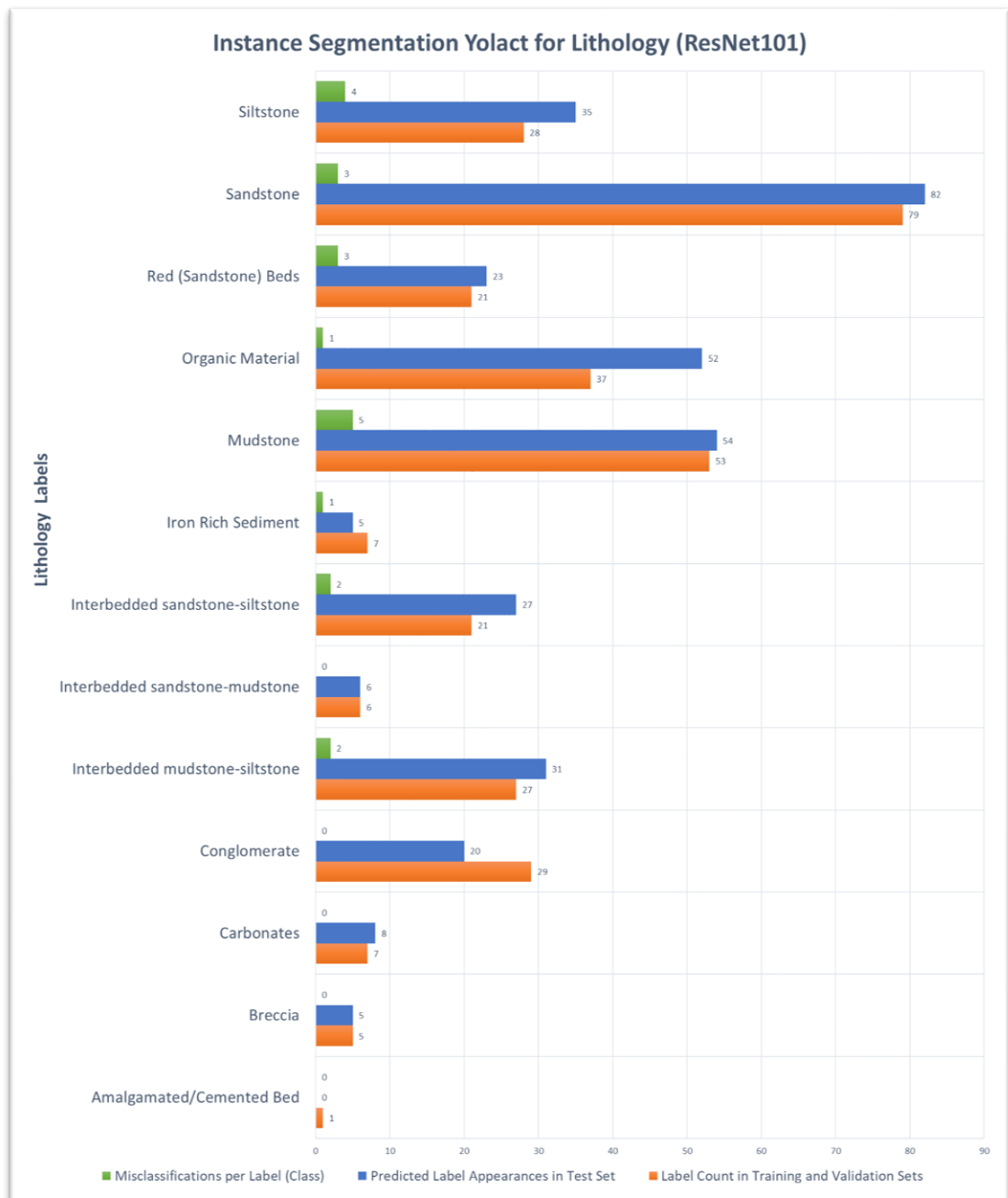


Figure 7-26: Quantitative Results of the YOLACT (ResNet101) model. The model was trained on dataset 10b and tested on outcrop images to segment the various sedimentary structures present.

A single image can contain multiple predictions of different or the same class, meaning that the number of predicted labels is expected to be higher than the total number of images in the training and validation sets, depending on the size of the test set. In this case, the test dataset included 50 images of outcrops. In these 50 images, there are 348 total appearances of geological objects, lithology types in this case. The total percentage of misclassifications across the test set was calculated as the ratio of the sum of

misclassifications per class over the sum of the predicted label appearances in the test set for each table, respectively. The resulting percentage of misclassifications in the test set adds up to 6% (Table 7-10), giving YOLACT a 94% accuracy for the particular test set.

A sample of the qualitative results of this section can be found in Figure 7-27. The YOLACT (ResNet101) was tested with the same twelve outcrop images used in sections 7.4.2.2 and 7.4.2.4 to estimate the lithology types in each image. Most predictions but two are correct when compared with the ground truth labels. The right-hand image from the 1st row demonstrates an example in which the model assigned the wrong label to the image. However, the masks and bounding boxes were still assigned to the correct place, showing improved mask fitting.

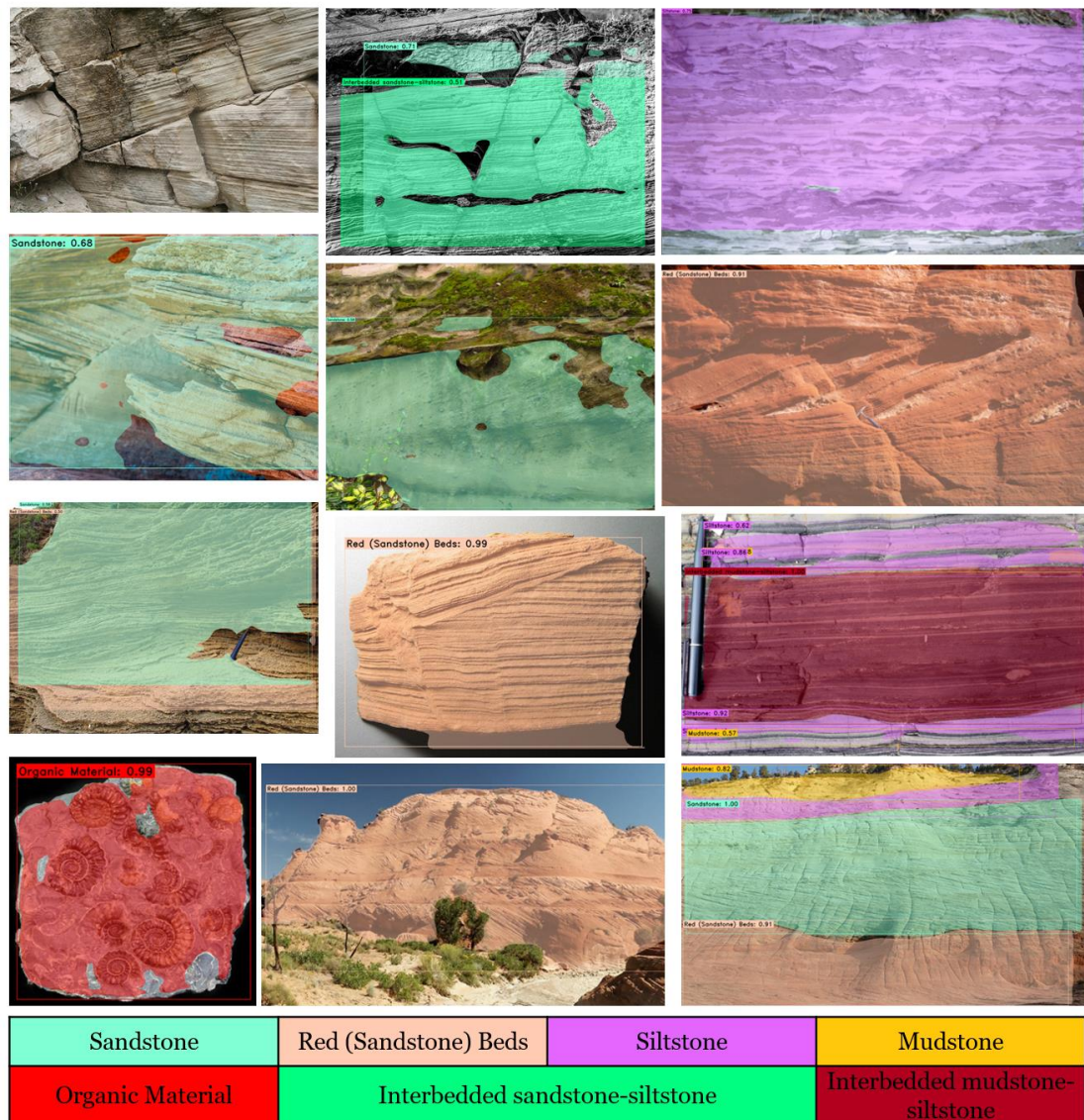


Figure 7-27: Qualitative Results of the YOLACT (ResNet101) model on the same twelve outcrop images as in sections 7.4.2.2 and 7.4.2.4.

Furthermore, in some images, the masks fit perfectly around the objects, accurately contouring the different lithology layers. Finally, in other images, the masks are sparser, indicating that the model was challenged to assign the masks. Overall, this model shows significant improvement compared to the model and results described in *section 7.4.1.2*.

7.4.3.2 Testing the YOLACT (ResNet101) model for Sedimentary Structures Segmentation

This subsection shows a quantitative and qualitative analysis of the abovementioned segmentation model's results when tested on a set of images. This particular test set includes the same 50 images as before. Table 7-11 and Figure 7-28 show the quantitative results of the YOLACT (ResNet101) model when trained on Dataset 10b and tested on outcrop images to segment the various sedimentary structures present.

Instance Segmentation Yolact for Sedimentary Structures (ResNet101)				
Labels/Classes	Label Count in Training and Validation Sets	Predicted Label Appearances in test data	Misclassifications per Label (Class)	Percentage of misclassifications per class
Bioturbation	48	32	0	0
Clasts	52	16	0	0
Convoluted/Irregular Lamination	2	1	0	0
Convoluted/Irregular Bedding	2	2	0	0
Cross Bedding/Stratification	39	36	8	22
Cross Lamination/Climbing Ripples	22	13	1	8
Dessication Cracks	5	4	0	0
Erosive Contacts/Bases	75	27	0	0
Erosive Features	31	11	1	9
Faults	16	7	0	0
Flame Structures	6	3	0	0
Flaser Lamination	3	3	2	67
Flute Marks	42	0	0	0
Fossils	12	11	0	0
Herringbone Cross Stratification	9	3	2	67
Hummocky Cross Stratification	10	2	0	0
Lenticular Bedding	8	4	0	0
Lenticular Lamination	5	3	0	0
Planar/Parallel Bedding	45	23	1	4
Planar/Parallel Lamination	29	12	0	0
Scour Marks	6	2	1	50
Structureless	56	32	0	0
Swaley Cross Stratification	4	1	0	0
Syneresis Cracks	2	1	0	0
Wave Ripples/Lamination	6	7	1	14
Wavy Bedding	5	3	0	0
Total	540	259	17	
Total Percentage of misclassifications for Test set, %				6.56

Table 7-11: Quantitative Results of the YOLACT (ResNet101) model. The model was trained on dataset 10b and tested on outcrop images to segment the various sedimentary structures present.

Table 7-11 provides a detailed breakdown of the model's performance per class/label present in Dataset 10b. The Table shows the number of labels present in the training and validation sets, the number of label appearances in the test data, and the number of misclassifications per label. The highlighted values, in yellow colour, in the Tables represent the higher percentages (>50%) of the misclassifications per class. The last row

(orange colour) of the Table shows the total percentage of misclassifications for the entire Test set.

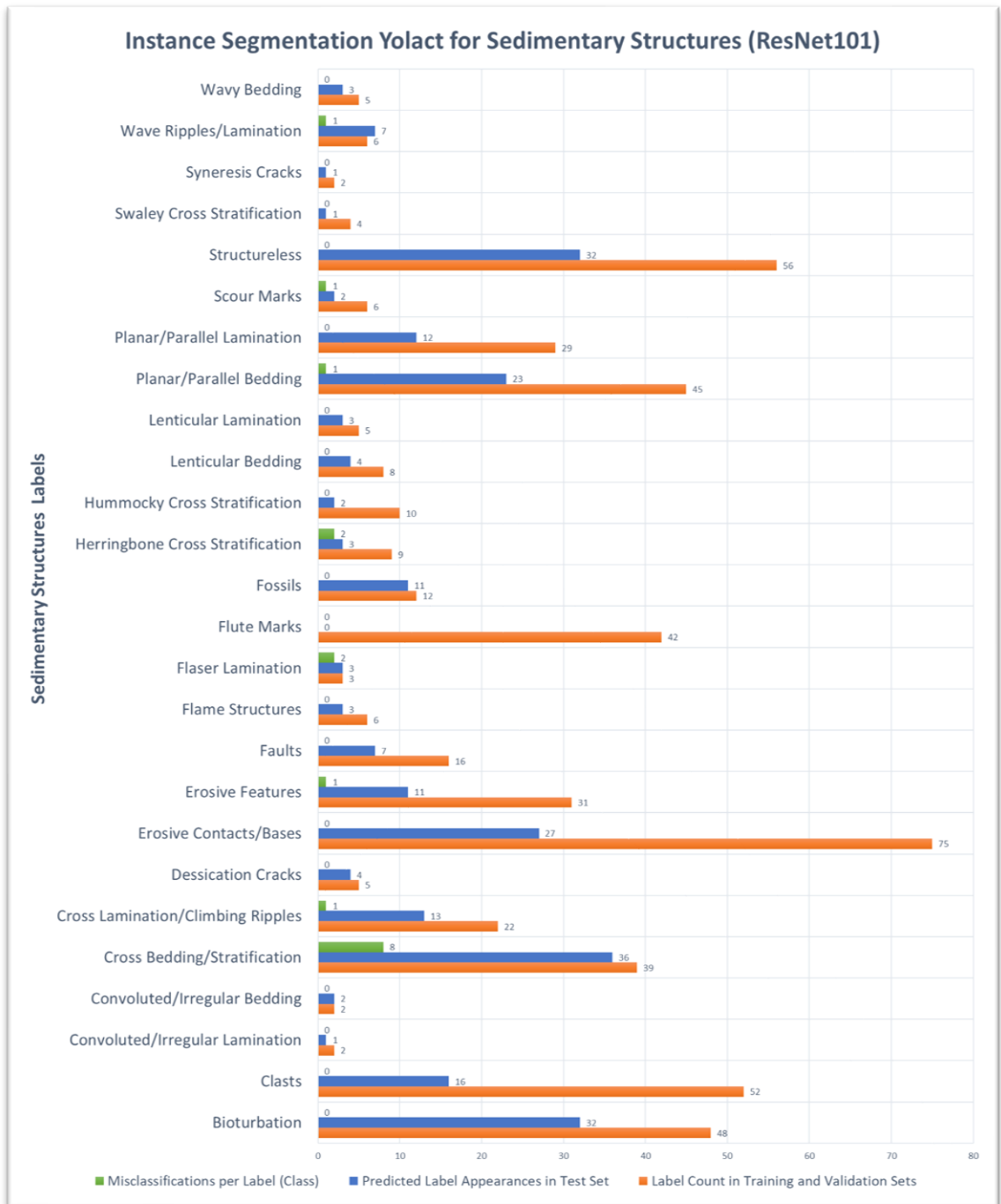


Figure 7-28: Quantitative Results of the YOLACT (ResNet101) model. The model was trained on dataset 10b and tested on outcrop images to segment the various sedimentary structures present.

According to Figure 7-28, the top misclassified class is again the Cross Bedding/Stratification class, with eight misclassifications. Although cross-bedding is

easy for a human geologist to identify, it seems to be a challenge for the Machine Learning model. As discussed in Chapter 6, cross-bedding proved to be difficult to detect and segment in this case, as the pattern of cross-bedding is widespread in nature as a full feature or part of other entities.

The results showed that some of these predictions were wrong but not far from the ground truth, while others were predicted wrong. This is due to the lack of data available to train the segmentation model but also to the inability of machine learning in general to obtain a holistic understanding of the image.

Similar to the previous section, the test dataset included the same 50 images of outcrops. In these 50 images, there are 259 total appearances of geological objects, sedimentary structures in this case. The total percentage of misclassifications across the test set was calculated as the ratio of the sum of misclassifications per class over the sum of the predicted label appearances in the test set for each Table, respectively. The resulting percentage of misclassifications in the test set adds up to 6.6% (Table 7-11), giving YOLACT a 93.4% accuracy for the particular test set.

A sample of the qualitative results of this section can be found in Figure 7-29. The YOLACT (ResNet101) was tested with the same twelve different outcrop images used in Figure 7-27 to segment the sedimentary structures in each image.

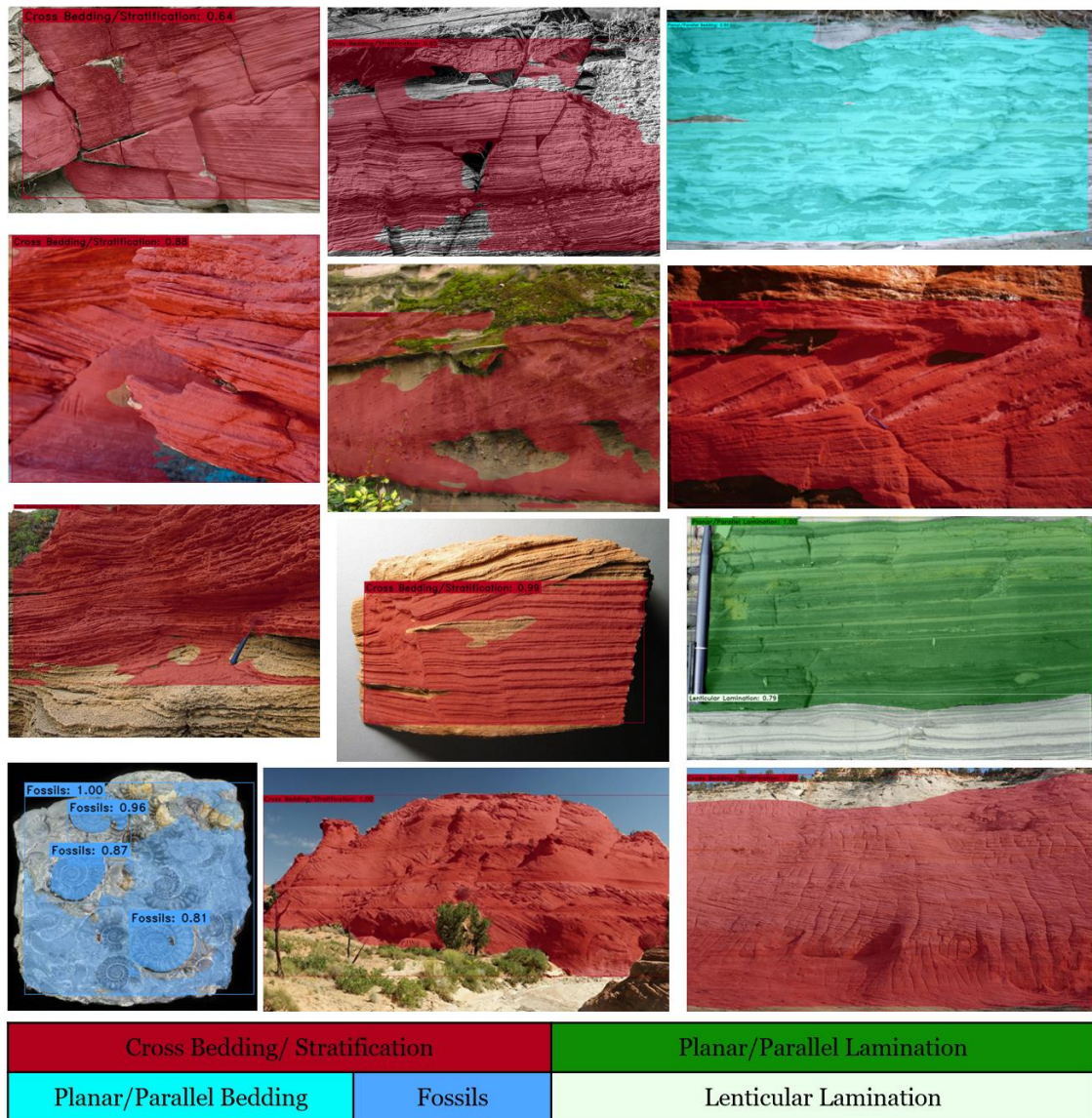


Figure 7-29: Qualitative Results of the YOLACT (ResNet101) model on the same twelve outcrop images as in sections 7.4.2.2 and 7.4.2.4.

Most predictions but three are correct when compared with the ground truth labels. The middle and right-hand images in the first row of images display two misclassifications according to the ground truth.

Furthermore, in all images in Figure 7-29, the masks fit well around the objects, accurately contouring the different structures. Overall, this model shows significant improvement compared to the model and results described in section 7.4.1.2.

7.4.3.3 Comparative Study: cDarkNet53 versus ResNet 101

A comparative study has been produced by training the lithology and sedimentary structures models by using two different backbones while keeping all the hyperparameters the same. In addition to the base backbone of ResNet-101 (He, et al., 2016), cDarkNet53, the modified version of DarkNet-53 (Redmon & Farhadi, 2018), was applied to obtain even faster results. If higher speeds are preferred, especially if the aim is to build real-time applications for geological Segmentation, the use of the cDarkNet-53 backbone with a combination of reduced training image size is recommended based on the results achieved in this study. A comparison of the mAP scores is shown in Table 7-12 for the two Segmentation models over the number of iterations.

Geology	Method	Backbone	Input Image Dimensions	mAP Score		Number of Iterations
				BBox	Mask	
Lithology	YOLACT-550	ResNet101	550x550	31.98	24.82	13333
	YOLACT-512	cDarkNet53	512x512	30.11	23.81	13333
Sedimentary Structures	YOLACT-550	ResNet101	550x550	58.37	53.59	32000
	YOLACT-512	cDarkNet53	512x512	57.27	52.09	32000

Table 7-12: Backbone comparison between the ResNet101 and cDarkNet53 backbones, respectively, for both the Lithology and Sedimentary Structures Segmentation models.

It is evident that both models have almost the same performance in terms of the mAP scores, with ResNet101 showing a slightly higher accuracy (Table 7-12) compared to the custom DarkNet53 (cDarkNet53). The ResNet101 tends to fit the masks better around the geological object. In contrast, the cDarkNet53 shows increased speed over the real-time inference, according to Table 7-13. The cDarkNet53 is significantly faster for real-time inference, almost by 10 frames per second (FPS) on average, making it more suitable for real-time applications. At the same time, ResNet101 may provide more robust masks and class prediction on static data (2D images). According to Figure 7-30, considering that for real-time inference, where 30+ FPS is needed (Bolya, et al., 2019), 10 fps is a significant difference.

Geology	Method	Backbone	FPS on Outcrop Data	FPS on Core Data
Lithology	YOLACT-550	ResNet101	31.34	34.1
	YOLACT-512	cDarkNet53	42.56	38.72
Sedimentary Structures	YOLACT-550	ResNet101	29.02	28.1
	YOLACT-512	cDarkNet53	39.42	38.65

Table 7-13: An FPS comparison between the two models we use on video data for real-time predictions.

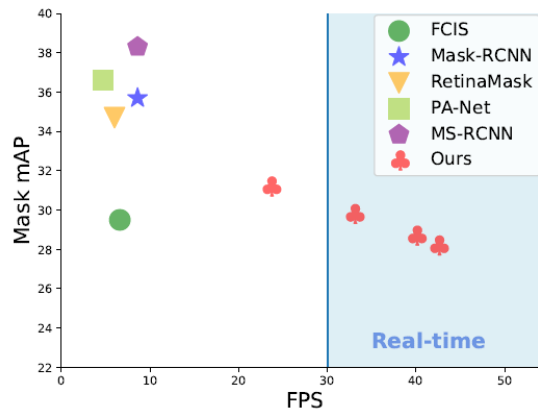


Figure 7-30: Speed-performance trade-off for various instance segmentation methods on COCO. To our knowledge, ours is the first real-time (above 30 FPS) approach with around 30 mask mAP on COCO test-dev (Bolya, et al., 2019).

Figure 7-31 shows the comparison of the two backbones when the model is applied on the same outcrop for the Segmentation of the lithology and sedimentary structures. ResNet101 segments the lithology slightly more accurately by applying a better fit of the masks around the centre part of the image as shown in Figure 7-32. Regarding the Segmentation of the sedimentary structures, both backbones yield the same results in terms of accuracy and mask fit. Both Segmentation results are accurate in terms of label prediction with ResNet101 providing more robust masks and class prediction on static data (2D images). The predictions (labels and masks) are almost identical regarding map scores (1% difference). But the ResNet101 fits the masks slightly better than cDarknet53 as shown in Figure 7-31 and Figure 7-32.

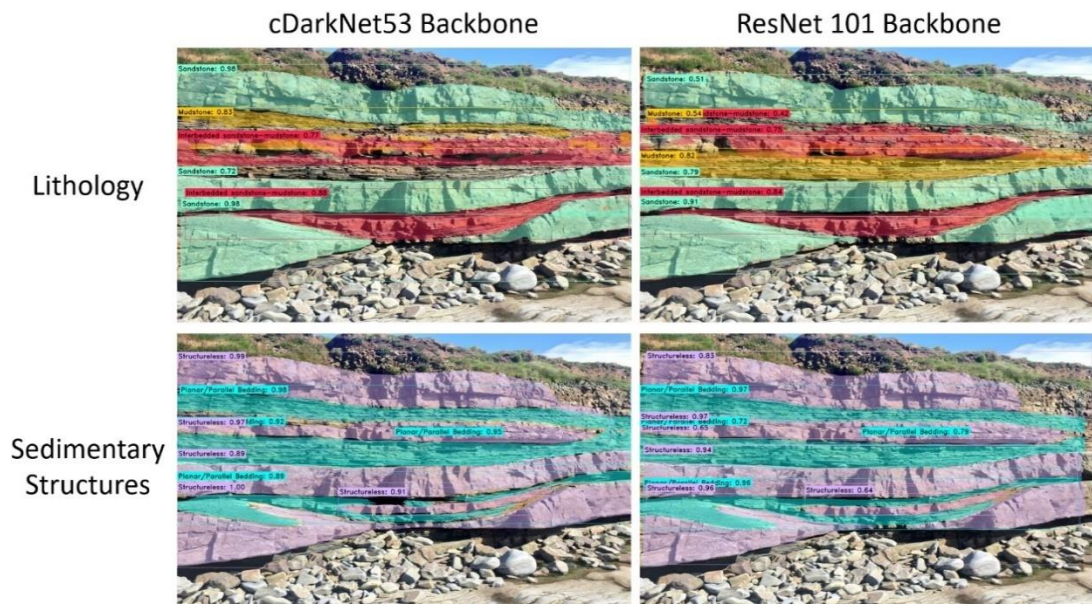


Figure 7-31: Backbone Comparison for the Lithology and Sedimentary structures models.

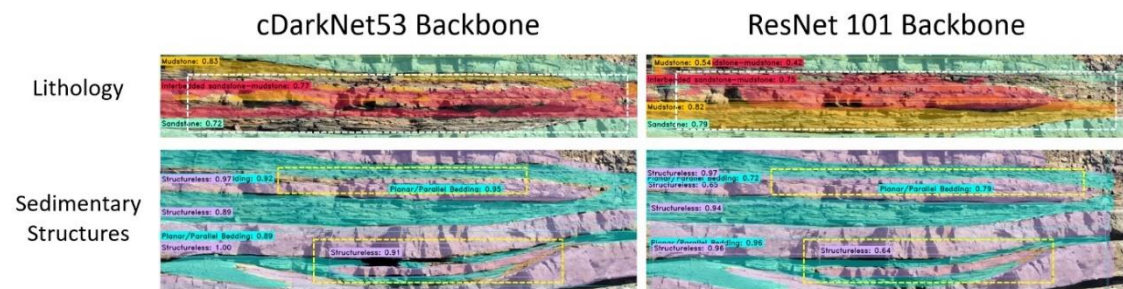


Figure 7-32: Backbone Comparison for the Lithology and Sedimentary Structures models depicted in more detail.

The last part of this experiment consists of a final comparison between all the three models described so far in this chapter, YOLACT (DarkNet53), YOLACT (cDarkNet53), and YOLACT (ResNet101). All three models were tested against the same outcrop images and the output images were juxtaposed. The comparison is shown in Figure 7-33 and Figure 7-34.

In Figure 7-33, the original outcrop image shows an outcrop (bedding scale) with big red sandstone beds (lithology) and very prominent cross-bedding (sedimentary structure). The YOLACT (DarkNet53) model fails to identify both the lithology type and the sedimentary structures present by assigning a completely wrong label and by misfitting the masks, not covering the objects well, while capturing unwanted information in the image such as the trees and vegetation. In contrast, both the YOLACT (cDarkNet53) and

the YOLACT (ResNet101) correctly predict the lithology and sedimentary structures present in the outcrop by assigning the correct labels, and by perfectly fitting the masks and bounding boxes around the geological objects, excluding all the unnecessary information.

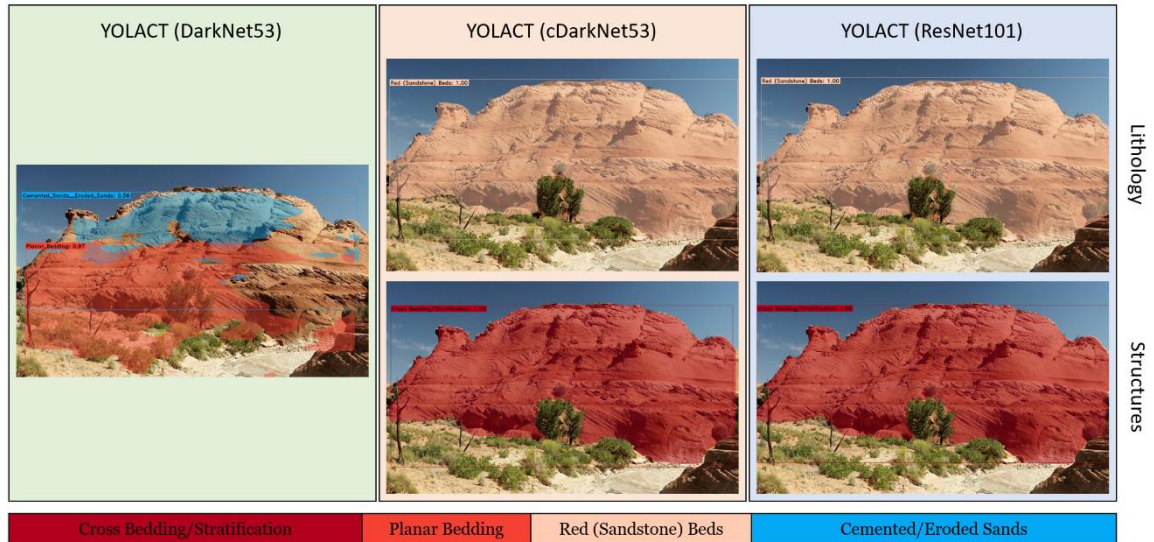


Figure 7-33: Comparison between all three models YOLACT (DarkNet53), YOLACT (cDarkNet53), and YOLACT (ResNet101) on a new outcrop image.

In Figure 7-34, the original outcrop image shows an image (laminae scale) with parallel laminated siltstone-mudstone and some lenticular lamination near the base of the image. The YOLACT (DarkNet53) model fails to identify both the lithology types and the sedimentary structures present by assigning a completely wrong label and by misfitting the masks, not covering the objects well, but most importantly, the prediction of Planar Bedding on a laminae scale is a serious error as it indicates scale mismatch. In contrast, both the YOLACT (cDarkNet53) and the YOLACT (ResNet101) correctly predict the lithology and sedimentary structures present in the outcrop. The two models correctly assigned the geological labels, and by perfectly fitting the masks and bounding boxes around the geological objects, clearly defined the layers and structures present in the image, with consistency regarding the scale of the objects.

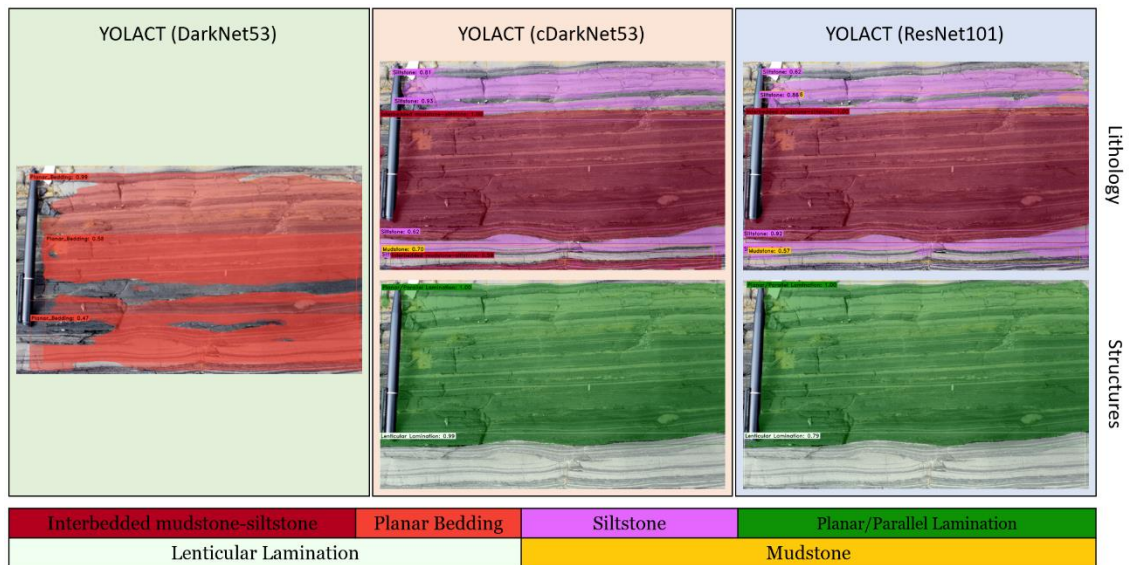


Figure 7-34: Comparison between all three models YOLACT (DarkNet53), YOLACT (cDarkNet53), and YOLACT (ResNet101) on another new outcrop image.

The conclusions derived from this third experiment is that depending on the task we are interested in we should choose the appropriate backbone for YOLACT. According to Table 7-12 and Table 7-13, if slightly better accuracy and fit of the masks are desired, then the best model to choose is YOLACT (ResNet101). If real-time predictions, faster inference and FPS performance is preferred, then the YOLACT (cDarkNet53) model is the best option. Finally, according to Figure 7-33 and Figure 7-34, when all three models are tested against two outcrop images, the YOLACT (DarkNet53) model is completely wrong in its predictions, while the other two models, YOLACT (cDarkNet53) and YOLACT (ResNet101), consistently make the correct predictions for all the lithology and sedimentary structures present in the images.

7.4.4 Experiment 4: Application of the YOLACT (cDarkNet53) model on core images

The last experiment of Chapter 7 deals with the application of the custom YOLACT (cDarkNet53) model on core images this time, to test the model's ability to generalize over not just different outcrops, but on a different geologic data type. Geologists examine the outcrops to ultimately infer the environment of deposition by linking all the different observations and knowledge extracted from the outcrops. To support and make their interpretation more robust, they combine different kinds of geological data. Such types of additional data include seismic data, well logs, geochemical data, and also core data. In comparison to the outcrop data that are laterally extensive and contain a lot of

information and features, the core barrels are only a meter tall and 10-15 cm wide. But, the process of interpreting the core is not much different from the interpretation process of an outcrop. Thus, this experiment aims to apply the knowledge the model accumulated from its training on the outcrops from the previous steps. Dataset 10 includes only outcrop images and not a single image of a core sample. The model aims to predict the same features present in dataset 10 but on a much smaller scale.

The segmentation model was tested on the same 11 images used previously in Chapter 6, section 6.4.3.2 in Figure 6-19. Several versions of the same core sample were generated by applying different image filters and effects on the image to alter its texture. Then YOLACT (cDarkNet53) was applied to all of the instances of this new core example, along with three additional core images, as shown in Figure 7-35 and Figure 7-36, for the segmentation of the lithology and sedimentary structures, respectively.

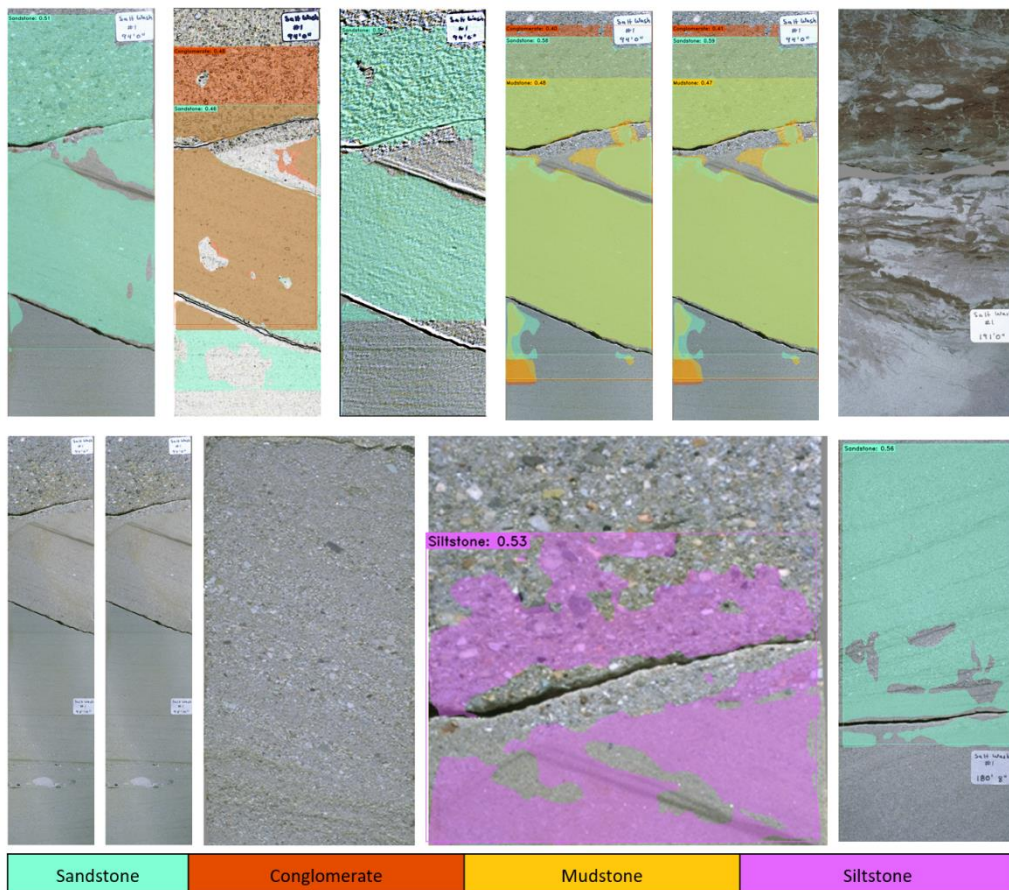


Figure 7-35: Application of the YOLACT (cDarkNet53) model for the segmentation of the lithology on core images.

Figure 7-35 shows that the model does not perform very well on the segmentation of the lithology on the particular core images. In the majority of images, the model predicts the sandstone label, which is correct according to the ground truth. However, there is mask overlap in some of the images, while in others, there are no predictions at all.

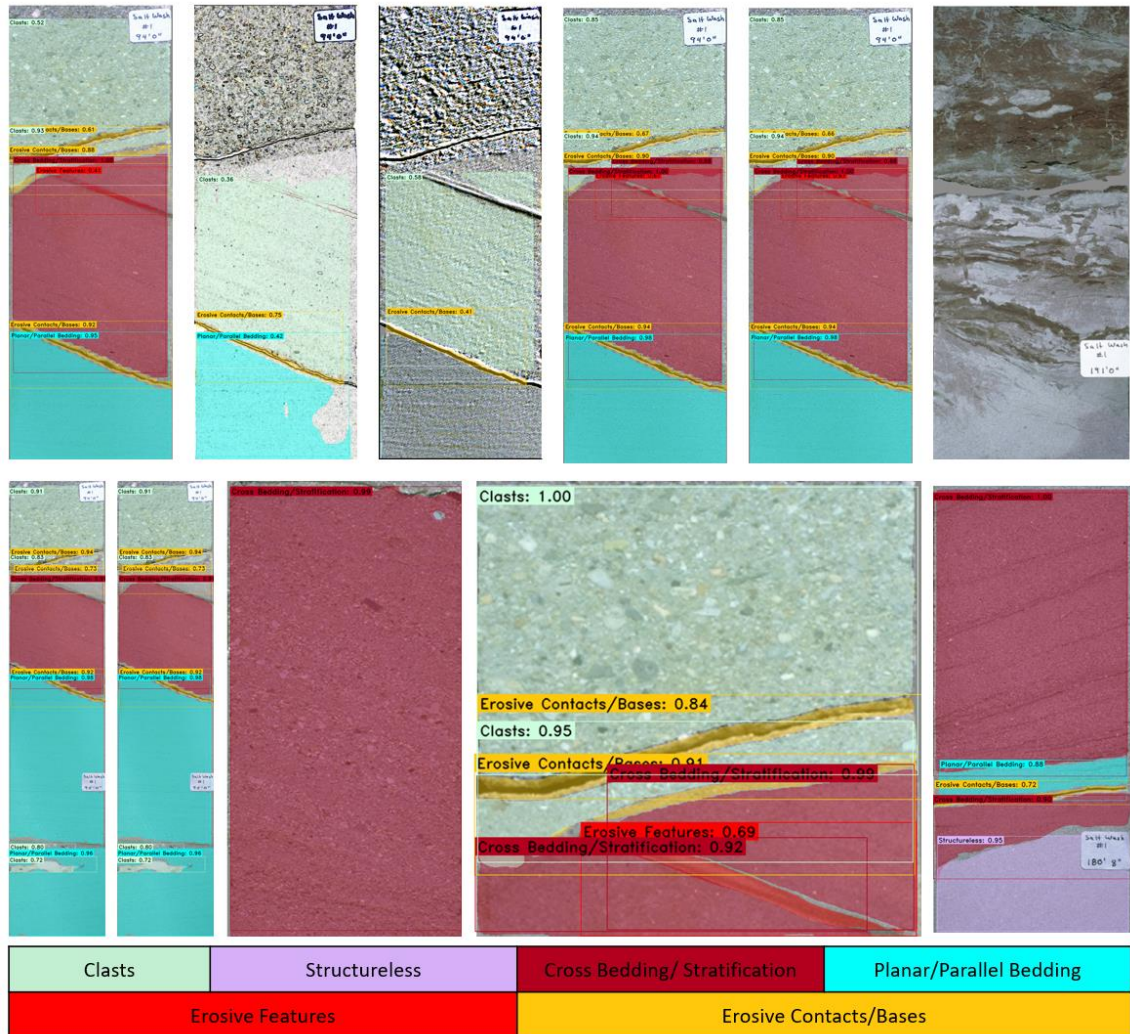


Figure 7-36: Application of the YOLACT (cDarkNet53) model for the segmentation of sedimentary structures on core images.

On the contrary, Figure 7-36 shows much better results for the segmentation of the sedimentary structures. The model displays correct predictions with detailed masks for all images but the 2nd and 3rd image (top row) in this figure. An important observation is that the second and third images in row one, trick the model and misguide its predictions, possibly due to the artificial change in texture and contours within the individual images.

Studying the results of the Instance Segmentation and Object Detection models, on the same 11 core images, we can conclude that both models can generalize well on their

predictions, as they can transfer their learning from the outcrop images directly to the core images without using any core images in their training. Therefore, if we incorporate core images in the training of both models, their performance will be significantly enhanced.

In order to showcase the advancements made in this chapter regarding the adaptation of the YOLACT model for geological applications, we conducted an experiment where the discussed model was applied to a video of core data. A snippet of the segmented video is shown in Figure 7-37, while the media file will be provided separately from the thesis document.

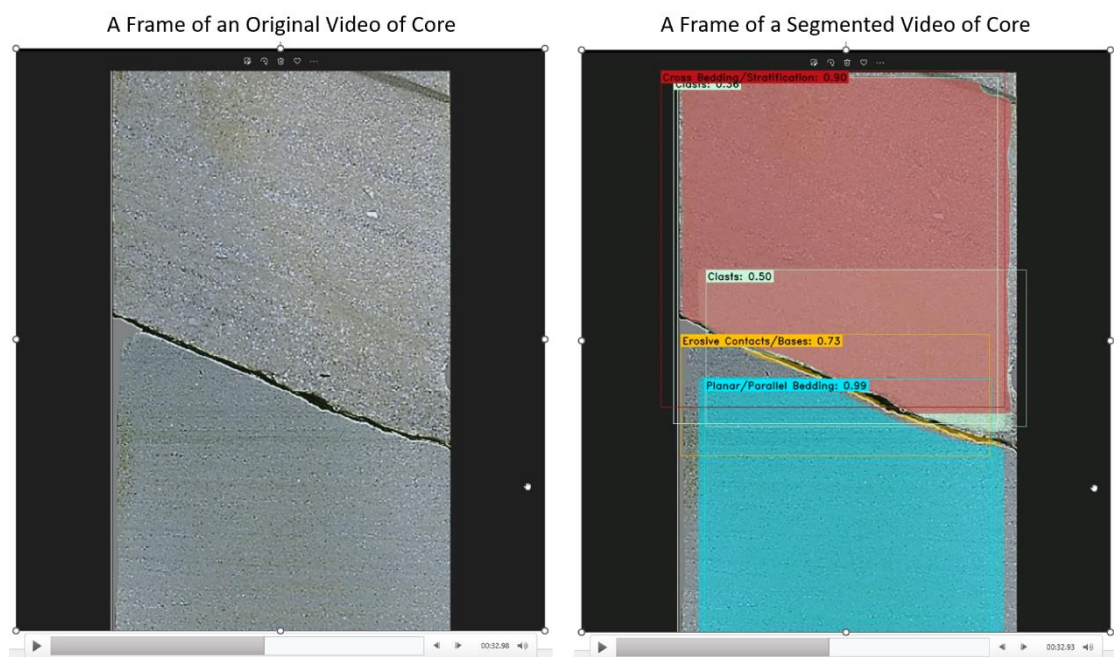


Figure 7-37: Real-time application of the default YOLACT (cDarkNet53) model on core data video.

Surprisingly, the real-time results of applying the model to this experiment were remarkably good, especially considering that the model had never been trained with core images before. This outcome demonstrates the model's generalizability and robustness, as it can learn from larger scale and more extensive outcrops and effectively apply that knowledge to fragmented core data. The model accurately predicts the presence of sedimentary structures, highlighting its capability to adapt and perform well in diverse scenarios.

As part of our future work, we intend to exclusively train this model using core images. This approach will further enhance the model's accuracy and enable us to incorporate lithology and fossil segmentation into its predictions.

From the comparison between the two Computer Vision methods, it is found that YOLOv6-S is more suitable for the detection of geological features from the core images when the models' training sets do not include any core images.

7.5 Chapter's Conclusions

This chapter presented the third Computer Vision method used in this thesis, Instance Segmentation, covering the 3rd step of the high-level workflow of the thesis presented in Chapter 1 (Figure 1-3). The chapter's results were derived from four different experiments to conclude in the selection of the appropriate data, model backbone, and configuration to segment the geological features of an outcrop successfully. There are various segmentation models available, but the chosen model was YOLACT, as it provides good results both for static and real-time inference.

7.5.1 Experiment 1 Conclusions

Experiment 1 clearly demonstrated that when using the YOLACT (DarkNet53), the model misclassifies image features and generates masks with significant overlap, likely due to its attempt to predict lithology and sedimentary structures simultaneously. The purpose of this experiment was to assess YOLACT's suitability for outcrop geology segmentation.

7.5.2 Experiment 2 Conclusions

In Experiment 2, the default YOLACT model was enhanced by using a modified version of the Darknet53 backbone called cDarkNet53, along with modifications to hyperparameters and the dataset used for training.

To refine and improve the segmentation outputs, the backbone of the default YOLACT (DarkNet53) model was modified into a shallower version (cDarkNet53), by removing a convolutional layer, leading to a smaller number of the model's total parameters.

The YOLACT model with the custom backbone (cDarkNet53), was trained twice. Once with Dataset 10a to segment the lithology and separately with Dataset 10b to segment the

sedimentary structures. Dataset 10 (a and b) contains images and labels from multiple outcrops and across both the laminae and bedding scales to enhance the variability in the dataset and, therefore, the generalization of the model's predictions on both seen and unseen data.

For the given test images, the lithology and the sedimentary structures are segmented separately. Dataset 9, due to its low variability, hinders the model's predictions, not allowing for good generalization on unseen data. Dataset 10, as it is more variable, improves the model's predictions on the same unseen data.

The higher the variability of the dataset, the better and more generalized results on unseen data. When the model is trained on Datasets 10a and 10b correctly identifies the patterns present in the tested outcrops, avoiding severe mask overlapping and improving the model's results' interpretability and accuracy.

7.5.3 Experiment 3 Conclusions

Experiment 3 introduced the YOLACT model with the ResNet101 backbone, which was trained and tested on the same data previously used with YOLACT (cDarknet53). This was done to conduct a comparative study between the two models. The choice of these backbones was based on their state-of-the-art usage in Segmentation and Object Detection tasks. ResNet101 is adept at extracting textures and patterns, and it is primarily used for Image Classification, while DarkNet53 is good at locating and classifying objects fast and is mainly used for Object Detection.

The hyperparameters for both backbones were the same, and both were tuned in the same way. The experiment's conclusions suggest that the appropriate backbone for YOLACT should be chosen based on the task at hand. Table 7-12 and Table 7-13 indicate that if the goal is slightly better accuracy and mask fit, the YOLACT (ResNet101) model should be chosen. On the other hand, if real-time predictions, faster inference, and FPS performance are preferred, the YOLACT (cDarkNet53) model is the best option.

Finally, Figure 7-33 and Figure 7-34 demonstrate that when all three models are tested against two outcrop images, the default YOLACT (DarkNet53) model makes completely incorrect predictions, while the other two models, YOLACT (cDarkNet53) and YOLACT (ResNet101), consistently predict all the lithology and sedimentary structures present in the images accurately.

7.5.4 Experiment 4 Conclusions

Experiment 4 presents the application of the trained YOLACT (cDarkNet53) model on core images to test its ability to generalize over not just different outcrops but also different types of geologic data. The study examines the results of the Instance Segmentation and Object Detection models on the same 11 core images, tested in Chapter 6.

The results of the experiment indicate that both models are capable of generalizing well on their predictions. They can transfer their learning from the outcrop images directly to the core images without using any core images in their training. This finding suggests that both the Instance Segmentation and Object Detection models are highly adaptable and can perform well on diverse geological datasets, which is a crucial aspect of their practical applicability.

The most reliable way to assess the model's performance is by obtaining feedback from a human geologist rather than relying solely on the mAP scores and confidence in the final predictions. This is because even if some labels or annotations are incorrect, the model may still predict the wrong label with high accuracy according to the numerical prediction confidence scores. However, in reality, the predicted label may be geologically incorrect.

Therefore, it is essential to have a human geologist evaluate the model during two specific stages of the general workflow of this chapter; the first stage is during the annotation step to ensure the model's practical understanding of the geology and the second stage is during the testing step to help improve the model's robustness. Obtaining feedback from a human geologist at these stages will help to ensure the accuracy and reliability of the model's predictions.

7.6 Summary of the Three Computer Vision Methods

The three Computer Vision Methods described so far in Chapters 5, 6 and 7 cover the first three steps of the thesis's high-level workflow and are all utilised to extract and identify from outcrop images any visual evidence, such as sedimentary structures and lithology types. The next and final chapter of the thesis will take all this information and cross-reference it with the literature to establish sequences of these features and their meaning. Such combinations and arrangements of features are what comprise the interpretation of a depositional environment.

Image Classification tends to be successful only on smaller scales regarding geological classification tasks, where there is only one feature present in the image. If the Image Classification model is presented with an entire outcrop displaying multiple features, it will fail because an outcrop cannot be classified or interpreted based only on single observations or features present. Object Detection and Instance Segmentation are good for identifying multiple features at once, while Image Classification is good for single-feature predictions.

As no model is perfect, the Segmentation of certain outcrops or parts of outcrops may sometimes provide insufficient information or even wrong predictions. At this stage, if additional details are needed, such as fossil description to distinguish between smaller geological features within images at a laminae scale, we can use image classification to get a more detailed description and classification.

The Image Classification model is much easier to set up and train because it is not computationally expensive. Although it provides a single prediction, it compares the confidence of that prediction across all the available classes.

The Object Detection model, YOLOv6-S, described in Chapter 6, was used to detect geological structures in different images, including images of outcrops, cores, and fossils. This chapter accentuates the importance of human evaluation during the annotation and testing stages to ensure the model's practical understanding of geology. The study concluded that YOLOv6-S has the potential to transfer geological knowledge from outcrop to core data and generalize well across different geological data types. Although YOLOv6-S's performance is impressive, one thing that it is not able to do is predicting the lithology in the outcrop images. It can undoubtedly assign bounding boxes and labels around the lithology types and layers; however, this will not be of much use as the geologists need to know and be able to define clear boundaries and distinctions between the various lithologies, an essential element of the outcrop interpretation and, consequently, that of the depositional environment.

Saying that there is a need for another Computer Vision method able to take Object Detection a step further by counting and capturing the detailed shape of each object in the image, in addition to its localization and label that Object detection offers.

Such a method is Instance Segmentation. Chapter 7 demonstrates the effectiveness of Instance Segmentation in accurately delineating the boundaries of various sedimentary

structures and lithology types in 2D outcrop images, alongside their recognition and localization. Instance Segmentation also provides more detailed information regarding the shape and location of each geological object and enables the estimation of lithology by using the masks in addition to the prediction and segmentation of the sedimentary structures. Furthermore, applying this model to real-time data makes it a novel approach and a valuable tool used in the field for outcrop segmentation on the fly.

Establishing a custom adaptation of the YOLACT model indicates that the segmentation model is capable of generalizing very well on its predictions and can transfer its learning from the outcrop images directly to the core images without using any core images in their training. This finding shows that such a model if trained with a larger and more diverse dataset, can provide great results for the segmentation of different geological data types.

CHAPTER 8 - INTERPRETING MULTIPLE DEPOSITIONAL ENVIRONMENTS BASED ON AVAILABLE DATA AND KNOWLEDGE WITH NLP AND NEURAL NETWORKS

8.1 Introduction

This chapter provides a comprehensive explanation of steps D and E within the high-level workflow (depicted in Figure 1-3), introduced in Chapter 1. It demonstrates how the results of Chapters 5, 6, and 7, integrated with Natural Language Processing (NLP) and Neural Networks (NN), can be employed to generate diverse geological concepts utilising existing published textual interpretations of the geology and observations collected from the outcrops. These interpretations and conceptual scenarios can capture interpretational uncertainties associated with a range of subsurface problems.

The first part of this chapter shows the application of a custom NLP model in a set of published pdf texts to extract geological labels of sedimentological features and their assemblage with the labels extracted from the outcrops previously with the Computer Vision methods.

The second part employs a custom neural network to combine all the available information by embedding geological knowledge into the model to predict the depositional environment by identifying features and finding the best fit of the label combinations.

Lastly, the third step displays the model's results and involves the use of a custom graphical user interface, producing and displaying several interpretations of the geology, each with an assigned probability, according to the user's input.

8.2 Natural Language Processing (NLP)

The objective of this section is to automate the text extraction process by leveraging a custom list of keywords, which encompasses various geological terms and features. Furthermore, the developed NLP model transformed the extracted information into a clean text format comprising strings of text. This clean data format has enabled the integration of these extracted geological insights into a custom neural network, which will be discussed in the following section (8.3).

Published geological literature provides geologists and machine learning models with knowledge, data, and analysis techniques. It serves as a vital resource in advancing geological understanding and enhancing the capabilities of machine learning models in geology. However, manually extracting relevant geological terms and features from this vast corpus of text is a time-consuming and labour-intensive task. To overcome these challenges, this section proposes the application of a custom Natural Language Processing model to automate the extraction process. Particularly, the geological terms I am seeking to extract from the text are terms that provide useful information for the interpretation of depositional environments, including lithology and fossil types, sedimentary structures, facies assemblages, etc.

In this thesis, Document Processing and Information Extraction techniques are employed to extract text from heritage geologic texts. The objective is to process PDF documents and identify specific geological keywords or expressions of interest, utilising a customized file containing these terms. The compilation of this keyword list draws upon my personal background knowledge, experience in the field, and the results obtained from computer vision methods previously discussed in the thesis.

Within the literature, numerous publications provide descriptions and interpretations involving various combinations of these terms. It is through these combinations that plausible interpretations of depositional environments are derived. Thus, the extraction of pertinent information related to these geological terms offers valuable insights into the processes, environmental conditions, and paleoenvironments associated with sediment deposition.

Through the analysis and interpretation of sedimentary structures, valuable insights into the paleoenvironment can be gained, including information about ancient currents, proximity to shorelines, and the overall depositional context. Cross-bedding and ripple marks, for example, indicate the influence of water currents, while the presence of mud cracks suggests periods of desiccation.

Lithology also provides important clues regarding the depositional processes and environmental conditions that affect sedimentary deposition. Different types of environments, such as fluvial systems, marine settings, or deltaic environments, can be inferred based on the identification of specific minerals or sedimentary textures, such as sandstone, shale, limestone, or conglomerate.

Fossil analysis within a sedimentary sequence further aids in understanding paleoenvironmental conditions. The presence of marine fossils within a limestone deposit suggests a marine environment, whereas terrestrial plant fossils within a coal seam indicate a swampy or deltaic setting. The types of fossils present serve as indicators of the environments in which the organisms lived.

Moreover, facies assemblages, which encompass the spatial distribution and associations of sedimentary features and rock units with distinct characteristics, play a crucial role in identifying the depositional environment. A combination of coarsening-upward sequences, cross-bedded sandstones, and marine fossils within a sedimentary sequence, for instance, can suggest a transition from a nearshore environment to a deeper marine setting.

The NLP identifies if these words or combinations of these words exist in the examined document, extracts them, and puts them into an Excel file, making them readable and in a clean text format, as shown in Figure 8-1.

Figure 8-1 demonstrates how a basic NLP pipeline processes the text from pdf files and extracts only the necessary geological information, following the steps of Tokenization, Text Cleaning, POS Tagging, Stop words, and Lemmatization.

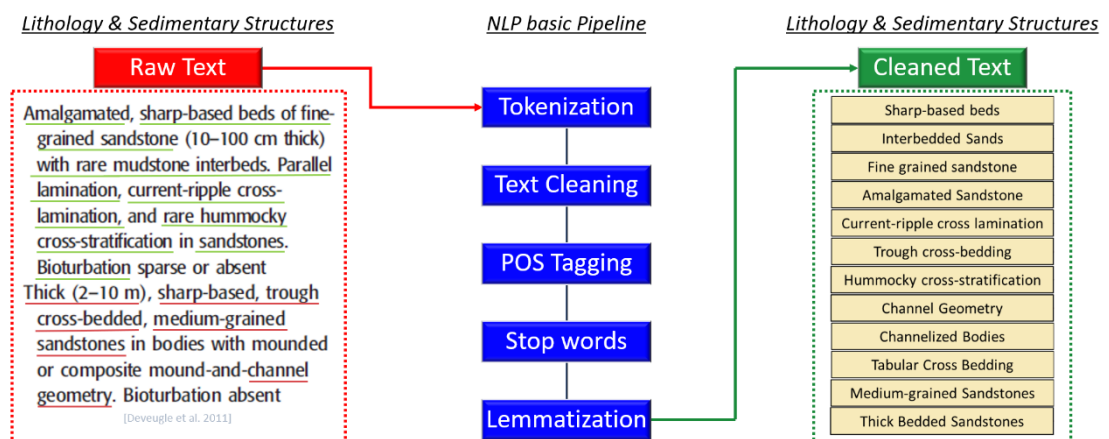


Figure 8-1: The figure shows the process of transforming unstructured geological text into structured by utilising Information Extraction and Document Processing. These methods helped to extract the relevant geologic terms and transform them into a string text format using a basic NLP pipeline.

Figure 8-1 depicts a systematic application of natural language processing (NLP) techniques to extract sedimentological features from a PDF file sourced from the geological literature (Deveugle, et al., 2011). The figure is divided into three main components:

The first component displays a section of text taken from the PDF file, which contains geological descriptions and observations related to sedimentological features, encompassing a broad range of geological terms.

The second component illustrates a comprehensive NLP pipeline tailored to geological analysis. This workflow focuses on identifying and extracting specific geological terms from the text. These terms may include lithological descriptions (e.g., sandstone, shale), depositional features (e.g., cross-bedding, ripple marks), sedimentary structures (e.g., channels, deltas), or other sedimentological indicators that are relevant to this thesis. The steps of the workflow are as follows:

Tokenization: The text is tokenized, breaking it down into individual words or tokens. This step ensures that each term and word is treated as a separate entity for further analysis.

Text Cleaning: The text undergoes a cleaning process specifically designed for geological content. This involves removing unnecessary noise, such as extraneous characters, symbols, or punctuation marks that might hinder the accurate extraction of sedimentological features.

POS Tagging with Geological Tags: Part-of-Speech (POS) tagging is performed, where each word is assigned a grammatical label specific to geological terms. This helps identify the role and context of each word in the sedimentological domain, allowing for more precise analysis.

Stop Words Removal: Stop words, such as common English words ("the," "and," "is"), are eliminated from the text to focus on the significant geological content. This step ensures that only relevant words and terms are considered for feature extraction.

Lemmatization: The words are lemmatized to reduce them to their base or root form within the geological context. This process aids in consolidating words with similar meanings, enabling consistent identification and analysis of sedimentological features.

The last component of the figure presents the outcome of the NLP *pipeline*: a clean and organized text format containing the extracted sedimentological features. This output represents a processed and structured version of the initial geological text from the PDF file.

These aforementioned five components are used to form a custom NLP workflow (Figure 8-2) to extract critical geological terms and knowledge from the selected pdf files. Teaching a computer model to understand the meaning of different geological features and their arrangements, as well as what their presence or absence indicates in a geological setting, is a difficult task, while for humans is an easier task due to our cognitive abilities.

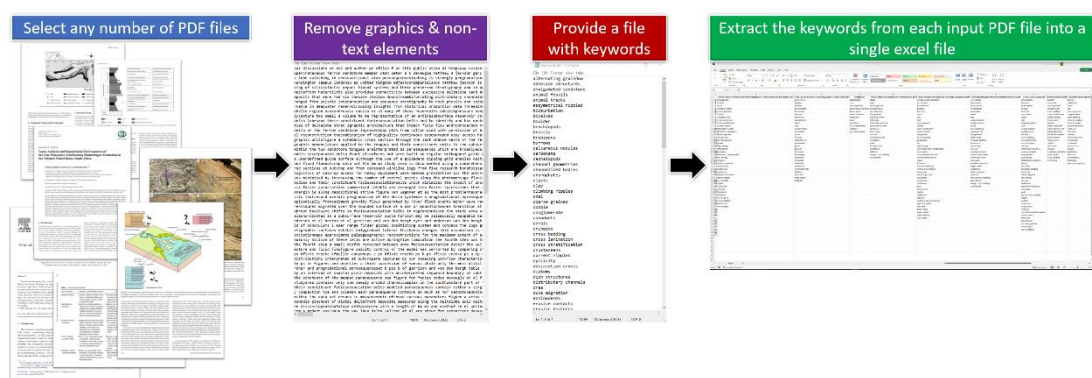


Figure 8-2: The main steps followed to extract the geological information from multiple geological pdf files into a single Excel file.

As shown in Figure 8-2, after selecting any number of PDF files, the NLP algorithm is to simplify the PDF files by removing graphics and non-text elements, ensuring that the subsequent NLP processes focus only on the relevant textual content. Following the simplification step, the NLP algorithm incorporates POS tagging and lemmatization as part of the text-processing phase. After extracting the textual content from the simplified PDF files, each word/token can be assigned a part-of-speech tag using POS tagging. Additionally, lemmatization can be applied to reduce words to their base forms, as mentioned earlier. Once the text has been processed through POS tagging and lemmatization, the NLP algorithm can search for the desired geological keywords specified in the custom-made ('keywords') text file. The keywords can be extracted based on the identified POS tags or by comparing the lemmatized forms of the words with the custom keyword list. The extracted keywords for each PDF file can be stored in a data frame, where each column represents the extracted keywords from a specific PDF file. This data frame is then exported into an Excel file, as mentioned in the workflow

description. After the Excel file is generated, which is the result of my NLP model, a manual review of the results is conducted to ensure grammar, spelling, and geological correctness, allowing for validation and refinement of the extracted keywords.

8.3 Custom Artificial Neural Network and Graphical User Interface to Interpret the Geology

The previously generated Excel file, including strings of text, is used by the custom artificial neural network explained in this section to embed the domain knowledge into my model. The custom model combines all this domain knowledge extracted with the help of NLP and the Computer Vision results of Chapters 5, 6, and 7, which provided visual evidence from the outcrops. The results of the model are produced and displayed through the use of a Graphical User Interface. The GUI allows a user-model interaction, through which the user can choose combinations of different text inputs from a long list of geological features compiled based on the strings of text present in the Excel file. Then, the network will analyse the inputs and produce several scenarios, each with an assigned probability.

8.3.1 Custom Artificial Neural Network

This section presents the construction and application of my custom Neural Network (NN) designed to process textual inputs derived from the previous Excel file and generate a diverse set of interpretations regarding depositional environments. To facilitate this process, an additional sheet was manually created within the Excel file, incorporating potential facies assemblages and geological feature combinations. These combinations were derived from a variety of sources, including relevant literature, personal expertise, and labels extracted from outcrops using computer vision (CV) methods.

Subsequently, the custom Neural Network was trained using this text dataset, utilising the concatenated input text strings. This training process considered both the individual features and their plausible combinations to generate multiple interpretations of depositional environments. By combining geologic keywords such as sedimentary structures, fossils, and lithology types, the Neural Network produces a range of depositional interpretations in the form of text strings, along with corresponding prediction probabilities.

Each prediction probability is calculated independently by the Neural Network based on the specific combination of individual inputs. These probabilities range from 0 to 1 and are indicative of the likelihood of a given interpretation. The final output comprises a list of potential interpretations, ranked in descending order of probability, thereby providing an ordered set of plausible depositional environment interpretations.

The proposed feedforward Neural Network model is built using the Keras API for Python (Keras, 2015), incorporating five layers of neurons that operate nonlinearly to transform the input data. This NN model comprises four dense layers with various activation functions and dropout layers that minimize overfitting. A dense layer, also known as a fully connected layer, is a type of Artificial Neural Network layer in which all the neurons or nodes in the layer are connected to all the neurons in the previous layer (Goodfellow, et al., 2016). In other words, each neuron in a dense layer receives inputs from every neuron in the previous layer, and its output is connected to every neuron in the next layer. The weights of these connections are learned during the training of the neural network through a process called backpropagation, in which the errors in the network's output are propagated backward to adjust the weights and improve the accuracy of the predictions.

The output layer features a number of units matching the categories of the geological dataset, employing a sigmoid activation function. The model uses the categorical cross-entropy loss function, the Adam optimizer, and the accuracy metric to adjust the neurons' weights in each layer and minimize the discrepancy between the predicted and actual output.

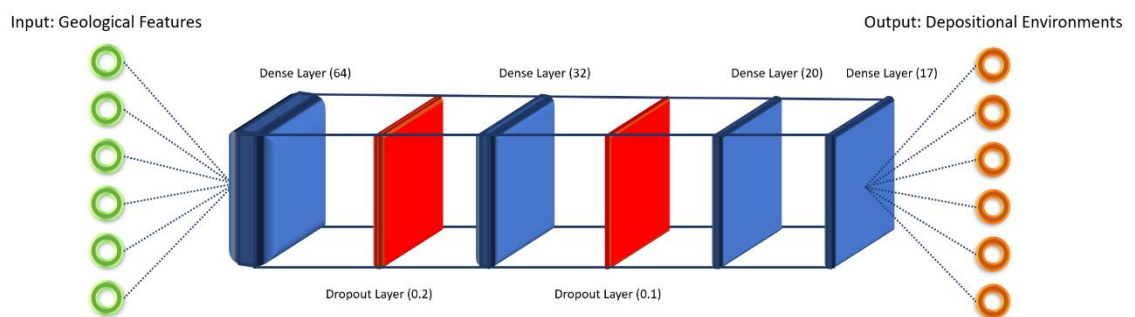


Figure 8-3: Custom NN model's architecture.

Figure 8-3 summarizes the described model's architecture, visualizing the layers shown in Figure 8-4. First, the necessary modules are imported to build this neural network model. Sequential is imported to create a sequential model, and Dense and Dropout are

imported to define the model's layers. An instance of the Sequential class is created and stored in the model's variable. The layers of the model are added one by one using the add method of the Sequential class.

The first layer added is a Dense layer with 64 neurons and the sigmoid activation function. The next layer is a Dropout layer with a rate of 0.2, which means that 20% of the input neurons will be randomly dropped during training to prevent overfitting. The third layer is another Dense layer with 32 neurons and the ReLU activation function. The fourth layer is a Dropout layer with a rate of 0.1, dropping randomly another 10% of the neurons. The fifth layer is another Dense layer with 16 neurons and the ReLU activation function. Finally, the output layer is a Dense layer with a number of neurons equal to the number of categories in the dataset and the sigmoid activation function.

The model is compiled using the compile method with the categorical_crossentropy loss function (Keras, 2015), the Adam optimizer, and the accuracy metric. During training, the model adjusts the neurons' weights in each layer to minimize the categorical cross-entropy loss between the predicted and actual outputs. The Adam optimizer is used to optimize the learning process, and the accuracy metric is used to evaluate the model's performance. The model is then trained using the fit method, with the training data and 100 epochs. The validation data split is also set to 20% of the training data.

Finally, the summary method is called to print the model's summary, as shown in Figure 8-4. This neural network model is a valuable tool for classifying strings of text resembling geological features into meaningful geological sequences. By learning from patterns in the training data, the model can combine text strings into coherent sequences that can be used for further analysis and interpretation.

Layer (type)	Output Shape	Param #
dense (Dense)	(32, 64)	10112
dropout (Dropout)	(32, 64)	0
dense_1 (Dense)	(32, 32)	2080
dropout_1 (Dropout)	(32, 32)	0
dense_2 (Dense)	(32, 20)	660
dense_3 (Dense)	(32, 17)	357

Total params: 13,209
 Trainable params: 13,209
 Non-trainable params: 0

Figure 8-4: Custom NN model's summary.

Once the model is trained, the weight file of the training is saved, encapsulating all the model's learning. The model is trained with strings of text and learns to combine certain strings of text to produce several outputs. The outputs are again strings of text describing the environments of deposition. In Figure 8-5, all these concepts are put in a geological context with a synthetic example to show visually how the custom neural network works. Figure 8-5 shows a synthetic example, utilising the data previously structured into a clean format with the NLP in Figure 8-1. Assuming that the model was trained with such data, the results would be similar to those shown in Figure 8-5. In order to create a similar visualization with real-world data, as shown in the next subsection, I created a custom graphical user interface that integrates my model with input data provided by a user to generate and display the model's predictions.

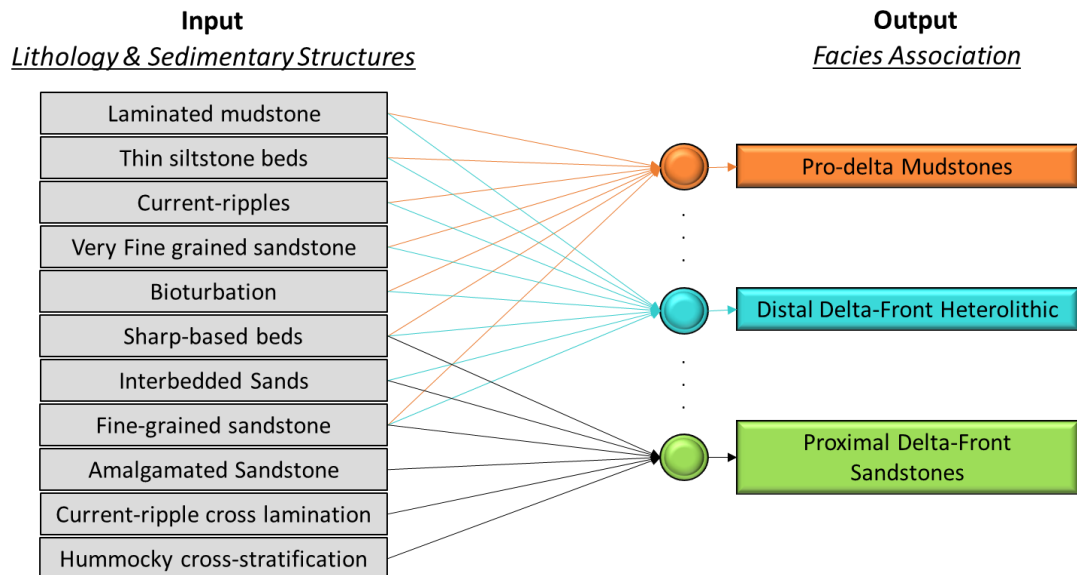


Figure 8-5: The results from the NLP model (Figure 8-1) are combined based on established interpretations from the literature (Deveugle, et al., 2011) to form plausible interpretations.

8.3.2 Streamlit Graphical User Interface

A custom Graphical User Interface (GUI) is the last component of the AI system described in this thesis. The GUI model for this thesis was named G.A.I.A, standing for Geological Assisted Interpretation Application, and was used to run the trained neural network and display its predictions.

To build this custom Graphical User Interface, a) the Excel file containing all the extracted information from the literature and b) the weights file resulting from the training of the NN are utilised in a final script, combining both elements to generate multiple interpretations of the depositional environment. This script was developed utilising the Streamlit Python library to build a custom Graphical User Interface (GUI).

The GUI accepts input from the user, who can select between several characteristics of depositional environments, sedimentary structures, and types of fossils and lithologies. Any number and combination of inputs can be selected, and according to the selection, the GUI will display a sentence describing the top prediction of the depositional environment, followed by a list of all the possible predictions with their corresponding probabilities.

The prediction of the model and the probability for each depend on the combination of the user's selected input. Each probability is independent for every prediction and ranges from 0-1, calculated with up to three decimal places. These probabilities facilitate the understanding of the relative certainty or uncertainty of each interpretation, enabling geologists to assess the robustness of the results and make well-informed decisions considering the inherent uncertainties in the process of depositional environment interpretation.

8.4 Chapter's Results & Discussion

In order to generate multiple interpretations of the depositional environment based solely on the text descriptions derived from the results of the NLP and computer vision models earlier in this thesis, I trained a custom artificial neural network (ANN) on this text data. To this end, the NLP results were recorded in an EXCEL file and incorporated into the ANN's training. The resulting ANN outputs were visualized using a graphical user interface (GUI) presented in this section.

To validate the proposed approach, the ANN was further challenged to interpret the depositional environment not only from outcrop images but also from core images and sedimentary logs. Three distinct test cases were presented utilising the GUI model to interpret the environment of deposition based on the aforementioned geological data types.

To better understand the results that follow, please refer to Figure 8-6, which showcases an example of the user interface and its corresponding components. On the left-hand side of the figure, you will see the user's input, which originates from the predicted labels of the computer vision models. On the right-hand side of the figure, you will find the predictions of the depositional environment along with their associated probabilities. The 'Test Image Area' on the following figures is not part of the GUI's layout or results. It has been manually added to provide visual evidence of the geological features present and as a reference point for interpreting the depositional environment.

It is worth noting that a test image has been provided for each example presented in all the results sections, showcasing the corresponding inputs. Please keep in mind that the test outcrop and core images have already been discussed in Chapters 6 and 7.

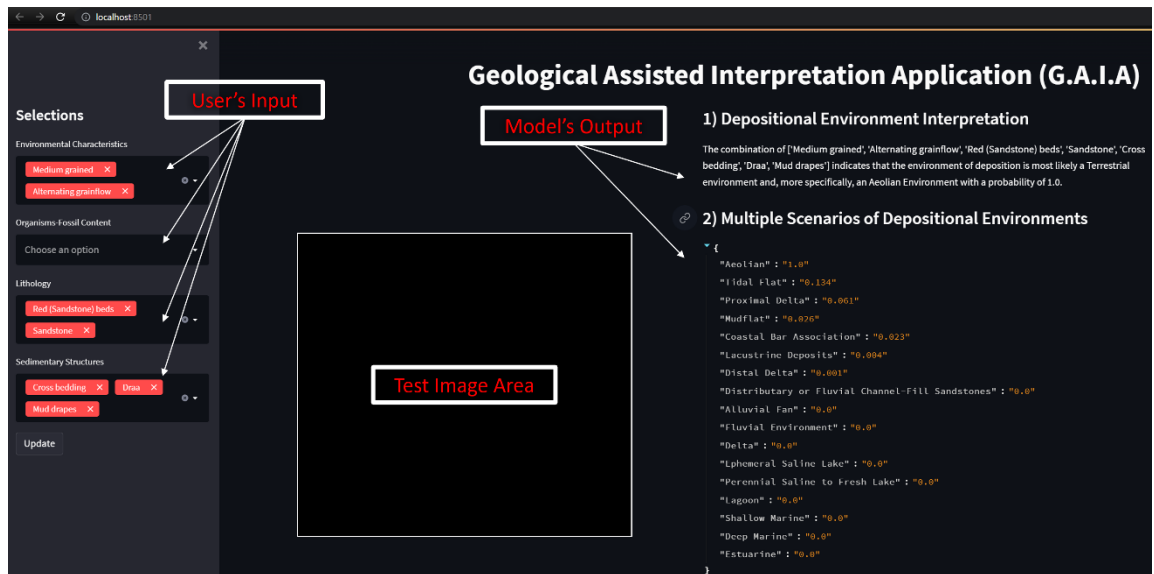


Figure 8-6: Sample of the Graphical User Interface layout with the fields and outputs.

Please note that Figure 8-6 is demonstrating how the concept described in Figure 8-5 is applied to real test data and user input.

8.4.1 Test Case 1: Interpret the Depositional Environment from Outcrop Images

The GUI model was initially tested with four different outcrop images. For all the tested examples, the interpretation of the depositional environment is validated with the published literature. The GUI predicts a list of possible interpretations, which is then compared to the ground truth to determine the model's robustness, as shown in Table 8-1.

Examples	Data Type (Images)	Ground Truth	Top 1 Prediction	Top 1 Probability	Top 2 Prediction	Top2 Probability
Example 1	Outcrop	Deep Marine/ Turbidites	Deep Marine	0.999	Delta	0.966
Example 2	Outcrop	Deep Marine/ Turbidites	Deep Marine	0.993	Delta	0.891
Example 3 (Dawlish, UK)	Outcrop	Fluvial/ Aeolian	Fluvial	1.0	Aeolian	0.273
Example 4 (Exmouth, UK)	Outcrop	Fluvial/ Alluvial	Fluvial	1.0	Aeolian	0.554

Table 8-1: GUI model's predictions based on outcrop images.

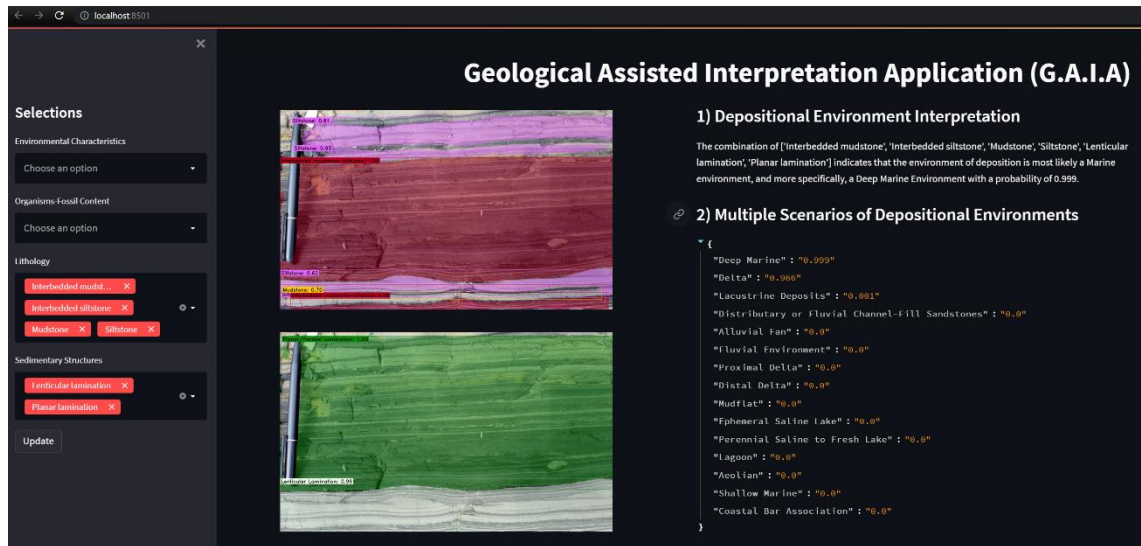


Figure 8-7: Example 1 of the GUI model's predictions based on outcrop images.

The first outcrop example, Figure 8-7, is an outcrop image with geological features from a deep marine environment interpreted by an expert geologist as turbidites. This image had previously been segmented in Chapter 7 to obtain the lithology and sedimentary structure labels using Instance Segmentation. These labels are now used as input in the user interface, as shown in Figure 8-7 on the left-hand side. The possible fields for geological features include general characteristics of depositional environments, fossil content, as well as lithology types, and sedimentary structures.

According to the user's input and the geological features present in the images, the G.A.I.A. output, shown on the right-hand side of the figure, is multiple predictions of depositional environments, with the top prediction being a Deep Marine environment with a probability of 0.999, which is the correct answer. The second top prediction is that the environment of deposition is a Delta, with a probability of 0.996. In neither case the term turbidites was shown in the model's results. This is because the term was never encoded in the model. In the future, the model will include more geological terms as more geological information will be embedded into the model. Nevertheless, the model's top1 prediction was correct because it classified the environment of deposition as a Deep Marine environment (Turbidites indicate a deep marine environment).

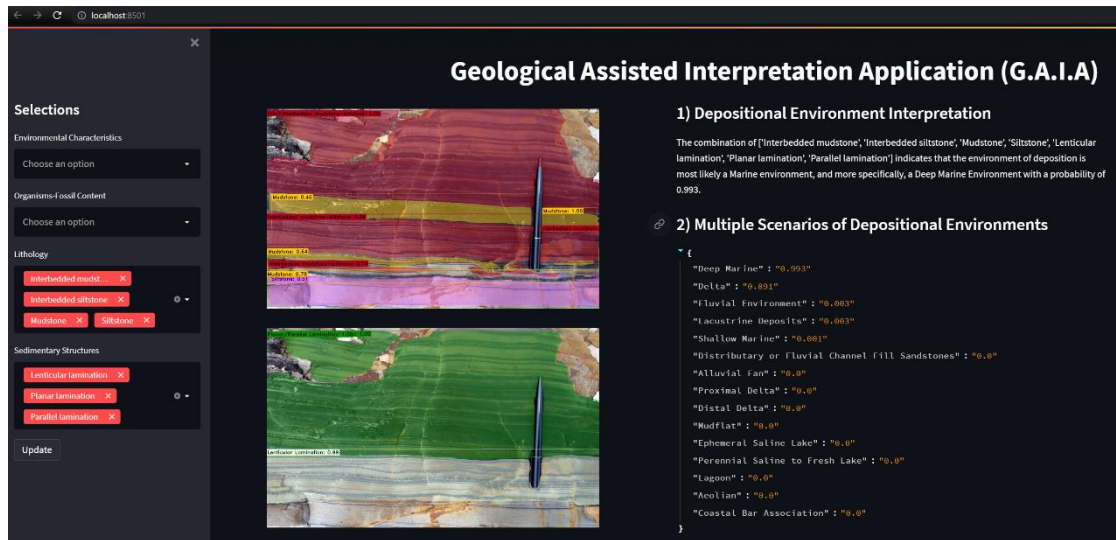


Figure 8-8: Example 2 of the GUI model's predictions based on outcrop images.

The second outcrop example, Figure 8-8, is an outcrop image with geological features again from a deep marine environment interpreted by an expert geologist as turbidites. This image had also been segmented in Chapter 7 to obtain the lithology and sedimentary structure labels using Instance Segmentation. These labels are used as input in the user interface, as shown in Figure 8-8 on the left-hand side.

The G.A.I.A. output, shown on the right side of the figure, is multiple predictions of depositional environments, with the top prediction being a Deep Marine environment with a probability of 0.993, which is the correct answer. The second top prediction is that the environment of deposition is a Delta, with a probability of 0.891. In neither case the term turbidites was shown in the model's results for the reasons mentioned in the first example. Nevertheless, the model's top1 prediction was correct because it classified the environment of deposition as a Deep Marine environment (Turbidites indicate a deep marine environment).

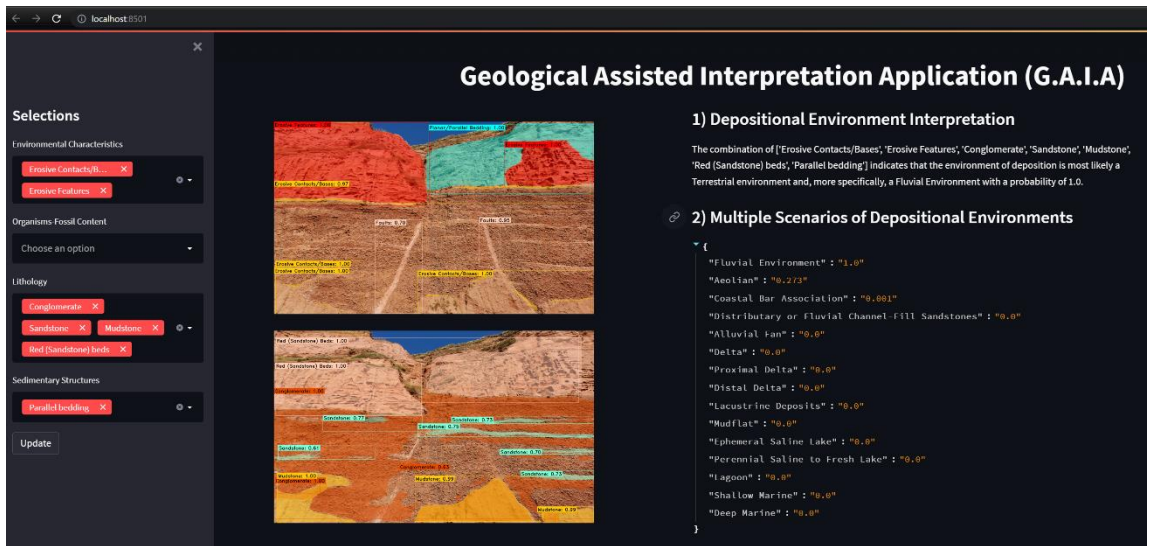


Figure 8-9: Example 3 of the GUI model's predictions based on outcrop images.

The third outcrop example, Figure 8-9, is an outcrop image with geological features from a fluvial/aeolian environment interpreted by an expert geologist as a fluvial. This image was segmented in Chapter 7 to obtain the lithology and sedimentary structure labels using Instance Segmentation. These labels are used as input in the user interface, as shown in Figure 8-9 on the left-hand side.

The G.A.I.A. output, shown on the right side of the figure, is multiple predictions of depositional environments, with the top prediction being a Fluvial environment with a probability of 1.0, which is the correct answer. The second top prediction is that the environment of deposition is an Aeolian, with a probability of 0.891.

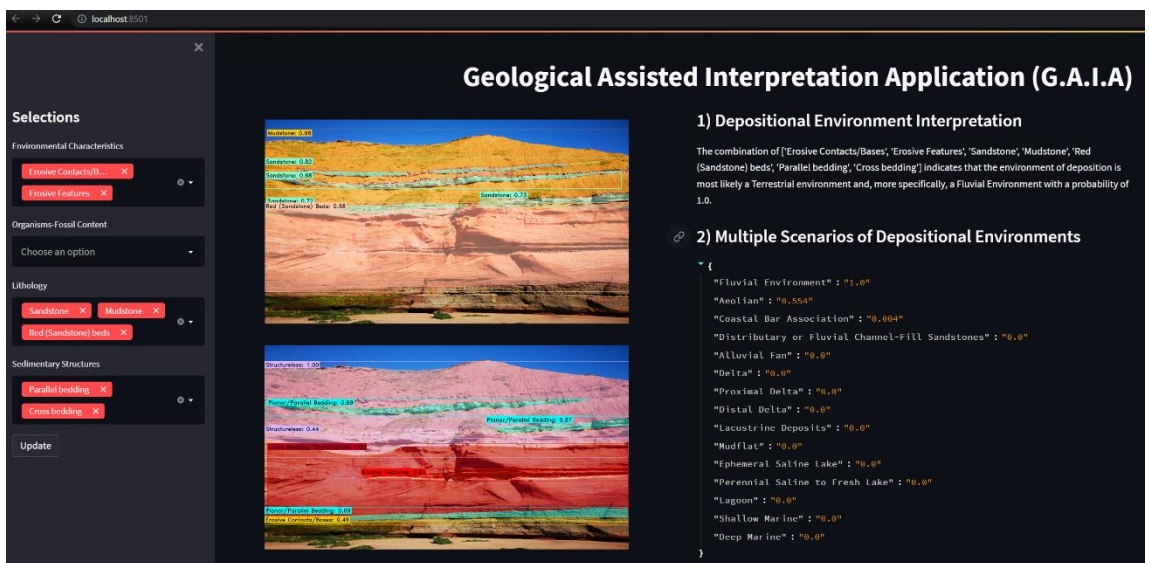


Figure 8-10: Example 4 of the GUI model's predictions based on outcrop images.

The fourth outcrop example, Figure 8-10, is an outcrop image with geological features from a fluvial/aeolian environment interpreted by an expert geologist as a fluvial. This image was segmented in Chapter 7 to obtain the lithology and sedimentary structure labels using Instance Segmentation. These labels are used as input in the user interface, as shown in Figure 8-10 on the left-hand side.

The G.A.I.A. output, shown on the right side of the figure, is multiple predictions of depositional environments, with the top prediction being a Fluvial environment with a probability of 1.0, which is the correct answer. The second top prediction is that the environment of deposition is an Aeolian, with a probability of 0.554, which is not correct when compared to the ground truth.

Overall, the results of this test case demonstrated that the G.A.I.A. model correctly predicts the depositional environment on its top 1 prediction 4/4 times, given four different outcrop examples. In the next two sections, the G.A.I.A. model will be tested with two additional data types, core images and sedimentary logs.

8.4.2 Test Case 2: Interpret the Depositional Environment from Core Images

In this section, the GUI model was tested with labels extracted from Figure 6-19, Figure 7-34 and Figure 7-35, displaying core images. For all the tested examples, the interpretation of the depositional environment is validated with the published literature. The GUI predicts a list of possible interpretations, which is then compared to the ground truth to determine the model's robustness, as shown in Table 8-2.

Examples	Data Type (Images)	Ground Truth	Top 1 Prediction	Top 1 Probability	Top 2 Prediction	Top2 Probability
Example 5 (Salt Wash, USA)	Core	Fluvial	Fluvial	1.0	Delta	0.739
Example 6 (Salt Wash, USA)	Core	Fluvial	Fluvial	1.0	Delta	0.028

Table 8-2: GUI model's predictions based on core images.

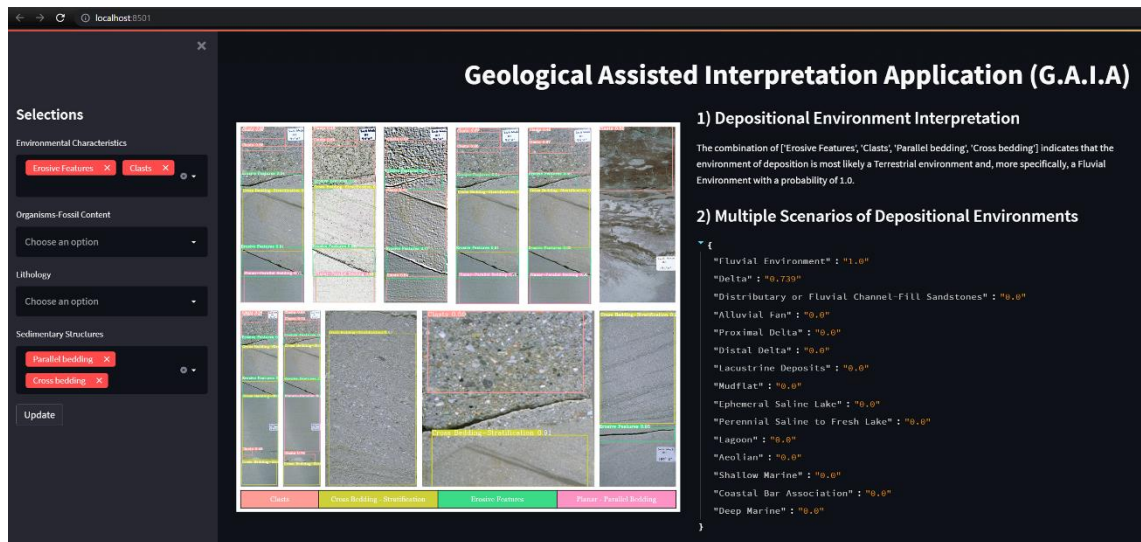


Figure 8-11: Example 5 of the GUI model's predictions based on core images.

The fifth example, Figure 8-11, is a core image with geological features from a fluvial environment, according to the interpreted core data from SEPM (Society for Sedimentary Geology). This image was used in Chapter 6 to obtain the sedimentary structure labels using Object Detection. These labels are used as input in the user interface, as shown in Figure 8-11 on the left-hand side.

The G.A.I.A. output, shown on the right side of the figure, is multiple predictions of depositional environments, with the top prediction being a Fluvial environment with a probability of 1.0, which is the correct answer. The second top prediction is that the environment of deposition is a Delta environment, with a probability of 0.739, which is not correct when compared to the ground truth.

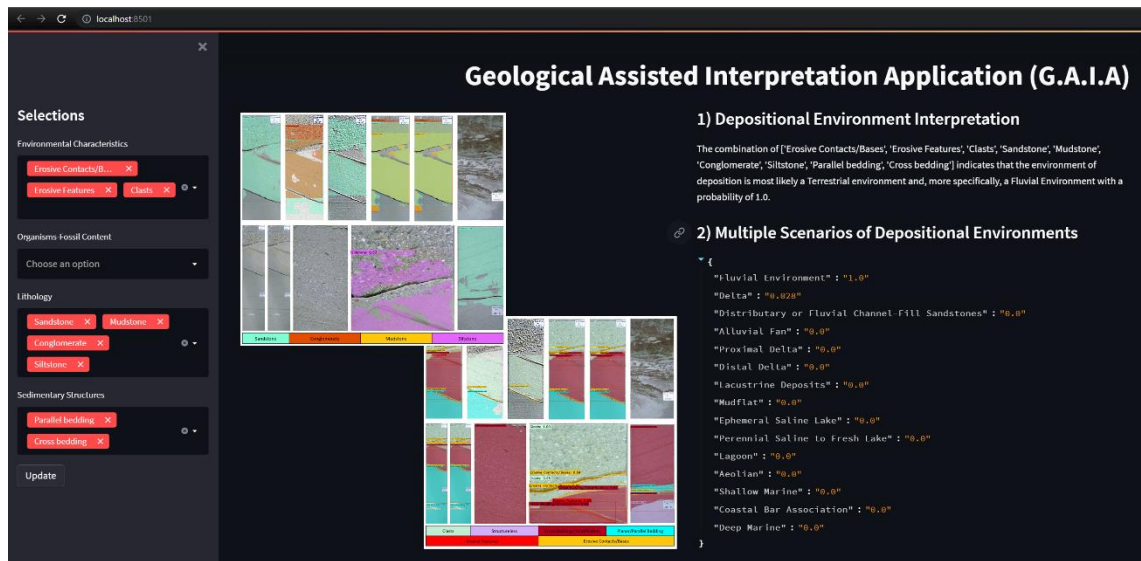


Figure 8-12: Example 6 of the GUI model's predictions based on core images.

The sixth example, Figure 8-12, is a core image with geological features from a fluvial environment, according to the geological literature. This image was segmented in Chapter 7 to obtain the lithology and sedimentary structure labels using Instance Segmentation. These labels are used as input in the user interface, as shown in Figure 8-12 on the left-hand side.

The G.A.I.A. output, shown on the right side of the figure, is multiple predictions of depositional environments, with the top prediction being a Fluvial environment with a probability of 1.0, which is the correct answer. The second top prediction is that the environment of deposition is a Delta environment, with a probability of 0.028, which is not correct when compared to the ground truth.

In both examples five and six, for the same test image, the top 1 prediction of the G.A.I.A. model was the same, a Fluvial environment with a probability of 1. However, looking at the top 2 predictions for both cases, it is obvious that the prediction probability of each is much different, 0.739 versus 0.028. The lower probability in the second case indicates that the model is more certain of its predictions overall. This is due to the higher number of inputs provided by the user in example 6 (Figure 8-12) compared to example 5 (Figure 8-11). This supports a statement made earlier in the thesis about why instance segmentation is necessary and the best method out of the three CV methods described in this thesis to extract geological features from the outcrop images. Better segmentation of the geology would mean more labels to use and incorporate with the G.A.I.A. model.

Overall, the results of this test case demonstrated that the G.A.I.A. model correctly predicts the depositional environment on its top 1 prediction 2/2 times, given two different core examples. In the next section, the G.A.I.A. model will be tested with a sedimentary log.

8.4.3 Test Case 3: Interpret the Depositional Environment from Sedimentary Logs

Finally, the GUI model was tested with six different sections of a sedimentary log. This sedimentary log was created and discussed with expert geologists during a field trip in Spain as part of my Ph.D. program. This log describes the Isona outcrop, showing a geological formation located in the northeastern region of Spain, specifically in the province of Lleida in Catalonia. It is a well-known and studied outcrop that has become famous for its rich fossil record and stratigraphic significance. The entire sedimentary log is shown in Figure 8-13.

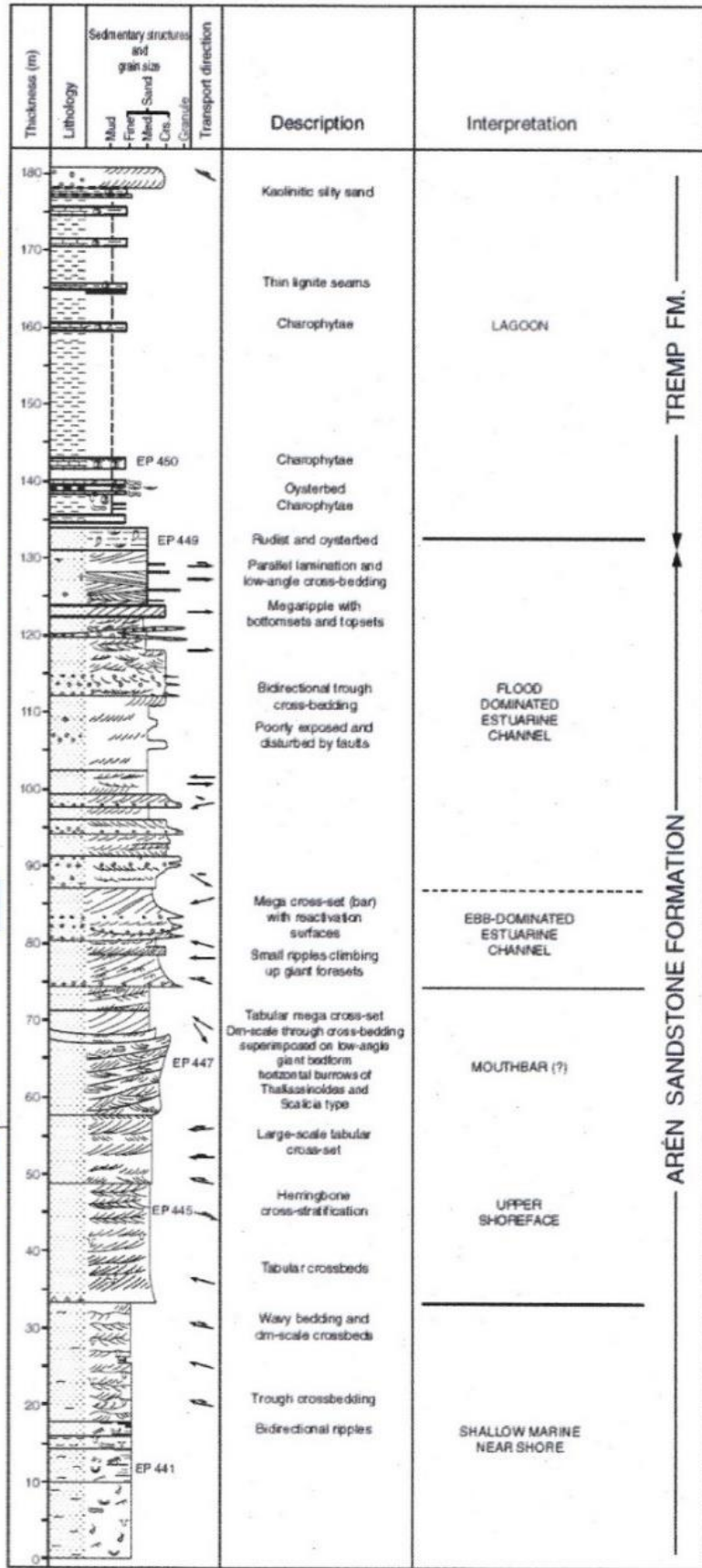
Isona



40



40



TREMP FM.

AREN SANDSTONE FORMATION

Figure 8-13: Sedimentary log of the Isona outcrop.

From Figure 8-13, we can see that the log is split into six different sections by the geologists according to the depositional environment interpretation. For this reason, I took snapshots of each section to examine and interpret individually with the G.A.I.A. model. The details for each of the sections are shown in Table 8-3.

For all the tested examples, the interpretation of the depositional environment is validated by the expert geologist. The GUI predicts a list of possible interpretations, which is then compared to the ground truth to determine the model's robustness, as shown in Table 8-3. The layout of the user interface is the same as before, with all the user's input on the left of the figure and the model's results on the right.

Examples	Data Type (Sedimentary Log)	Ground Truth	Top 1 Prediction	Top 1 Probability	Top 2 Prediction	Top2 Probability
Example 7 (Isona, ES)	Sedimentary Log (0-34m)	Shallow Marine/ Near Shore	Shallow Marine	0.98	Coastal Bar Association	0.384
Example 8 (Isona, ES)	Sedimentary Log (34-57m)	Shallow Marine/ Upper Shoreface	Shallow Marine	1.0	Fluvial	0.985
Example 9 (Isona, ES)	Sedimentary Log (57-75m)	Mouthbar	Fluvial	0.937	Lacustrine Deposits	0.915
Example 10 (Isona, ES)	Sedimentary Log (75-87m)	Ebb-Dominated Estuarine Channel	Shallow Marine	0.999	Estuarine	0.999
Example 11 (Isona, ES)	Sedimentary Log (87-131m)	Flood-Dominated Estuarine Channel	Estuarine	1.0	Coastal Bar Association	0.986
Example 12 (Isona, ES)	Sedimentary Log (131-180m)	Lagoon	Lagoon	1.0	Deep Marine	0.177

Table 8-3: GUI model's predictions based on sedimentary logs.

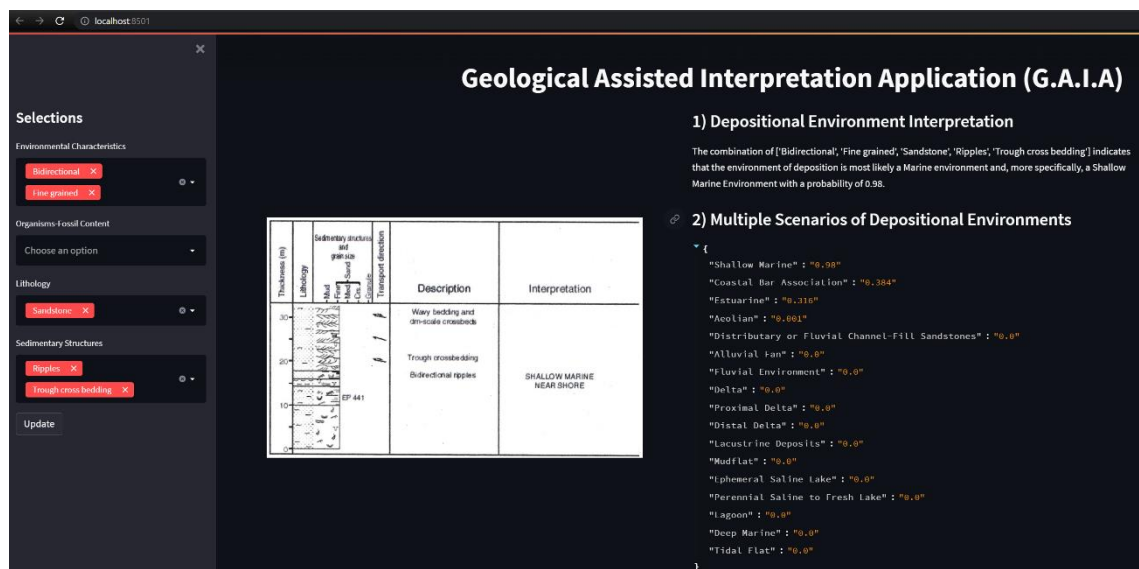


Figure 8-14: Example 7 of the GUI model's predictions based on sedimentary logs.

The seventh example, Figure 8-14, is the bottom of the sedimentary log (0-34m) with geological features from a Shallow Marine/ Near Shore environment. The various lithology and sedimentary structure labels are shown in the snapshot in Figure 8-14.

The G.A.I.A. output is multiple predictions of depositional environments, with the top prediction being a Shallow Marine environment with a probability of 0.98, which is the correct answer. The second top prediction is that the environment of deposition is a Coastal Bar Association, with a probability of 0.384.

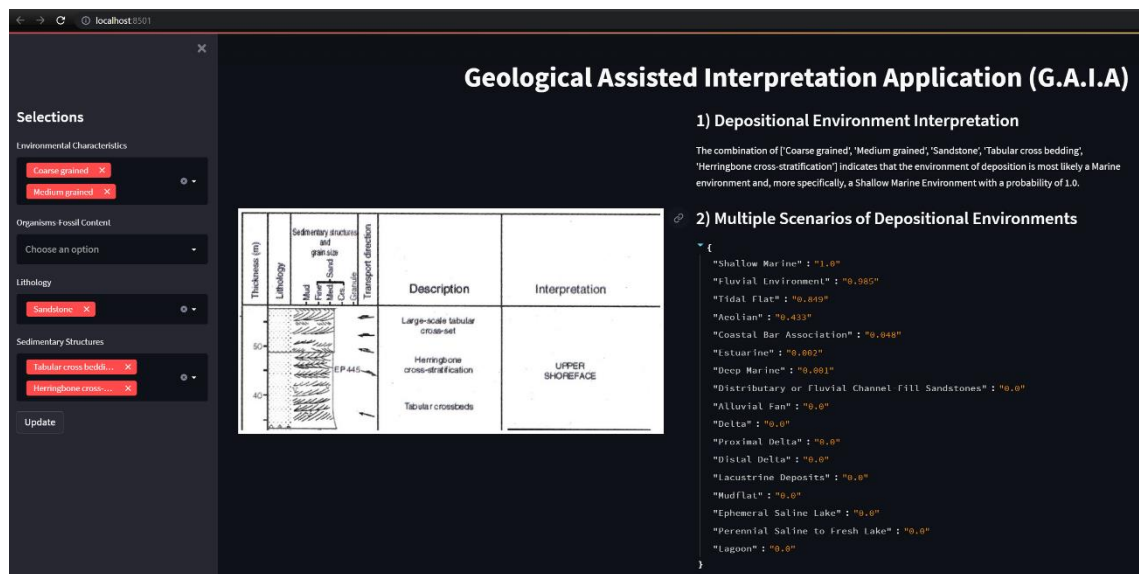


Figure 8-15: Example 8 of the GUI model's predictions based on sedimentary logs.

The eighth example, Figure 8-15, is the 34-57m section of the sedimentary log with geological features from a Shallow Marine/ Upper Shoreface environment. The various lithology and sedimentary structure labels are shown in the snapshot in Figure 8-15.

The G.A.I.A. output is multiple predictions of depositional environments, with the top prediction being a Shallow Marine environment with a probability of 1.0, which is the correct answer. The second top prediction is that the environment of deposition is a Fluvial environment, with a probability of 0.985, which is not correct when compared to the ground truth.

In both examples 7 and 8, the top one prediction of the model is correct. However, the model cannot predict the level of detail the sedimentary log shows in the interpretation, meaning that the model does not have the context of 'Near Shore' and 'Upper Shoreface'

just yet. As more data is to be embedded into the model, as part of future work, and the model improves, it will be able to predict a higher level of complexity.

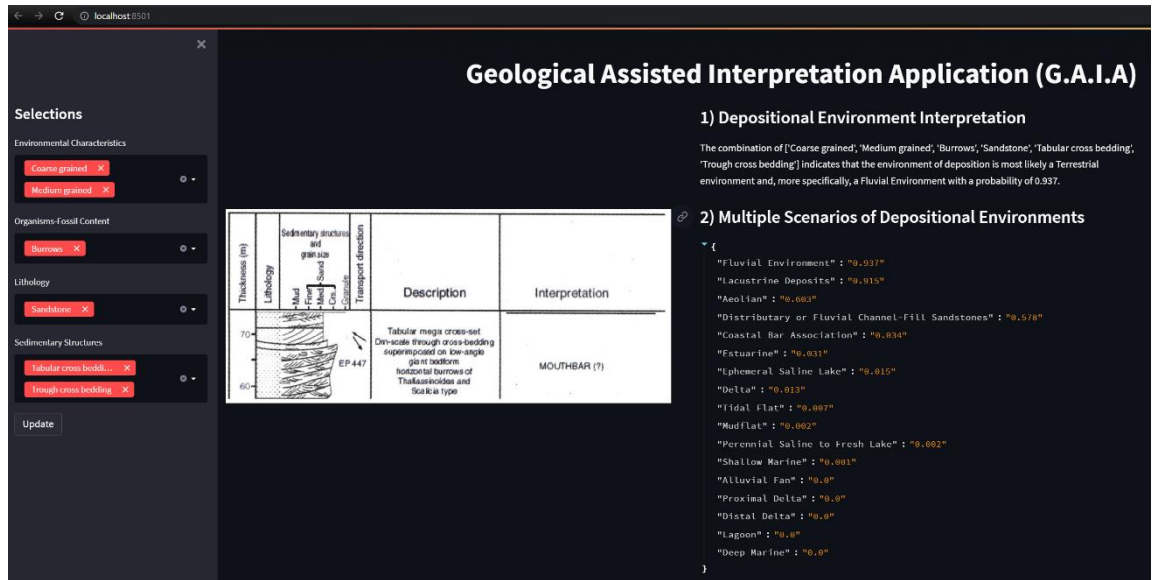


Figure 8-16: Example 9 of the GUI model's predictions based on sedimentary logs.

The ninth example, Figure 8-16, is the 57-75m section of the sedimentary log with geological features from a Mouthbar environment. In geology, a Mouthbar is a type of bar that is formed at the mouth of a river or an estuary, where the river meets the sea. It is a depositional landform that is created by the deposition of sediment carried by the river, which is then distributed and reworked by the waves and currents of the ocean. The various lithology and sedimentary structure labels are shown in the snapshot in Figure 8-16.

The G.A.I.A. output is multiple predictions of depositional environments, with the top prediction being a Fluvial environment with a probability of 0.937. The second top prediction is that of Lacustrine Deposits, with a probability of 0.915. Both predictions are not correct when compared to the ground truth, but both predictions hold some correct elements as a Mouthbar is associated with rivers and also less saline water than can be associated with a lake.

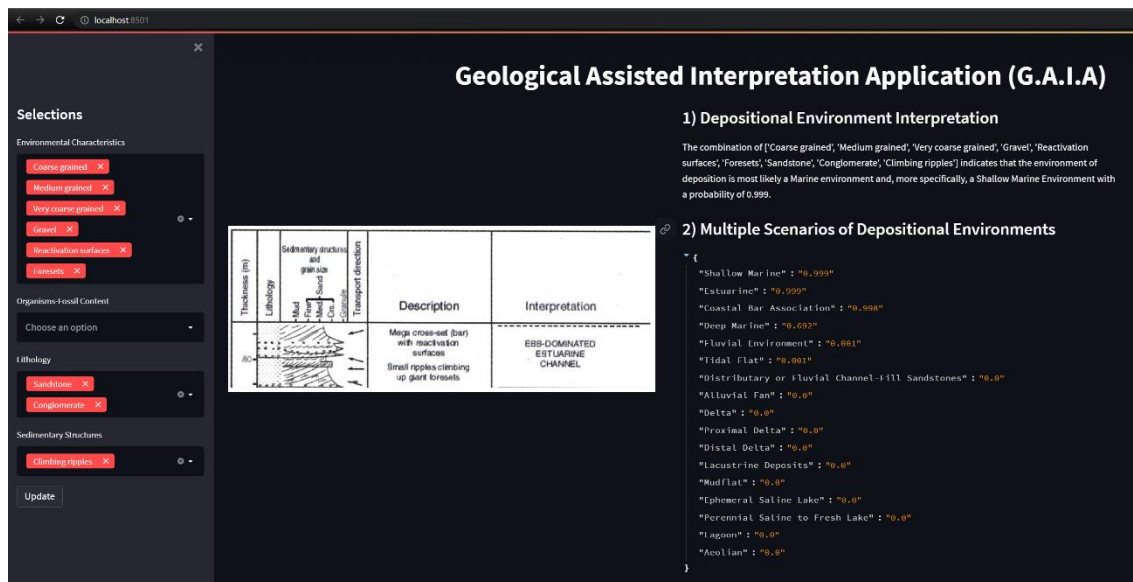


Figure 8-17: Example 10 of the GUI model's predictions based on sedimentary logs.

The tenth example, Figure 8-17, is the 75-87m section of the sedimentary log with geological features from an Ebb-Dominated Estuarine Channel environment. An ebb-dominated estuarine channel is a type of channel that is found in estuaries, which are partially enclosed coastal bodies of water where freshwater from rivers mixes with seawater from the ocean. In ebb-dominated channels, the flow of water is dominated by the outgoing tide, which carries sediment and water out of the estuary and into the ocean. The various lithology and sedimentary structure labels are shown in the snapshot in Figure 8-17.

The G.A.I.A. output is multiple predictions of depositional environments, with the top prediction being a Shallow Marine environment with a probability of 0.999. The second top prediction is that of an Estuarine environment, also with a probability of 0.999. Both predictions are correct, but according to the ground truth, the second top prediction is the most correct one.

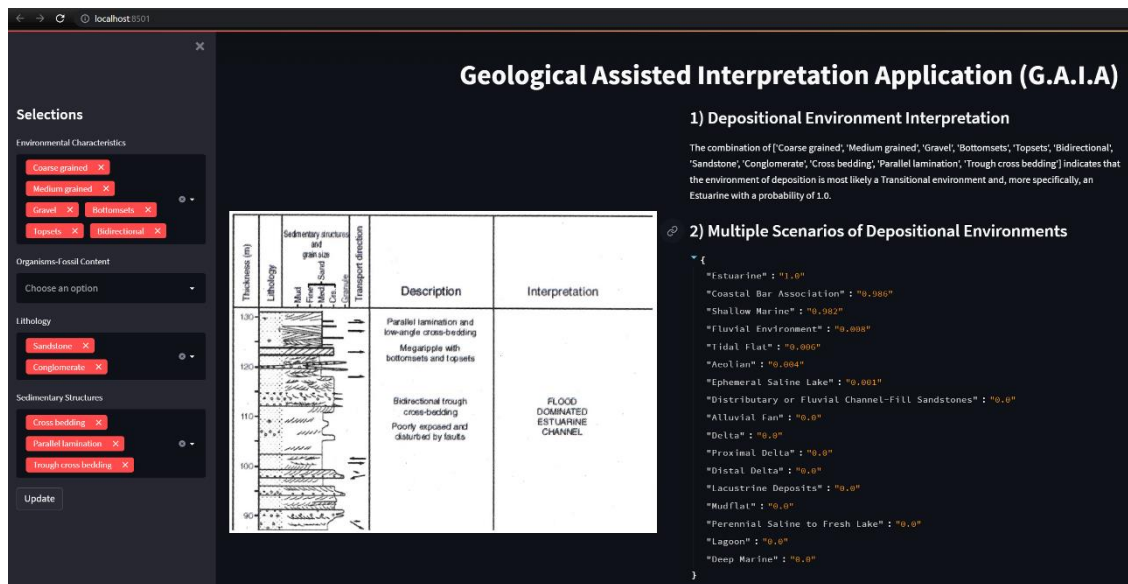


Figure 8-18: Example 11 of the GUI model's predictions based on sedimentary logs.

The eleventh example, Figure 8-18, is the 87-131m section of the sedimentary log with geological features from a Flood-Dominated Estuarine Channel environment. A flood-dominated estuarine channel is a type of channel that is found in estuaries, which are partially enclosed coastal bodies of water where freshwater from rivers mixes with seawater from the ocean. In flood-dominated channels, the flow of water is dominated by the incoming tide, which carries sediment and water into the estuary. Flood-dominated channels are typically wider and shallower than ebb-dominated channels, with gently sloping sides and a U-shaped cross-section. The various lithology and sedimentary structure labels are shown in the snapshot in Figure 8-18.

The G.A.I.A. output is multiple predictions of depositional environments, with the top prediction being an Estuarine environment with a probability of 1.0. The second top prediction is that of a Coastal Bar environment, with a probability of 0.986. According to the ground truth, the top 1 prediction is the correct one.

In both examples 10 and 11, the model correctly predicts the ground truth label. However, it cannot understand the difference between the Ebb-Dominated Estuarine Channel and the Flood-Dominated Estuarine Channel. As discussed in earlier examples, this level of complexity and details will be embedded into the model at a later stage, beyond this Ph.D. project.

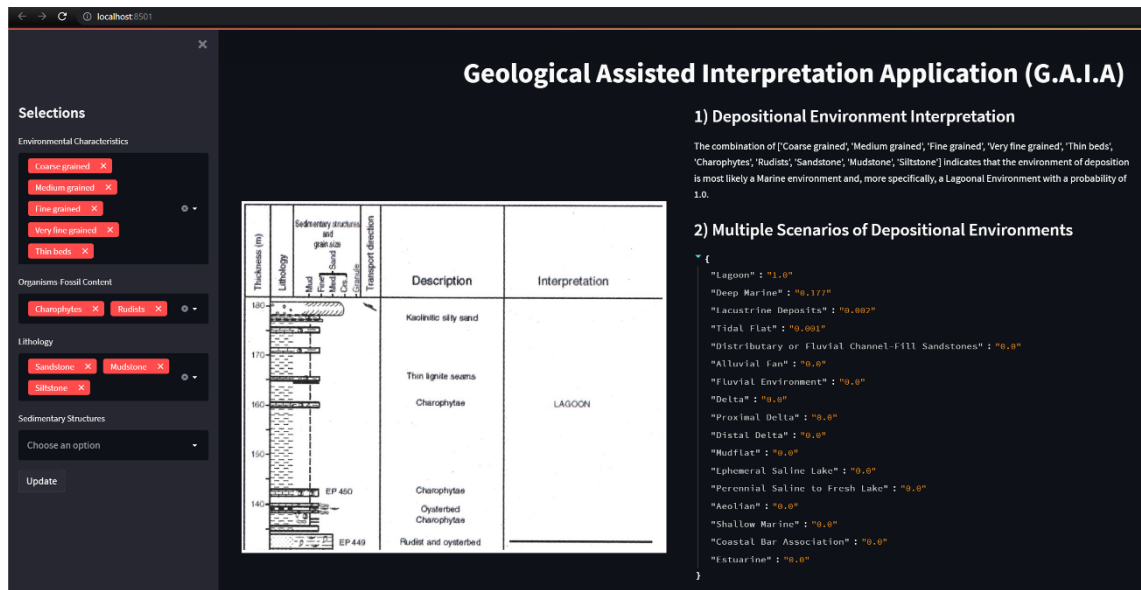


Figure 8-19: Example 12 of the GUI model's predictions based on sedimentary logs.

The twelfth example, Figure 8-19, covering 131-180m, represents the last section of the sedimentary log with geological features from a Lagoon environment. A lagoon is a shallow body of water that is separated from the open ocean by a narrow strip of land, such as a barrier island or a coral reef. Lagoons can be found in a variety of coastal environments, including tropical and subtropical regions, and can vary in size from small ponds to large, expansive bodies of water. The various lithology and sedimentary structure labels are shown in the snapshot in Figure 8-19.

The G.A.I.A. output is multiple predictions of depositional environments, with the top prediction being a Lagoon environment with a probability of 1.0, which is correct. The second top prediction is that of a Deep Marine environment, with a probability of 0.177. According to the ground truth, the second top prediction is wrong as a Lagoon is a sub-category of a Shallow Marine environment; however, the probability of the second top prediction is quite low.

This last test case showed the application of the G.A.I.A. model in its attempt to interpret the depositional environment from a sedimentary log. The model was quite successful in its prediction when compared to the ground truth and showed its ability to make predictions across different data types and depositional environments.

The results obtained from this model, overall, are crucial in capturing interpretational uncertainty within the context of depositional environment interpretations. The

prediction probabilities assigned by the custom Neural Network play a key role in assessing and quantifying this uncertainty.

The prediction probabilities associated with each interpretation generated by the model reflect the level of confidence or certainty attributed to that particular interpretation. A higher probability suggests a stronger indication that the interpretation aligns well with the input data and the learned patterns within the Neural Network. Conversely, a lower probability indicates a higher degree of uncertainty associated with the interpretation.

By ranking the interpretations based on their prediction probabilities in descending order, the model provides a clear representation of the relative certainty or uncertainty of each interpretation. The top-ranked interpretations are associated with higher probabilities, indicating a higher confidence level, while lower-ranked interpretations reflect increased uncertainty.

8.5 Chapter's Conclusions

This chapter concludes all the results in this thesis by demonstrating how the complete AI system developed in this Ph.D. works and how it interprets the geology across different data types. Geological interpretation is a very complicated task, and for a machine learning system to interpret it, the interpretation process was split into three main components, as shown in Figure 8-20.

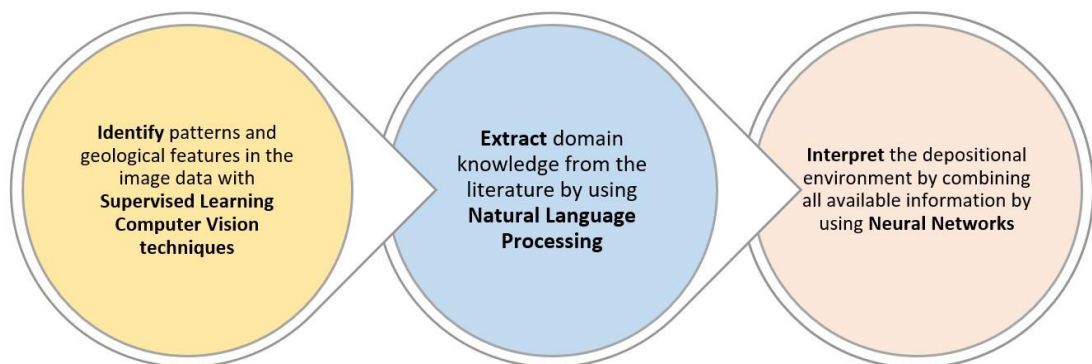


Figure 8-20: AI system steps for interpreting the depositional environment from an outcrop.

The three steps of Figure 8-20 were accomplished by using the five different models described in this thesis.

For the first step, three Computer Vision models were utilised to predict and identify geological patterns based on training on outcrop images and geological sketches.

For the second step, NLP was used to extract the knowledge of feature combinations associated with depositional environments from the literature.

The third last step was to combine all the available information by embedding geological knowledge into a neural network that predicts the depositional environment by identifying features and finding the best fit of the label combinations. The second and third steps were the focus of this chapter.

The Geological Assisted Interpretation Application (G.A.I.A.) is the GUI model developed for this thesis. The model is built by utilising three key elements: a) the labels of the geological features extracted from the outcrop images by using Image Classification, Object Detection, and Instance Segmentation; b) an Excel file containing all the information extracted from the literature, and c) the weights' file resulting from the training of the NN. These three elements are combined in a final script to generate multiple interpretations of the depositional environment. The script is developed using the Streamlit Python library to create a custom Graphical User Interface (GUI).

The GUI is designed to accept user input and offers several characteristics to choose from, including depositional environments, sedimentary structures, types of fossils, and lithologies. Users can select any number and combination of inputs. The GUI displays a sentence describing the top prediction of the depositional environment, followed by a list of all possible predictions with their corresponding probabilities. The model's prediction and the probability for each prediction depend on the combination of user-selected inputs. Each probability is calculated independently for every prediction and ranges from 0 to 1, with up to three decimal places.

Across three different test cases, the model was tested on three different data types, including outcrop and core images, as well as on sections of a sedimentary log. The G.A.I.A. model interpreted a range of depositional environments with very good accuracy. Although the model has only been trained on outcrop images and text data, it

was shown that G.A.I.A.'s applicability and interpretive skills extend beyond the outcrop images to other data types, such as core images and sedimentary logs.

The results of this chapter show the capability of the developed AI system to interpret the geology based on an outcrop image and showcase the system's transferability and proficiency in applying its geological knowledge to understand and interpret geology from diverse data types and sources.

Furthermore, the ranking of interpretations based on probability allows geologists to understand the range of possible depositional environment interpretations while also acknowledging the level of uncertainty associated with each interpretation. It provides valuable insights into the variations and potential alternative explanations for the given set of input data, which is essential in understanding the uncertainty of the subsurface.

The next and final chapter of this thesis will summarize the thesis's findings, highlight the key challenges associated with this Ph.D. project and how they were addressed, and conclude with recommendations for future work and potential improvements.

CHAPTER 9 - SUMMARY, CONCLUSION, AND FUTURE WORK

9.1 Summary & Conclusions

Geological interpretations, due to the inherent complexity and variability of geological systems, are always subject to a degree of uncertainty. Geological data, sourced either from the surface (outcrop data) or the subsurface (core, well logs, etc.), can be ambiguous, allowing for multiple valid interpretations. Different geologists or scientists may analyse the same data and arrive at different conclusions based on their subjective interpretations, expertise, or biases. This subjectivity contributes to the uncertainty in geological interpretations. To mitigate the risks associated with this uncertainty, multiple depositional environment interpretations based on outcrops are valuable as they recognize the potential for multiple plausible interpretations and highlight the range of uncertainty associated with each interpretation.

A thorough examination of multiple interpretations of the outcrop data enables geologists to identify the various possibilities and limitations inherent in each interpretation. For instance, in the case of a meandering versus a braided river, there are several possible interpretations of the sedimentary structures or facies observed in the outcrop. The interpretation of meandering and braided river depositional environments can be uncertain due to overlapping sedimentary structures such as point bars and cross-bedding, the presence of transitional facies, complexities introduced by avulsion deposits, preservation biases, scale dependencies, and heterogeneity within river systems. With a rigorous quantification and characterization of such uncertainties, geologists can make more informed decisions and reduce the risks associated with exploration and production activities.

In Bond et al.'s 2007 study, the researchers examined a synthetic seismic image and found that a wide range of interpretations can arise from a single dataset (Bond, et al., 2007). This variability highlights the inherent uncertainty in seismic interpretation. Out of 412 geoscientists who participated in the study, only 21% were able to accurately interpret the tectonic setting depicted in the image. These findings demonstrate the significant diversity of interpretations that can emerge from a single dataset or image.

Expanding on this work to address the interpretational uncertainty related to subsurface problems, in this thesis, a new AI system has been developed to learn valuable geological

information from surface data (outcrop images), transfer this knowledge to the fragmented data of the subsurface (core data), and finally, link all the extracted information with the heritage geological texts to produce plausible interpretations of the depositional environment based on a single image.

To set the stage for our proposed novel workflow, we exploited the concept of interpretational uncertainty and the significance of accurate and comprehensive depositional environment interpretations, emulating the work of a geologist who relies on outcrop observations to infer subsurface geology. Our workflow integrates methods from various aspects of Artificial Intelligence and Machine Learning methods.

We systematically approached each issue by breaking down the larger problem into simpler tasks, leveraging the most appropriate methods and technologies to achieve optimal results. To identify patterns and geological features in the image data, three Supervised Learning Computer Vision techniques were employed: i) Image Classification, ii) Object Detection, and iii) Instance Segmentation. To extract geological features from the heritage text Natural Language Processing was used, and finally, a custom Neural Network was utilised to assemble the information collected into meaningful sequences, constrain these sequences according to the rules of geology, and generate multiple interpretational scenarios.

Eleven customized image datasets, including exclusively outcrop images, were created specifically for the Computer Vision models. The reason for having eleven datasets is that each dataset had a distinct purpose and was utilised within and between the three Computer Vision methods and their corresponding experiments.

In Chapter 4, a comprehensive methodology is presented for the construction of datasets customized to meet the specific requirements of Computer Vision (CV) models in the field of geology. The workflow outlined in this chapter highlights the importance of each step involved in the dataset-building process and provides a meticulous account of the outcrop datasets employed in this thesis.

The CV models trained on these datasets were aimed at extracting a range of geological features, including, but not limited to, sedimentary structures, lithology, and fossil types from outcrop images or videos. The integration of sketched interpretation data with photographic datasets of geological outcrops (Datasets 4 and 7) has the potential to significantly enhance the accuracy of sedimentary structure classification, even with

smaller volumes of data. By combining sketches of sedimentary structures and fossils with natural outcrop photos, the CNN Image Classification model can more precisely categorize specific geological structures based on the distinctive features present in each image. It is recommended to strike an optimal balance between sketches and outcrop photos in the training data to facilitate the model's learning process. Remarkably, it has been observed that incorporating a proportion of sketch/outcrop images ranging from 40% to 67% leads to a substantial improvement in the model's accuracy compared to training it solely with outcrop images. This blending approach not only enhances the interpretative quality by leveraging the knowledge and simplicity inherent in the sketches but also takes into account the complexity of real-world conditions depicted in the photos.

The sketched templates function as realistic representations that assist image classification models in learning the desired geological patterns while disregarding extraneous features such as colour, surface textures, shadows, and vegetation. By incorporating these sketches into image classification datasets, the robustness and prediction accuracy of the model can be enhanced. This integration directs AI pattern recognition toward relevant features portrayed by the sketches while disregarding any irrelevant or unimportant features present in the images.

Through experiments conducted using the custom datasets created in this thesis, the efficacy of Supervised Computer Vision methods in geological applications was demonstrated. For all three Computer Vision methods used, based on the results of the individual chapters describing the methods, the variability and richness of the datasets used for training and validation have a direct impact on the model's performance. For instance, using images representing cross-bedding structures from different outcrops and locations under multiple lighting conditions forms a more diverse and extensive training set, leading to better predictions and generalization. However, there were three reasons why it was not feasible to combine all subclasses of sedimentary structures, lithology, and fossil types into a single dataset. Firstly, there was a limited data issue, which refers to the scarcity of high-quality photos and annotated outcrop images that had been verified by an expert. As a result, I had to manually collect and gradually assemble the images for my datasets, as the initial phase of my Ph.D. lacked outcrop images suitable for training computer vision models. Secondly, the availability of computational resources was limited throughout the duration of my Ph.D. Thirdly, the findings of Chapter 7 indicated that training the segmentation model separately for sedimentary structures and lithology led to improved prediction accuracy.

- I. Chapter 5 delves into the potential of utilising Image Classification techniques for the purpose of classifying geological structures and fossils (Figure 9-1 left side). The custom image classification model proposed can help geologists distinguish individual sedimentary structures and fossils by classifying outcrop/fossil images according to the predominant sedimentary feature represented in each image. The investigation presented in this chapter elucidates that a blended dataset comprising 2D outcrop images and simplified geological sketches can augment the predictions and learning capabilities of the model, specifically with regard to the identification of various sedimentary structures and fossils. Furthermore, this model can provide a list of other possible predictions ranked based on the probability of their occurrence, showing the confidence of the model for its predictions. Image Classification tends to be successful only on smaller scales (1cm-4m) regarding geological classification tasks, where there is only one feature present in the image reaching a test accuracy of 82%. If the Image Classification model is presented with an entire outcrop displaying multiple features, it will fail because an outcrop cannot be classified or interpreted based only on single observations or features present. Although it provides a single prediction, it compares the confidence of that prediction across all the available classes. This custom image classification model is easy to set up and train and is not computationally expensive due to its customized setup.
- II. Chapter 6 addresses the intricate task of identifying and localizing multiple geological features from 2D images of outcrops, cores, and fossils, utilising the Object Detection model YOLOv6 (Figure 9-1 left side). The YOLOv6-S model, a version of YOLOv6, was exclusively trained and validated on outcrop and fossil images and subsequently tested on previously unseen outcrop and fossil images. While the model reaches a test accuracy of 88% when tested on outcrop images, we observe that it cannot predict lithology either in outcrops or in core images/videos. Although it can assign bounding boxes and labels around lithology types and layers, this does not provide sufficient information to geologists who need to define clear boundaries and distinctions between various lithologies to reach an interpretation of an outcrop. Additionally, the model was challenged to identify sedimentary structures on core data, which represents subsurface fragmentary geological evidence on a smaller scale than outcrops. The outcomes exemplify the YOLOv6-S model's

applicability in exploiting geological expertise from outcrops to predict geology at varying scales accurately, with a test accuracy of 79%. The YOLOv6-S model is able to predict the occurrence of sedimentary structures and fossils across two distinct data sources, outcrop, fossil, and core images/videos. It was found that the YOLOv6-S model has the potential to transfer geological knowledge from outcrops to the smaller core data and generalize well across different geological data types by making accurate predictions.

III. Chapter 7 shows how the delineation of sedimentary structures and lithology types in 2D outcrop images is achieved with the application of the YOLACT, an Instance Segmentation method (Figure 9-1 left side). In addition to the localization and labeling that Object detection offers, the YOLACT model accurately delineates the boundaries of various sedimentary structures and lithology types present in outcrop images by applying colourful, binary masks. To segment the geological features of an outcrop successfully, the selection of the appropriate data, model backbone, and configuration is necessary. The appropriate backbone for YOLACT should be chosen based on the task at hand. If the goal is to achieve better accuracy and mask fit, the YOLACT (ResNet101) model should be chosen, providing test accuracies of 94% and 93.44% for the segmentation of the lithology and sedimentary structures, respectively. On the other hand, if real-time predictions, faster inference, and FPS performance are preferred, the YOLACT (cDarkNet53) model is the best option, which yields test accuracies of 96.65% and 91.25% for the segmentation of the lithology and sedimentary structures, respectively. Establishing a custom adaptation of the YOLACT model indicates that the segmentation model is capable of generalizing very well on its predictions and can transfer its learning from the outcrop images directly to the core images without using any core images in their training, achieving a test accuracy of 88.7% for the sedimentary structures.

Both the Instance Segmentation and Object Detection models are highly adaptable and can perform well on diverse geological datasets, which is a crucial aspect of their practical applicability. When YOLACT attempts to simultaneously predict both lithology and sedimentary structures from a single outcrop image, it tends to misclassify the geological features and generate masks with substantial overlap. In contrast, when the model is trained separately for each task, once to segment lithology and separately to segment sedimentary structures, it correctly identifies

the patterns present in the tested outcrops, avoiding severe mask overlapping and improving the model's results' interpretability and accuracy.

The best way to evaluate an Object Detection and an Instance Segmentation model for a geological task is by obtaining feedback from a human geologist rather than relying solely on the mAP scores and confidence in the final predictions. This is because even if some labels or annotations are incorrect, the model may still predict the wrong label with high accuracy according to the numerical prediction confidence scores. However, in reality, the predicted label may be geologically incorrect. The geologist's evaluation is essential only during the annotation step to ensure the model's practical understanding of the geology. Nonetheless, to ensure the proper functionality and robustness of the models, I conducted evaluations during both the annotation step and the test stage.

The interpretational uncertainty associated with the depositional environment interpretation is influenced by the confidence of the individual predictions generated by the three Computer Vision (CV) models employed. The uncertainty inherent in the predictions of image classification, object detection, and instance segmentation models contributes to the overall uncertainty in the interpretation of the depositional environment. When the individual predictions from the CV models exhibit higher levels of confidence, it tends to reduce the overall uncertainty in the interpretational process. Conversely, if the individual predictions are characterized by lower confidence, it introduces a higher level of uncertainty in the final interpretation. The confidence in the CV models' predictions serves as an important factor in assessing the reliability and accuracy of the results, consequently affecting the degree of uncertainty associated with the interpretation of the depositional environment. By considering the uncertainty propagated from the individual predictions of the CV models, it becomes crucial to account for the reliability and confidence of these models' outputs when utilising them as inputs for the ANN model described in Chapter 8.

In Chapter 8, the elaborate and multi-layered process through which geologists formulate their interpretations and conceptual scenarios to comprehend the heritage geological knowledge of the Earth's surface and subsurface is examined. The chapter introduces a novel custom Neural Network model that leverages domain knowledge extracted in the form of text from geological publications through Natural Language Processing (NLP)

(Figure 9-1 right side) and visual evidence from Computer Vision to form multiple interpretations of the depositional environment.

The implementation of the model offers a flexible approach for users to choose different geological features from a long list compiled from the combined NLP and Computer Vision results using a customized Graphical User Interface (GUI) (Figure 9-1 bottom side). The Neural Network analyses the selected text inputs and generates several scenarios, each with an assigned probability. These probabilities range from 0 to 1 and are indicative of the likelihood of a given interpretation. The final output comprises a list of potential interpretations, ranked in descending order of probability, thereby providing an ordered set of plausible depositional environment interpretations.

The workflow of Figure 9-1 comprises three different parts, starting with the Computer Vision methods on the left-hand side, trained to analyse and segment images of outcrops and automatically learn, identify, and extract features such as rock textures and classify different types of sedimentary structures and lithology types. The Natural Language Processing component, shown on the right-hand side, is used to elicit expert knowledge from the corpus of geological publications. Finally, the custom Neural Network (GUI) component shown at the bottom of the figure utilises the results of both Computer Vision and Natural Language Processing networks to generate several different interpretations to predict the likelihood of an outcrop being formed by various depositional environments.

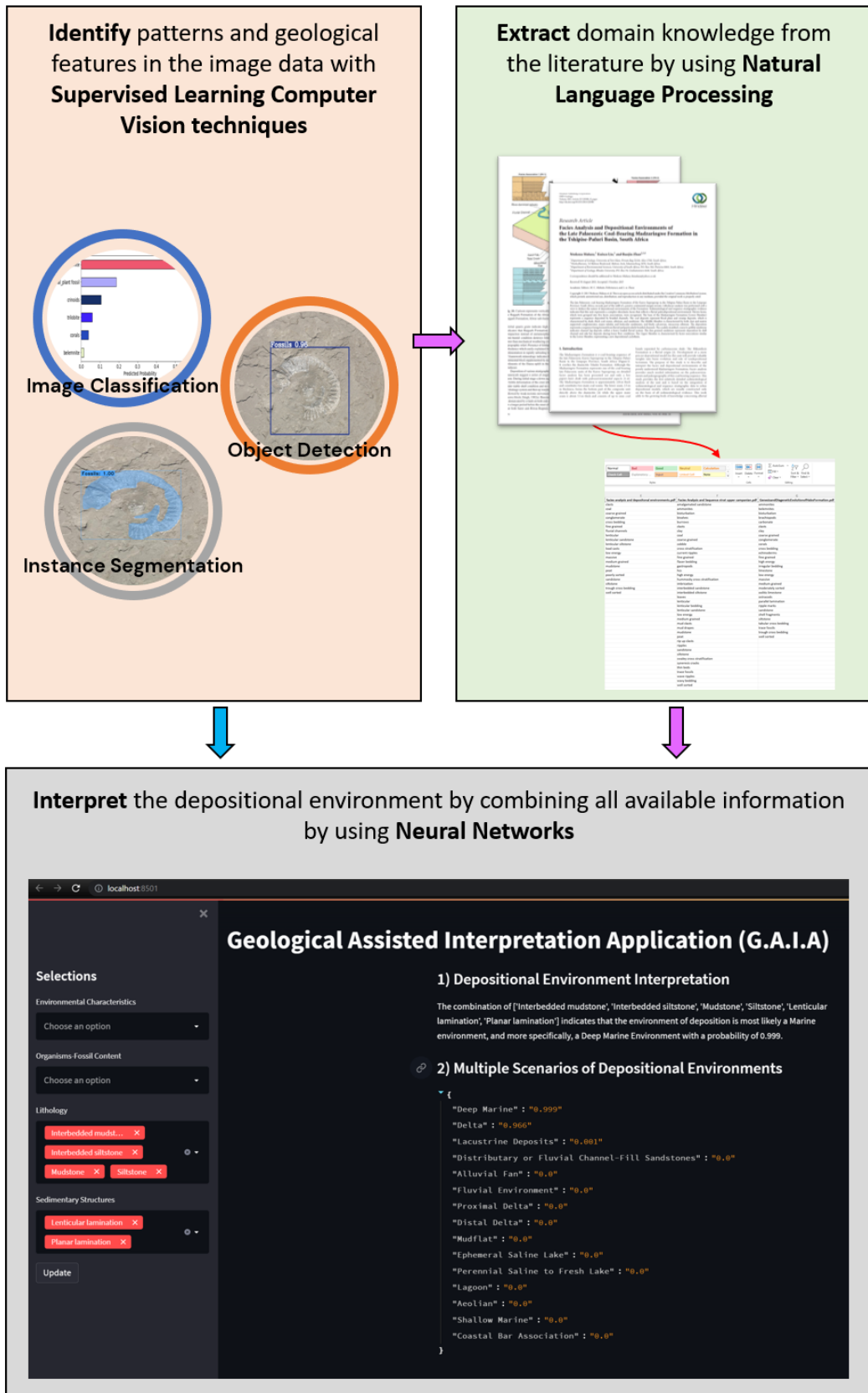


Figure 9-1: Conceptual summary of the overall workflow developed in this thesis.

Multiple depositional environment interpretations were achieved through the use of a custom Graphical User Interface (GUI) model developed for this thesis, named Geological Assisted Interpretation Application (G.A.I.A.). This model pieces together three elements: a) the outputs of the CV methods with b) a simple NLP pipeline to extract in strings of text the knowledge of feature combinations associated with depositional environments from the geological literature, and c) a custom neural network that predicts the depositional environment by identifying features and finding the best fit of the label combinations. The G.A.I.A. model interpreted a range of depositional environments with very good accuracy. Although the model has only been trained on outcrop images and text data, with this experiment, it was shown that G.A.I.A.'s applicability and interpretive skills extend beyond the outcrop images to other data types, such as core images and sedimentary logs.

The prediction probabilities associated with each interpretation generated by the model reflect the level of confidence or certainty attributed to that particular interpretation. A higher probability suggests a stronger indication that the interpretation aligns well with the input data and the learned patterns within the Neural Network. Conversely, a lower probability indicates a higher degree of uncertainty associated with the interpretation.

By ranking the interpretations based on their prediction probabilities in descending order, the model provides a clear representation of the relative certainty or uncertainty of each interpretation. The top-ranked interpretations are associated with higher probabilities (between 0.9-1), indicating a higher confidence level, while lower-ranked interpretations reflect increased uncertainty.

Overall, this thesis introduces a novel methodology for dealing with interpretational uncertainty in geology, accomplished through the creation of an AI system capable of assimilating critical geological insights from surface data, particularly from outcrop images. These insights can then be linked to fragmented subsurface data, such as core data, and ultimately leveraged to interpret the depositional environment. The broad array of interpretations that emerges from this approach captures the interpretational uncertainty inherent in geology and provides geologists with a wider range of viable options and scenarios for further exploration using reservoir modelling and simulations. This thesis makes a valuable contribution to the holistic understanding of the role of AI in geology, particularly in clastic sedimentology and outcrop interpretation, and lays a foundation for further research and development in this field.

9.2 Challenges & Recommendations for Future Work

Certain limitations and challenges were encountered at various stages of this thesis. The most significant of these challenges include:

1. One of the most critical challenges related to all the computer vision models' performance is the data availability and variability, as there is a need for high-resolution images and a highly variable dataset, including training images depicting multiple instances of geological features. Lack of data is one of the biggest and most common challenges when building custom computer vision models. Without a sufficient amount of data, it can be difficult to train a model to accurately identify and classify objects that are complicated, rare, and domain-specific. This challenge was partially addressed with the addition of geological sketches and the use of augmentation in the training datasets.
2. Image quality is another issue related to data. Even when data is available, it may not be of high quality, which can negatively impact the performance of a computer vision model. For example, images may be blurry, have low contrast, or contain artifacts, making it difficult for the model to identify the object of interest correctly. Images selected for my datasets were very carefully selected, ensuring the image quality was meeting the model's standards.
3. Certain domains, such as geology, may present unique challenges that need to be addressed when building custom computer vision models. For example, outcrop images may require more than one model to extract the geological features accurately and to handle complex images or identify subtle differences in texture. Furthermore, the scale of the geological features is indeed a unique challenge for the computer vision models in this thesis. In order to reflect the appropriate scale of the geological features for Object Detection and Instance Segmentation tasks, the models used in Chapters 6 and 7 incorporated a Feature Pyramid Network (FPN) architecture. This architectural choice was made to address the challenge of scale variation within the geological data. By integrating the FPN architecture, the models were able to extract features at multiple scales, enabling them to detect and segment geological features accurately across different size ranges. Additionally, the annotations of the geological features were utilised to implicitly embed the scale information within the models. The inclusion of the FPN

architecture and the utilization of annotations for scale embedding were essential strategies employed in these chapters to ensure that the models were capable of effectively handling the diverse scale variations present in geological data.

4. Labeling and annotation is another challenge, particularly when dealing with large datasets or complex annotations. It is important to ensure that the labeling and annotation process is consistent and accurate to ensure high-quality training data. A great amount of time has been spent labeling the images and data preparation which is a key contribution to the work. The varying geology presents a challenge, as it is sometimes difficult to distinguish between similar-looking rock formations of different sizes and shapes. Furthermore, there is a need for a team of expert geologists to oversee and evaluate the work of the annotators to ensure that accurate information is fed into the computer vision models.
5. The architecture of a computer vision model can greatly impact its performance and predictions. Selecting an appropriate architecture can be challenging and require experimentation with different models and hyperparameters. Chapters 5-7 are a very good example of this statement, as part of the results presented in this thesis dealt with finding a suitable configuration for each model in order to tailor it for our geological problem.
6. Lastly, training the computer vision models and especially the segmentation model, was computationally intensive, which posed a challenge as I was working with limited resources. In some instances, models took as long as a week to train (on a single GPU), acting as a barrier to rapid prototyping and iteration. Depending on the dataset size, the image resolution, the number of labels, and the models' architecture and complexity, the computing power required to train and evaluate the models varies and can get computationally expensive.

There are several options to improve or extend the research discussed in this thesis. Some only require using different testing data sets or trying different backbones for the computer vision models, whereas others require further codebase development. Some potential recommendations for future research and improvements are:

1. The custom image datasets and Supervised Computer Vision methods employed in this thesis could serve as a benchmark for future research in geological applications of Computer Vision. These models could form a set of baseline

models that will be used to compare the performance of other algorithms. The custom datasets could be further enhanced by adding more sedimentological features or adding more variations of the existing ones to enrich the variability and enhance the data quality. As the results in Chapter 7 indicate, if the segmentation model is trained with a larger and more diverse dataset, it can provide significantly improved results for the segmentation of different geological data types while enhancing its generalisability.

2. Another idea is to make use of the described computer vision models and workflows to extract the geological features from photogrammetry data instead of only 2D images. Acquiring and incorporating photogrammetry data into these workflows might lead to a significant increase in the models' accuracy of the predictions as it will be easier to account for the scale of the geological features explicitly due to the spatial features photogrammetry data offer.
3. The study conducted in Chapter 7 demonstrated that the custom-trained YOLACT model has the potential to provide valuable insights when applied to real-time video data. Building upon the initial results, a proposed project involves adapting the instance segmentation model into a drone by integrating it with both the software and hardware of the drone. This proposed drone-based tool could enable geologists to rapidly scan outcrops and obtain quick estimations of the sedimentological features present in the outcrops. Such a tool could significantly enhance the efficiency and effectiveness of the data acquisition process, providing geologists with critical information to inform their research and analysis. For instance, the drone using the proposed YOLACT (cDarkNet53) can scan remote areas inaccessible to geologists and provide imagery with the segmenting geology, enhancing the geologist's fieldwork results and data collection. This model, once refined, could be used as a valuable interpretation and teaching tool in the field for outcrop interpretation on the fly.
4. An apparent extension to this work, related to the above recommendation, would be to test the developed AI system on actual field data, including outcrops or cores, along with a team of subsurface experts and sedimentologists. This would allow a more detailed fine-tuning of all the models used and allow testing on a wider range of data. Applying the methods to real field data would also allow for

better verification of the results and conclusions made in this thesis and highlight the true value of such a model.

References

- Ali, A. & Sharma, S., 2017. *Content based image retrieval using feature extraction with machine learning*. Madurai, India, IEEE, pp. 1048-1053.
- Allen, J., 2014. *Sedimentary structures, their character and physical basis, volume 2*. eBook ISBN: 9780080869445 ed. s.l.:Elsevier Science \& Technology.
- Almklov, P. G., Hepsø & Vidar, 2011. Between and beyond data How analogue field experience informs the interpretation of remote data sources in petroleum reservoir geology. *Social Studies of Science*, 41(4), pp. 539-561.
- Alpaydin, E., 2010. *Introduction to Machine Learning*. 2 ed. London, England: MIT Press.
- Anders, K. et al., 2016. 3D Geological Outcrop Characterization: Automatic Detection of 3D Planes (AZIMUTH and DIP) Using LiDAR Point Clouds. *ISPRS Annals of Photogrammetry Remote Sensing and Spatial Information Sciences*, Volume III-5, pp. 105-112.
- Baheti, P., 2021. *Train Test Validation Split: How To & Best Practices [2023]*. [Online] Available at: <https://www.v7labs.com/blog/train-validation-test-set#h1> [Accessed November 2021].
- Bengio, Y., Courville, A. & Vincent, P., 2013. Representation Learning: A Review and New Perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8).
- Bird, S., Klein, E. & Loper, E., 2009. *Natural language processing with python*. Sebastopol, CA: O'Reilly Media.
- Birgenheier, L. P. et al., 2020. Climate impact on fluvial-lake system evolution, Eocene Green River Formation, Uinta Basin, Utah, USA. *Geological Society of America Bulletin*, pp. 562-587.
- Bochkovskiy, A., Wang, C.-Y. & Liao, H.-Y. M., 2020. YOLOv4: Optimal Speed and Accuracy of Object Detection. *CoRR*, Volume abs/2004.10934.

- Boggs, J. S., 2013. *Principles of sedimentology and stratigraphy: Pearson new international edition*. 5 ed. London, England: Pearson Education.
- Boggs, S., 2006. *Principles of Sedimentology and Stratigraphy*. 4 ed. Upper Saddle River: Pearson Education Inc..
- Bolya, D., Zhou, C., Xiao, F. & Lee, Y. J., 2019. *YOLACT: Real-time Instance Segmentation*. s.l., 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 9157-9166.
- Bond, C. E., Gibbs, A. D., Shipton, Z. K. & Jones, S. a. o., 2007. What do you think this is? ``Conceptual uncertainty" in geoscience interpretation. *GSA today*, 17(11), p. 4.
- Canny, J., 1986. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6), pp. 679-698.
- Chaitanya, K., 2020. *Understanding NLP Pipeline: An introduction to phases of NLP pipeline*. [Online]
Available at: <https://medium.com/analytics-vidhya/understanding-nlp-pipeline-9af8cba78a56>
[Accessed September 2021].
- Chowdhary, K., 2020. *Fundamentals of Artificial Intelligence*. New Delhi: Springer, New Delhi.
- Coleman, J. M., 1981. *Deltas: Processes of Deposition and Models for Exploration*. s.l.:Burgess.
- Cordts, M. et al., 2016. *The Cityscapes Dataset for Semantic Urban Scene Understanding*. Las Vegas, NV, USA, IEEE, pp. 3213-3223.
- Dalrymple, R. W., Boyd, R. & Zaitlin, B. A., 1994. History of Research, Types and Internal Organisation of Incised-Valley Systems: Introduction to the Volume. In: *Incised-Valley Systems. Origin and Sedimentary Sequences*. s.l.:SEPM Society for Sedimentary Geology, pp. 3-10.
- Davies, R., 2011. Depositional environments, lithostratigraphy and biostratigraphy: concepts and pitfalls. *Geological Society*, 364(1), pp. 17-34.

Deng, J. et al., 2009. *ImageNet: A large-scale hierarchical image database*. Miami, FL, USA, IEEE, pp. 248-255.

Deng, L., 2012. The MNIST Database of Handwritten Digit Images for Machine Learning Research Best of the Web. *Signal Processing Magazine, IEEE*, Volume 29, pp. 141-142.

Deveugle, P. E. K. et al., 2011. Characterization of stratigraphic architecture and its impact on fluid flow in a fluvial-dominated deltaic reservoir analog: Upper Cretaceous Ferron Sandstone Member, Utah. *American Association of Petroleum Geologists Bulletin*, 95(5), pp. 693-727.

Dubey, S. R., 2022. A Decade Survey of Content Based Image Retrieval Using Deep Learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(5), pp. 2687-2704.

Dunlop, H., 2006. *Automatic Rock Detection and Classification in Natural Scenes.*, s.l.: Carnegie Mellon University.

Everingham, M. et al., 2010. The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2), pp. 303-338.

Francis, R., McIsaac, K., Thompson, D. & Osinski, G., 2014. *Autonomous Rock Outcrop Segmentation as a Tool for Science and Exploration Tasks in Surface Operations*. Pasadena, CA, American Institute of Aeronautics and Astronautics.

Geiß, M. et al., 2023. Automatic bounding box annotation with small training datasets for industrial manufacturing. *Micromachines (Basel)*, 14(2).

Géron, A., 2019. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. 2nd ed. s.l.:O'Reilly Media, Inc..

Ghiasi, G., Lin, T.-Y. & Le, Q. V., 2019. *NAS-FPN: Learning scalable feature pyramid architecture for object detection*. Long Beach, CA, USA, IEEE.

Goodfellow, I., Bengio, Y. & Courville, A., 2016. *Deep Learning*. London, England: The MIT Press.

Haralick, R. M. & Shapiro, L. G., 1992. *Computer and Robot Vision*. Upper Saddle River, NJ: Pearson.

He, K., Gkioxari, G., Dollár, P. & Girshick, R., 2017. *Mask R-CNN*. s.l., s.n.

He, K., Gkioxari, G., Dollár, P. & Girshick, R., 2017. *Mask R-CNN*. Venice, Italy, IEEE, pp. 2980-2988.

He, K., Zhang, X., Ren, S. & Sun, J., 2016. *Deep Residual Learning for Image Recognition*. Las Vegas, NV, USA, IEEE, pp. 770-778.

Hosna, A. et al., 2022. Transfer learning: a friendly introduction. *Journal of Big Data*, 9(1), pp. 1-19.

Jurafsky, D. & Martin, J. H., 2023. *Speech and Language Processing*, s.l.: s.n.

Karpatne, A. et al., 2019. Machine Learning for the Geosciences: Challenges and Opportunities. *IEEE Transactions on Knowledge and Data Engineering*, 31(8), p. 1544–1554.

Keras, 2015. *Keras: Deep Learning for humans*. [Online]
Available at: <https://keras.io/>
[Accessed September 2021].

Keymakr, I., 2021. *Polygon Image Annotation for Computer Vision*. [Online]
Available at: <https://keymakr.com/blog/polygon-annotation-for-computer-vision/>
[Accessed July 2021].

Kingma, D. P. & Ba, J., 2014. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations*.

Kirillov, A. et al., 2019. *Panoptic Segmentation*. Long Beach, CA, USA, s.n., pp. 9396-9405.

Kornblith, S., Shlens, J. & Le, Q. V., 2018. Do Better ImageNet Models Transfer Better?. *CoRR*.

- Kozyrkov, C., 2018. *The AI hierarchy of needs*. [Online]
Available at: <https://cloud.google.com/blog/products/gcp/the-ai-hierarchy-of-needs>
[Accessed December 2019].
- Krizhevsky, A., Nair, V. & Hinton, G., 2009. *The CIFAR-10 and CIFAR-100 datasets*,
Toronto, CA: s.n.
- Krizhevsky, A., Sutskever, I. & Hinton, G. E., 2012. *ImageNet Classification with Deep Convolutional Neural Networks*. Lake Tahoe, NV, USA, Curran Associates, Inc..
- Krizhevsky, A., Sutskever, I. & Hinton, G. E., 2017. ImageNet classification with deep convolutional neural. *Communications of the Association for Computing Machinery (ACM)*, 60(6), pp. 84-90.
- Kwok, C. Y. T. et al., 2018. *Deep learning approach for rock outcrops identification*. Xian, IEEE.
- Lazebnik, S., Schmid, C. & Ponce, J., 2006. *Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories*. New York, NY, USA, IEEE, pp. 2169-2178.
- LeCun, Y., Bengio, Y. & Hinton, G., 2015. Deep Learning. *Nature*, 521(7553), pp. 436-444.
- LeCun, Y., Huang, F. & Bottou, L., 2004. *Learning methods for generic object recognition with invariance to pose and lighting*. Washington, DC, USA, IEEE, pp. 97-104 .
- Li, C. et al., 2022. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. *arxiv.org*, pp. 1-17.
- Li, C. et al., 2018. Robust Flow-Guided Neural Prediction for Sketch-Based Freeform Surface Modeling. *ACM Transactions on Graphics*, Volume 37.
- Liddy, E. D. & Liddy, J. H., 2001. *An NLP Approach for Improving Access to Statistical Information*, Syracuse: School of Information Studies - Faculty Scholarship.
- Lin, T.-Y. et al., 2017. *Feature Pyramid Networks for Object Detection*. Honolulu, HI, USA, IEEE, pp. 936-944.

- Lin, T.-Y. et al., 2017. *Focal Loss for Dense Object Detection*. Venice, Italy, IEEE, pp. 2999-3007.
- Lin, T.-Y. et al., 2015. *Microsoft COCO: Common Objects in Context*. s.l.:arXiv.
- Li, Q. & Dehler, S. A., 2019. Seismic data quality control and interpolation using principal component analysis. *International Journal of Geosciences*, Volume 10, pp. 950-966.
- Li, X. et al., 2018. H-DenseUNet: Hybrid Densely Connected UNet for Liver and Tumor Segmentation From CT Volumes. *IEEE Transactions on Medical Imaging*, 37(12), pp. 2663-2674.
- Long, J., Shelhamer, E. & Darrell, T., 2015. *Fully convolutional networks for semantic segmentation*. Boston, MA, USA, IEEE, pp. 3431-3440.
- Lowndes, A., 2015. Deep Learning with GPU Technology for Image and Feature Recognition.
- Luo, L., Xiong, Y., Liu, Y. & Sun, X., 2019. *Adaptive Gradient Methods with Dynamic Bound of Learning Rate*. s.l., s.n.
- Malik, O. A., Puasa, I. & Lai, D. T. C., 2022. Segmentation for Multi-Rock Types on Digital Outcrop Photographs Using Deep Learning Techniques. *Sensors*, Volume 22.
- Manning, C. D., Raghavan, P. & Schütze, H., 2008. Text Processing. In: *Introduction to Information Retrieval*. s.l.:Cambridge University Press, p. Chapter 2.
- Miall, A. D., 2015. The interpretation of sedimentary environments: twenty years on. *Canadian Journal of Earth Sciences*, 52(10), pp. 903-928.
- Mishra, U., 2021. *Binary and Multiclass Classification in Machine Learning*. [Online] Available at: <https://www.analyticssteps.com/blogs/binary-and-multiclass-classification-machine-learning> [Accessed September 2021].
- Nathanail, A., Demyanov, V., Arnold, D. & Gardiner, A., 2021. *The Importance of Blending Different Data Types to Train Machine Learning Classifiers for Sedimentary*

Structure Detection. Amsterdam, NL, European Association of Geoscientists & Engineers, pp. 1-5.

Neubeck, A. & Van Gool, L., 2006. Efficient Non-Maximum Suppression. *Proceedings of International Conference on Pattern Recognition*, Volume 3, pp. 850-855.

Nichols, G., 2009. *Sedimentology and Stratigraphy*. London: Blackwell Science Ltd..

Nikhil, b., 2017. *Medium.com*. [Online]

Available at: <https://becominghuman.ai/image-data-pre-processing-for-neural-networks-498289068258>

[Accessed September 2021].

Otsu, N., 1979. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1), pp. 62-66.

Pascual, A. D., 2019. *Autonomous and Real Time Rock Image Classification using Convolutional Neural Networks*, s.l.: s.n.

Polanyi, M., 1966. *The Tacit Dimension*. London: Routledge & Kegan Paul.

Pont-Tuset, J. et al., 2017. Multiscale Combinatorial Grouping for Image Segmentation and Object Proposal Generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(1), pp. 128-140.

Prothero, D. R. & Schwab, F., 2013. *Sedimentary geology: An introduction to sedimentary rocks and stratigraphy*. s.l.:W. H. Freeman and Company.

Provost, F. & Fawcett, T., 2013. *Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking*. s.l.:O'Reilly Media, Inc..

PyTorch, 2016. *PyTorch*. [Online]

Available at: <https://pytorch.org/>

[Accessed May 2021].

Randle, C. H., Bond, C. E., Lark, R. M. & Monaghan, A. A., 2019. Uncertainty in geological interpretations: Effectiveness of expert elicitations. *Geosphere*, 15(1), pp. 108--118.

- Reading, H. G., 1996. *Sedimentary Environments: Processes, Facies, and Stratigraphy*. 3 ed. Cambridge: Blackwell Science.
- Redmon, J., Divvala, S., Girshick, R. & Farhadi, A., 2016. *You only look once: Unified, real-time object detection*. Las Vegas, NV, USA, IEEE, pp. 779-788.
- Redmon, J. & Farhadi, A., 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.
- Rezatofighi, H. et al., 2019. *Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression*. Long Beach, CA, USA, IEEE, pp. 658-666.
- Ridzuan, F. & Zainon, W. M. N., 2019. A Review on Data Cleansing Methods for Big Data. *Procedia Computer Science*, Volume 161, pp. 731-738.
- Rosebrock, A., 2016. *Intersection over Union (IoU) for object detection*. [Online] Available at: <https://pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/> [Accessed July 2021].
- Russakovsky, O. et al., 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3), pp. 211-252.
- Russell, B. C., Torralba, A., Murphy, K. P. & Freeman, W. T., 2008. LabelMe: A Database and Web-Based Tool for Image Annotation. *International Journal of Computer Vision (IJCV)*, 77(1-3), pp. 157-173.
- Saif, J. A. M., Hammad, M. H. & Alqubati, I. A. A., 2016. Gradient Based Image Edge Detection. *IACSIT International Journal of Engineering and Technology*, 8(3), pp. 153-156.
- Samuel, T., 2021. *Roboflow: Converting Annotations for Object Detection*. [Online] Available at: <https://medium.com/analytics-vidhya/converting-annotations-for-object-detection-using-roboflow-5d0760bd5871> [Accessed January 2022].
- Sathya, R. & Abraham, A., 2013. Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification. *International Journal of Advanced Research in Artificial Intelligence(IJARAI)*, 2(2).

- Schwartz, T. M. & Tracy, R. J., 2020. Interpretation of geological outcrops. *Encyclopedia of Geology*, Volume 2, pp. 529-537.
- Serra, O., 1984. Fundamentals of Well Log Interpretation: The acquisition of logging data. In: *Developments in Petroleum Science*, 15A. s.l.:Elsevier, p. 160.
- Shorten, C. & Khoshgoftaar, T. M., 2019. A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, 6(1), p. 60.
- Siddiqui, N. et al., 2017. Shallow-marine Sandstone Reservoirs, Depositional Environments, Stratigraphic Characteristics and Facies Model: A Review. *Journal of Applied Sciences*, Volume 17, pp. 212-237.
- Simonyan, K. & Zisserman, A., 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv e-prints*, p. arXiv:1409.1556.
- Smith, A. R., 1978. Color gamut transform pairs. *Computer Graphics. Computer graphics.*, 12(3), pp. 12-19.
- Snell, J., Swersky, K. & Zemel, R., 2017. Prototypical Networks for Few-shot Learning. *CoRR*, Volume abs/1703.05175.
- Sobel, I. & Feldman, G., 1973. An Isotropic 3×3 image gradient operator. *Pattern Classification and Scene Analysis*, pp. 271-272.
- Sun, Z., Bebis, G. & Miller, R., 2004. Object detection using feature subset selection. *Pattern Recognition*, 37(11), pp. 2165-2176.
- Szeliski, R., 2010. Algorithms and Applications. In: *Computer Vision*. London: Springer, p. 812.
- Ting, K. M., 2017. Confusion Matrix. In: *Encyclopedia of Machine Learning and Data Mining*. Boston, MA: Springer.
- Tucker, M. E., 2001. *Sedimentary petrology: an introduction to the origin of sedimentary rocks*. s.l.:John Wiley & Sons.

- Vasuki, Y., Holden, E.-J., Kovesi, P. & Micklethwaite, S., 2017. An interactive image segmentation method for lithological boundary detection: A rapid mapping tool for geologists. *Computers & Geosciences*, Volume 100, pp. 27-40.
- Wang, C.-Y., Bochkovskiy, A. & Liao, H.-Y. M., 2021. *Scaled-YOLOv4: Scaling Cross Stage Partial Network*. Nashville, TN, USA, s.n., pp. 13024-13033.
- Wang, C.-Y. et al., 2019. CSPNet: A New Backbone that can Enhance Learning Capability of CNN. *eprint arXiv:1911.11929*, pp. 1-14.
- Wellmann, F. & Caumon, G., 2018. 3-D Structural geological models: Concepts, methods, and uncertainties. In: *Advances in Geophysics*. s.l.:Elsevier, pp. 1-121.
- Wicander, R. & Monroe, J. S., 2012. *Historical Geology: Evolution of Earth and Life Through Time*. 7 ed. Florence, AL: Cengage Learning.
- Williamson, K., 2002. *Research methods for students, academics and professionals*. 2 ed. s.l.:Woodhead Publishing.
- Young, T., Hazarika, D., Poria, S. & Cambria, E., 2018. Recent Trends in Deep Learning Based Natural Language Processing [Review Article]. *IEEE Computational Intelligence Magazine*, 13(3), pp. 55-75.
- Zhang, L., 2021. Hand-drawn sketch recognition with a double-channel convolutional neural network. *EURASIP Journal on Advances in Signal Processing*, 2021(1), p. 73.
- Zhang, S. et al., 2017. Single-Shot Refinement Neural Network for Object Detection. *CoRR*, Volume abs/1711.06897.
- Zhang, Z. & Sabuncu, M., 2018. Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels. *CoRR*, Volume abs/1805.07836.
- Zhao, Z.-Q., Zheng, P., Xu, S.-t. & Wu, X., 2018. Object Detection with Deep Learning: A Review. *CoRR*, Volume abs/1807.05511.
- Zheng, Y. et al., 2021. Sketch-specific data augmentation for freehand sketch recognition. *Neurocomputing*, Volume 456, pp. 528-539.