Classifying barchan outlines into morphological classes using convolutional neural networks: a proof of concept

Barend Jacobus van der Merwe

Submitted in partial fulfillment for the degree
Doctor of Philosophy (Geography)

In the Faculty of Natural and Agricultural Sciences

Supervisor: Prof Serena M. Coetzee

Department of Geography, Geoinformatics
and Meteorology
University of Pretoria
2022

ABSTRACT

Remotely sensed imagery is a valuable source of data for studying barchan morphology. However, manual methods of data extraction constrain both the spatial and temporal resolution of studies because they are time consuming to carry out. Therefore, to effectively use the increasing availability of remotely sensed imagery, novel methods need to be developed that can extract the desired data from imagery automatically. Convolutional Neural Networks (CNNs) have shown promise in identifying landforms from imagery, but its suitability for barchan research remains untested. Since CNNs are strongly influenced by the texture of the image, it can be questioned whether the classification is based on the image's texture (which can vary due to solar angles and atmospheric disturbances) or the geometry of the landform. Additionally, deviations in shape and other morphometric properties can manifest as subtle alterations to the barchan's geometry. This poses a challenge for CNNs which have difficulty in distinguishing between similarly shaped landforms. Using a small sample of dunes from the Kunene region in Namibia, it is shown that CNNs can: distinguish between different morphologic classes of barchans in the absence of image texture with accuracies exceeding 80%, and distinguish between similarly shaped landfroms. When used along with methods of barchan outline extraction, a suitably trained CNN can automatically extract barchan morphologic data from remotely sensed imagery. This can increase both the temporal and spatial resolution of barchan research.

Research Outputs

Journal Articles

During the course of this study two peer reviewed articles have been published along with five oral conference presentations at both a national and an international level. A list of these is provided below. Additionally, the portions of the thesis which they relate to are indicated. For both articles and presentations, the material covered is not directly copied from the thesis but involves a reworking of the data to fit the topic of the article or presentation.

1. **Van der Merwe, B.J.**, 2021: The relationship between barchan size and barchan morphology: a case study from Northern Namibia, *South African Geographical Journal*, 103, 119 – 138..

   This article covered the following aspects of the study:

   (a) It provides a description of the wind regime of the study site (§5.2).

   (b) It formalises the concept and method of bilateral asymmetry. This was needed to create suitable labels for training data and to develop suitable tasks (§1.5).

   (c) It describes the morphometric parameters of the barchans within the study site (§6.2). This is needed in order to prepare the data for use in a CNN (§1.5).

   (d) It adds supporting evidence on the influence of size on barchan asymmetry (§2.4.5).

2. **Van der Merwe, B.**, Pillay, N., and Coetzee, S., 2022: An application of CNN to classify barchan dunes into asymmetry classes, *Aeolian Research*, 56, 100801.

   This article covered the following aspects of the study:

   (a) It provides a summary of the different measurements carried out on barchans. Such a summary has not yet been published in the barchan literature. It also forms part of the motivation for the development of the research tasks (§1.5).

   (b) It motivates for the use of the different CNNs that are used in this study (§5.10).

   (c) It provides details on how the models are constructed, trained, and evaluated (§5.9).

(d) It provides and contextualizes the results of four out of the six tasks: barchan shape, side of elongation, magnitude of elongation, and bilateral asymmetry (§6.4.1, §6.5.1, §6.8.1, and §6.9.1).

Conference Presentations

1. Using Geometric Morphometrics to Quantify Barchan Shape from Satellite Imagery, Oral Presentation, South African Association of Geomorphologists (SAAG) 2015 Conference, 19-20 September, Sani Mountain Lodge, Lesotho.
   This presentation introduces the idea of using geometric morphometrics (GM) as a method for describing barchan shape. GM is a central part of calculating the bilateral asymmetry used in this study (§5.7.6).

2. Barchan shape as an aid to process explanation: An historical overview, Society of South African Geographers (SSAG) centennial conference 2016, 25-28 September, Stellenbosch, South Africa.
   This presentation reviews the different approaches to describing the shape of barchan dunes. It formed the foundation upon which the more detailed investigation of barchan shape description (§2.2.2) is built.

3. A bibliometric study of the role of artificial intelligence in geomorphology, Oral Presentation, South African Association of Geomorphologists (SAAG) 2017 Conference, 25-28 July, University of Swaziland, Kwaluseni, Swaziland.
   In this presentation, an overview of the use of artificial intelligence within geomorphology was presented. While it was not specifically focussed on AI, it formed a starting point upon from which §4.2.1 was developed.

4. The performance of different CNN architectures on barchan asymmetry classification, Webinar Presentation, International Association of Geomorphologists, IAG Webinar Africa, 2 March 2021.
   The results of the CNN on classifying barchan asymmetry was presented for four out of the six tasks (Chapter 6).

5. Classifying barchan shape and asymmetry using convolutional neural networks, Oral Presentation, Society of South African Geographers & Southern African Association of Geomorphologists 2021 Joint Biennial Conference (6-8 September).
   The results of the CNN on classifying barchan asymmetry was presented for four out of the six tasks (Chapter 6).

Declaration

I declare that the dissertation/thesis, which I hereby submit for the degree **PhD Geography** at the University of Pretoria, is my own work and has not previously been submitted by me for a degree at this or any other tertiary institution.

Ethics Statement

The author, whose name appears on the title page of this dissertation/thesis, has obtained, for the research described in this work, the applicable research ethics approval (**Ethics Clearance Number: 180000081**).

The author declares that he has observed the ethical standards required in terms of the University of Pretoria's Code of Ethics for Researchers and the policy guidelines for responsible research.

Signature:

Student Name: Barend van der Merwe

Date: 08/2022

Acknowledgements

The University of Pretoria is gratefully acknowledged for financial assistance, in the form of tuition rebates, and for providing the resources needed for this project, both computational and educational. Without their contributions, this research would not have been possible.

I want to extend my heartfelt gratitude to my current supervisor Prof. Serena Coetzee for her invaluable assistance in navigating the world of publishing and academic writing. I am indeed fortunate to have benefited from her kindness, eye for detail, and wealth of experience. My gratitude is also extended to my former supervisor Prof. Paul Sumner who, although circumstances prevented him from seeing the project through to the end, was instrumental in placing me on my current path. I also want to thank one of my co-authors Prof. Nelishia Pillay who provided invaluable advice during the construction, training and evaluation of the CNN models. To my colleagues at the Department of Geography, Geoinformatics, and Meteorology I extend a heartfelt thanks for their support and enlightening discussions over the years that it took to complete this research. Without the assistance of these academics this project could not have reached its current form.

A word of appreciation also needs to be extended to the reviewers of the two articles that stemmed from this project. Their comments and recommendations on the individual articles made meaningful contributions to the final form of this thesis.

My love and appreciation is also extended to my family whose support and continued interest in my work, even though they did not always understand it, greatly helped to keep me motivated to see the project through to completion. Lastly, and especially, I want to thank my person and my best friend Suzie Viljoen. Without her love, support, and encouragement I would not have been able to maintain the mental and psychological fortitude to complete this work.

Contents

List of Figures

List of Tables

List of Acronymns

ACC Accuracy.

ADAM Adaptive Moment Estimation.

AI Artificial Intelligence.

ANN Artificial Neural Networks.

CNN Convolutional Neural Network.

DM Digital Model.

DP Drift Potential.

ELU Exponential Linear Unit.

FN False Negative.

FP False Positive.

GM Geometric Morphometrics.

GPA Generalized Procrustes Analysis.

KDE Kernel Density Estimation.

MCC Matthew's Correlation Coefficient.

MSE Mean Square Error.

NCAR National Center for Atmospheric Research.

NCEP National Centers for Environmental Modeling.

PPV Positive Predictive Value.

R-CNN Regional Convolutional Neural Network.

RDD Resultant Drift Direction.

RDP Resultant Drift Potential.

ReLU Rectified Linear Unit.

RS Remotely Sensed.

TAR Transverse Aeolian Ridges.

TN True Negative.

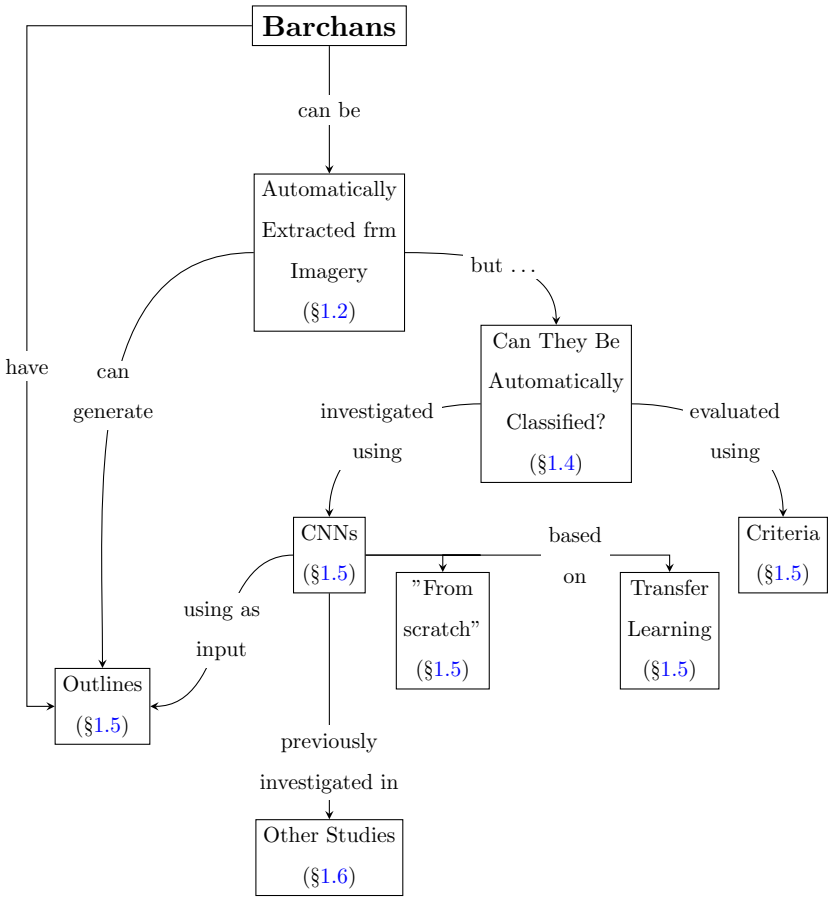
TP True Positive.

TPR True Positive Rate.

Chapter 1

Introduction

1.1 Chapter Overview



1.2 Problem Background

The shape of a landform contains information on the functionality, composition and genesis of that landform (Minár and Evans, 2008; MacMillan and Shary, 2009). For barchans, studies have found potential links between the shape, or asymmetry, of the dune and environmental variables associated with the genesis, composition, and processes operating on the dune (Table

Category	Variable	Example Study
Processes	Winds	Bourke (2010)
	Landform transition	Tsoar and Parteli (2016)
	Sediment	Zhang *et al.* (2014)
Genesis	Sediment supply	Parteli *et al.* (2014)
	Collisions	Lv *et al.* (2016)
	Topography	Parteli *et al.* (2014)
	Wind regime	Bourke (2010)
Composition	Soil conditions	Zhang *et al.* (2018)

Table 1.1: Variables that have been linked to either the shape and asymmetry of a barchan, or to changes in the shape or asymmetry. A more detailed overview is provided in Chapter 2.

1.1). This suggests that it is possible, in principle, to infer these variables based on the analysis of a barchan's shape. This is, admittedly, an oversimplification of reality but this approach has been used on Mars where environmental data is limited (Hayward *et al.*, 2007; Gardin *et al.*, 2012; Bourke and Goudie, 2009). The interpretative value of barchan shape and asymmetry is expanded upon in Chapter 2. Aside from interpreting the environment on planetary surfaces, more data on barchan morphology can also contribute to an improved understanding of barchan processes (Bourke and Goudie, 2009). For example, the different barchan width and height relationships that are present within the literature (Hesp and Hastings, 1998) suggest that a more useful generalization can be achieved when using a more diverse data set.

It is, therefore, evident that increasing the sample size can lead to improvements in understanding. This stems from both improved generalizations based on statistical methods (Richards, 1996) and from improvements in the predictive availability of geomorphological models (Hjort and Marmion, 2008). However, there are challenges associated with increasing the sample size. Data on shape can either be collected through field visits (e.g. Elbelrhiti, 2012), or by using remotely sensed imagery (see Hugenholtz *et al.*, 2012). Field work provides valuable data but, unfortunately, is very time consuming and expensive (Azzaoui *et al.*, 2020; Parteli *et al.*, 2014; Moosavi *et al.*, 2014). Extracting data from remotely sensed data can overcome some of these constraints. Continuous technological advancement has led to an increase in the amount of available remotely sensed data (Maggiori *et al.*, 2017; de Lima and Marfurt, 2020; Boulila *et al.*, 2021; Kattenborn *et al.*, 2021). This makes it possible to collect data on barchans faster and with less cost than conventional field visits. However, even though data can be more rapidly collected than with field surveys, the extraction of morphological data from remotely sensed imagery is still dominated by manual methods (Maxwell *et al.*, 2020). Therefore, it is still a time consuming task (Gafurov and Yermolayev, 2020; Aydda *et al.*, 2020; Azzaoui *et al.*, 2020).

This is particularly problematic when the scales of observations are large (Parteli *et al.*, 2014) or when the frequency of observations is high (Azzaoui *et al.*, 2019).

This has led to the development of automated data extraction methods. These methods attempt to combine the interpretive abilities of a human analyst with the processing volume of automated systems (i.e. computer software). Earlier studies have been successful in automatically extracting landform data for a variety of different landforms. These studies used techniques such as object based image analysis (e.g. Vaz *et al.*, 2015) and machine learning algorithms, such as Convolutional Neural Networks (CNNs) (e.g Carbonneau *et al.*, 2020), to rapidly and accurately extract feature information from imagery. In the case of barchans, however, little work has been done. Azzaoui *et al.* (2020) used AlexNet (a type of CNN) to detect barchans and barchan collisions in high resolution imagery. Rubanenko *et al.* (2021) combined image segmentation techniques with a mask regional CNN to detect and outline barchan dunes on Mars. Other studies have also placed emphasis on detecting barchans and generating outlines (Azzaoui *et al.*, 2019; Dakir *et al.*, 2016) although these studies did not make use of neural networks. Although progress has been successful on detecting and isolating barchan dunes in imagery, it remains to be seen whether CNNs are capable of extracting morphological information about the barchans.

1.3 Problem Statement

It has been demonstrated that CNNs can be used to delineate barchans from remote sensing imagery, however, it remains to be shown that they are capable of extracting usable morphometric data from individual dunes. This is a necessary step in order to effectively use remote sensing big data to drive data intensive research on barchan dynamics.

1.4 Research Aim

The aim of this research is to determine if barchan shape and asymmetry information can be automatically extracted by a CNN from a simplified image of a barchan and, if so, how the performance of the network can, potentially, be enhanced. It is important to emphasise that this project is a proof of concept. It should not be viewed as complete model that can extract data from any image of a barchan. To achieve this would require a substantial investment of time and resources. However, that being said, it is argued that before large programs should be implemented, small–scale trials should be conducted in order to assess the feasibility of the project. It is this need that the present study seeks to fulfill.

The type of shape data to be extracted must be usable in conventional studies on barchan morphology in order to ensure its relevance to the large aeolian geomorphology community. Barchans were selected for several reasons:

1. The majority of studies emphasise identifying the location of features (e.g. Palafox *et al.*, 2017; Huang *et al.*, 2018; Robson *et al.*, 2020), or detecting the sub-categories of a landform (Gao *et al.*, 2021), leading to a relative deficit in research focusing on classifying landform attributes such as shape.

2. Previous studies (Dakir *et al.*, 2016; Azzaoui *et al.*, 2019) have already shown promise in extracting the outlines of barchans from imagery.

3. Barchans are relatively simple dunes (Sauermann *et al.*, 2000; Qian *et al.*, 2021) with a similar overall planform, but considerable amounts of variations in within this form. Since high levels of similarity can pose a challenge for CNNs (Cheng *et al.*, 2018; Li and Hsu, 2020), the classification of barchans provides a suitable benchmark against which the performance of CNNs can be assessed.

4. Barchan asymmetry serves a proxy for several important environmental variables: local wind regime (Lv *et al.*, 2016; Zhang *et al.*, 2018), geomorphologically effective winds (Bourke, 2010), local sediment dynamics (Bourke and Goudie, 2009; Parteli *et al.*, 2014) planetary soil conditions (Zhang *et al.*, 2018), the presence of dune collisions (Wang and Anderson, 2018; Tsoar and Parteli, 2016), and inclined topography (Bourke, 2010). Although asymmetry should not be the sole source of information, it is nonetheless useful in cases where other data may be lacking.

5. Barchans are considered one of the most studied types of dunes (Lv *et al.*, 2016; Tsoar and Parteli, 2016; Livingstone *et al.*, 2007) and a considerable amount of literature is already in place regarding the causes of asymmetry.

6. The widespread distribution of barchans on Earth (Goudie, 2020; Breed and Grow, 1979) increases the likelihood of obtaining good quality imagery using Google Earth ™.

7. Due to the migratory nature of barchans they are considered a natural hazard (Hamdan *et al.*, 2016; Aydda *et al.*, 2020) that pose a risk to residential areas (Moosavi *et al.*, 2014) and irrigation schemes (Abdu, 1976). Although Bourke (2010) found that asymmetry does not impact barchan migration, an image of dune calving, where new barchans form at the horns of mature barchans, provided by Worman *et al.* (2013) clearly shows calving associated with an asymmetrical barchan.

Although there are many types of neural networks, CNNs have been identified as a useful tool in studying remote sensing imagery (Shakya *et al.*, 2021; Bhuiyan *et al.*, 2020). CNN models serve as benchmarks for image classification tasks due to their efficiency and success

rate (Kalita and Biswas, 2019; de Lima and Marfurt, 2020). These models recognise objects by identifying the spatial features that best describe a feature (such as edges, corners, and textures) with a target feature (Kattenborn *et al.*, 2021). This makes it possible for CNNs to identify features based on their shape (Kubilius *et al.*, 2016). There is some uncertainty, however, about whether they can relate distant features of an image to each other or whether it is restricted to local considerations only (Baker *et al.*, 2018, 2020). Nevertheless, CNNs are useful tools in geomorphology for identifying features within the landscape (Gao *et al.*, 2021).

1.5 Objectives

The following objectives are needed in order to evaluate the suitability of CNNs to classify barchan asymmetry:

1. Collate data on the type of morphometric data that is typically used within the literature.
2. Develop tasks that cover a wide range of barchan morphological investigations with an emphasis on those that rely on barchan shape and asymmetry.
3. Identify the various hyperparameters that are known to significantly influence the performance of CNNs.
4. Select a suitable site that meets the following criteria: it must have high resolution imagery so that small barchans can be clearly delineated and it must contain isolated barchans.
5. Prepare the imagery obtained from the site in order to remove the potential textural bias and to meet the input requirements of the CNNs.
6. Develop and train several different CNN architectures that encompass the "from scratch" and transfer learning approaches. The former consists of variations in model architecture using the hyperparameters identified in objective 3.
7. Evaluate the performance of each model against a defined set of criteria.
8. Interpret the results and make recommendations regarding the use of CNNs in barchan research.

1.6 Related Work

A similar comparative study was carried out by de Lima and Marfurt (2020) who also compared CNNs trained "from scratch" with those that were pretrained on a different dataset. They found that pretrained CNNs performed better than CNNs that were trained by the user. The present

study differs from this study in key aspects. First, rather than subsetting an "established" network to create shallower networks (as was done by de Lima and Marfurt (2020)) completely customized shallow networks are created that do not, necessarily, resemble any established network. Second, the focus in the present study is shifted away from modifications to the optimizer, which was kept the same for all "from scratch" networks, to the modification of hyperparameters. The difference is substantial. By altering the optimizer, the strategy that is followed to reduce the loss function of the network, the primary means through which the network learns, is modified. The types of features learned by the network is, for the most part[1], kept unaltered. Modifying the hyperparameters alters the capacity of the model to learn features that may aid in successful classification. Therefore, it is a shift away from the learning process itself, towards generating the features used during the learning process. A third difference is the application. The work of de Lima and Marfurt (2020) is aimed at scene classification. Conceptually, this is similar to the goal in the present study. The difference lies in the scenes that are intended to be classified. In the earlier study, the scenes were substantially different from each other (e.g. airplanes vs forest). In the present study, all of the images belong to the same class, i.e. barchans. The difference, therefore, is in distinguishing between differently shaped barchans rather than distinguishing a barchan from another unrelated feature class.

Although not directly related to the present work, Cheng *et al.* (2018) compared the classification performance of custom regulizer to both transfer learning and CNN with fine tuning. This work difference from the present work since they modified the learning metric itself. This was achieved by creating a metric regularization which resulted in a substantial improvement of the network without requiring the architecture to be altered (Cheng *et al.*, 2018). In the present study, no customised regularizations are developed and the architectures are the variables that are modified.

1.7 Outline of Document

The structure of the document is represented in figure 1.1.

Chapter 2 Barchans:

This chapter provides background information on barchan dunes. It covers aspects such as the distribution, origin, general morphology, and sediment dynamics of barchan dunes. It also presents the different methods that have been used to describe barchan morphology (Objective 1), the different causes of asymmetry (Objective 2), and a section discussing the automatic extraction of barchans from images.

[1]Because of the stochastic nature of barchans it cannot be said that the capacity is identical in all cases.

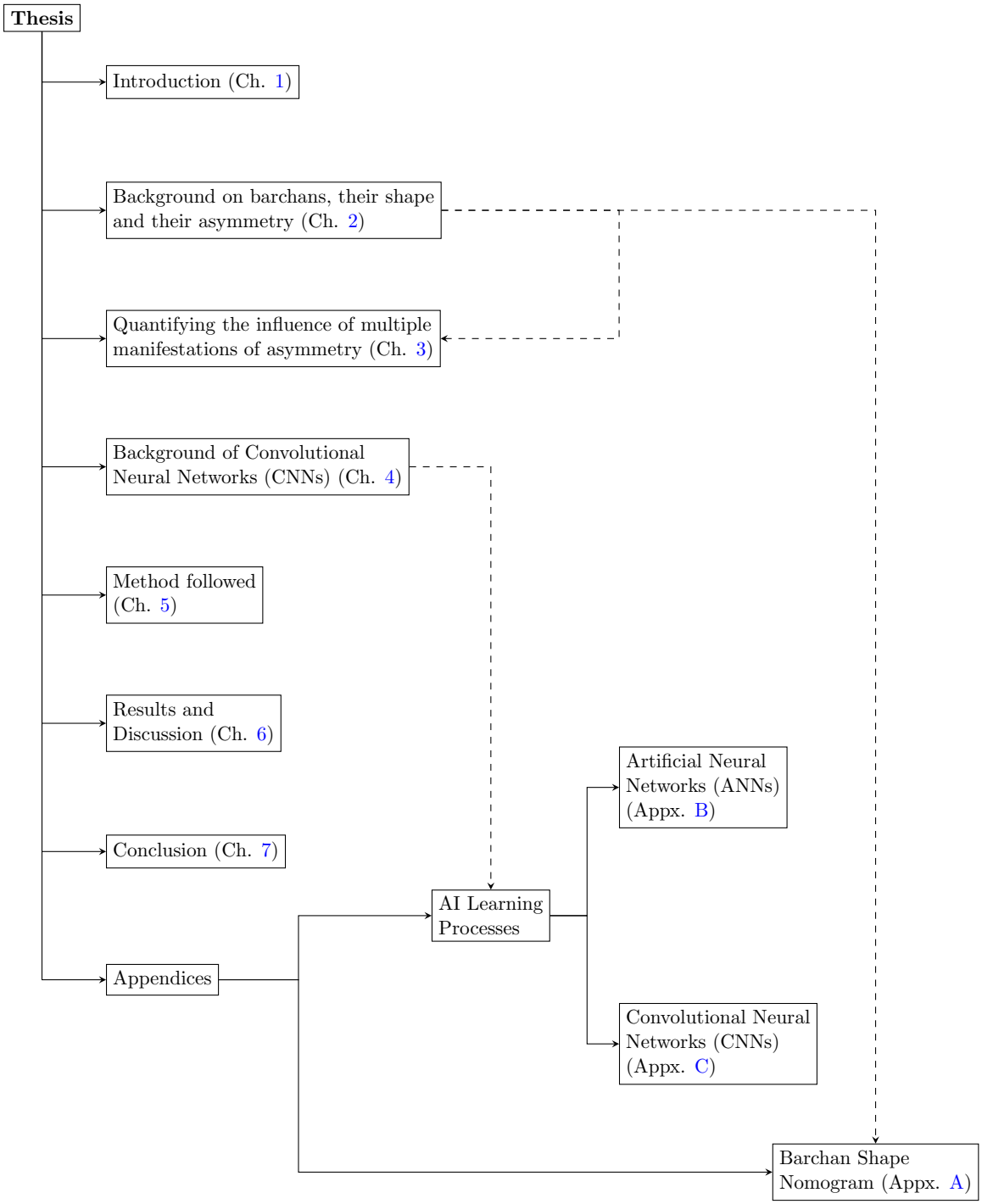


Figure 1.1: An outline of the structure of the thesis. The solid arrows indicate the main parts of the thesis while the dashed arrows indicate conceptual relationships between different chapters.

Chapter 3 Bilateral Asymmetry:

Since no existing method is available to evaluate the combined influence of several different types of asymmetry (e.g. elongation combined with lateral extension) there is a need to develop a method that can carry out this form of analysis. The conceptual basis, and mathematical treatment of this process, termed bilateral asymmetry, are discussed in this chapter. It forms the basis on which the one of the tasks (Objective 2) is based.

Chapter 4 Convolutional Neural Networks:

Provides general background on Convolutional Neural Networks. The topics covered include the characteristics of a CNN that differentiate it from other types of neural network, the data flow within a CNN, and how a CNN improves its performance through training. The different hyperparameters that play a role in the perfromance of a CNN are also discussed (Objective 3).

Chapter 5 Method:

This chapter describes the site that was selected as the source of the imagery (Objective 4) in terms of its location, general topography, and wind regime at the time of image acquisition. The methods used to prepare the imagery is then discussed (Objective 5). The chapter concludes with a section describing how the architectures of the different models, the training parameters, and the evaluation criteria that is used to assess model performance (Objective 6).

Chapter 6 Results and Discussion:

Here, the results of the model evaluation are presented (Objective 7) along with their respective interpretations (Objective 8). For ease of interpretation, this chapter combines the results and discussion sections. Each task identified in Objective 2 is discussed separately with their own "results" and "discussion" subsections. The chapter concludes with a holistic overview that focuses on common themes related to CNNs that were identified during the study.

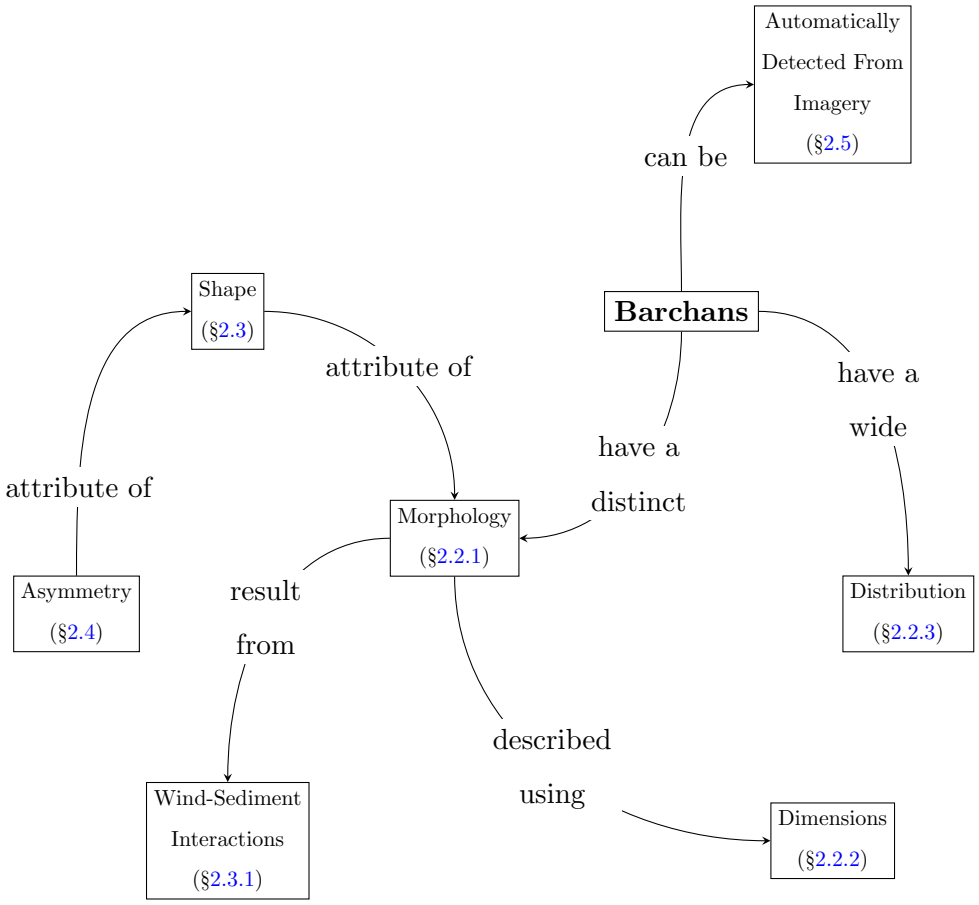
Chapter 7 Conclusion:

The concluding chapter of the thesis provides a synopsis of the research aim, the method followed, and the results obtained. The significance of the work is then contextualised within the barchan research community and recommendations are made regarding future research efforts. This chapter also includes a section on the new contributions made through this research.

Chapter 2

Barchans

2.1 Overview



This chapter provides a background on barchan dunes as landforms. Their distribution, general morphology, and emergence from an initial pile of sand is briefly discussed. In spite of, or potentially because of, their relatively simple shape, a number of different strategies have been used to describe their morphology (i.e. shape). A brief overview of these is provided. The

chapter then proceeds to discuss barchan shape and asymmetry in more detail, with a focus on the $\frac{a}{c}$ ratio, and the main causes of asymmetry and the effect they have on barchan shape.

2.2 Introduction

2.2.1 Morphology

Barchans[1] are a type of dune (Parteli *et al.*, 2014; Zhang *et al.*, 2018; Bourke and Goudie, 2009; Qian *et al.*, 2021; Tsoar and Parteli, 2016; Worman *et al.*, 2013) (Figure 2.1). Some key properties of barchans can be extracted from this classification. First, barchans are simple dunes which, according to the classification scheme by Tsoar (2001), means that individual barchans are spatially separated from other barchans. This view of barchans is widely present within the literature (Sauermann *et al.*, 2000; Qian *et al.*, 2021; Barnes, 2001; Bourke and Goudie, 2009). Second, barchans originate due to surface or aerodynamic roughness (in principle the latter would result from the former). In this classification, it is used to describe dunes that are unvegetated and, essentially, are self-accumulating Tsoar (2001). Barchans are also classified as transverse dunes (Bishop *et al.*, 2002) which are dunes that are formed roughly perpendicular to the dominant winds within a region and are mobile (Bourke and Goudie, 2009; Bishop *et al.*, 2002; Worman *et al.*, 2013) meaning that they migrate. This migration has important implications for asymmetry (§2.4). Therefore, it is to be expected that there exists a strong correlation between the environmental factors and the shape of the barchans, but also some form of feedback between the barchan and itself since the barchan is self-accumulating.

The main morphologic components of a barchan dune are the stoss slope, the left and right horns, the slip face, and the brink line (Figure 2.2). The stoss slope captures sediment from upwind sources (Worman *et al.*, 2013). Erosion occurs along the stoss slope through wind action (Lv *et al.*, 2016; Elbelrhiti, 2012; Zhang *et al.*, 2014) which transports sediment to the crest/brink of the dune and the horns. The brink line represents the beginning of the slip face and introduces a sharp flow separation in the airflow across the dune (Herrmann *et al.*, 2005; Zhang *et al.*, 2014). This flow separation results in the localised accumulation of sediment (Elbelrhiti, 2012; Wiggs, 2013) which, when the angle of repose is exceeded, leads to gravitational collapse and avalanching of sediment down the slip face (Lv *et al.*, 2016; Pelletier *et al.*, 2015; Zhang *et al.*, 2014). This sediment is then incorporated into the main body of the dune as the barchan migrates (Tsoar, 2001). Sediment on the horns is derived largely from the periphery of the upstream region (Alvarez and Franklin, 2018; Durán *et al.*, 2010) and can be significantly coarser than the rest

[1]The term "barchan" is believed to be of Russian or Turkic origin (Lewin, 2015). Other terms encountered in the literature include: medanos (Von Tschudi, 1847), burkhan (McMahon, 1906), barkhan (Lessar, 1885), barchane (Cornish, 1897), barcan (Melton, 1940), and lunar dune (MacDougal and Sykes, 1907).

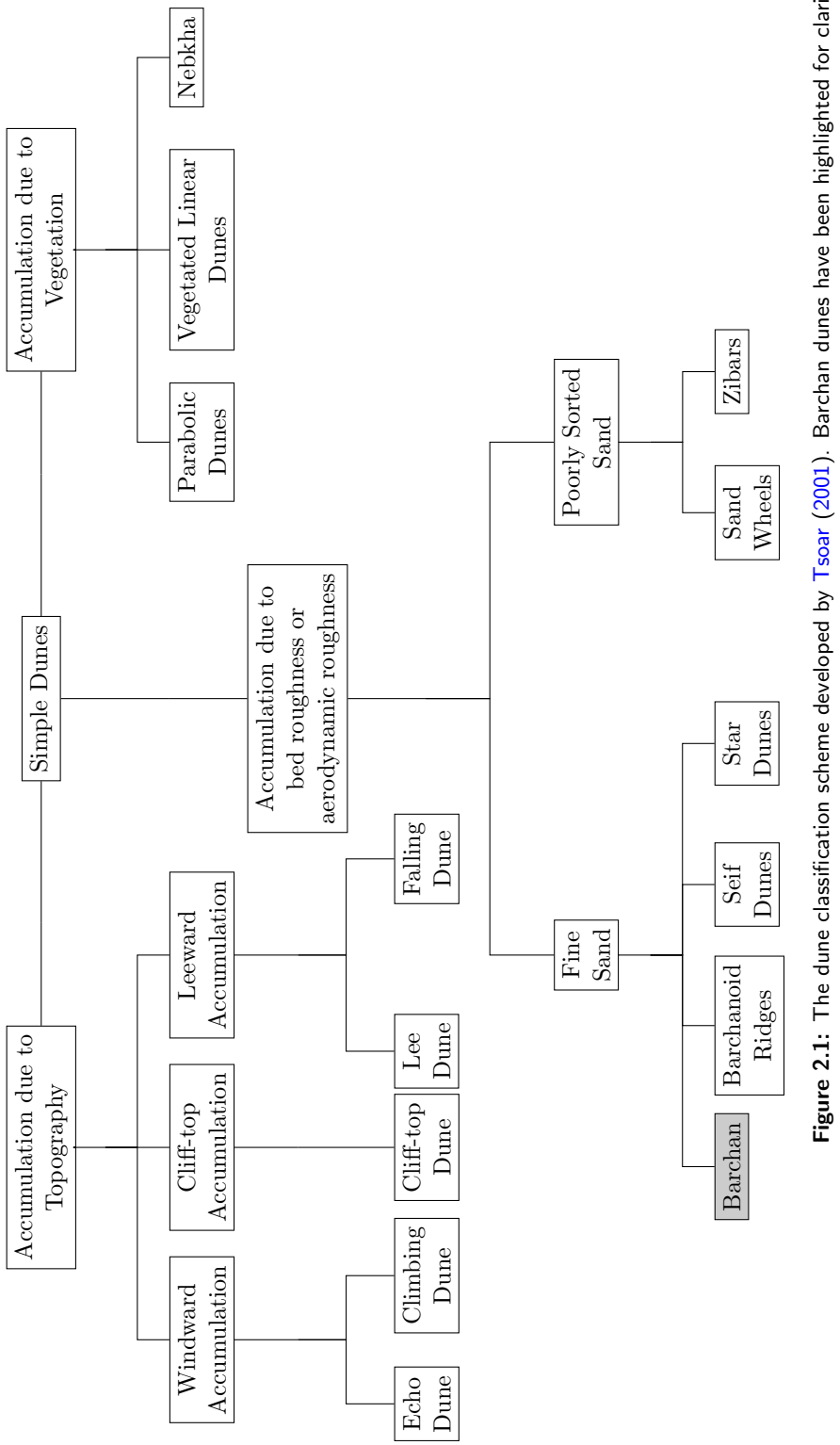


Figure 2.1: The dune classification scheme developed by Tsoar (2001). Barchan dunes have been highlighted for clarity

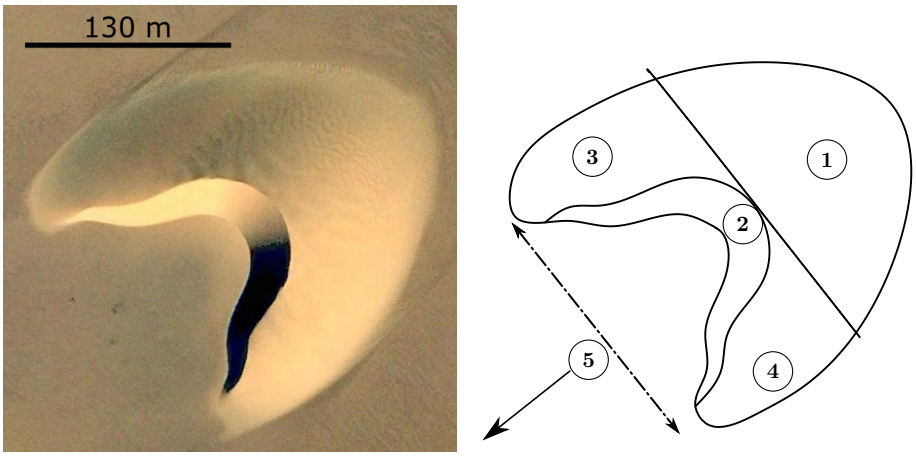


Figure 2.2: A simplified representation of the main morphological components of a barchan dune: Stoss slope (1), slip face (2), right horn (3), left horn (4), and the direction of the prevailing wind (5). The slip face is separated from the stoss slope by the brink line. The line tangent to the brink line is used as a reference point for identifying the horns.

of the dune in cases where the horn is elongated (Barnes, 2001). The low topographical profile of the horns prevents the formation of a slip face (Zhang *et al.*, 2014) which, in turn, prevents sediment from being trapped (as is the case on the slip face). This results in sediment being lost to downwind regions from the tips of the horns (Tsoar, 2001; Elbelrhiti *et al.*, 2008). These sediment pathways are illustrated in Figure 2.3.

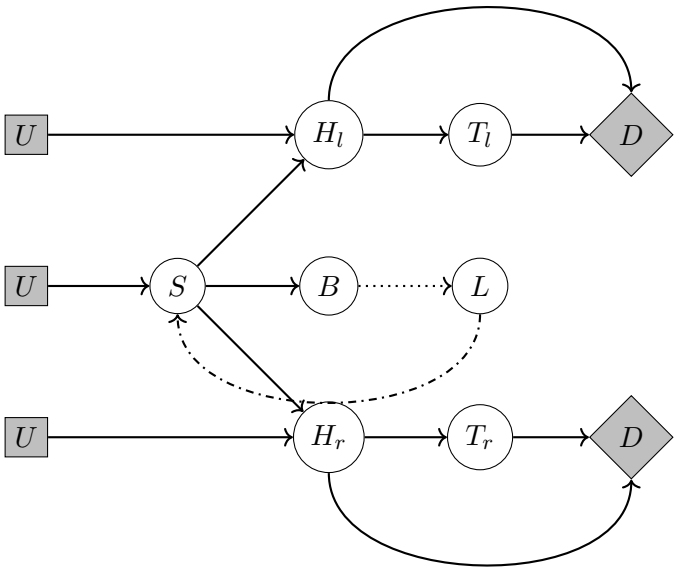


Figure 2.3: Sediment pathways along a barchan dune: U = upwind sources, S = stoss slope, B = brink line, H_l = left horn, H_r = right horn, T_l = left horn tip, T_r = right horn tip, L = slip face, and D = downwind sediment sources. The gray boxes represents upwind sources of sediment while the gray diamonds represent downwind sediment sinks. The solid line represents wind transported sediment, the dotted line represents gravitational processes, and the dash-dot line represents sediment that is contained within the barchan and exposed later as the barchan migrates.

The preceding is relevant to this research in several ways. First, it identifies barchans as individual landforms. This makes it easier to isolate individual forms that can be provided to the CNN for classification. In cases where the barchans are merged, a separation line becomes subjective with the obvious correct interpretation (Hugenholtz *et al.*, 2012). This adds an additional layer of complexity for the CNN that is not warranted at this early stage of development. Second, it establishes the sediment dynamics that occur on a barchan dune. This has important implications for the development of asymmetry (§2.4).

2.2.2 Dimensions

A number of different approaches have been used to measure barchan dimensions (Figure, 2.4 and Table 2.1). Reasons for these different approaches are not expressly stated in the literature. Instead, it appears that the selection of method is largely based on the data required by the study. The approaches differ in both the number of measurements used to describe the barchan and the points where these measurements are made. It is also reasonable to assume that the larger the diversity of approaches that can be automated, the more useful an automated system will be to the barchan research community. The approaches differ from cases where only a single measurement is made (e.g. 1a) to nine measurements being made (method 9). The deciding factor is the goal of the measurement where fewer measurements tend to be used in a descriptive fashion while more measurements are used when more detailed morphological parameters are desired.

There is, unfortunately, a substantial disadvantage associated with this diversity. Because measurements are not guaranteed to be between common points, it does introduce an element of error when comparisons are made between studies. For example, horn length can be expressed either using the brink line as a reference (Wang *et al.*, 2007; Burrough *et al.*, 2012), or using the base of the slip face as a reference (e.g Franklin and Charru, 2011; Ma *et al.*, 2014). Also, the distance between the horns can be expressed as the straight line, i.e. Euclidean, distance between the tips of the horns (Norris, 1966) or the perpendicular distance between the tips (Sauermann *et al.*, 2000). The magnitude of error that this introduces is not evaluated in this study and, therefore, its impacts on comparative research is unknown.

Bearing the limitations of such comparisons in mind, an analysis of the published literature (Bourke *et al.*, 2009; Jimenez *et al.*, 1999; Dong *et al.*, 2000; Wang *et al.*, 2007; Hamdan *et al.*, 2016; Embabi, 1982; Al-Dousari and Pye, 2005; Sauermann *et al.*, 2000; Barnes, 2001; Abdu, 1976; Finkel, 1959; Al-Harthi, 2002; Sagga, 1998; Lorenz *et al.*, 2013; Long and Sharp, 1964; Norris, 1966) indicates that barchans occur in a variety of different sizes (Figure 2.5 and Table 2.2). Several key points can be observed. There is a considerable variation between different

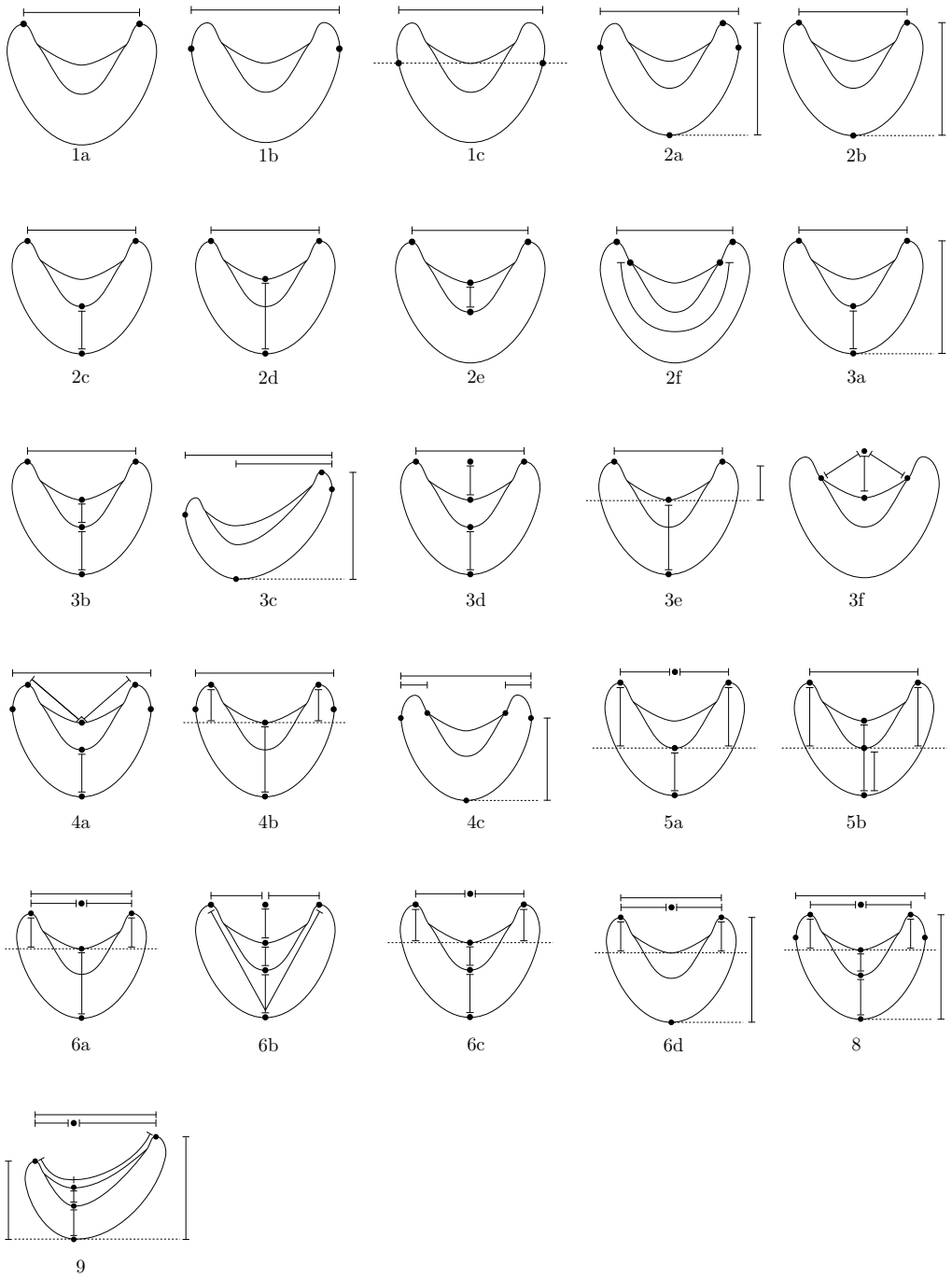


Figure 2.4: Different approaches to recording barchan dimensions. The integer represents the number of measurements that is made on the dune and the character is used to distinguish between the different approaches based on the sequence they were encountered in the literature.

Table 2.1: Studies that made use of the different methods (cf. Figure 2.4) to record barchan dimensions.

Approach	References
1a	Bagnold (1931, 1933); Simmons (1956); Norris and Norris (1961); McCulloch and Janda (1964); Hesp and Hastings (1998); Gay (1999); Bourke *et al.* (2006); Boulghobra (2016)
1b	Bourke *et al.* (2006); Durán *et al.* (2009)
1c	Gay (1999)
2a	Beadnell (1910); Parteli and Herrmann (2007); Parteli *et al.* (2007); Durán *et al.* (2010); Parteli and Pöschel (2017)
2b	Lonsdale and Malfait (1974); Breed and Grow (1979); Hersen *et al.* (2002)
2c	Norris (1966); Wippermann and Gross (1986); Wiggs *et al.* (1996); Jimenez *et al.* (1999); Barnes (2001); Sauermann *et al.* (2003); Parteli *et al.* (2005); Bourke *et al.* (2009); Bourke and Goudie (2009); Chojnacki *et al.* (2011); Lorenz *et al.* (2013); Hamdan *et al.* (2016); Boulghobra and Dridi (2016)
2d	Breed (1977); Cacchione *et al.* (1987); Courrech du Pont (2015)
2e	Hesse (2009)
2f	Silvestro *et al.* (2010)
3a	Long and Sharp (1964)
3b	Sagga (1998); Al-Harthi (2002)
3c	Parteli *et al.* (2014)
3d	Rempel (1936)
3e	Franklin and Charru (2011)
3f	El belrhiti and Douady (2011)
4a	Douglass (1909)
4b	Andreotti *et al.* (2002)
4c	Elbelrhiti *et al.* (2008)
5a	Wang *et al.* (2007)
5b	Burrough *et al.* (2012)
6a	Finkel (1959)
6b	Bailey (1906); Douglass (1909)
6c	Sauermann *et al.* (2000); Daniell and Hughes (2007); Moosavi *et al.* (2014); Nai *et al.* (2016)
6d	Ma *et al.* (2014)
8	Todd (2005)
9	Maghsoudi *et al.* (2010)

Table 2.2: Means and standard deviations of published barchan dimensions. T = Terrestrial Barchans, M = Marine Barchans.

Planet	Environment	Dimension	Mean (m)	Standard Deviation
Earth	M	Dune Width	337.66	603.45
Earth	M	Length of Stoss Slope	878.68	514.39
Earth	M	Total Length	78.33	92.25
Earth	T	Dune Width	130.66	108.64
Earth	T	Length of Stoss Slope	100.49	91.65
Earth	T	Total Length	56.59	56.26
Mars	T	Dune Width	127.79	32.29
Mars	T	Total Length	180.48	39.52

regions in barchan dimensions. The barchans in Peru appear to be smaller than the barchans in Namibia. The variability in barchan sizes within regions is also not consistent. For example, the barchans of Peru have a narrower range of sizes than the barchans from Egypt. On Earth, terrestrial barchans are substantially smaller than marine barchans. Lastly, barchans on Mars tend to be larger than the barchans on terrestrial environments here on Earth.

This section contributes in two key ways. First, it provides a brief overview of the different methods that have been used to describe the morphology of barchans (i.e. it completes Objective 1). As stated earlier, the goals of this study is to evaluate the performance of CNNs. In order for this to be achieved, suitable labels need to be generated that describe the different morphological categories. These labels, in turn, are obtained based on the methods illustrated in this section. A second contribution is the description of the different sizes of barchans. This is needed in order to properly contextualise the role of size in barchan asymmetry (§2.4.5).

2.2.3 Distribution

Barchans have been identified in a number of different countries around the world (Table 2.3). Additionally, they have been observed on Mars (Bandeira *et al.*, 2013; Chojnacki *et al.*, 2011; Cardinale *et al.*, 2012), on Saturn's moon Titan (Ewing *et al.*, 2014; Radebaugh *et al.*, 2010), and on the dwarf planet Pluto (Parteli and Pöschel, 2017). Barchans can also be found along the sea floor (Daniell and Hughes, 2007; Lonsdale and Malfait, 1974; Ma *et al.*, 2014).

Within these regions, barchans also exhibit local spatial groupings in the form of dune fields (Durán *et al.*, 2009; Worman *et al.*, 2013; Elbelrhiti *et al.*, 2008; Sauermann *et al.*, 2000). Both the size and spatial distribution of barchans differ between dune corridors (Elbelrhiti *et al.*, 2008). Spatial distribution patterns can also differ within a given dune field (Embabi and Ashour, 1993) but appears to be non-random (Durán *et al.*, 2009; Bishop, 2007). A given corridor can also contain barchans of varying sizes (Hugenholtz and Barchyn, 2012). Furthermore, there does appear to be a relationship between the density and size distribution of dune within a

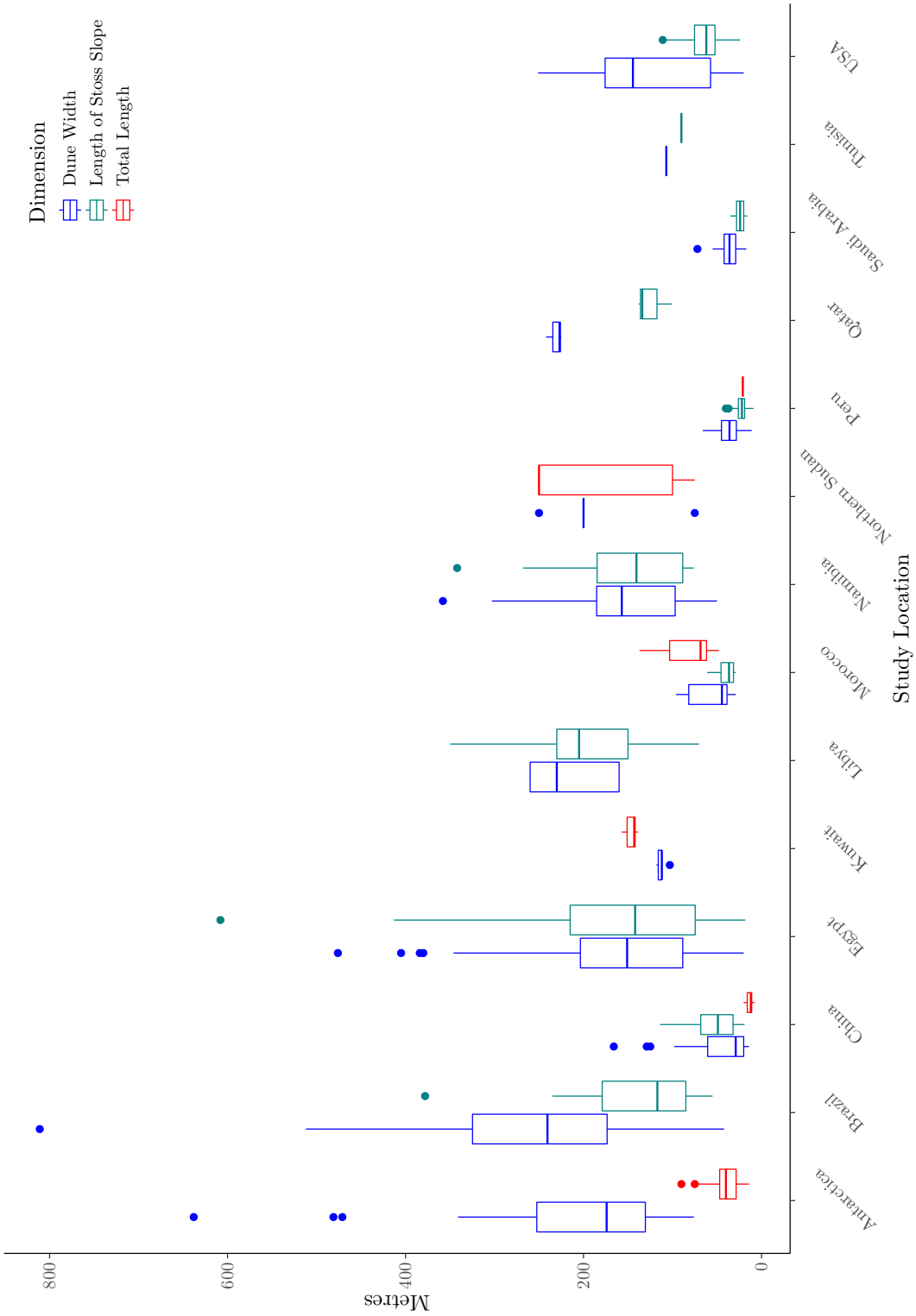


Figure 2.5: Published dimensions (1959–2016) for barchans found on Earth (excluding marine barchans).

Table 2.3: Different regions on Earth were barchan have been identified.

Region	Reference	Region	Reference
Afghanistan	McMahon (1906)	Morocco	Azzaoui *et al.* (2019)
Algeria	Boulghobra (2016)	Namibia	Lancaster (2014)
Antarctica	Bourke *et al.* (2009)	Oman	Wiggs *et al.* (1996)
Australia	Bishop (2001)	Peru	Tsoar and Parteli (2016)
Botswana	Grove (1969)	Poland	Borówka (1980)
Brazil	Pelletier *et al.* (2015)	Qatar	Engel *et al.* (2018)
Chad	Gao *et al.* (2015)	Russia	Labuz *et al.* (2018)
China	Zhang *et al.* (2018)	Saudi Arabia	Schatz *et al.* (2006)
Egypt	Hamdan *et al.* (2016)	Spain	Gómez-Ortiz *et al.* (2009)
India	Wadhawan (1996)	Sudan	Lv *et al.* (2016)
Iran	Maghsoudi *et al.* (2010)	Tunisia	Lorenz *et al.* (2013)
Israel	Tsoar and Blumberg (2002)	Turkmenistan	Lessar (1885)
Kuwait	Al-Dousari *et al.* (2015)	United Arab Emirates	Abu-Zeid *et al.* (2001)
Libya	Bagnold (1933)	United States of America	Pelletier (2013)
Lithuania	Labuz *et al.* (2018)	Uzbekistan	Morgan (1881)
Mexico	Inman *et al.* (1966)		

dune field with narrow size distributions being associated with dense fields and a larger size distribution occurring in sparse fields (Durán *et al.*, 2009). The spatial proximity of upwind barchans can influence the shape of downwind barchans (Parteli *et al.*, 2014; Lv *et al.*, 2016), however, it remains to be examined how the spatial organisation of barchans affect the types of morphologies that are present within the dune field.

The implications of the barchan distribution on the present research is as follows: it provides data on the distribution of barchans around the globe which, when combined with the dimension data, help identify regions where suitable satellite imagery may be found. It also highlights a potential application of using CNNs in barchan research. By combining the techniques developed in this study with those used to extract barchans from satellite imagery, it will be possible to investigate the relationship between spatial organisation of barchans within a dune field and the accompanying range of morpholgies that are present. This idea is explored further in §6.4.1.

2.3 Barchan shape

For the purposes of this study, it is important to have some understanding about the processes that lead to the development of the barchan's characteristic shape. As will be seen later in the section on asymmetry (§2.4), the processes that result in the development of asymmetry are, in effect, the same types of processes that lead to the development of the barchan. This is because, fundamentally, any barchan shape or asymmetry is merely the result of sediment trasnport along different pathways or with different intensities. Since the envisaged application of this research is focused on extracting shape information with the goal of inferring environmental characteristics, it must first be demonstrated that the environment is a determining factor on the shape (or asymmetry) of a barchan.

2.3.1 Wind and Sediment Interaction

Several different environmental factors need to be in place in order for barchan genesis to occur:

1. The winds need to be either unidirectional (Elbelrhiti *et al.*, 2008; Barnes, 2001; Bourke, 2010; Durán *et al.*, 2010; Sauermann *et al.*, 2000) or narrowly bimodal (Bourke and Goudie, 2009) with low directional variability Wiggs (2013).

2. The region must be sediment limited (Barnes, 2001; Bourke, 2010; Bourke and Goudie, 2009; Durán *et al.*, 2010; Sauermann *et al.*, 2000; Tsoar, 2001; Wiggs, 2013; Hugenholtz and Barchyn, 2012) on a non-erosive surface (Barnes, 2001; Tsoar, 2001) with an upwind sediment source (Elbelrhiti *et al.*, 2008).

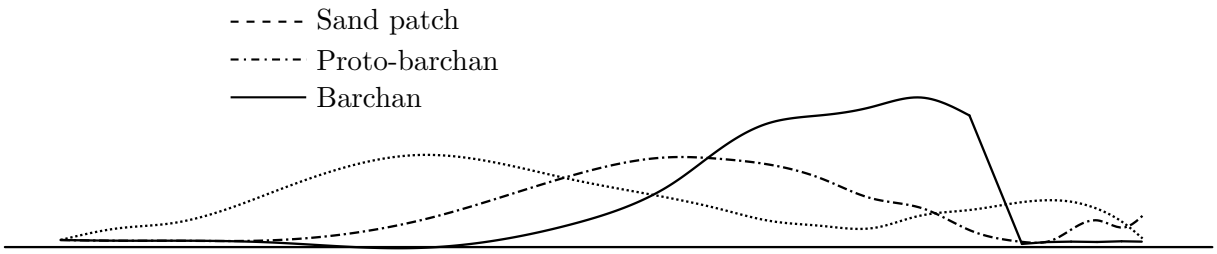


Figure 2.6: Longitudinal profiles for barchan evolution from an initial sand patch.(Adapted from data published in Elbelrhiti (2012)).

3. The region needs to have low precipitation and sparse vegetation (Barnes, 2001; Bourke and Goudie, 2009).

As dunes, the genesis of barchans involve the interaction between granular matter and fluid flow (Alvarez and Franklin, 2018; Durán *et al.*, 2010). Barchans can form from an initial accumulation of sediment (Lv *et al.*, 2016; Qian *et al.*, 2021; Engel *et al.*, 2018; Kocurek *et al.*, 1992) (Figure 2.6). The dynamics associated with sand patches are, as yet, not properly understood (Baddock *et al.*, 2018). These sand patches are sub-elliptical to sub-rectangular in shape, possess a distinct rounded upwind boundary with a less distinct lee boundary[2], and their long axis is orientated along the direction of dominant winds (Lancaster, 1996). Continuous wind action transports sediment from the windward slope of the dune (Elbelrhiti, 2012; Zhang *et al.*, 2014) and deposits it on the crest of the dune (Lv *et al.*, 2016; Wiggs, 2013). As this material continues to accumulate, the angle of repose is eventually exceeded (Lv *et al.*, 2016; Wiggs, 2013). This leads to the formation of a slip face (Zhang *et al.*, 2014) as a result of gravitational collapse and marks the transition from a protodune into a barchan dune (Elbelrhiti, 2012). It is worth noting that the angle at which collapse occurs is not constant and is influenced by the wind speed (Pelletier *et al.*, 2015).

2.3.2 Emergence of Barchan Shape

In the previous section, the emergence of the slip face is discussed. A second key characteristic of barchan dunes is the presence of horns. The development of a barchan's horns are a direct consequence of interaction between the wind and the geometry of the sand pile. Consider the situation depicted in Figure 2.7 which represents a conical sand pile at an angle of repose (taken here as 34°). Although it is unlikely that a sand patch or sand pile in nature will have such a shape[3], this configuration has been used in studies dealing with barchan shape simulation (e.g.

[2]The lee boundary, in this context, refers to the region of the barchan dominated by the gravitational collapse of accumulated sediment.

[3]Sand patches that evolve into barchans are subelliptical to subrectangular in shape (Lancaster, 1996).

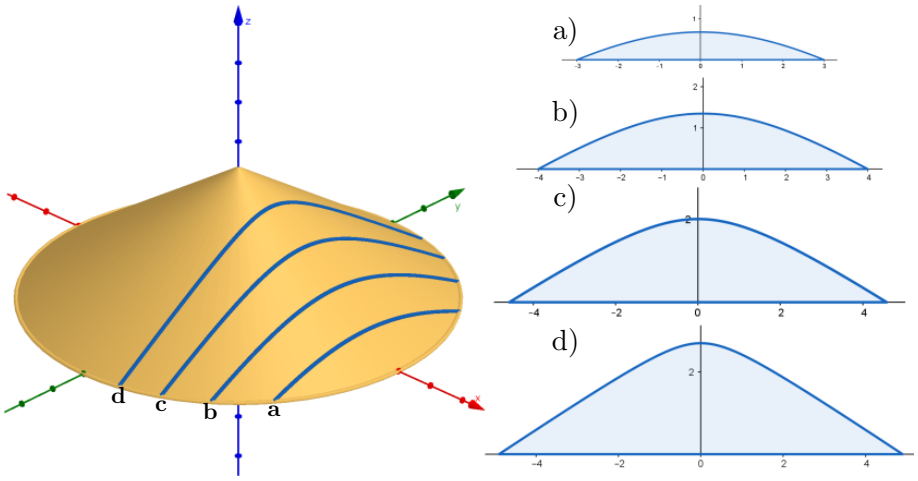


Figure 2.7: Sections through a cone shape sand pile. The curves on the right represents the cross sections of the sand pile in the left of the image.

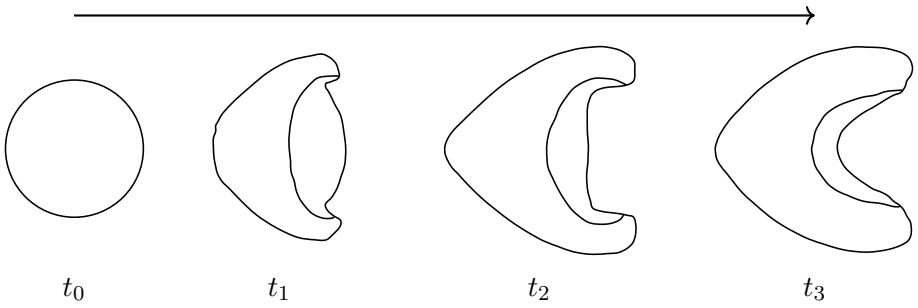


Figure 2.8: The shape evolution from an initial pile of sand (t_0) to a barchan dune (t_3) (adapted from Durán *et al.* (2010)). At t_1 the slip face merges and the horns of the barchan start elongating at t_3. The wind direction is indicated by the arrow above the image.

Wippermann and Gross, 1986; Alvarez and Franklin, 2018; Hersen *et al.*, 2004) and as such can provide some insights.

Moving from the flanks of the sand pile towards the central peak (i.e. slice a to slice d) the area underneath each slice increases. In three dimensions, this will equate to an increase in volume. A larger volume is associated with a larger mass which, in turn, is associated with a larger reconstitution time (Wiggs, 2013). As a result, the lateral flanks of the barchans are displaced faster than the central portion[4] leading to the development of the barchan's horns. Additionally, as the winds moving over the windward slope of the dune get displaced towards the horns, local acceleration occurs (Tsoar, 2001) leading to further displacement of the horns downwind. A visual summary of the evolution of a barchan from an initial sand pile is given in Figure 2.8.

[4]This has been documented as far back as the work of Forsyth (1876)

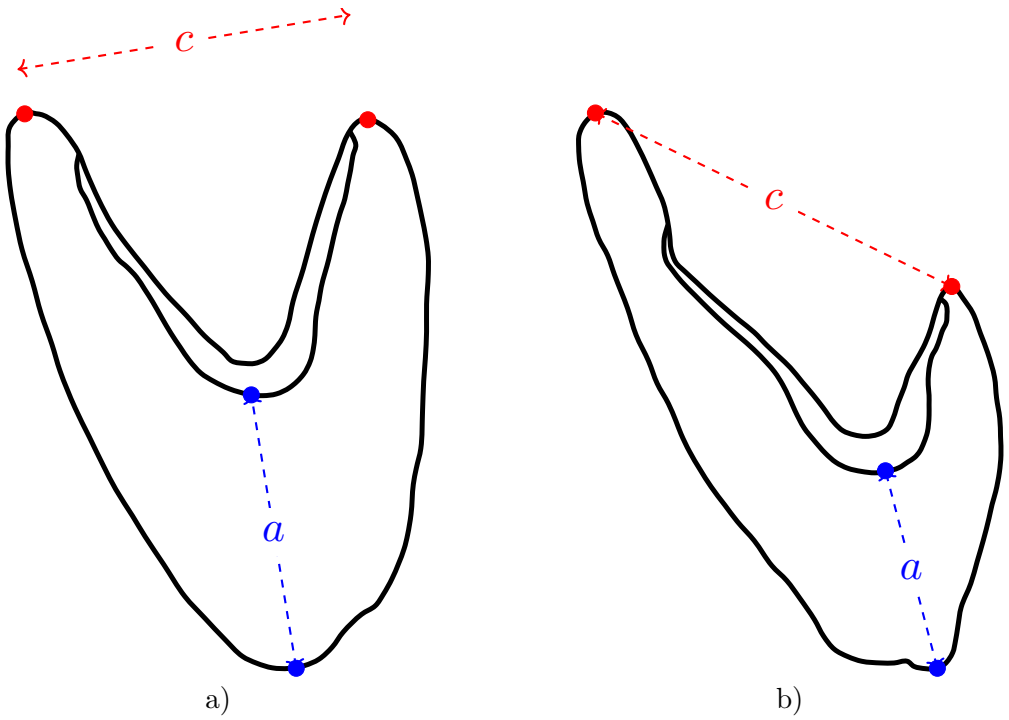


Figure 2.9: Two different approaches in calculating the $\frac{a}{c}$–ratio: a) expressing the distance between the horns perpendicular to the longitudinal axis, and b) expressing it as the straight line distance.

2.3.3 Description of Barchan Shape

The $\frac{a}{c}$ ratio (Figure 2.9) has been used as a descriptive tool in several studies on barchans (Hamdan *et al.*, 2016; Bourke and Goudie, 2009; Barnes, 2001; Boulghobra and Dridi, 2016; Lorenz *et al.*, 2013; Long and Sharp, 1964; Chojnacki *et al.*, 2011; Norris, 1966)[5]. It relates the length of the stoss slope to the distance between the tips of the horns. The relationship between these two variables is illustrated by the nomogram in figure 2.10. By connecting the measurement obtained for the stoss slope (a) with the measurement of the distance between the horns (c), the $\frac{a}{c}$–ratio, and its corresponding shape category, can be determined. Unfortunately, there is some ambiguity in the original diagrams provided by Long and Sharp (1964), who developed this method. Their use of a symmetrical barchan as an illustration creates uncertainty since the distance between the horns can either be interpreted as the straight line distance between the tips of the horns, or as the distance perpendicular to the central axis.

This ratio has been related to migration speed and sediment dynamics (Parteli *et al.*, 2007). Hamdan *et al.* (2016) found that there is an inverse relationship between migration speed and the $\frac{a}{c}$ ratio with smaller ratios migrating more rapidly than larger ratios. The work of Parteli *et al.* (2007) found that as the sediment influx to a barchan increases, the dune transitions from being *slim* to becoming *fat*. This can be demonstrated using the nomogram. Assuming that the

[5]There is an element of irony in this widespread use since the original authors considered their study to be a "quick and dirty investigation"(Long and Sharp, 1964, :150)

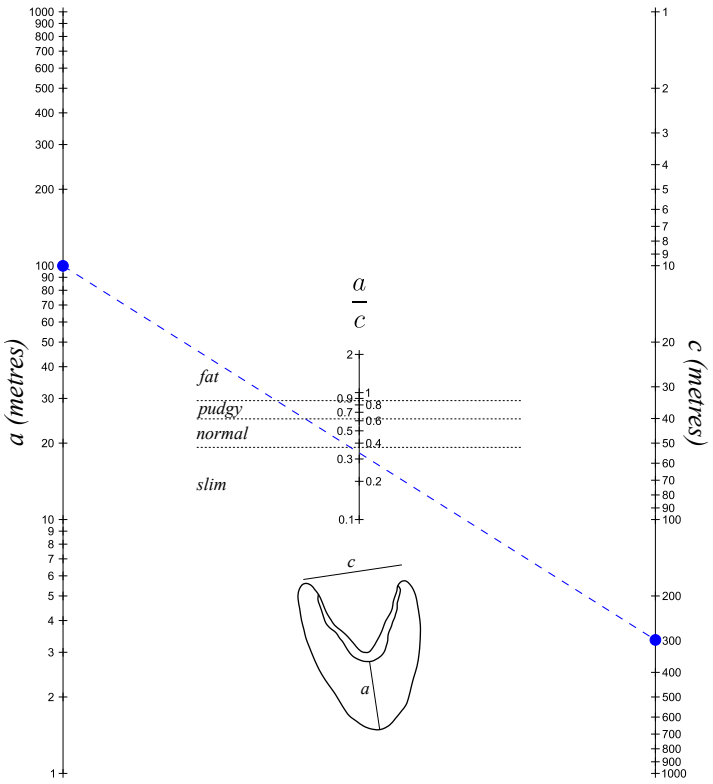


Figure 2.10: A nomogram showing the relationship between the a and c dimensions of a barchan and its classification as being either *slim*, *normal*, *pudgy*, or *fat*. This nomogram is based on the classification system developed by Finkel (1959) and expanded upon by Bourke and Goudie (2009). In this example, a barchan with an a–axis of 100m and a c–axis of 300m resulting in a classification into the *slim* category. A larger version of this nomogram is available in Appendix A.

c–axis remains constant, additional material that enlarges the a–axis, will result in a shift of the line joining these two points towards the *fat* category. It must be noted that this is, admittedly, an over-simplification since any additional material added to the stoss slope will alter the airflow in that region. This could influence the lateral airflow as well which, in turn, can result in horn displacement.

The determination of the shape of a barchan (as expressed in its $\frac{a}{c}$–ratio) is one of the tasks that the CNN is expected to successfully complete. Therefore, it is important to know how this value is calculated so that the labels to be used by the CNN can be properly assigned.

2.4 Barchan Asymmetry

As physical objects that cause increased relief, the interaction between barchans and a uniform fluid flow results in the formation of regions of altered flow (Bauer, 2013). Alterations in fluid flow are associated with changes in sediment transport capability and therefore the distribution of granular material is modified. Since the shape of a dune is the result of the distribution of granular material, any change in the distribution of granular material is also associated with a change in shape. This change in shape does not have to be to the extent where a reclassification of the dune is warranted, although this can happen in the case of barchans transitioning into seif dunes (Tsoar and Parteli, 2016; Lv *et al.*, 2016). However, this change in shape is often associated with barchan asymmetry.

Symmetrical versions of barchans are not prevalent in nature (Parteli *et al.*, 2014) and the asymmetric version predominates on both Earth and Mars (Bourke, 2010; Lv *et al.*, 2016; Tsoar and Parteli, 2016; Zhang *et al.*, 2018). Barchan asymmetry refers to several morphological changes occurring on the dune. First, one horn can be extended in a downwind direction (Bourke, 2010). Second, one horn can be wider than the other horn (Elbelrhiti *et al.*, 2008). Third, the lateral side of a barchan can be extended in the down-slope direction (Parteli *et al.*, 2014). Lastly, any combination of the preceding forms can be present to different extents. The following sections will discuss these in more detail.

This section provides support for using asymmetry to interpret sediment processes which forms part of the practical application of this research. If the relationship between process and shape is understood, in the absence of equifinality (Haines-Young and Petch, 1983) and using abductive reasoning (Inkpen, 2005), then it is possible to work "backwards" and derive processes from observed shape. Additionally, it also introduces the concepts associated with asymmetry upon which the different tasks are based.

2.4.1 Side of Horn Elongation

As mentioned earlier, one of the barchan horns can be elongated relative to the other (Parteli *et al.*, 2014; Zhang *et al.*, 2018; Lv *et al.*, 2016; Tsoar and Parteli, 2016; Barnes, 2001). These changes are not consistently present for all dunes within the same dune field (in the sense that it is always the same horn that is elongated) (Barnes, 2001) and both symmetric and asymmetric barchans can be present within the same dune field (Abdu, 1976). Several factors have been identified as causing this elongation: bimodal winds (Bagnold, 1954; Tsoar, 1984; Melton, 1940; Bourke, 2010), topography (Parteli *et al.*, 2014), collisions (Tsoar and Parteli, 2016; Zhang *et al.*, 2018; Boulghobra, 2016), modifications to upwind airflow (Barnes, 2001), and changes in the direction of the prevailing wind direction (Simmons, 1956).

In terms of bimodal winds, three models have been proposed to explain this process (Figure 2.11). One model was proposed by Bagnold (1951). In this model, the barchans are aligned along the gentler winds and it is a stronger secondary wind that causes the extension of one of the horns. The extended horn is the one facing closest to the oncoming stronger wind. In an alternate model, proposed by Tsoar (1984), barchans are orientated along the direction of the strongest winds and it is the gentler secondary wind that causes the elongation of the horn opposite the wind. The factor that determines which of these two models will be valid is the angle between the primary and the secondary wind (Lv *et al.*, 2016). A third model, which has not been widely discussed in the literature, was proposed by Melton (1940). In this model the barchans are not aligned to any particular wind and the extension of the horn is directly downwind of the stronger and gentler winds with the magnitude of elongation depending on which horn is elongated by which wind.

Asymmetry can also result from an asymmetrical sediment supply (Zhang *et al.*, 2018; Parteli *et al.*, 2014; Bourke, 2010; Lv *et al.*, 2016). Since barchans are not necessarily isolated features, and therefore affected by surrounding dunes (Barnes, 2001), they do exist in a state of receiving sediment from upwind sources and losing sediment to downwind sources (Figure 2.3). There are two contrasting views with regards to limb elongation under these conditions. Lv *et al.* (2016) states that the horn closest to the greater sediment supply will be elongated while Parteli *et al.* (2014) demonstrates, based on computer simulations, that the limb opposite to the one receiving the additional influx will be elongated (Figure 2.12). Considering that the additional influx will increase the volume of material, and the extension of horns is related to the rate of sediment displacement, it is more likely that the latter view will occur.

Bachan corridors are characterized by collisions (Hugenholtz and Barchyn, 2012). Barchans are migratory dunes (Tsoar, 2001) and their rate of migration is inversely proportional to their

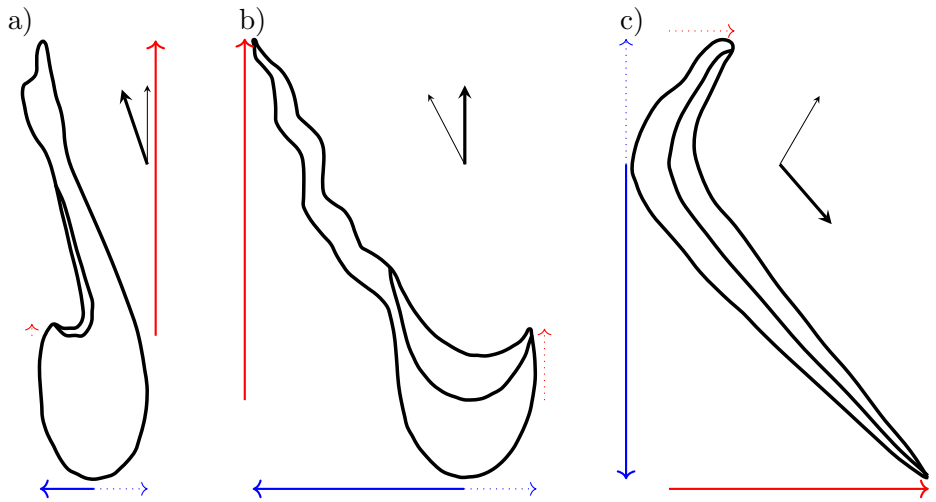


Figure 2.11: Asymmetry due to wind regime according to: a) Bagnold (1954), b) Tsoar (1984), and c) Melton (1940). The thick arrow represents the dominant wind direction while the thinner arrow represents the secondary wind. The coloured arrows show the asymmetry of the dunes in this example with blue representing the lateral extension and red the extension of the horn in the downwind direction. Solid colour arrows represent the largest value.

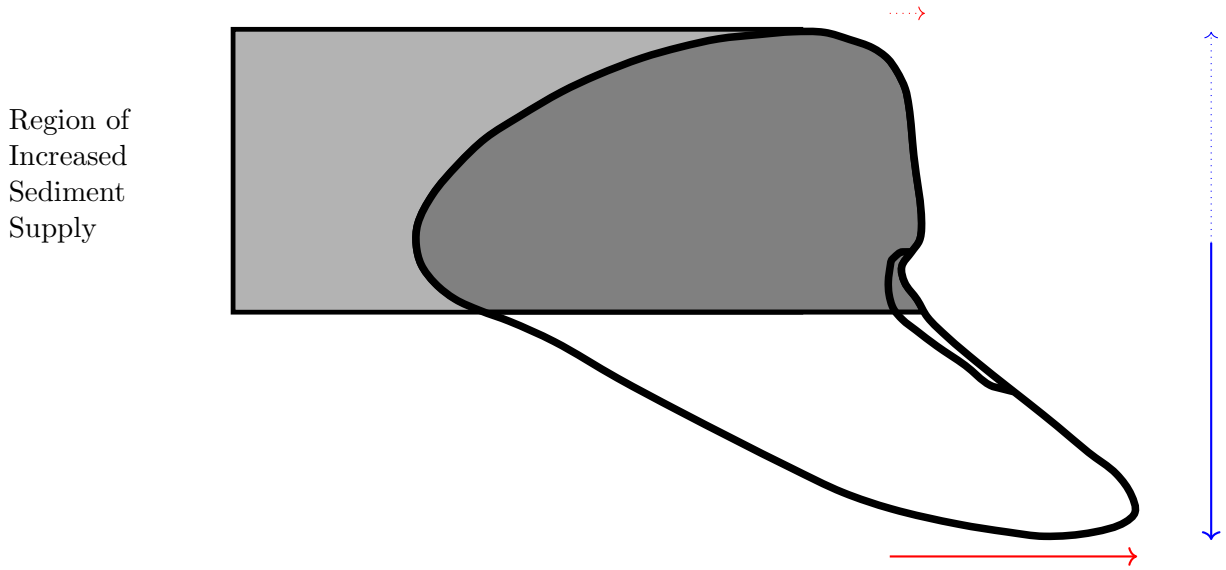


Figure 2.12: Asymmetry that resulted from an asymmetrical sand supply. The shaded region represents the region that is receiving the additional influx of sediment (adapted from Parteli *et al.* (2014)). The difference in right and left lateral extension is indicated by the blue arrows, while differences in horn length are represented by the red arrows. Solid lines indicate the larger value.

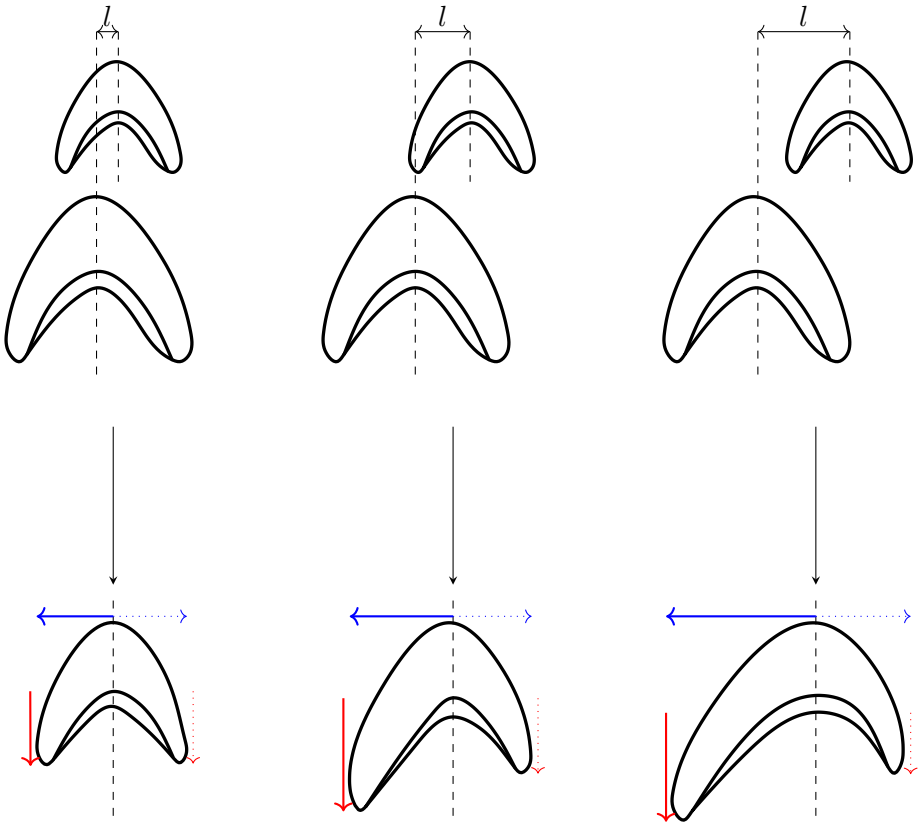


Figure 2.13: Simulated outcomes of two barchan dunes with different amounts of lateral offset (l) (adapted from Parteli *et al.* (2014)). The lateral extension of the resultant barchan is indicated by the blue arrows, while the red arrows indicate the horn lengths. The solid arrows indicate largest values. As the value of l increases, the resulting asymmetry increases.

height (Norris, 1966; Abdu, 1976; Sauermann *et al.*, 2000; Worman *et al.*, 2013; Wiggs, 2013; Gay, 1999). Because a dunefield contains barchans of a variety of different sizes (Durán *et al.*, 2009; Momiji *et al.*, 2002), the migration rates are unequal. Therefore, faster barchans can "catch-up" to slower barchans resulting in collisions (Figure 2.13). These collisions lead to the development of asymmetrical barchans (Parteli *et al.*, 2014; Zhang *et al.*, 2018; Bourke, 2010), which can manifest as both an elongation of one of the horns and the lateral extension of the resultant barchan.

2.4.2 Differences in Horn Width

Asymmetry in the context of horn width is understudied. It refers to a situation where the width of the horns are unequal (Elbelrhiti *et al.*, 2008) (Figure 2.14). Given that sediment is lost from the horns (Tsoar, 2001; Hersen, 2004; Zhang *et al.*, 2014) for potential intercept by downwind barchans, this can have a great impact on the morphology of downwind barchans. The work of Elbelrhiti *et al.* (2008) related the horn width to the amount of sediment flux between an upwind dune and a downwind dune. Keeping the distances between these two dunes constant, wider horns will result in an increase in the amount of sediment flux.

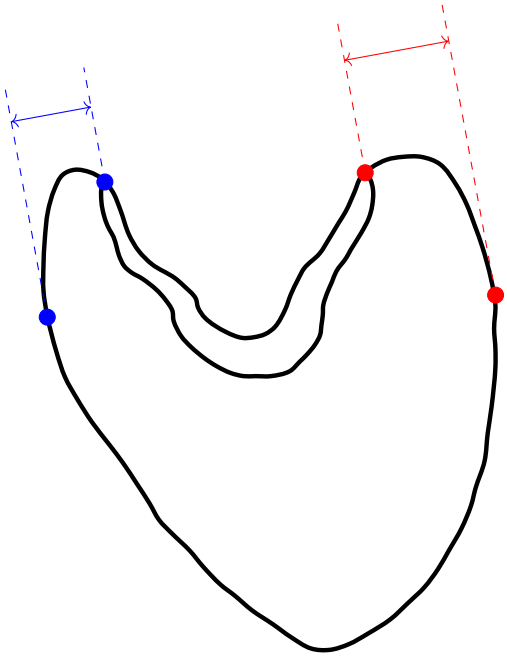


Figure 2.14: An illustration of the concept of horn width asymmetry used by Elbelrhiti *et al.* (2008). In this example, the right horn (red) is wider than the left horn (blue).

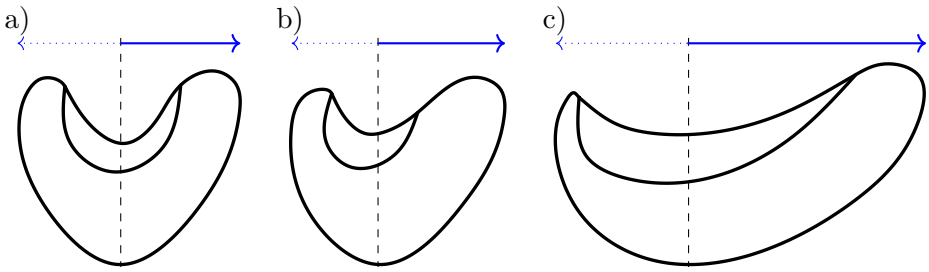


Figure 2.15: Increase in barchan asymmetry associated with an increase in the inclination of the slope (adapted from Parteli *et al.* (2014)). The surface tilt of the dunes are: a) $11.3°$, b) $19.8°$, and c) $26.6°$ and the direction and magnitude of displacement is indicated by the arrows. The dashed line represents an approximation of the longitudinal axis of the dune.

2.4.3 Lateral Extension of Barchan

The influence of topography in alternating the shape of barchans was first proposed by Finkel (1959) who linked it to the lateral spread of barchans in Peru. Later studies have also agreed that the local topography of the surface on which the barchan occurs can lead to asymmetry (Zhang *et al.*, 2018; Parteli *et al.*, 2014; Bourke, 2010; Lv *et al.*, 2016). In the case of an inclined surface, computer simulations by Parteli *et al.* (2014) showed that sediment transport along the downhill horn has an additional downhill component leading to an extension of the horn in that direction. In contrast, any downhill movement of the upslope horn will be trapped by the slip face resulting in no net downslope deformation (Parteli *et al.*, 2014) . As the tilt of the surface increases, so the level asymmetry observed also increases (Figure 2.15).

2.4.4 Magnitude of Horn Elongation

The magnitude of horn elongation refers specifically to the differences in length between the left and the right horn. The mechanisms by which this occurs is the same as those discussed earlier (Section 2.4.1). Parteli *et al.* (2014) states that, provided the angle between primary and secondary winds do not change[6], the longer the duration of the secondary wind, the more elongated a horn becomes. A similar explanation is given by Bourke (2010). Differing sediment supply can also play a role where an increase in sediment supply can lead to the horn receiving the additional influx becoming shorter than the horn without the additional influx (Parteli *et al.*, 2014), or the opposite way around according to Lv *et al.* (2016). Collisions can also manifest in severe elongation of one of the horns where a smaller downwind barchan gets "absorbed" by a larger upwind barchan at one of its horns (Bourke *et al.*, 2009). Bourke (2010) also discusses how horns that are funnelled through a gap in the topography can exhibit an elongated horn (i.e. the horn that is being funnelled elongates relative to the other horn).

2.4.5 Influence of Size on Asymmetry

It has been shown that there is a potential relationship between the size of barchans and its shape and asymmetry (van der Merwe, 2021). The mean shape of barchans tend to change as the size of the barchan increases hinting at the possibility of barchan allometry. Boulghobra (2016) also links increasing barchan size with increasing levels of asymmetry. However, Hesp and Hastings (1998) state that barchans follow scaling laws that tend towards isomtery.

2.5 Automated Detection of Barchans

For reasons that will be discussed in more detail in Chapter 4, the input data used in the CNN models developed here are intentionally limited in the amount of texture information that they contain. This is mainly to determine if CNNs are able to classify barchans when only supplied with images containing outlines. Therefore, to place this work into a larger context, previous work that attempted to extract barchan outlines from imagery need to be consulted.

Rubanenko *et al.* (2021) used a Regional Convolutional Neural Network (R-CNN) to detect and construct outlines of barchans on Mars. The output of their model is a segmented mask corresponding to the position of the barchan within the image. Image segmentation is a technique where a collection of neighbouring pixels are homogenised based on some shared attribute (Lakshmanan, 2005). This, in effect, separates the image into distinct areas based on texture (Armi and Fekri-Ershad, 2019). This is very different from the outlines used in this study and does impose a limitation. It is not possible, using a segmentation mask, to distinguish the mor-

[6]Here, the barchans are aligned along the direction of the primary winds.

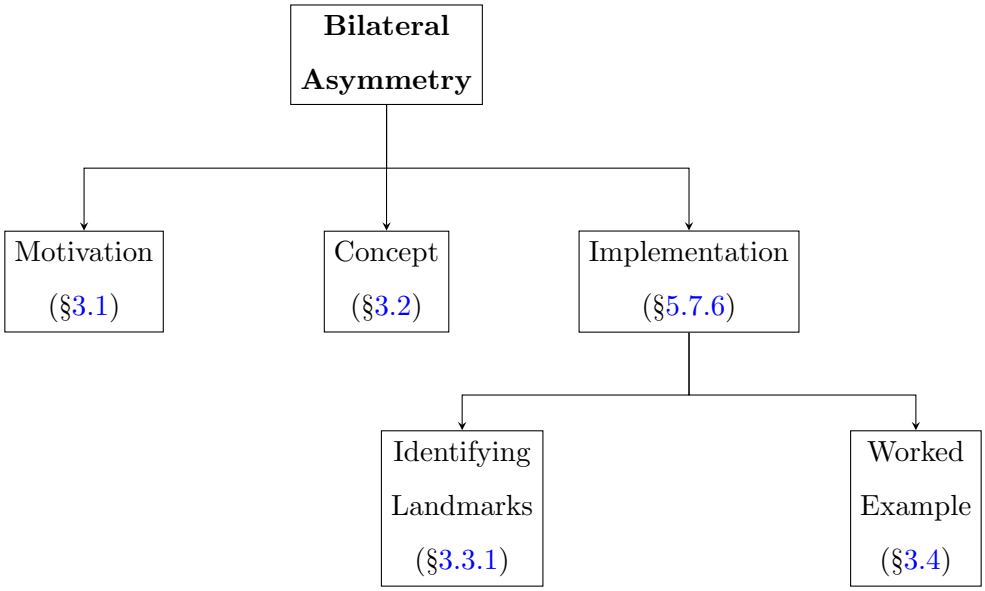
phological components of the barchan such as the stoss slope and the slip face (Figure 2.2). This would still be the case if the raster image[7], which represents the mask, is converted to a vector[8] polygon of just the outline. Since a large number of morphometric descriptions are based on these regions (Figures 2.4 and §2.4), the mask data generated by Rubanenko *et al.* (2021) cannot be used here. Similar work was also carried out by Azzaoui *et al.* (2020) and Vaz *et al.* (2015).

Although these studies do not directly reproduce the input images required for the CNNs investigated here, the separate techniques needed have been developed. The barchan masks that are automatically generated (Rubanenko *et al.*, 2021; Azzaoui *et al.*, 2020; Vaz *et al.*, 2015) can be converted to vector polygons. This makes it possible to have a planform outline of the barchan. But, as already mentioned, the morphological components of the barchan cannot be differentiated under these conditions. The work of Shumack *et al.* (2020) has demonstrated that ti is possible to extract dune crests (their work did not focus on barchans) from images. Therefore, combining these two approaches would, in principle, make it possible to extract the needed data for the models developed here.

[7] In a raster image the information is contained in values stored within pixels.
[8] In a vector image the information is stored as points and the equations of lines connecting different points.

Chapter 3

Bilateral Asymmetry



3.1 Motivation

This section describes the motivation behind developing a new method to describe complex asymmetry, i.e. the combined influence of several different types of asymmetry such as downwind elongation and lateral extension. The asymmetry is discussed from a vector-based perpsective which indicates that changes in shape can be viewed as a displacement of landmarks with an x and y vector component. Next, the limitations of using existing methods is discussed before the new proposed method is introduced. The section concludes with a worked example to illustrate the new method.

3.1.1 A Vector-Based Approach to Barchan Asymmetry

It is highly unlikely that a barchan would experience only one of the asymmetries discussed in Chapter 2. It is more likely that the asymmetry of a barchan would manifest as complex pattern of downwind elongation and lateral extension. Using common reference points, the change from

Table 3.1: The results of a linear model where the y–coordinate is the dependent variable and the x–coordinate is the independent variable.

Region	**Linear model**	R^2	p–value
Left Horn Tip	$y = 0.38x + 0.01$	0.86	< 0.001
Right Horn Tip	$y = -0.37x - 0.05$	0.68	< 0.001
Left Max Lateral Extent	$y = 0.3x + 0.44$	0.62	< 0.001
Right Max Lateral Extent	$y = -0.28x - 0.45$	0.42	< 0.001
Join between left horn and slip face	$y = 0.45x - 0.13$	0.87	< 0.001
Join between right horn and slip face	$y = -0.45x + 0.11$	0.67	< 0.001

a perfectly symmetrical barchan to an asymmetric barchan is illustrated in Figure 3.1. The displacement of a landmark[1] can be expressed as changes on both x and y components. This diagonal displacement is therefore the resultant vector of these two components. While not conclusive, it does provide some *prima facie* support for the claim being made.

Further support for this claim is provided by plotting corresponding landmarks from the sampled dunes (Figure 3.2). By using a baseline registration approach (Zelditch *et al.*, 2004; Rohlf, 2000; Webber and Hunda, 2007) provided by the *shapes* R package (Dryden and Mardia, 2016), it is possible to depict the positions of the landmarks in relation to fixed points. Here, the fixed points are the toe of the barchan and a point where the longitudinal axis intersects the brink line. The result of this transformation is that the influence of size, orientation and position are removed, and the shape variation is restricted to the points of interest (Webber and Hunda, 2007). In this case, the points of interest are the tips of the horns, the point of maximum lateral extent, and the meeting point between the slip face and the horns. Although the manner in which the longitudinal axis is determined (§3.3.1) does have a role in the perceived symmetry across the longitudinal axis, it is still clear that the all of the dunes have a distinct diagonal displacement of their landmarks. This is supported by the results of a linear model where the y–coordinate is predicted as a function of the x–coordinate (Table 3.1).

3.1.2 Limitations of Current Methods

Current approaches to describing barchan morphology (§2.2.2) are based on distances measured between two points. Although such a distance-based approach can, conceivably, be used to describe such complex displacements, it does have significant shortcomings. First, the measurement styles only tend to record one vector component of the shape (the vector component recorded is based on whichever component is associated with the downwind displacement of the landmark). A notable disadvantage of such an approach is that the lateral displacement of the landmark can assume any value without altering the measured vector (Figure 3.3). In this example, a lateral displacement of a horn tip, which can occur on an inclined surface (Parteli *et al.*,

[1] A landmark is a point of correspondence (Dryden and Mardia, 1993) (see §3.3.1).

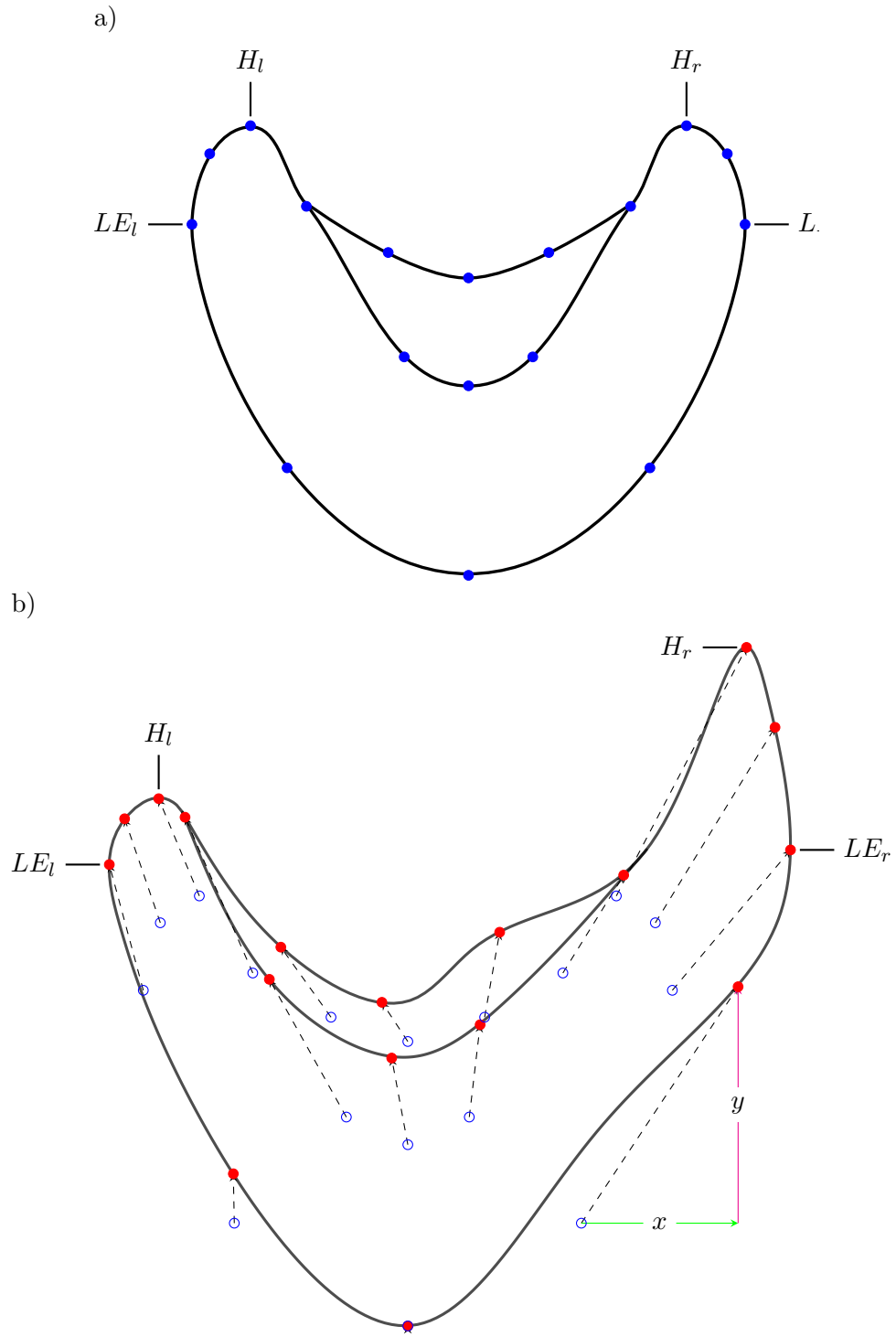
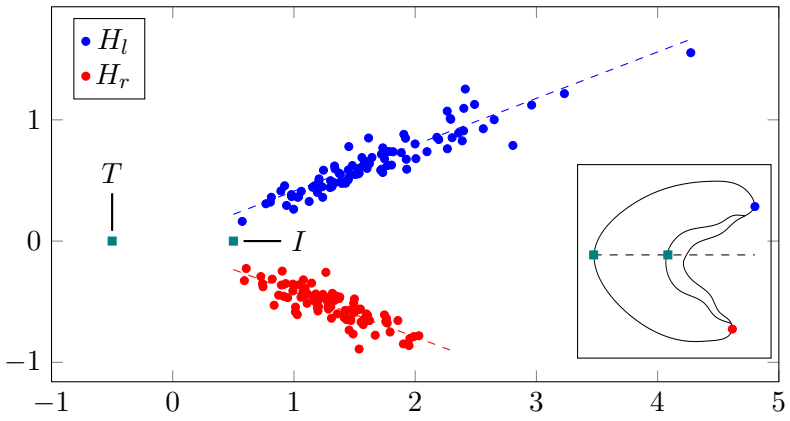
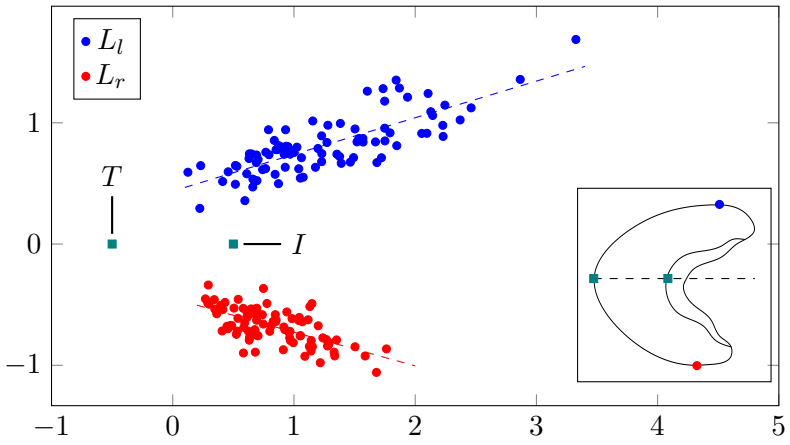


Figure 3.1: An example of how barchan asymmetry is a collection of different displacements. a) A perfectly symmetrical barchan with landmarks indicated. b) An asymmetric barchan with landmarks indicated in red. The blue circles indicate the positions of the perfectly symmetrical barchan (the toe of the barchan is treated as a common point) while the dashed arrows indicate the direction of displacement of the landmark between the two configurations. The labels correspond to regions of interest on the barchan: H_l= left horn tip, H_r= right horn tip, LE_l= most lateral extension of the barchan towards the left, and LE_r= the most lateral extension of the barchan towards the right. The vector components of the displacements are indicated by the green (x–component) and magenta (y–component) arrows. Note that the images are not in Procrustes superimposition in order to emphasize the displacement (see Section 3.4).

(a) Horn tips.



(b) Maximum lateral extent.



(c) Meeting between horns and slip face.

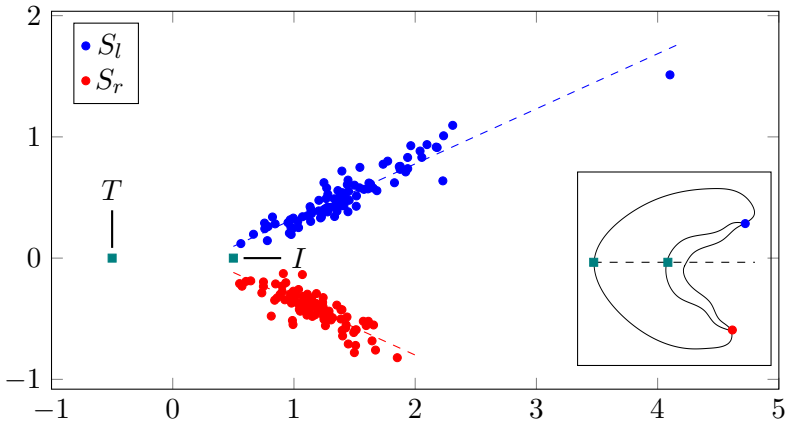


Figure 3.2: The position of selected landmarks of the dunes sampled in the Kunene region of northern Nambia. All samples are plotted using Bookstein baseline registration (Zelditch *et al.*, 2004) using the barchan toe (T) and the intersection between the brink line and the longitudinal axis (I) as registration points (squares). Dashed lines represent the results of a generalized linear model fitted to the data (cf. Table 3.1). The variables used are: H_l, H_r for the left and right horn tips respectively, L_l, L_r for the left and right points of maximum lateral extent, and S_l, S_r for the left and right points that represent the join between the slip face and the respective horns.

2014), cannot be determined using a measurement approach that relies on recording only along

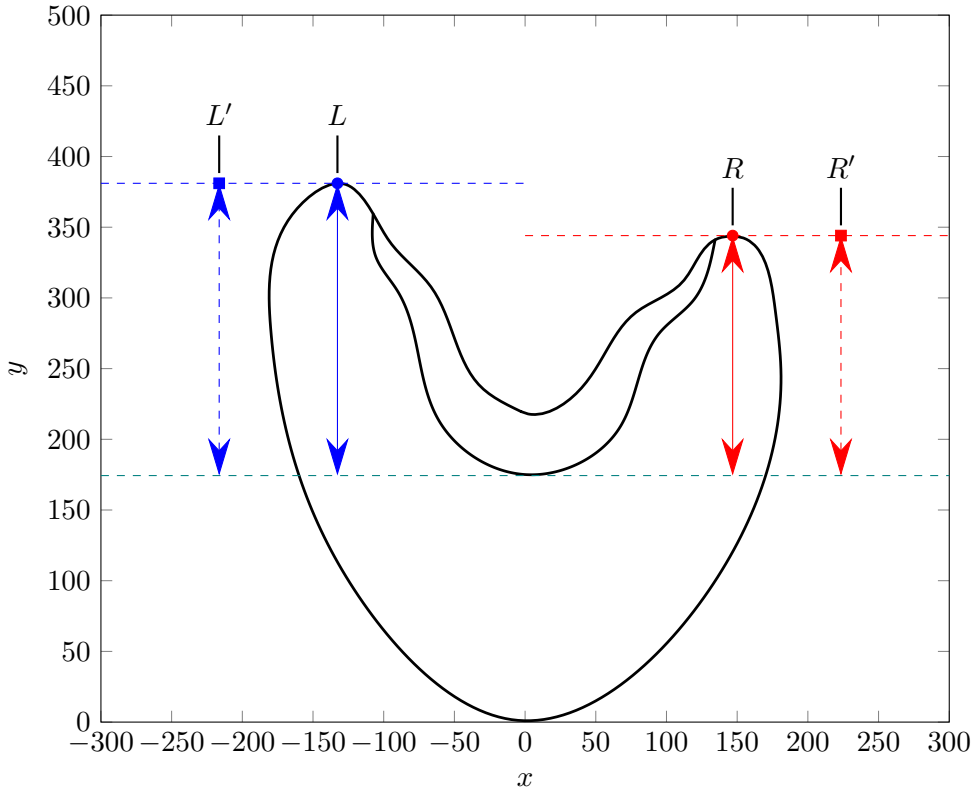


Figure 3.3: When only a single vector is used (in this case the downwind component) any asymmetry associated with the second vector component, for example lateral displacement, cannot be determined. The x and y axis represent pixel coordinates. The points L and R indicate the left and right horn tips respectively while the double-headed arrows represent distance measurements (cf. Figure 2.4). The points L' and R' represent landmarks that have been laterally displaced while still maintaining the same distance measurement.

a single vector (e.g. approaches $5a$ and $5b$ (Figure 2.4)). This is because those measurements are, in effect, made between two parallel lines.

Some approaches, such as $6b$ record a diagonal distance from an origin point to a designated landmark. However, this is also not sufficient (Figure 3.4). Because only a single distance measurement is used, a landmark can be located at any point along the boundary of a circle and still have an apparent equal measurement value. While the single downwind measurement can result in undetectable fluctuation along one axis, it is possible that this measurement style will miss displacement along two axes (i.e. both the y and x components). If the distance measurement is accompanied by a bearing then it is possible to precisely place the point. However, it will also increase the amount of data that needs to be processed. Alternatively the distances to two different landmarks may be considered, but since the position of the second landmark cannot be accurately obtained, since it is susceptible to the same constraints mentioned earlier, this is not a feasible solution.

It is therefore clear that these approaches is not sufficient to describe the positional change of a barchan landmark. An alternative approach is to quantify the positions of the landmarks with

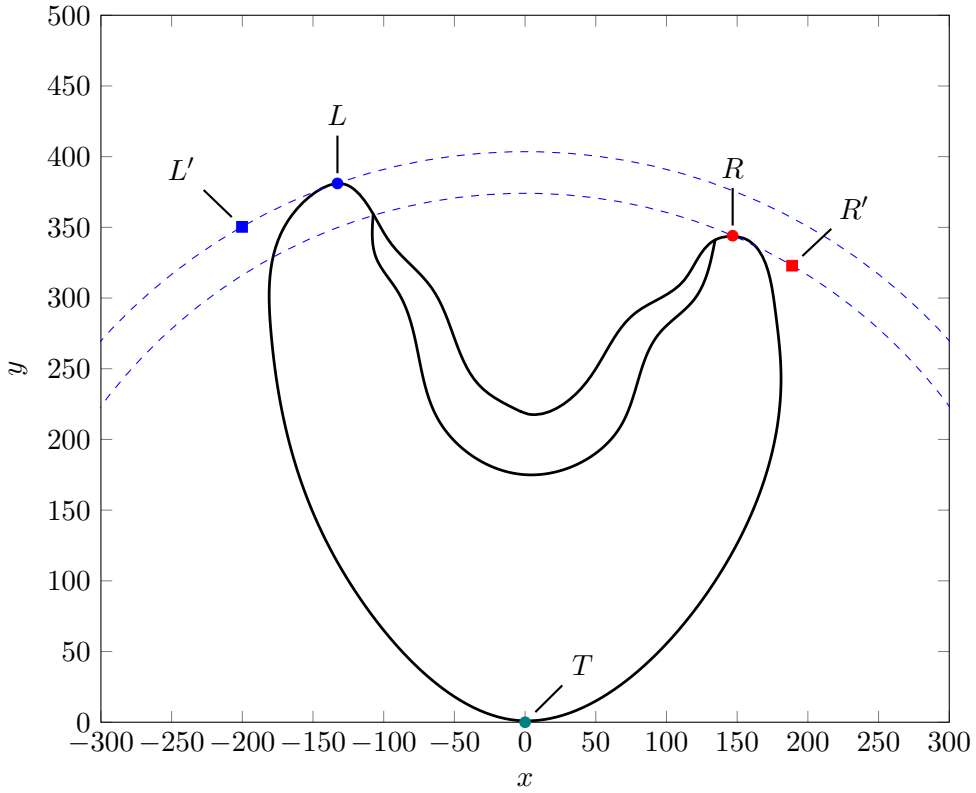


Figure 3.4: When a diagonal line is used to record the distance between two landmarks, the one landmark can be located on any point along the boundary of a circle whose radius is equal to the distance measured. The x and y axis represent pixel coordinates. The points L and R indicate the left and right horn tips respectively while the dashed circles represent their distance from the toe of the barchan (T) (cf. Figure 2.4). The points L' and R' represent landmarks that have been laterally displaced but are still the same distance away from the origin.

regards to each other. In other words, to measure the distance between a landmark and all of the other landmarks in the configuration repeated across all of the landmarks recorded. Strauss and Bookstein (1982) found that there are three types of approaches that can be used to completely describe the configuration of n landmarks: triangulation, truss, and global redundancy. These approaches rely on the measurement of the distances between pairs of landmarks. Because of this, such an approach by definition is multi-variable with one variable representing a distance between two landmarks. As the number of landmarks under consideration increases, so to do the number of variables (Figure 3.5).

This approach scales rapidly with an increase in the number of landmarks leading to an increase in the number of distance measurements. While this approach can, in principle, describe the change based solely on distance measurements, it is susceptible to the influence of size. Larger dunes will have larger distances. While size can, in principle, be controlled for by calculating ratios, it is not immediately obvious which value(s) to use as the denominator in these cases. It is also not easy to use this multi-variate data to describe the asymmetry as a single variable. Therefore, a new method had to be developed for the purposes of this study.

3.2 Conceptual Foundations

This section describes the conceptual foundations (i.e. geometric morphometrics (GM)) of the method that was developed to describe complex asymmetry. First, it is necessary to review the concept of shape from the perspective of GM. Within this framework, shape is considered as the geometric information of an object that is invariant under the following transformations: translation, scaling, and rotation (Mitteroecker *et al.*, 2013; Klingenberg, 2013; Slice, 2007; Adams *et al.*, 2004; Dryden and Mardia, 1993). This definition refers specifically to the spatial information that is contained within landmark points (Webster and Sheets, 2010; Márquez *et al.*, 2012).

These are points that can be precisely located on all forms with a clear one-to-one correspondence (Klingenberg, 2010; Viscosi and Cardini, 2011; Richtsmeier *et al.*, 2002) (e.g. the point that corresponds to the horn tip on one barchan should not be the same point that is considered the toe of another barchan). Collectively, the landmarks are referred to as a configuration. Three types of landmark are used: biological landmarks (whereby points correspond by virtue of their biological function), mathematical landmarks (where correspondence occurs via a common geometric property), and pseudo-landmarks which are placed in-between other landmarks (such as points evenly spaced between mathematical landmarks) (Dryden and Mardia, 1993). These landmarks are often assigned based on specific rules (Jensen, 2003) and highly dependent on the hypothesis being tested (Viscosi and Cardini, 2011). Further, landmarks should, ideally, be easy

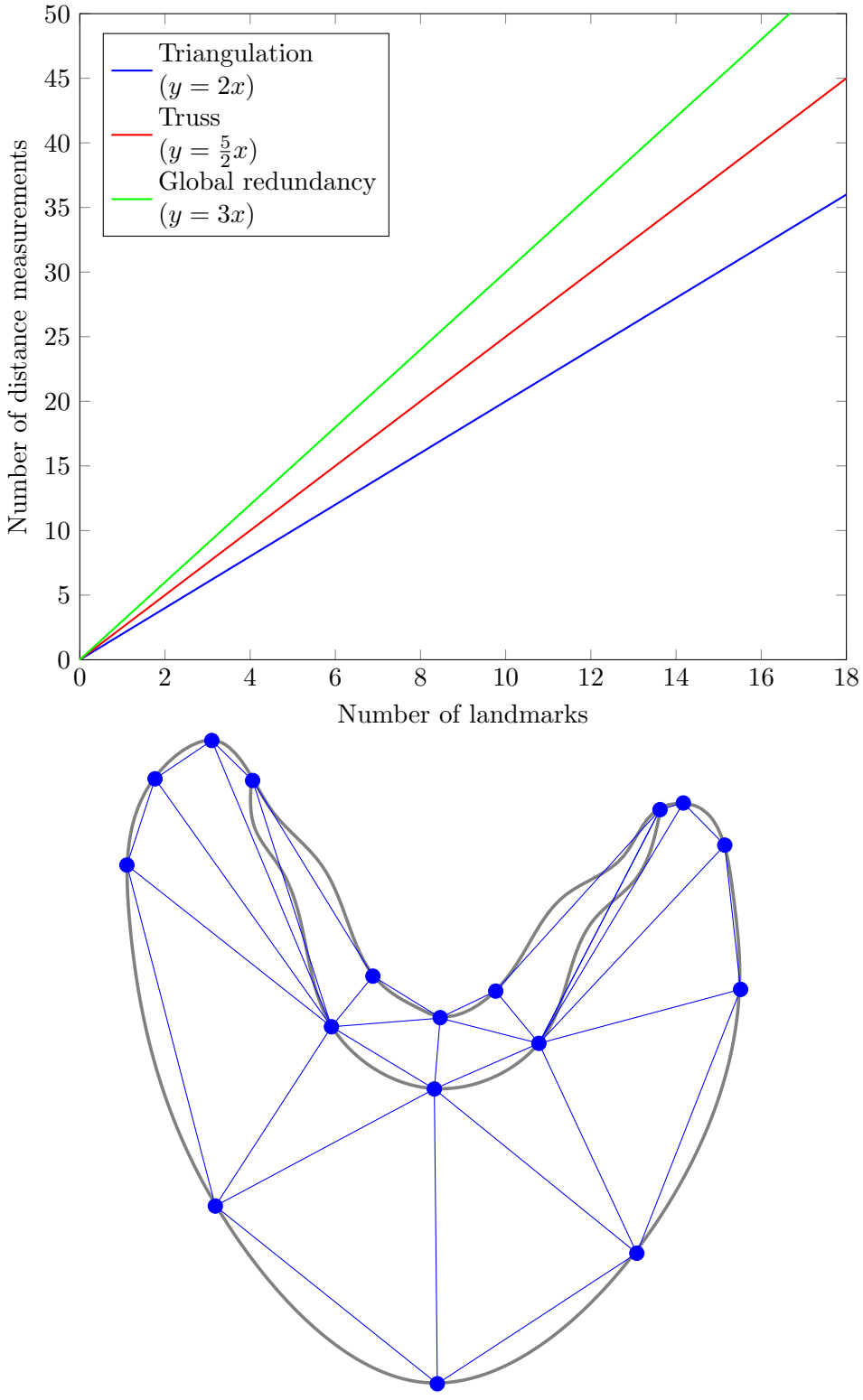


Figure 3.5: Top: The number of distance measurements needed to describe the landmark configuration as a function of the number of landmarks present. For 17 landmarks the following number of distance measurements are needed: triangulation (34), truss (43), and global redundancy (51). Bottom: An example of the triangulation approach as applied to the landmarks used in this study.

to locate and be abundant enough to adequately describe the morphology of an object (Webster and Sheets, 2010). Landmarks can either be expressed as two- or three-dimensional coordinates

(Richtsmeier *et al.*, 2002) and contain information on shape, size, position, and the orientation of an object (Mitteroecker *et al.*, 2013). Therefore, in order to use landmarks to investigate shape, the non-shape variables need to be removed.

Generalized Procrustes Analysis (GPA) removes the non-shape variables through superimposition based on a least-squares criterion (Klingenberg, 2013). In this case, the least-squares criterion is used to minimise the Euclidean distance between corresponding landmarks (Jensen, 2003; Slice, 2007; Adams *et al.*, 2004). This is achieved via three congruent transformations of the matrix containing the landmark coordinates. These transformations are scaling, translation, and scaling mentioned as part of the shape definition earlier. During translation, the centroids of the respective configurations are moved to the same point (Mitteroecker *et al.*, 2013) which is typically the origin (i.e. they are assigned an x,y coordinate of $(0, 0)$(Adams *et al.*, 2004). Afterwards, the configurations are scaled to a common unit size by dividing with the centroid size (Adams *et al.*, 2004). The centroid size is the square root of the sum of squared distances from each landmark to the centroid of the configuration (Mitteroecker and Gunz, 2009; Zelditch *et al.*, 2004). In effect, this serves to scale all of the configurations so that they have a centroid size of 1.0 (Klingenberg, 2013). Lastly, the configurations are rotated into an optimal position by minimising the squared differences between corresponding landmarks (Adams *et al.*, 2004; Zelditch *et al.*, 2004). At this point, the configurations are in superimposition.

Once in superimposition, the shape difference is quantified by the Procrustes distance which is the squared distances between landmarks (Mitteroecker *et al.*, 2013; Webster and Sheets, 2010; Adams *et al.*, 2004). If two objects are identical in shape, then their respective configurations will have the exact same landmark coordinates once in superimposition (Dryden and Mardia, 1993). This results in a Procrustes distance of 0 with increasing values reflecting larger deviations from similarity (Mitteroecker *et al.*, 2013).

Using GM it is possible to avoid the constraints on describing the displacements of landmarks mentioned earlier. Any point on a plane can be represented by two coordinates (x,y). As discussed earlier, the displacement of landmarks, on a plane, include both an x–component vector and a y–component vector. When these component vectors are added to the coordinates of a landmark (x_1, y_1) and new coordinate is created (x_2, y_2) where the changes to the coordinates correspond to the vectors. The new landmark still exists in the plane and as such has a coordinate. Therefore, following from the previous discussion, it contains information about the shape (in this case the new shape). It is thus possible to use the information contained within the landmark to investigate shape and, since asymmetry is just an attribute of shape, it is possible to investigate asymmetry. The following section elaborates on this in more detail.

3.3 Proposed Solution

This section describes how the more complex asymmetry is quantified using a new measure: "bilateral asymmetry". This nomenclature is intended to reflect the bilateral nature of barchans along their longitudinal axis. The section starts with a discussion on how landmark points are to be determined before proceeding to a discussion on how GM can be used to quantify the level of asymmetry.

3.3.1 Identification of Barchan Landmarks

This section describes the approach used to identify landmarks on the outlines of barchan dunes. This is a required step in order to generate the labels that are to be used to train the CNNs. As stated earlier, a key requirement of geometric morphometrics is that landmarks need be located accurately and consistently. A key challenge, in this regard, is assigning landmarks to curved surfaces (Slice, 2007). As mentioned previously, landmarks need to be placed consistently between different specimens in order for the shape comparison to be valid. This is relatively easy for points of maximum curvature or where two different portions of an object join. However, on a curve this becomes more complicated since it cannot be ensured that the landmarks will be placed consistently. Ros *et al.* (2014) made use of sliding semi–landmarks[2] spaced equally between two landmarks in order to capture the shape information of a curve. Semi–landmarks, by definition, contains information dependent on the position of another landmark (Claes *et al.*, 2012). The "sliding" aspect of the semi–landmarks used by Ros *et al.* (2014) refers to landmarks whose positions are allowed to change, based on a specified criterion, during the analysis process (Slice, 2007). However, for the purposes of this study such a feature is not considered desirable since it cannot be guaranteed that the positioning of the landmarks will be consistent between dunes. This would, therefore, introduce a variable that cannot be sufficiently controlled for.

A different strategy based on mathematical landmarks (Dryden and Mardia, 1993). These landamrks are placed based on a geometric criterion and are suitable for the analysis of outlines (Dryden and Mardia, 1993). Barchan landmarks were placed on each outline as indicated in figure 3.6. The benefit of using landmarks in this analysis is that it allows for all of the desired morphometric variables to be calculated using coordinate geometry[3]. Seventeen to 18 landmarks were used per barchan. The reason for the discrepancy is due to the calculation of the $\frac{a}{c}$–ratio. The c value requires a distance to be calculated between the two of the barchan and the brink line along the longitudinal axis. For some dunes, this point coincided with the position of landmark

[2]A semi–landmark is a landmark that is intended to capture information about an object's curve (Webster and Sheets, 2010).

[3]Coordinate geometry involves the study of geometry by making use of the Cartesian coordinates.

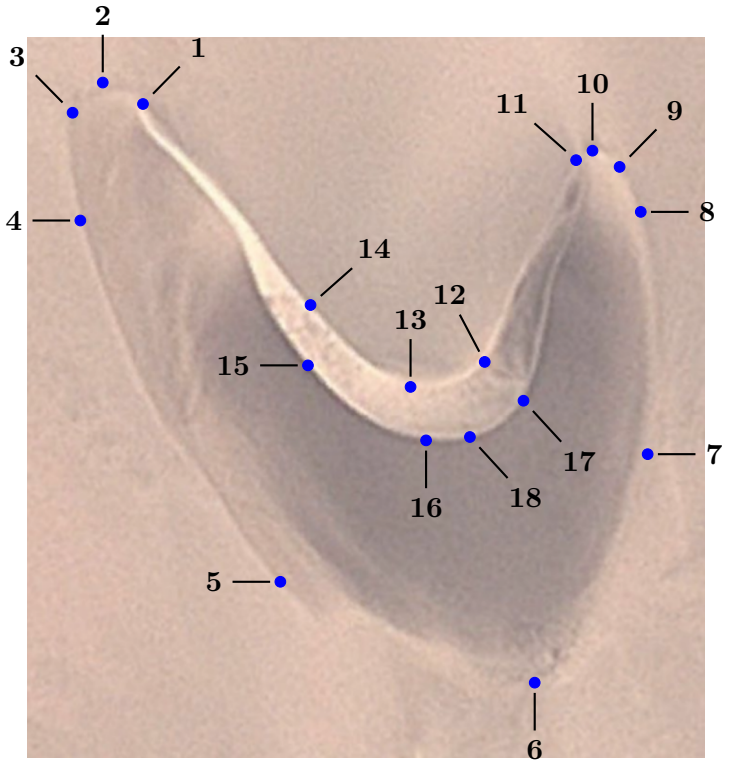


Figure 3.6: Landmarks used in the description of barchan morphology. Note that landmark 18 was not needed in all cases since, for some barchans, landmark 18 and 16 were in the same position.

16 and, in those cases, landmark 16 was used in the calculation. In cases were this was not the case, the 18^{th} landmark was added.

Some points along the barchan are easily identifiable such as the tips of the horns, the toe and the meeting point between the stoss slope and the horns. Other points located along the curved portion are harder to identify consistently between specimens using unguided visual assessment. To place landmarks consistently at these points two assumptions need to be made. First, it is assumed that the boundary of a barchan can be represented by a polynomial curve (the order of this is not relevant but has to be ≥ 2). This is not new since barchan dune boundaries have been broken into separate parabolic curves by earlier authors (Sauermann *et al.*, 2003; Moosavi *et al.*, 2014). The second assumption is that this curve is differentiable and continuous. With these two assumptions, the problem of consistently identifying the positions of the landmarks can be reduced to finding the point along the curve where the first derivative of the function is zero. To illustrate this, consider the curve in Figure 3.7. Here, a portion of a barchan is represented by a polynomial function (Eq. 3.1). Using the first derivative of this function (Eq. 3.2) the minima of this function can be calculated (Eq. 3.3).

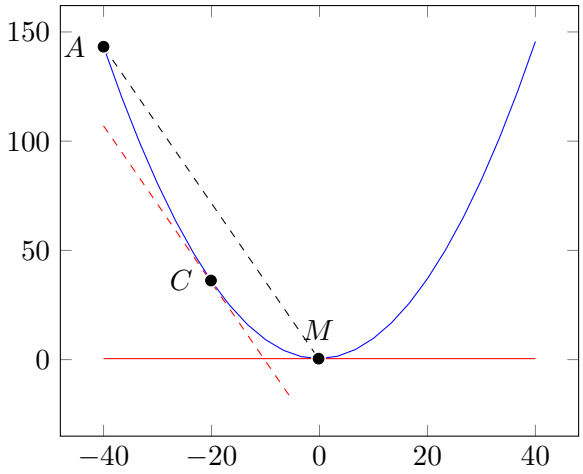


Figure 3.7: Identifying the points of curvature on a barchan boundary. The plotted function is $f(x) = 0.09x^2 + 0.03x + 0.44$. The minimum of $f(x)$is indicated by point M. A line (red) through this point is parallel to the x–axis. The equation of the red dashed line is $g(x) = -3.59x - 36.562$ and is parallel to AM and tangent to $f(x)$ at C.

$$f(x) = 0.09x^2 + 0.03x + 0.44 \quad \text{(Eq. 3.1)}$$

$$f'(x) = 0.18x + 0.03 \quad \text{(Eq. 3.2)}$$

$$0 = 0.18x + 0.03$$

$$0.18x = -0.03 \quad \text{(Eq. 3.3)}$$

$$x = -0.17$$

A property of a line through this minimum is that it is parallel to the x–axis. This makes it possible to calculate points of curvature along the given curve between two points. For example, to calculate the point of maximum curvature (i.e. a maxima) along the provided function between points $A = (-40, 143.24)$ and $M = (-0.17, 0.44)$, the function of the line through these points (Eq. 3.4) needs to be calculated. Afterwards, the line parallel to the line AM can be calculated that is also a tangent line to the function $f(x)$ (Eq. 3.5). The point of tangency can then also be calculated. This process can then be repeated between successive points until the desired number of points are obtained.

$$y = -3.59x - 0.1703 \quad \text{(Eq. 3.4)}$$

$$g(x) = -3.59x - 36.562 \quad \text{(Eq. 3.5)}$$

Although this process does provide a consistent means of identifying points along a smooth curve of a barchan, the function which describes a barchan's outline cannot be known *a priori*. Therefore, a different technique was applied to the barchan outlines. As stated earlier, one of the assumptions of this approach is that the boundary (i.e. the outline) of a barchan is continuous and differentiable. Hesp and Hastings (1998) discuss that barchans are aerodynamically maintained. This, in itself, suggests that barchans must have a smooth and continuous curvature since: it must exist at all points along its curve (i.e. it is continuous), and any properties that would make it non-differentiable (such as sharp cusps marking drastic changes in direction) would be smoothed out. This latter aspect results from the granular nature of a dune which does not afford substantial resistance, in the way that bedrock does, to the action of winds. Further support for this assumption is provided by figure 3.8 which shows how different parts of the barchan's boundary can be represented by differentiable polynomial functions. Each of these polynomials is a good generalizations of the barchan's boundary (based on the R^2 values).

Using this information, it is possible to follow a geometric approach to identifying landmarks (Figure 3.9 and Table 3.2). It is based on the fact that the line through either the minima or maxima of a function is parallel to the x–axis. Starting from the identification of landmarks that can be assigned visually (the toe (landmark 6), the horn tips (landmarks 2 and 10), and the meeting point between the slip face and the horns (landmarks 1 and 11)), the remaining landmarks can be determined by refining the x–axis locally to exist between each of the known points. For example, by redefining the z–axis to exist between landmarks 1 and 11 the point of maximum curvature (i.e. landmark 13) is the point where a line, parallel to a line connecting landmarks 1 and 11, is tangent to the barchan outline. A full description of the process is given in Table 3.2 and illustrated in Figure 3.9.

3.3.2 Bilateral Asymmetry

As mentioned earlier, bilateral asymmetry makes it possible to evaluate more complex changes in barchan shape. Therefore, it is possible to quantify the combined effect of more than one manifestation of asymmetry. There is no method currently in use within barchan research that can combine these values, aside from using a form of multi-variable analysis which becomes increasingly complex the more changes being considered. However, it is possible to use the principles of geometric morphometrics to quantify this combined effect.

Geometric morphometrics treats shape as a configuration of points, termed landmarks (Webster and Sheets, 2010; Mitteroecker *et al.*, 2013; Márquez *et al.*, 2012). These landmarks are points that need to be located precisely on all specimens and there must be a one-to-one correspondence for all landmarks between specimens (Klingenberg, 2010). In other words, the toe of

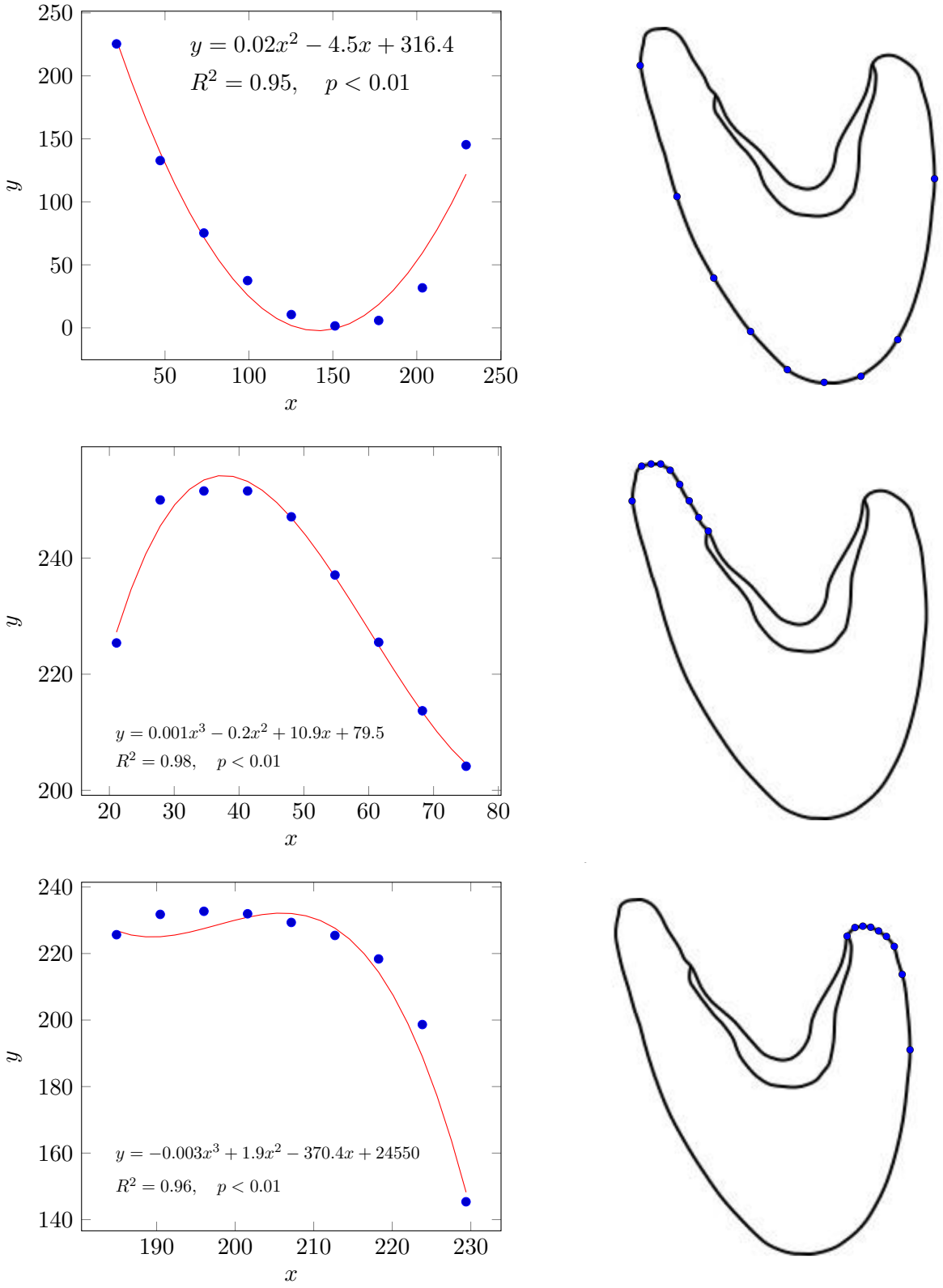


Figure 3.8: Continuous polynomial functions, along with R^2 and significance values, fitted to the different portions of a barchan dune. The x and y axes refer to measurements on the image with 1 unit representing approximately 0.08mm.

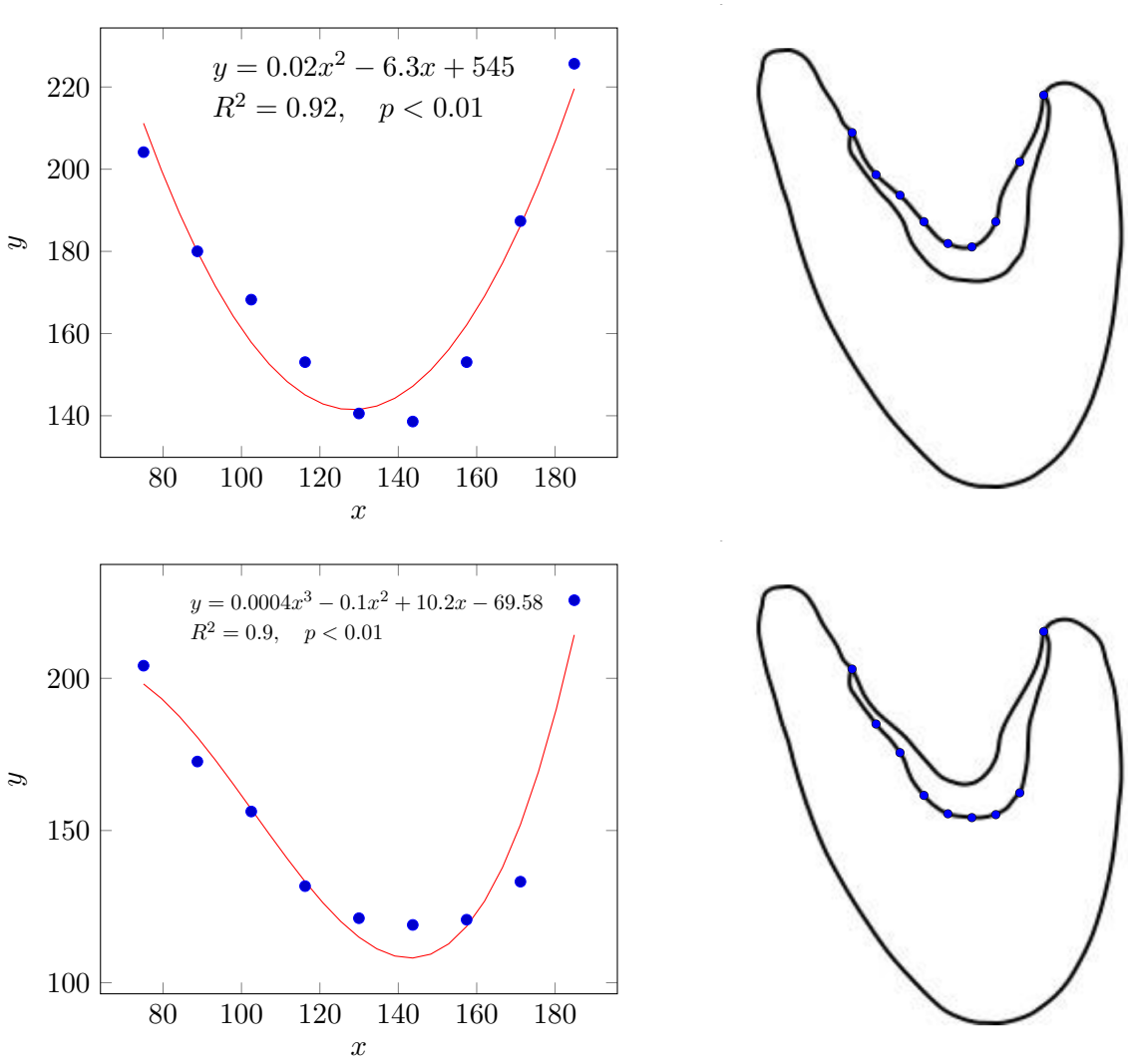


Figure 3.8: *Cont.*

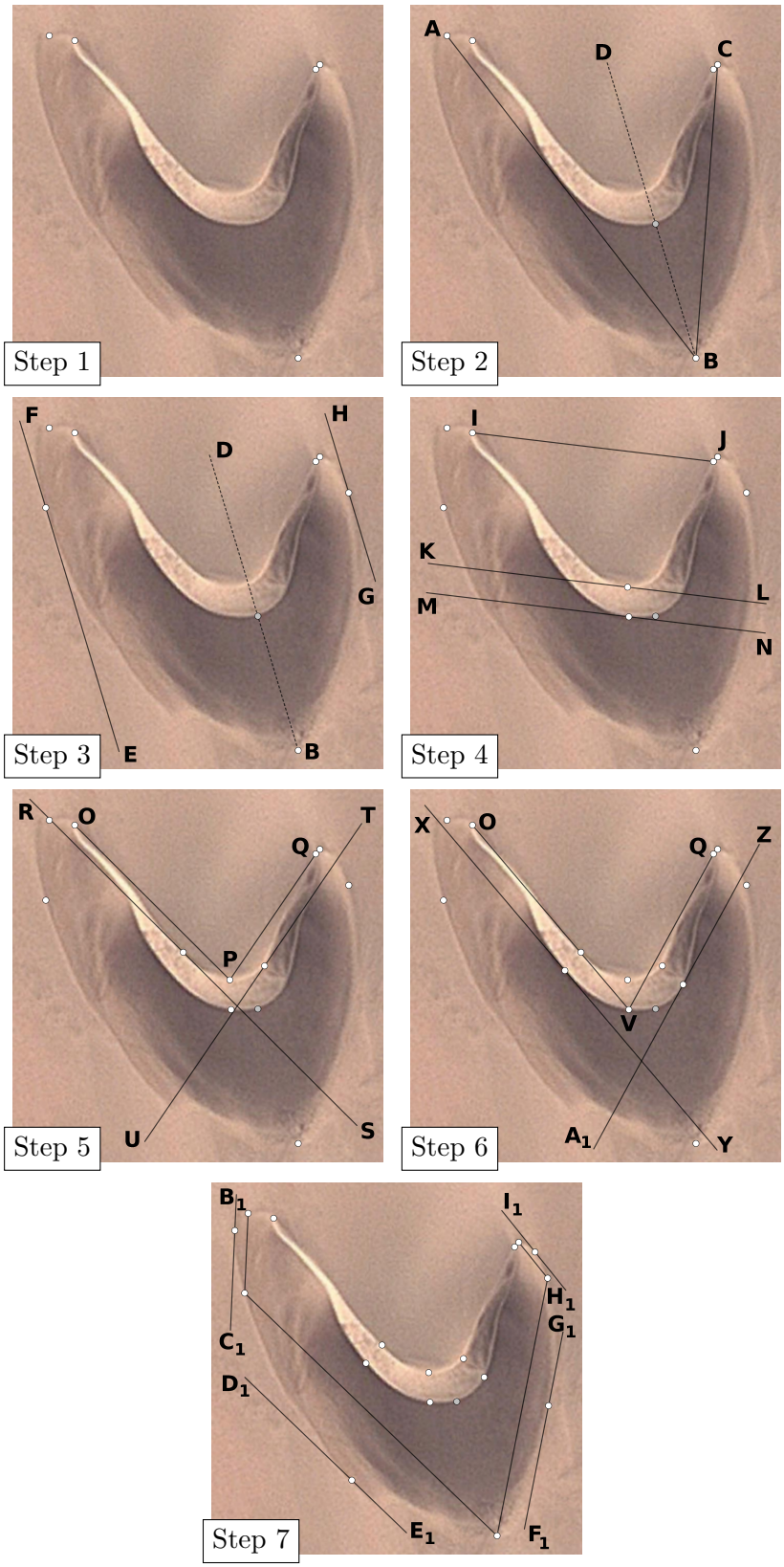


Figure 3.9: The steps involved in assigning landmarks to a barchan. See Table 3.2 for a description of each step. This was carried out on barchan outlines and is shown here on a satellite image to provide some additional context.

Table 3.2: A description of the steps involved in assigning landmarks to the barchan's boundary (see Figures 3.9 and 3.6).

Step	Description
1	Place landmark 6 at the toe of the barchan, 2 and 10 at the horn tips, and landmarks 1 and 11 at the points where the base of the slip face intersects the horns.
2	Construct lines $\overline{AB}, \overline{CB}$ connecting the horn tips to the toe. Construct line $\overline{DB}$ which bisects the angle $\angle DBC$. This line corresponds to the longitudinal axis of Finkel (1959). Add landmark 18 at the point where the $\overline{DB}$ intersects the brink line.
3	Construct $\overline{EF}$ and $\overline{GH}$ parallel to $\overline{DB}$ and tangent to the barchan's boundary. Add landmarks 4 and 8 at the tangent points.
4	Construct line $\overline{IJ}$ connecting landmarks 1 and 11. Construct line $\overline{KL}$ parallel to $\overline{IJ}$ and tangent to the base of the slip face. Add landmark 13 to the tangent point. Construct $\overline{MN}$ parallel to $\overline{IJ}$ and tangent to the brink line. Add landmark 16 to the tangent point.
5	Construct lines $\overline{OP}$ and $\overline{QP}$ that connects landmark 13 to landmarks 1 and 11. Construct $\overline{RS}$ parallel to $\overline{OP}$, and $\overline{TU}$ parallel to $\overline{QP}$, both tangent to the base of the slip face. At landmarks 14 and 12 to the points of tangency.
6	Construct $\overline{OV}$ and $\overline{QV}$ connecting landmarks $1, 16$, and 11. Construct $\overline{XY}$ parallel to $\overline{OV}$, and $\overline{ZA_1}$ parallel to $\overline{QV}$. Add landmarks 15 and 16 where the lines are tangent to the brink line.
7	Finally, connect the landmark pairs $(2,4), (4,6), (6,8)$, and $(8,10)$ with segments and construct the lines $\overline{B_1C_1}, \overline{D_1E_1}, \overline{F_1G_1}$, and $\overline{H_1I_1}$ parallel to each segment and tangent to the outer boundary of the barchan. Add landmarks $3, 5, 7$, and 9 to the tangent points.

the dune must be at the same position on the dune for all specimens. Because of the manner in which each of the landmarks are defined (Table 3.2) this requirement can be easily met.

Once the landmarks have been identified, the shape of each object is then expressed in the form of a $k \times m$ matrix (Eq. 3.6) where k is the number of landmarks and m the number of dimensions used to express the location of the landmark. In other words, for the two dimensional case (which is followed here) the number of dimensions is two which corresponds to the xy–plane. However, in addition to the shape information, this matrix also contains additional non–shape information such as the position, orientation, and size of the object within the xy–plane. These non–shape variables need to be removed from the coordinates in order to extract the shape information. Three steps are used to achieve this (Zelditch *et al.*, 2004): translation to remove the influence of position, rotation to remove the influence of orientation, and scaling to remove the influence of size. The shape of an object, therefore, refers to the geometric information contained within the collection of landmarks that are invariant to the influences of translation, scaling, and orientation (Slice, 2007; Kendall, 1977; Dryden and Mardia, 1993). An example of this process is provided in Section 3.4.

$$\begin{bmatrix} 176 & 619 \\ 182 & 629 \\ 158 & 614 \\ 125 & 584 \\ 11 & 379 \\ 18 & 283 \\ 75 & 282 \\ 156 & 341 \\ 158 & 350 \\ 156 & 357 \\ 140 & 359 \\ 89 & 350 \\ 66 & 375 \\ 144 & 579 \\ 137 & 578 \\ 56 & 381 \\ 80 & 335 \end{bmatrix} \quad \text{(Eq. 3.6)}$$

Two objects are considered to have the same shape if these landmarks can be rotated, scaled, and translated in such a way as to match exactly (Mitteroecker *et al.*, 2013). A popular method for carrying out this shape comparison is Generalized Procrustes Analysis (GPA) which is described extensively in the literature (Zelditch *et al.*, 2004; Mitteroecker *et al.*, 2013; Adams and Otárola-Castillo, 2013; Klingenberg and McIntyre, 1998; Klingenberg, 2010; Slice, 2007). In GPA, once the shape information of an object has been extracted, the landmarks of each configuration is reduced to a single point that is embedded on the surface of a hypersphere (Zelditch *et al.*, 2004), which is termed "shape space" (Klingenberg, 2010; Mitteroecker *et al.*, 2013). The difference in shape can then be calculated as the partial Procrustes distance (ρ_p) between these points (Webster and Sheets, 2010; Mitteroecker *et al.*, 2013). This can then be further refined to the full Procrustes distance (ρ) (Zelditch *et al.*, 2004) which is the metric used in this study. Distances that are 0 occur only for shapes that are identical and values larger than 0 represents increasing differences in shape (Mitteroecker *et al.*, 2013). An example of ρ values are given in Figure 3.10 using quadrilaterals as an example.

The ρ value can also be used to express asymmetry by creating a mirror image of the object and calculating the ρ between the two configurations (Klingenberg and McIntyre, 1998).

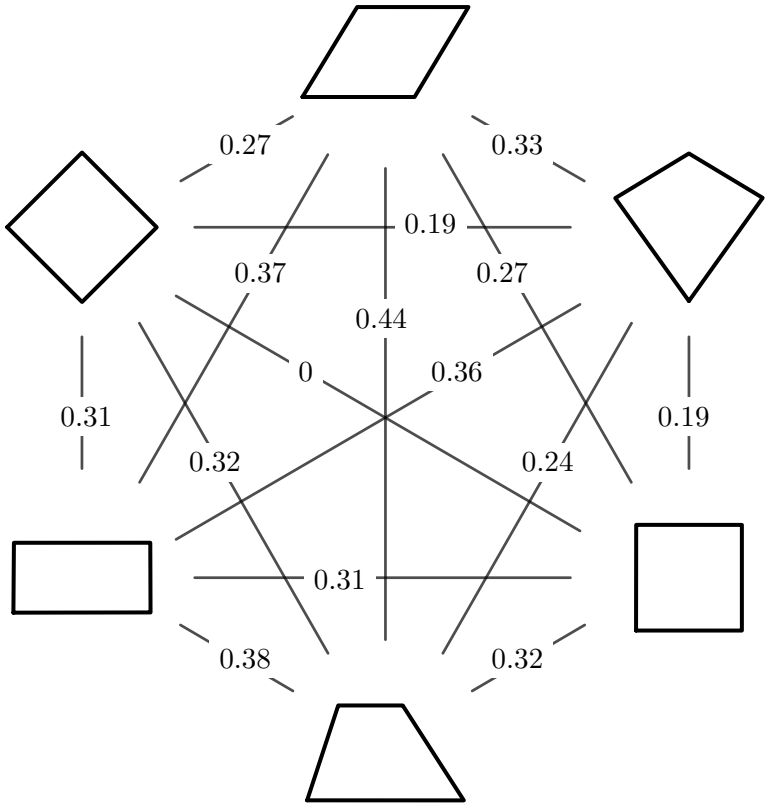


Figure 3.10: The full Procrustes distance (ρ) between quadrilaterals following GPA.

Following this approach, the ρ between each barchan and its mirror image (along the y-axis) was calculated using the assigned landmarks. A perfectly symmetrical object will, when flipped across the y-axis, have a $\rho = 0$ and as ρ becomes larger, it indicates increasing levels of asymmetry between the object and its mirror image. Because the mirror image of the barchan in figure 3.6 would have the landmark A on the right horn instead of the left horn as in the original configuration. A one-to-one correspondence does not exist in such a situation requiring the relabeling of the landmarks (Figure 3.11).

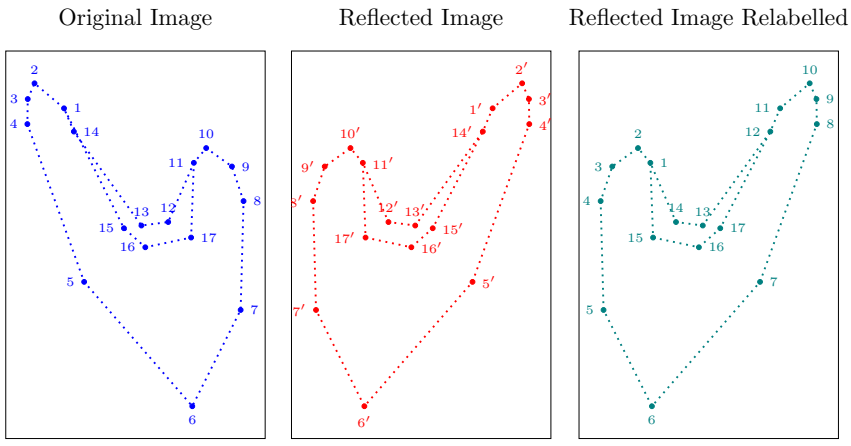


Figure 3.11: An example of creating a mirror image of a barchan and relabeling the landmarks prior to analysis. The uncorrected labels are indicated by l' where l represents the landmark number.

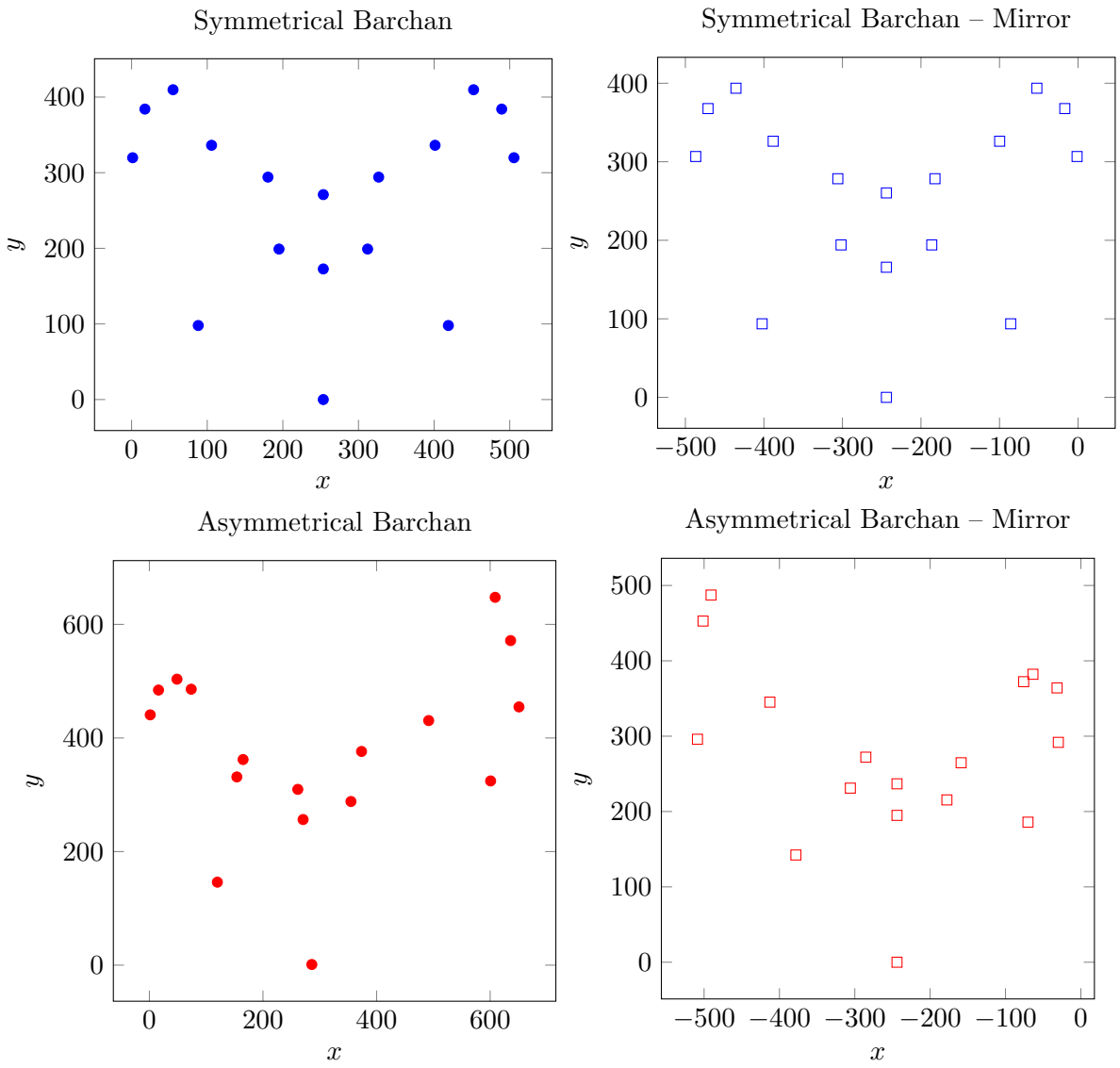


Figure 3.12: Hypothetical symmetrical and asymmetrical barchans along with their mirror images. For this example, the y–axis is used as the line of reflection (cf. Figure 3.11).

3.4 Worked Example

The majority of this section is based on the calculations provided in Zelditch *et al.* (2004) and the analysis is carried out using the *shapes* package from R (Dryden and Mardia, 2016). To illustrate how bilateral asymmetry is determined using GM, the hypothetical[4] dunes created in figure 3.1 are used.

The landmark information for each barchan is contained in a 17×2 matrix since there are 17 landmarks whose coordinates consist only of an xy position wihtin the plane. An example of such a matrix (for the asymmetrical barchan $\mathbf{A}$ and its mirror $\mathbf{M}$) is given in (Eq. 3.7). It is

[4]By hypothetical it is meant barchans that were generated computationally. This is done in order to simplify the example and draw emphasis to the differing results.

expected that these matrices would have different values since the barchans differ in position[5]. During Procrustes Superimposition these non-shape variable are removed through translation, rotation, and scaling (Klingenberg, 2013). This converts the "normal" spatial coordinates into shape variables (Adams *et al.*, 2004). Although there is no need to remove the influence of scale and orientation in this example, the underlying approach is still discussed for the sake of completeness.

$$\mathbf{A} = \begin{bmatrix} 75.92 & 372.31 \\ 63.60 & 382.16 \\ 31.57 & 364.09 \\ 29.92 & 291.81 \\ 70.17 & 185.85 \\ 244.00 & 0.00 \\ 378.19 & 142.32 \\ 508.79 & 295.92 \\ 501.40 & 452.80 \\ 490.72 & 487.30 \\ 412.69 & 345.20 \\ 285.37 & 272.10 \\ 244.00 & 236.78 \\ 158.88 & 264.71 \\ 177.77 & 215.42 \\ 244.00 & 194.89 \\ 305.91 & 231.03 \end{bmatrix} ; \mathbf{M} = \begin{bmatrix} -412.69 & 345.20 \\ -490.72 & 487.30 \\ -501.40 & 452.80 \\ -508.79 & 295.92 \\ -378.19 & 142.32 \\ -244.00 & 0.00 \\ -70.17 & 185.85 \\ -29.92 & 291.81 \\ -31.57 & 364.09 \\ -63.60 & 382.16 \\ -75.92 & 372.31 \\ -158.88 & 264.71 \\ -244.00 & 236.78 \\ -285.37 & 272.10 \\ -305.91 & 231.03 \\ -244.00 & 194.89 \\ -177.77 & 215.42 \end{bmatrix} \quad \text{(Eq. 3.7)}$$

3.4.1 Translation

In translation the barchans are "moved" so that their centroids correspond to the same point. Therefore, it is necessary to calculate the current centroids of each barchan. The centroids are calculated using (Eq. 3.8). The calculated centroid for **B** is $(113, 435)$ and for **R** is $(212.6, 95.1)$.

[5] In this instance orientation and size are not different since the one configuration is just a mirror image of the other.

$$x_c = \frac{1}{k}\sum_{j=1}^{k} x_j$$
$$y_c = \frac{1}{k}\sum_{j=1}^{k} y_j \qquad \text{(Eq. 3.8)}$$

Translation involves ensuring that the centroids have a coordinate of $(0, 0)$. Therefore, the calculated centroid for each configuration needs to substracted from the landmarks ((Eq. 3.9)). By "moving" the barchans to a common reference point (i.e. having their centroids in the same position) the effect of differing positions within a plane is removed.

$$\mathbf{T} = \begin{bmatrix} (x_1 - x_c) & (y_1 - y_c) \\ (x_2 - x_c) & (y_2 - y_c) \\ \vdots & \vdots \\ (x_k - x_c) & (y_k - y_c) \end{bmatrix} \qquad \text{(Eq. 3.9)}$$

3.4.2 Scaling

After the barchans have been translated to a common point, it is necessary to scale them to a uniform size. The scaling operation involves calculating the centroid size of each barchan landmark configuration ((Eq. 3.10)) where $\mathbf{X}$ is the matrix of landmark points and c the centroid $(0, 0)$. In order to ensure that the barchans are of the same size, each element within matrix is divided by the corresponding centroid size.

$$C = \sqrt{\sum_{i=1}^{k}\sum_{j=1}^{m}(\mathbf{X}_{ij} - c_j)^2} \qquad \text{(Eq. 3.10)}$$

3.4.3 Rotation

Lastly, the influence of orientation needs to be removed. This is achieved by rotating the configurations in order to minimize the square error between each of the corresponding landmarks (i.e. landmark 1 from $\mathbf{A}$ and $\mathbf{M}$). This rotation is carried out by choosing one configuration as the reference and the other as the target. The angle (θ) by which the target should be rotated in order to minimise the squared difference between the landmarks is given by (Eq. 3.11) where x_{Rj} and x_{Tj} refers to the x coordinate of the reference and target configuration respectively and y_{Rj} and y_{Tj} refers to the y coordinate. The adjustment that needs to be made to the target matrix is given by (Eq. 3.12).

$$\theta = \arctan\left(\frac{\sum_{j=1}^{k} y_{Rj}x_{Tj} - x_{Rj}y_{Tj}}{\sum_{j=1}^{k} y_{Rj}x_{Tj} + x_{Rj}y_{Tj}}\right) \quad \text{(Eq. 3.11)}$$

$$(x\cos\theta - y\sin\theta), (x\sin\theta + y\cos\theta) \quad \text{(Eq. 3.12)}$$

3.4.4 Superimposition

Once all of the transformations have been carried out, the configurations are said to be in Procrustes Superimposition. An example of this is given in Figure 3.13. The resultant configurations now contain only the shape variable since the influence of position, size, and orientation have already been removed. Note that, for the symmetrical barchan, the landmarks are placed directly on top of each other. This indicates that the two configurations, the symmetrical barchan and its mirror, can be considered to be the same shape (Dryden and Mardia, 1993; Mitteroecker *et al.*, 2013). This is not suprising since the barchan is completely symmetrical along its longitudinal axis and, therefore, its mirror image should be the same shape. In contrast, the landmarks of the asymmetrical barchan do not line up perfectly with those of its mirror image, indicating that they are not the same shape. Following from Klingenberg and McIntyre (1998) this indicates that the reference configuration (i.e. the asymmetrical barchan) is indeed asymmetrical.

3.4.5 Procrustes Distance

The partial procrustes distance (ρ_p) is a measure of how well the two configurations are aligned and is calculated using (Eq. 3.13). The full procrustes distance (ρ) can then be calculated ((Eq. 3.14)) which serves as an indication of the level of similarity between the two configurations and, by extension, the two barchans. For the symmetrical barchan, the full Procrustes distance between it and its mirror is 0, while for the asymmetrical barchan it is 0.31. This confirms the conclusions drawn following superimposition and also quantifies the extent of asymmetry as a single continuous variable.

$$\rho_p = \sqrt{\sum_{j=1}^{k}((x_{jR} - x_{jT})^2 + (y_{jR} - y_{jT})^2)} \quad \text{(Eq. 3.13)}$$

$$\rho = 2\arcsin\left(\frac{\rho_p}{2}\right) \quad \text{(Eq. 3.14)}$$

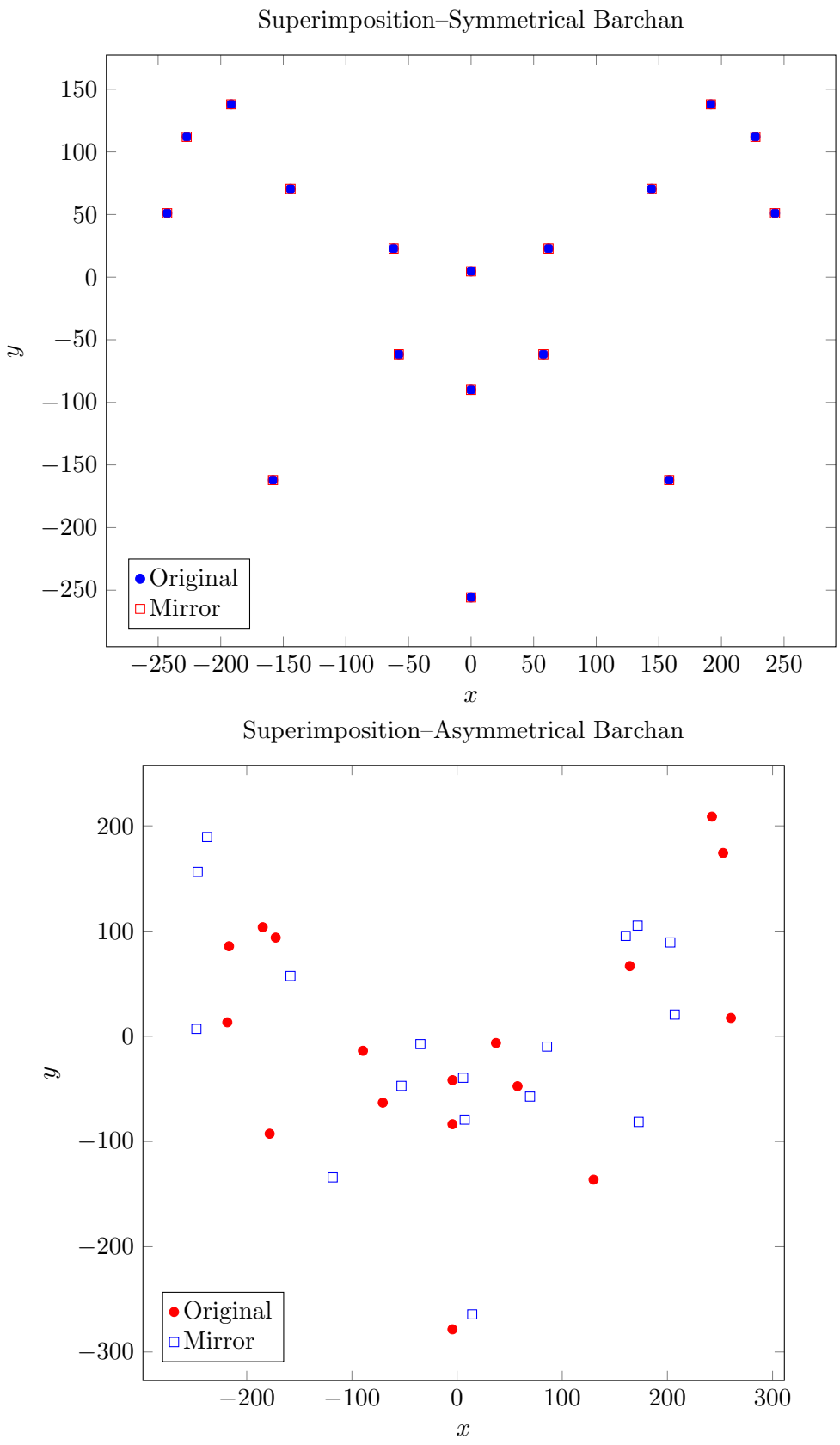
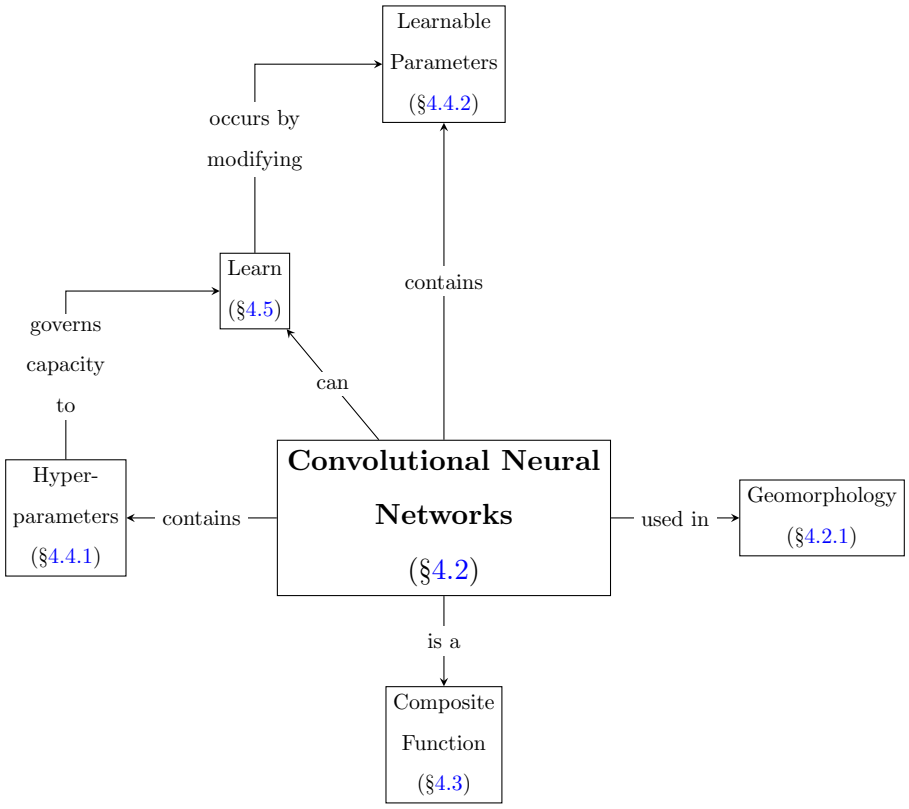


Figure 3.13: Top: The superimposition of the symmetrical barchan with its mirror image. Bottom: The superimposition of the asymmetrical barchan with its mirror image.

Chapter 4

Convolutional Neural Networks

4.1 Chapter Overview



Convolutional Neural Networks (CNNs) are a specific type of machine learning model that is specifically designed to classify images into categories. It is a composite mathematical function that consists of one or more stages, depending on the specifications contained within the hyperparameters, of image processing. This processing extracts key features within the image where a feature is a specific arrangement of pixels, such as straight and curved lines. These features, which form part of a feature map, are used to classify the image into two or more predefined categories. Since a CNN is a machine learning model, it is capable of updating some

of its internal values (i.e. its learnable parameters) based on exposure to known examples of the categories. Because of its ability to process image data, and its ability to learn features that are specific to that image, it is ideally suited to interpret images of barchan outlines.

This chapter provides an overview of the components of a CNN and how the model transforms data and improves its performance through learning. Since the study focuses on the use of CNNs to classify barchan outlines, it is important to gain an understanding into the functioning of a CNN. Additionally, it is only by understanding how a CNN applies image transformations that the choice of removing texture information, as part of the image preparation process (§1.5), can be contextualised.

4.2 Overview of CNNs

CNNs are neural networks inspired by human vision that contain convolution operations (Cady, 2017; Niepert *et al.*, 2016). However, it is believed that they differ from human vision in a key aspect. Baker *et al.* (2020) considers CNNs to rely more on object texture when making a classification as opposed to the human vision relying more on shape. This reliance on shape has also been stated by Geirhos *et al.* (2019). This reliance on texture has led to the generalization capabilities of CNNs to be questioned (Nguyen *et al.*, 2015). Such caution is understandable since CNNs have been found to misclassify images where the texture information has been altered to a level imperceptible to humans (Carrara *et al.*, 2018; De Cesarei *et al.*, 2021).

In the same way that the nervous system of animals consists of interconnected cells (Floreano and Mattiussi, 2008), neural networks consist of interconnected neurons that transfer data through the model. As such, they are attempts at imitating the input-output operations of biological networks (Clarke *et al.*, 2009; Forouzan and Mosharraf, 2012). By input-output it is meant that, in the same way an organism receives input from its environment which leads to an action, a neural network receives input from its environment which leads to a specific action (e.g. classifying, recognising etc.). The reason why this is so useful, is the adaptability that biological systems demonstrate when encountering a new stimulus. This is achieved through the process of learning whereby an organism can improve upon itself by acquiring new skills during its lifetime (Floreano and Mattiussi, 2008). Similarly, a neural network improves its performance, through learning, by being exposed to new data (Jordan and Mitchell, 2015). In biology, learning is believed to occur at the synaptic level rather than alterations to the physical structure of the nervous system (Ertel, 2011; Floreano and Mattiussi, 2008). The strength of the synaptic connections between neurons in the nervous system dictate the level of interaction between the neurons and contain the "intelligence" of the system (Rodvold *et al.*, 2001). These neurons also follow Hebb's rule (Floreano and Mattiussi, 2008) which states that when neuron

A continuously excites neuron B, a metabolic change can occur whereby the efficiency of A to excite B is increased (Hebb, 2002). As will be seen in this chapter, neural networks follow a similar process where the weights between connections are analogous to the synaptic weights in biological nervous systems.

CNNs differ from simple artificial neural networks[1] in the replacing of matrix multiplication (where inputs and weights are multiplied together) with convolution operations (Goodfellow *et al.*, 2016; Hao *et al.*, 2019). As such, CNNs are ideal for image based learning (Raghu *et al.*, 2020) achieving human level performance (De Cesarei *et al.*, 2021). This is because they are able to process information that exists in a grid-like topology (Goodfellow *et al.*, 2016) and can update several parameters in order to improve performance. The details of how this occurs is discussed in the respective subsections. However, despite all of this, CNNs are still considered to be in their infancy (Ghorbanzadeh *et al.*, 2019). This is not because of a lack of understanding regarding CNNs, but more of a consequence of the complexity of the models. This makes it impossible, at this stage, to develop an all encompassing set of guidelines that explicitly link the modifications of the hyperparameters to the performance of the model as measured in metrics such as accuracy. A consequence of this is that the hyperparameters of a CNN are determined using a trial and error approach (Thakkar and Lohiya, 2021; Zeiler and Fergus, 2014; Tajbakhsh *et al.*, 2016).

A CNN can be visualised as a flowchart where an input image proceeds through various stages of image processing (in this case convolutions) before being classified into one or more categories (Figure 4.1). In this example, an image ($\mathbf{X}$) is passed to the first convolution layer where it is transformed through a convolution with a kernel ($\mathbf{K}$). This produces a feature map ($\mathbf{F}$) which is then sub-sampled using a pooling operation ($\mathbf{P}$). Up until this point the input data still has a grid-like topology (i.e. it is either a matrix or an n–dimensional array). After the processing of the image in the last convolution layer, it is transformed from a grid-like topology into a vector and passed on to the first fully-connected layer. The outputs of each neuron in the fully connected layer (n) is passed onto all the neurons within the next layer ($n+1$) (hence the name "fully-connected"). The last vector is the output layer containing neurons that correspond to the number of categories. The details of these various stages are discussed in the appropriate sections below.

It is important to have some basic understanding of what a CNN is and what it aims to achieve. This helps to place the more detailed sections in context. The preceding section, therefore, contributes to the thesis by providing a high level overview of CNNs, their design

[1] This term is very vague but is often associated with a multi-layer perceptron. This convention will be followed here.

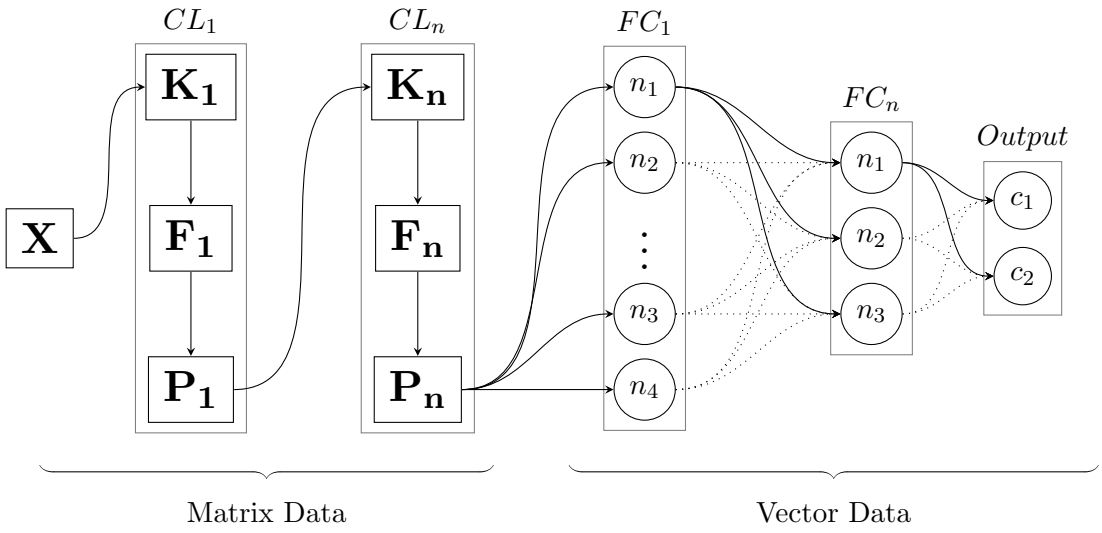


Figure 4.1: A flowchart of a CNN where an input image ($\mathbf{X}$) proceeds to one or more convolution layers (CL). Within a convolution layer it is transformed by two successive operations: convolution with a kernel ($\mathbf{K}$) which produces a feature map ($\mathbf{F}$) that is then subsampled in a pooling layer ($\mathbf{P}$). After the input image has been processed it is passed through several fully connected layers (FC) before an output is produced. Some connections are represented by dotted lines in order to make the diagram more clear, but these connections function the same way as those depicted by solid lines. The structure of the data at the various stages is also indicated.

philosophy, goals, and basic operation. The following sections will elaborate on the operation of CNNs in more detail. Ultimately, this will provide some idea as to why it can be expected that CNNs are up to the task of classifying barchan imagery.

4.2.1 Uses Within Geomorphology

Before proceeding to a discussion of the operation of CNNs, some background on their recent use within geomorphlogy is provided. This serves two purposes. First, it highlights that the application of CNNs within geomorphology is not new. Therefore, its relevance to the discipline, at least for some tasks, has been explored and shown to be promising. Second, it points out a gap that currently exists within the literature. It is this gap that is explored in this thesis.

This ability of CNNs to classify and categorise imagery has led to its use in a number of different studies of landforms within geomorphology (Table 4.1). These studies used CNNs mostly as a tool for semantic segmentation. Semantic segmentation attempts too classify the membership of every pixel within an image to a specified category. For example, the pixel is part of a dune or part of bare ground. This makes it possible to delineate a landform (i.e. map the boundary of the landform). It must be noted that a "standard" CNN cannot perform pixel-level semantic classification but requires the addition of a deconvolution layer to achieve this task (Maxwell *et al.*, 2020). A less frequently used application is feature detection. This attempts to identify whether a landform is present within an image without, necessarily, delineating it (i.e.

the exact position of the landform in the landscape is not determined). Images, or subsets of images, are merely classified based on whether the feature of interest is present or not.

U-Net, an architecture that is specifically designed for image segmentation (Ronneberger *et al.*, 2015), is a popular neural network architecture that has been used in several studies. Baumhoer *et al.* (2019) attempts to create a CNN that is capable of detecting the glacial and ice shelf fronts in Antarctica using a modified U-Net. Du *et al.* (2021) used U-Net to generate landslide susceptibility maps, but found it did not perform as well as other architectures such as DeepLab (Version 3). They speculate that it is possible that the complexity of a model (U-Net is more complex than DeepLab V3) leads to reduced performance in cases were limited training data is available (Du *et al.*, 2021). Gafurov and Yermolayev (2020) used U-Net to detect gullies from DigitalGlobe imagery. A problem that they identified with this approach is that, in a few cases, the gullies are discontinuous[2] because the semantic classification, since it is pixel based, can be negatively affected by vegetation cover. Another application of U-Net is Prakash *et al.* (2020) who applied it to map landslides using a combination of hillshade, derived from LiDAR rasters, and optical imagery. Like Ghorbanzadeh *et al.* (2019), they find that the presence of dense vegetation can hamper the performance of optical image analysis. Also, small landslides ($< 0.21km^2$) are difficult for CNNs to detect. Shumack *et al.* (2020) found that U-Net is able to, when used in conjunction with a Laplacian convolution[3], accurately detect linear dune ridges on Mars.

The suitability of CNNs to identify gullies, in this case thermokarst gullies, is investigated in Huang *et al.* (2018). The size of typical satellite imagery is well beyond the capacity of a number of CNN models. For example, the VGG16 model takes as an input an image of 224×224 pixels while a satellite image can be sized in the tens of thousands. To compensate for this, Huang *et al.* (2018) made use of overlapping sub-samples of the image in order to process large images.

The work by Gao *et al.* (2021) used a Mask R-CNN (He *et al.*, 2020) to classify yardangs from optical imagery. A Mask R-CNN, essentially, performs semantic segmentation. It can, however, be trained to only focus on specific regions of interest. For example, the model will perform a semantic classification of the pixels in the image with the express focus of identifying the target object. All pixels that constitute the target will then be labeled. It therefore follows two steps: detect the part of the image that contains the object (i.e. it creates a region of interest), and then classifies all of the pixels within the region of interest based on their membership to the target object (i.e. determine if they are part of the object or not). Additionally, Gao *et al.* (2021) also evaluates the performance of the model to classify the yardang in the image as belonging

[2]The gullies in the image are not discontinuous.
[3]This is a type of edge detector that increases the local contrast within an image (Russ, 2006).

Example Study	Task	Application
Baumhoer *et al.* (2019)	Semantic segmentation	Automatic extraction of seasonal glaciar ice shelf fronts from Sentinel-1 data.
Carbonneau *et al.* (2020)	Semantic segmentation	Classifying UAV imagery into classes of water, vegetation, and dry land.
Chen *et al.* (2018)	Feature detection	Detecting changes in the landscape, resulting from landslides, from remotely sensed imagery.
Du *et al.* (2021)	Semantic segmentation	Classifying pixels from optical satellite imagery as either landslide or non-landslide.
Gafurov and Yermolayev (2020)	Semantic segmentation	Detecting and delineating gullies from DigitalGlobe imagery.
Gao *et al.* (2021)	Semantic segmentation	Detecting, classifying, and delineating yardangs using Google Earth^{TM} imagery.
Ghorbanzadeh *et al.* (2019)	Semantic segmentation	Mapping landslides from RapidEye imagery.
Huang *et al.* (2018)	Semantic segmentation	Mapping thermokarst gullies from UAV imagery.
Ji *et al.* (2020)	Semantic segmentation	Detecting landslides from TripleSat imagery and digital elevation models.
Meena *et al.* (2021)	Semantic segmentation	Delineating landlsides that resulted from heavy rainfall.
Palafox *et al.* (2017)	Semantic Segmentation	Automatic detection of volcanic rootless cones and transverse aeolian ridges from HiRISE and CTX imagery.
Prakash *et al.* (2021)	Semantic segmentation	Mapping landslides from Earth observation data.
Robson *et al.* (2020)	Semantic segmentation	Delineating rock glaciers on Sentinel-2 optical imagery.
Shumack *et al.* (2020)	Semantic segmentation	Mapping dune crest patterns from digital surface models (ALOS).
Wilhelm *et al.* (2020)	Semantic segmentation	Creating a geomorphological map of the Martian surface using CTX imagery.
Witharana *et al.* (2020)	Semantic segmentation	Mapping the distribution of ice wedge polygons from remotely sensed imagery.

Table 4.1: Examples of the recent applications of CNNs within geomorphology to interpret imagery (either remotely sensed or generated as part of a terrain analysis).

to one of three categories: long-ridge, mesa, and whaleback. This latter part of their study is, conceptually, the closest to the goal attempted in this thesis. In general they found that CNNs are able to differentiate between the different types of yardangs, however, they note that one class (whaleback) was more difficult for the model to classify accurately. It is speculated that the more complex and diverse morphologies associated with the whaleback class led to an increased incidence of incorrect classifications and, therefore, poorer classification accuracies (Gao *et al.*, 2021).

CNNs are not necessarily used in isolation. Witharana *et al.* (2020) incorporated a Mask R-CNN in a workflow to extract shapefiles from imagery. In this case, the target feature was ice wedge polygons and the semantic segmentation derived from the Mask R-CNN was converted to a shapefile. This makes it possible to not only automatically extract a feature of interest from an image, but to also retain its spatial context.

Carbonneau *et al.* (2020) extends the application to of semantic segmentation by incorporating fuzzy classification. In a typical semantic segmentation task, pixels are assigned membership to mutually exclusive categories (e.g. a pixel can either be land or ocean, but not both). In a fuzzy classification system, a pixel has a level of membership to several categories. The higher the level of memberhip, the more similar to that category the pixel is. Through the use of fuzzy groupings new insights can be obtained in landscape analysis, particularly when the temporal dimension is considered. This is because it can identify gradual changes occurring within the landscape by looking at the changes in membership levels of individual pixels.

A novel application of CNNs is carried out in Chen *et al.* (2018). Rather than using a single CNN as a classification tool for a single input image, two separate images are supplied to two different CNNs. The resultant feature maps are then compared and the Manhattan distance[4] between them calculated. The underlying theory is that differences in content between the two images, in this case evidence of a landslide such as a slip scar, would translate into increased Manhattan distances between the two images. By using an appropriate threshold, to factor in non-landslide causes for changes in images such as the presence of clouds, it is then possible to detect if a landslide occurred in the intervening time between when the images were captured.

Although they made use of a fairly simply CNN architecture, Meena *et al.* (2021) made use of K-fold cross-validation during the training process. A key aspect of the development of a CNN is the creation of data sets containing images to be used for training, validation, and testing. Theses data sets are mutually exclusive and an image that is within the training set can not be present within the testing data. With K-fold cross-validation, the data is divided into k

[4]This Manhattan distance is calculated as the sum of absolute differences between pixels in the same position within the two images.

mutually exclusive categories. For each training and testing phase, a different subset is used as the testing data. As a result, the mean performance across all folds can serve as an indication of the architecture's performance on that task.

Ghorbanzadeh *et al.* (2019) also used a fairly simple CNN to identify landslides, but investigated the size of the input image (in terms of width and height) into the model. As objects with a spatial extent, the size of the "window" through which the CNN "views" the image can influence its performance. For example, a small window investigating a large landslide can be expected to perform differently from a large window investigating a small landslide. Their results showed that the performance of a CNN, in terms of a reduction in false positive and false negative cases, improves with increasing window size up until a critical point beyond which performance degradation results. Given the pixel-dependent behavior of a CNN this can be, potentially, explained by an increase in the amount of pixels that are not related to a part of the landslide as the window size increases. This can also potentially explain the findings of Prakash *et al.* (2020) where small landslides are difficult to detect.

Also investigating landslides, Ji *et al.* (2020) made use of attention mechanisms in order to improve the performance of the semantic segmentation. By using an attention mechanism, it is possible for the model to learn which features are more important to successfully classify the image. These useful features are then enhanced while the less useful features are suppressed (Weng *et al.*, 2020). By incorporating attention mechanisms within the model, Ji *et al.* (2020) is able to increase the accuracy of the classifications to above 90%.

Palafox *et al.* (2017) compared the performance of CNNs against support vector machines on a semantic segmentation task. Support vector machines are another type of classifier that is based on a linear function and allows class identity of an input to be determined (Goodfellow *et al.*, 2016). Palafox *et al.* (2017) found that their CNN has a higher accuracy than support vector machines in identifying rootless volcanic cones and transverse aeolian ridges from satellite imagery.

Using Sentinel-2 data, Robson *et al.* (2020) used a CNN to identify rock glaciers. Given the absence of supporting literature, they resort to a trial-and-error determination of optimal CNN depth and kernel size. A key finding is that landforms that resemble rock glaciers, such as debris flows, can result in the presence of misclassified regions. This indicates that detection of landforms from visual imagery is a challenge for CNNs.

Using fairly broad terrain categories, Wilhelm *et al.* (2020) demonstrated how CNNs can be used to generate geomorphological maps from optical satellite imagery. The best performing architecture for their work was DenseNet-161 which was pre-trained on the ImageNet data set.

The ImageNet data does not include imagery of the landforms of Mars. Therefore, this work shows that it is possible to use transfer learning for geomorphological applications.

It is therefore clear that there are a number of applications of CNNs within geomorphology. However, for the most part these applications focuses on a relatively narrow application, namely semantic segmentation. While this is beneficial, as shown in the studies, it does not investigate the full potential of CNNs. This leaves a gap in the literature that will be be at least somewhat addressed in this thesis.

4.3 CNN as a Composite Function

A CNN is essentially just a composite function[5] of the form $(f_1 \circ f_2 \circ \dots f_n)[x]$ where the variable x is an image and n denotes the number of inner functions. The goal of this composite function is to map a given input X to the correct category C. A conceptual representation of this mapping process is given in Figure 4.2. Here, the function $f(x)$ takes as input an image of a barchan. After all of the image processing has been carried out by the composite function, the produced output reflects the appropriate category. Technically, the output is a vector of probability of membership which reflects how likely the input is to correspond to each of the categories (de Lima and Marfurt, 2020). The variables associated with each of the functions $(f_1, f_2 \dots f_n)$ are updated during the training process in order to ensure that the mapping of X to C is accurate.

This classification is achieved by successive inner functions that either process part of an image, or transform a scalar value into a new scalar value. The latter occurs at individual neurons while the former occur through the use of kernels in convolution layers. An important characteristic of these inner functions is that the variables that determine the transformation and processing gets updated to an "optimal" value that is specific to that model and data type. This occurs through a process of learning where the model is exposed to known data from each category and iteratively adjusts the values within the function in order to match the input to the correct output. Once the model has learned the appropriate values for each parameter, it can then be used to classify imagery it has not seen before. This means that it is only necessary to train a model once and afterwards it can be reused and even distributed to others for their use.

This section is relevant to the thesis since it highlights the core idea behind CNNs that are used in classification applications. By realising that a neural network is just a mathematical function that aims to map an input to an output, it is possible to better understand how the model can be improved. For example, the CNN has variables that need to be updated in order to

[5] A composite function is a function that combines two or more functions in a specific order.

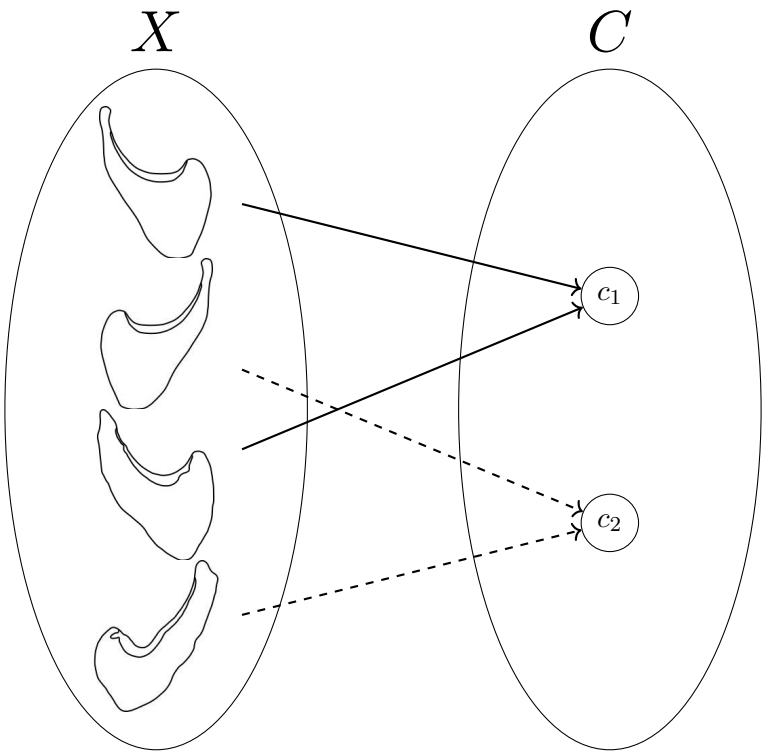


Figure 4.2: A conceptual representation of the mapping process where a barchan image (X) is mapped to its corresponding category C. The category c_1 contains barchans where the left horn is elongated while barchans where the right horn is elongated are placed in c_2.

perform optimally. This update is dependent on the hyperparameters of the model's architecture, as well the representatives of the training data. While still covering a broad territory of solutions, such a high level breakdown can form the starting point for improving model performance.

4.4 Model Parameters

The model parameters cover two important aspects of the CNN. First, the learnable parameters are those values within the composite function that are updated during the model training process. These include the weights of the connections between neurons, the values within the kernel that is used to process the input, and the biases of each neuron. The hyperparameters are the values that determine how the learning is executed, the types of data transformations that are used, and the amount of variables that the model has access to in order to optimise the composite function mentioned earlier. They are distinguishable from the learnable parameters in not being altered during the training process (i.e. the remain static). However, once the training process has been completed, the hyperparameters become static and do not change during subsequent use of the model. In the following sections these broad categories of parameters are discussed in more detail.

4.4.1 Hyperparameters

The performance of the CNN is determined by the hyperparameters (Ghorbanzadeh *et al.*, 2019; Shakya *et al.*, 2021). This is, however, only partially correct since the quality of the training data is also critically import in determining the success of a model. Despite this, it still takes skill and experience in order to properly select the proper hyperparameters (Gu *et al.*, 2018). Unfortunately, this means that a crucial aspect of the proper functioning of CNNs may, potentially, be unavailable to general users thereby limiting the widespread adoption of CNNs. During a typical use-case where a CNN is defined in TensorFlow[6], there are several hyperparameters that a user can set (Table 4.2). Together, these determine how the training will be carried out, the extent of image processing that is used, and several other aspects related to the functioning of the model. These are discussed separately under their respective subheadings.

4.4.1.1 Network Depth

By increasing the number of convolutional layers within a CNN, it is possible that higher level features can be extracted (Kattenborn *et al.*, 2021). In this context, a feature refers to a specific arrangement of pixel values that can include, but are not limited to, vertical, horizontal, and diagonal lines. Higher level features are therefore combinations of these lower level features (Tran *et al.*, 2018) similar to how the combination of a horizontal line ("–"), a line that is diagonally upwards ("/"), and a line that is diagonally downwards ("\") combine to form either an upwards triangle ("△") or a downwards triangle ("▽") depending on their spatial arrangement. In this example, the simpler straight line feature is identified earlier in the model while the composite feature (i.e. triangle) is identified deeper in the model.

Network depth is considered to be an important determining factor in model performance (He and Sun, 2015) and, in general, it is considered that increasing the depth of the CNN increases the accuracy of the model Shakya *et al.* (2021); Simonyan and Zisserman (2015); Wan *et al.* (2018); Ciresan *et al.* (2011). However, adding more layers to a model leads to an increase in the computational cost (He and Sun, 2015) and may be more difficult to optimise (Gu *et al.*, 2018). Unfortunately, it is also possible that overly increasing the depth of the network can lead to poorer performance (He and Sun, 2015; He *et al.*, 2016).

4.4.1.2 Number of Filters

A kernel is used to encode specific features of the input data (Chollet and Allaire, 2018; Tran *et al.*, 2018). The type of feature that the kernel encodes is determined by the values within the

[6]Tensorflow is a Python library that allows for the creation, training and execution of neural networks.

Hyperparameter	**Description**
Number of convolutional layers	Also referred to as the depth of the model. The number of layers can be any integer larger than 0.
Number of filters*	The number of kerenels that the relevant convolution layer uses to process the image. Can be any integer larger than 0.
Size of filters	The size of the $n \times n$ kernel. In principle it can be any integer larger than 3 provided the matrix is square.
Type of activation function	Any of several different activation functions. All the neurons within the same layer typically have the same activation function.
Number of epochs	The number of times the training data is passed completely though the model.
Learning rate	A constant that governs the magnitude of adjustments for each paramater during training.
Batch size	Specifies the number of samples in a batch.
Type of loss function	The function that is used to calculate the amount of error between the model's output and the actual data.
Dropout value	The fraction of activation functions in the model that outputs a value of 0.
Batch normalization	Adjusts the outputs of a layer in order to prevent the model from becoming imbalanced.
Type of optimizer	The algorithm that is used to adjust the parameter based on the value from the loss function.
Pooling	Specifies if, and how, the outputs of a given layer should be sub-sampled.
Regularization	Specifies what type of regularization should be added to the loss function.

Table 4.2: The different hyperparameters that define a CNN (adapted from the list provided by Igonin *et al.* (2020)).
* In this context the terms "filter" and "kernel" are interchangeable

kernel (Cady, 2017). Therefore, the more kernels within a convolution layer, the more features are identified within a given input (Gu *et al.*, 2018). The process by which the kernel extracts features is known as a convolution operation (Traore *et al.*, 2018; Niepert *et al.*, 2016; Lecun *et al.*, 2015) and occurs over a small region of the image (Russ, 2006; Gu *et al.*, 2018), known as the receptive field (Bera and Shrivastava, 2020). If the input image is larger than the kernel, the kernel slides across the input image horizontally and vertically (Tran *et al.*, 2018; Chevtchenko *et al.*, 2018) performing a convolution at each step (Figure 4.3). The convolution operation (Eq. 4.1) involves the matrix multiplication of the receptive field ($\mathbf{X}$) with the transpose of the kernel ($\mathbf{G}^T$) and taking the trace of the resultant matrix to produce a single pixel value $\mathbf{F}$ of the feature map. It is important to mention that although the number of kernels used in a model is a hyperparameter, the values of the kernel is a learned parameter (Maggiori *et al.*, 2017).

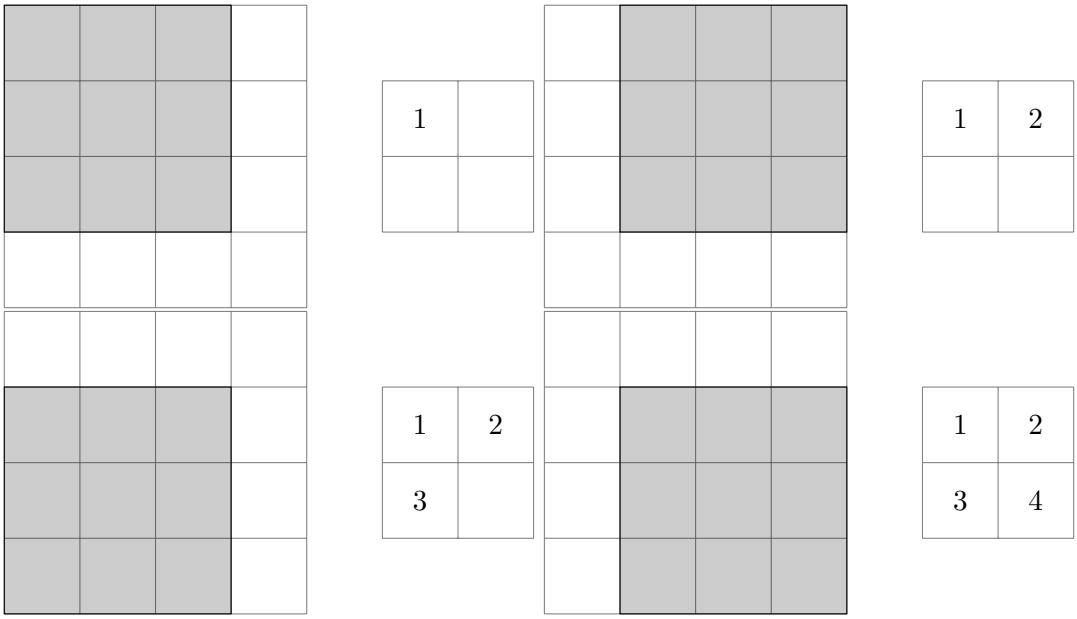


Figure 4.3: An example of a 3×3 kernel sliding over a 4×4 input image to produce a 2×2 feature map. The numbers show the sequence in which the feature map is generated and does not correspond, in this example, with the values contained within the input.

$$\mathbf{F} = tr(\mathbf{X}\mathbf{K}^T) \qquad \text{(Eq. 4.1)}$$

To illustrate how the kernel indicates the presence of a feature, consider a portion $\mathbf{X}$ of an input image (Eq. 4.2). The values within this input have been scaled from -1 to $+1$. This portion contains a diagonal line moving from the top left to the bottom right (indicated by the $+1$ values). If this input portion is evaluated by kernels with different configurations ($\mathbf{K_1}, \mathbf{K_2}, \mathbf{K_3}$), the resulting output values will also differ (Eq. 4.3) to (Eq. 4.5).

$$\mathbf{X} = \begin{bmatrix} 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{bmatrix} \tag{Eq. 4.2}$$

$$\mathbf{K_1} = \begin{bmatrix} 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{bmatrix} = 9 \tag{Eq. 4.3}$$

$$\mathbf{K_2} = \begin{bmatrix} -1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & -1 \end{bmatrix} = 1 \tag{Eq. 4.4}$$

$$\mathbf{K_3} = \begin{bmatrix} -1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{bmatrix} = -9 \tag{Eq. 4.5}$$

In cases where there is a strong agreement between a portion of the input image and the kernel (e.g (Eq. 4.3)), the resultant convolution (which ultimately forms part of the feature map) is high. Conversely if there is no match (i.e. (Eq. 4.5)) the output of the convolution will be low. Intermediate matches will therefore also have intermediate values (e.g. (Eq. 4.4)).

4.4.1.3 Size of Filters

This hyperparameter specifies the dimensions of the kernel (i.e. its size). In the examples (Eq. 4.3) to (Eq. 4.5) the size of the kernel is 3×3. Although the kernel can be any size, the sizes of 3×3 and 5×5 are typically used. A minimum size of 3×3 is needed in order to detect concepts such as left, right, up, down and centre (Simonyan and Zisserman, 2015). In principle larger kernels can detect more complex features but He and Sun (2015) found that deeper networks with smaller kernels performed better than shallower networks with larger kernels. Even though the values inside the kernel is learnable and updates during training, the size of the kernel is fixed and does not change once the model has been initialised.

4.4.1.4 Type of Activation Function

A number of different activation functions (φ) (sometimes referred to as transfer functions (Hagan *et al.*, 1996) or an input-output function (Suliman and Zhang, 2015)) are encountered in the literature (Table 4.3 and 4.4). The activation function determines what the neuron's output value will be (Rojas, 1996; Engelbrecht, 2007; Floreano and Mattiussi, 2008). Haykin (1994) argues that the activation function serves the additional role of limiting the amplitude

(i.e. range) of the neuron's output. This is because a number of activation functions, such as the sigmoid and the hyperbolic tangent, constrain the output values to $(0, 1)$ and $(-1, 1)$ respectively. However, not all activation functions perform this type transformation. Activation functions such as linear, Rectified Linear Unit (ReLU) and Exponential Linear Unit (ELU) are unbounded with positive values that approach infinity.

A second role of activation functions is that it introduces non–linearity into the model (Ravichandiran, 2019; Engelbrecht, 2007; Güner *et al.*, 2013). The characteristics of the activation function, particularly its first derivative, is important in determining how efficient the Artificial Neural Networks (ANN) is. For example, the weights of the connections that are updated during the training phase of the model (see §4.5) are updated proportionally to the first derivative of the activation function used (Suliman and Zhang, 2015).

4.4.1.5 Number of Epochs

An epoch is a unit of training during which the entire training data set has passed through the model once (Chollet and Allaire, 2018; Goodfellow *et al.*, 2016; Farzad *et al.*, 2019). As training progresses through the specified number of epochs, it is expected that the training accuracy, i.e. the ability to accurately classify the training images, will increase while the loss value (which represents the errors in classification) to decrease. A similar pattern is expected for the validation data. In practice this is not always the case and there can be a considerable amount of fluctuations between epochs. Additionally, Sinha *et al.* (2010) found that increasing the number of epochs that a model is allowed to train from 500 to 1100 increased the performance of the model, but beyond this point (i.e. using epochs between 1100 and 2000) lead to increasingly poorer performance. One possible explanation for this is that, as the number of epochs increase, the model may be prone to overfitting (Lamm and Unger, 2011) which means that its generalization ability decreases leading to poor performance.

4.4.1.6 Learning Rate

The learning rate is a scalar quantity (Goodfellow *et al.*, 2016) that determines the magnitude of an adjustment that is made to a parameter during training (Velasco *et al.*, 2018). If this value is high, then the model can potentially be optimized more quickly, however, the risk for oscillating around local minimums is also increased (Ertel, 2011). An example of this process is given in Figure 4.4. In this example, the optimal value for the parameter w is 2. Starting from an initial value of 5.19 the model attempts to reduce this value. If the learning rate is low (e.g. $\eta = 0.01$) the adjustment in the parameter is small. Increasing the learning rate results in larger adjustments and if the value is too large (e.g. $\eta = 1$) there is the risk that the weight will oscillate around the optimal value. Currently, there is no universally applicable solution to

Name	Equation	Description
Sigmoid	$\varphi = \frac{1}{1+\exp(-x)}$	Inefficient when compared to ReLU and hyperbolic tangent (Lu, 2020; Gu *et al.*, 2018) and susceptible to the vanishing gradient problem (Zheng *et al.*, 2020).
Hyperbolic tangent	$\varphi = \frac{1-\exp(-2x)}{1+\exp(-2x)}$	Considered to be superior to the sigmoid function (Zheng *et al.*, 2020) but less efficient than ReLU (Zhu *et al.*, 2017).
Linear	$\varphi = x$	Returns the output that it receives (Haykin, 2009). Since linear functions can only express linear mappings (Zheng *et al.*, 2020) this function cannot be used when non–linear solutions are required.
Softsign	$\varphi = \frac{x}{\lvert x\rvert+1}$	Developed by Elliott (1993) as an improvement to the sigmoid function. More robust to the vanishing gradient problem than the hyperbolic tangent function (Elsayed *et al.*, 2018).
Softplus	$\varphi = log(\exp(x)+1)$	Outperforms the sigmoid, hyperbolic tangent, and softsign functions on classification tasks (Ertam, 2017). However, it takes longer to train than the hyperbolic tangent and softsign functions (Elsayed *et al.*, 2018).
Gaussian	$\varphi = \exp(-x^2)$	Returns similar values for very large negative and very large positive inputs (Gundogdu *et al.*, 2016). Performs worse than the sigmoid function in predictive modelling tasks (Velasco *et al.*, 2018).
Swish	$\varphi = u_k \frac{1}{1+\exp(-x)}$	Developed to solve the vanishing gradient problem associated with the sigmoid function (Tripathi *et al.*, 2019). Considered by Jinsakul *et al.* (2019) to be an improvement on the ReLU function for classification tasks.
Softmax	$\varphi = \frac{\exp(x)}{\Sigma \exp(x)}$	Designed as a classifier (Ertam, 2017; Gu *et al.*, 2018) it calculates the probability distribution for each of the specified outputs (Chollet and Allaire, 2018; Lai *et al.*, 2019).

Table 4.3: Examples of activation functions where x is the output of the summing function.

Name	Equation	Description
Step	$\varphi = \begin{cases} 0 & : x \leq 0 \\ 1 & : x > 0 \end{cases}$	Forms part of a group of functions known as hard–limit functions that are typically used in classification and pattern recognition tasks (Negnevitsky, 2011).
ReLU	$\varphi = \begin{cases} 0 & : x \leq 0 \\ u_k & : x > 0 \end{cases}$	Rectified Linear Unit. Overcomes vanishing gradient problem associated with sigmoid (Zheng *et al.*, 2020). Considered to be the most popular activation function (Tripathi *et al.*, 2019; Lu, 2020; Lecun *et al.*, 2015).
Leaky ReLU	$\varphi = \begin{cases} \alpha x & : x \leq 0 \\ x & : x > 0 \end{cases}$	Designed to counter the dying ReLU problem (Gu *et al.*, 2018). Performs as well, or better, than ReLU (Zheng *et al.*, 2020).
ELU	$\varphi = \begin{cases} \alpha \exp(x) - 1 & : x < 0 \\ x & : x \geq 0 \end{cases}$	Exponential Linear Unit. Introduced by Clevert *et al.* (2016) to increase learning speed and to improve classification accuracies. More robust to noisy data (Gu *et al.*, 2018) but takes longer to train than ReLU (Qiumei *et al.*, 2019).

Table 4.4: Examples of piecewise activation functions where x is the output of the summing function.

finding the optimal learning rate and users have to resort to a trial and error approach (Velasco *et al.*, 2018).

4.4.1.7 Batch Size

This hyperparameter plays an important role during the learning process because of its influence on the optimisation algorithm. A mini–batch, sometimes simply referred to as batch depending on the context (Goodfellow *et al.*, 2016), is a small subset of the samples that are used to calculate the update that should be applied to each parameter (Chollet and Allaire, 2018; Raschka and Mirjalili, 2019). The batch size determines how many samples should be used within each batch. This value is often dictated by the available hardware due to the additional computational demand that larger batch sizes have (Planche and Andres, 2019). By using a sample of images, as opposed to a single image, during training, it could lead to better convergence[7] of the model during training by reducing the amount of variance in the parameters (Gu *et al.*, 2018). However, it is also possible that using batches that are too large will result in a decrease in performance (Keskar *et al.*, 2017).

[7]Convergence is reached when, after a certain amount of epochs, there is no significant change in the model's performance.

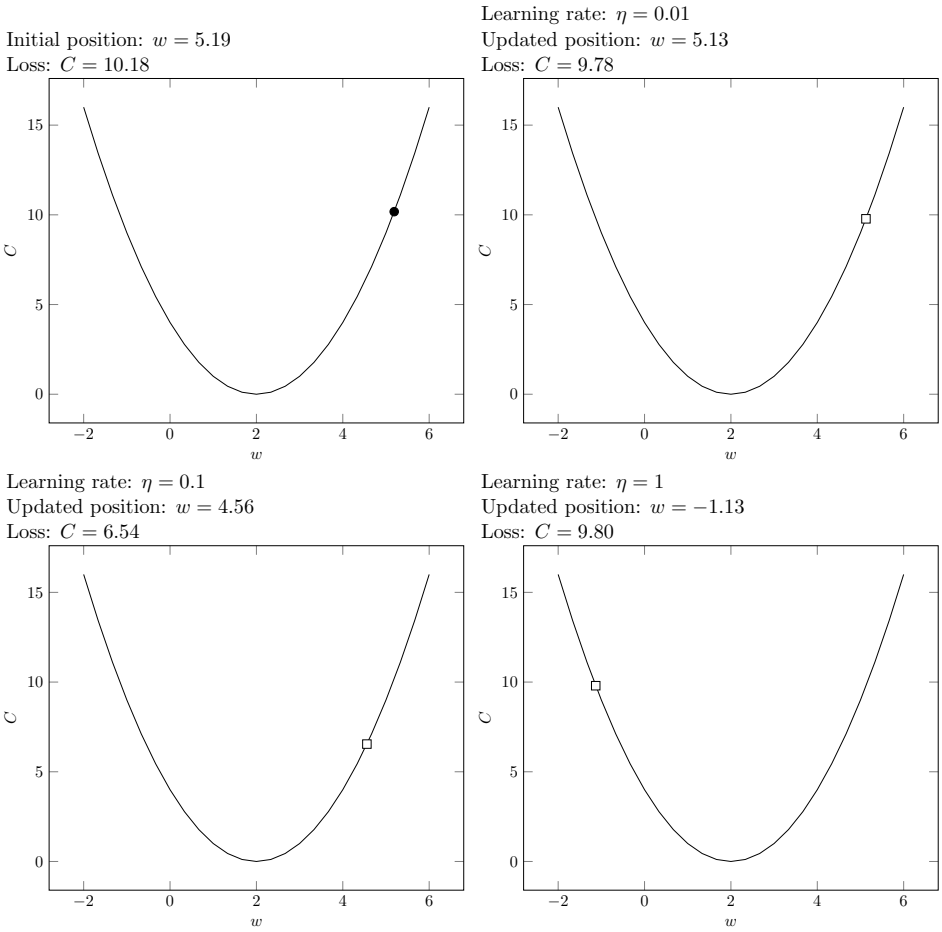


Figure 4.4: A simplified representation for the update of a learnable parameter for three different learning rates: $\eta = 0.01$. $\eta = 0.1$, and $\eta = 1$.

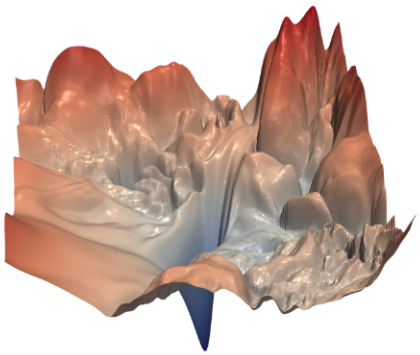


Figure 4.5: An example of part of the loss function for the ResNet56 model (Li *et al.*, 2018).

4.4.1.8 Type of Loss Function

During the training of the model it is important that the level of disagreement between the model's output and the true value be quantified. When a softmax activation function is used in the output layer, the CNN produces a probability value which represents the likelihood that the image belongs to each of the categories that were defined. In supervised learning, the actual classification of the image is known which means it is possible to quantify the level of correctness in the model's classification. This process is carried out by the loss function (Chollet and Allaire, 2018; Traore *et al.*, 2018). It is, essentially, a scalar field that plots the level of correctness (i.e. the loss)[8] for every possible combination of learnable parameters within the model. Also known as the objective function, cost function, or error function, the model seeks to minimise this function during training (Goodfellow *et al.*, 2016). The minimum point of an objective function represents the point where there is as much as possible agreement between the model and reality. In the context of evaluating remotely sensed data, there are a number of different loss functions that have been used, but those based on cross-entropy are most commonly used (Neupane *et al.*, 2021). These functions have been found to have a fast convergence (Maggiori *et al.*, 2017) suggesting that optimally performing models can be trained using fewer epochs.

It is important to note that this loss function is a high–dimensional, non–convex function (Li *et al.*, 2018) (Figure 4.5). It is currently not feasible for the majority of users to derive this surface for their models. This is because of the prohibitive computational costs involved in its generation (Li *et al.*, 2018). However, because this surface is differentiable, it is possible to use the principle of surface gradient to calculate the direction of a decease in this surface. This is carried out by the optimizer and is discussed in §4.4.1.11.

[8] A low loss value indicates a high level of agreement between the model and the actual classification.

4.4.1.9 Dropout Value

This hyperparameter is only adjusted when dropout layers are used in the CNN. A dropout layer assigns a random value of 0 to the output of a set of activation functions in a previous layer (Agarwal and Mittal, 2019). In other words, it causes the neurons to have an output of 0 thereby forcing the model to adjust other parameters to lower the loss function. Dropout layers are used to reduce potential overfitting within the model (Shakya *et al.*, 2021; Chevtchenko *et al.*, 2018) thereby making it possible for the model to become more generalized. The dropout value determines the fraction of the neurons that will have an output of 0. A typical value for this hyperparameter is 0.5 (De Cesarei *et al.*, 2021). It has been found that high dropout values can degrade the accuracy of the model leading to poorer performance (Shakya *et al.*, 2021).

4.4.1.10 Batch Normalization

Batch normalization adjusts the outputs of activation functions within a layer (Ball *et al.*, 2017) to have a mean of 0 and one unit of variance (Kattenborn *et al.*, 2021). In other words the outputs are normally distributed. This processes is carried out after each batch during training (Gu *et al.*, 2018) and can have a big impact on the performance of a CNN (Goodfellow *et al.*, 2016). In essence, batch normalization prevents the model from becoming imbalanced due to the presence of either very large or very small activation outputs (Kattenborn *et al.*, 2021). One of the benefits of this is that it makes the output of the model less sensitive to its initial configuration (i.e. the initial weights that are randomly assigned) (Ball *et al.*, 2017).

4.4.1.11 Type of Optimizer

Within the context of neural networks, optimization refers to the process of minimising some function $f(x_1, x_2 \dots x_n)$, which is the loss function, by altering $x_1, x_2 \dots x_n$ (Goodfellow *et al.*, 2016). In this oversimplified expression, the variables $x_1, x_2 \dots x_n$ represents the learnable parameters that result in the loss function. This task is carried out by an optimizer function which determines how the model will be updated (Chollet and Allaire, 2018). It therefore requires two important aspects to be determined: the direction of the change and the magnitude of the change. There are a number of different optimizers to choose from, but an example of four different approaches to optimization are presented in Table 4.5. In the first case, gradient descent, the update of the parameter is based solely upon the current gradient and the leaning rate. While this is more simple to calculate, it can result in the optimizer getting "stuck" in a local minimum (Kneusel, 2022) resulting in sub-optimal training and performance. To compensate for this a momentum term can be added which adds a scaled version of the previous update to the new update (Velasco *et al.*, 2018). With Nesterov momentum, the concept of momentum

is retained, but the position at which the gradient is calculated is changed (Gu *et al.*, 2018). The effect of this is to prevent any small fluctuations within the cost function to have a large influence on the direction of parameter adjustment (Pradhan *et al.*, 2020). Lastly, the Adaptive Moment Estimation (ADAM) optimizer belongs to a group of optimizers that are adaptive (Kneusel, 2022). This means that the learning rate is not kept constant throughout the training process (Bui *et al.*, 2019) leading to improved performance (Kneusel, 2022).

4.4.1.12 Pooling

Pooling is a means by which the data is downsampled in order to reduce the computational demands of the model (Chevtchenko *et al.*, 2018; Traore *et al.*, 2018; Wang *et al.*, 2019). This occurs after the convolution layer and provides a condensed feature map (Ghorbanzadeh *et al.*, 2019). It also aims to provide translation invariance by reducing the resolution of the feature maps (Bera and Shrivastava, 2020). The most popular pooling operation is max pooling (Bera and Shrivastava, 2020) which outputs the maximum value within a receptive area (Traore *et al.*, 2018; Wang *et al.*, 2019). Here, the receptive area is similar in principle to the receptive area discussed for filters (§4.4.1.2). As illustration, consider a feature map $\mathbf{F}$ (Eq. 4.6) that is processed by a pooling layer consisting of a 2×2 kernel. The feature map has a diagonal line from the upper left to the lower right (elements that are equal to 9). The kernel slides across the image (horizontally and vertically) with a step interval of one (i.e. similar to the convolution kernel discussed earlier). If a max pooling approach is used, the kernel will select the highest value that is present in the 2×2 kernel and produce that as an output (Eq. 4.7). The result is a sub-sampled feature map that, overall, has the same general pattern as the input where the downwards diagonal line is still present. But the advantage is that there are fewer elements within the matrix which, in turn, reduces the computational load.

$$\mathbf{F} = \begin{bmatrix} 9 & -9 & -9 & -9 \\ -9 & 9 & -9 & -9 \\ -9 & -9 & 9 & -9 \\ -9 & -9 & -9 & 9 \end{bmatrix} \tag{Eq. 4.6}$$

$$\mathbf{F_{sub}} = \begin{bmatrix} 9 & 9 & -9 \\ 9 & 9 & 9 \\ -9 & 9 & 9 \end{bmatrix} \tag{Eq. 4.7}$$

Table 4.5: Examples of different optimizer functions (adapted from Kneusel (2022)). The variables are: ρ – parameter to be updated, η – learning rate, υ – update value, ϑ – the running average of the first moment, φ – the running average of the second moment, β_1, β_2 – scalar weighting parameters, $\hat{\vartheta}, \hat{\varphi}$ – bias correction terms for first first and second moment averages, t – time step of training process, ϵ – constant.

Name	Equation	Description
Gradient Decent	$\rho_t = \rho_{t-1} - \eta \frac{\partial C}{\partial \rho_t}$	This optimizer is only dependent on the learning rate and gradient, meaning that it is susceptible to get stuck in sub-optimal local minimas within the cost function (Kneusel, 2022).
Gradient Descent with Momentum	$\upsilon_t = \mu \upsilon_{t-1} - \eta \frac{\partial C}{\partial \rho_t}$ $\rho_t = \rho_{t-1} + \upsilon_t$	Introduces the momentum term which incorporates the previous adjustment into the new adjustment (Kneusel, 2022). This is useful in situations where the gradient changes rapidly since the momentum will smooth out the differences to some extent (Shaf *et al.*, 2018).
Nesterov Momentum	$\upsilon_t = \mu \upsilon_{t-1} - \eta \frac{\partial C}{\partial(\rho_t + \mu \upsilon_{t-1})}$ $\rho_t = \rho_{t-1} + \upsilon_t$	Differs from gradient descent with momentum by calculating the gradient at an adjusted position within the cost function (Gu *et al.*, 2018; Pradhan and Lee, 2010).
Adam (Adaptive moment estimation)	$\vartheta = \beta_1 \vartheta + (1 - \beta_1) \frac{\partial C}{\partial x}$ $\varphi = \beta_2 \varphi + (1 - \beta_2) \left(\frac{\partial C}{\partial x} \right)^2$ $\hat{\vartheta} = \frac{\vartheta}{1 - \beta_1^t}$ $\hat{\varphi} = \frac{\varphi}{1 - \beta_2^t}$ $\rho = \rho - \frac{\eta}{\sqrt{\hat{\varphi}} + \epsilon} \hat{\vartheta}$	A popular optimizer (Kneusel, 2022) capable of adapting its learning rate during the training process (Bui *et al.*, 2019).

4.4.1.13 Regularizations

The regularizations covered by this hyperparameter are different in a very fundamental way from the dropout hyperparameter. Whereas the dropout hyperparameter is involved in sporadically changing the outputs of the model during training (by assigning a value of zero as the output of a neuron), the regularizations associated with this hyperparameter alter the value of the cost function that needs to be optimised (Gu *et al.*, 2018). This is also used as a strategy to reduce overfitting within the model (Thakkar and Lohiya, 2021; Tombe and Viriri, 2020; Lancashire *et al.*, 2009) and to reduce model complexity (Rubanenko *et al.*, 2021; Chollet and Allaire, 2018; Zhang *et al.*, 2017). In so doing, the model's predictive performance is increased.

According to (Goodfellow *et al.*, 2016) the term regularization refers to any modification made to a learning program that is intended to reduce its generalization error (i.e. to make it more accurate in classifying inputs it has not seen before). Two types of regularization are commonly used: L_1 and L_2 regularization. In L_1 regularization (Eq. 4.8), also known as sparsity regularization (Benuwa *et al.*, 2016), the additional cost added to the loss function is proportional to the absolute values of the weight coefficients (Chollet and Allaire, 2018). For L_2 regularization (Eq. 4.9), known as weight decay generalization (Benuwa *et al.*, 2016), the added cost is proportional to the square of the weight coefficients (Chollet and Allaire, 2018). In both cases the regulizer has a regularization strength parameter (λ) (Gu *et al.*, 2018; Tombe and Viriri, 2020; Thakkar and Lohiya, 2021) that determines the weight of the regularization in the final cost calculation.

$$L_1 = \lambda \sum_{j=1}^{p} |\beta_j| \qquad \text{(Eq. 4.8)}$$

$$L_2 = \lambda \sum_{j=1}^{p} \beta_j^2 \qquad \text{(Eq. 4.9)}$$

The purpose of this section, within the context of the thesis, is three-fold. First, it provides a brief overview of the different hyperparameters that can be modified in a typical CNN workflow (Objective 3). This is an important aspect to take into consideration since the hyperparpamters ultimately determine the performance of a model on a chosen metric. Second, by discussing the different hyperparameters and their possible configurations it is, hopefully, shown that there are many permutations possible when designing models. Much more than can be feasible carried by a human. Third, in addition to these multiple permutations, it is also not known how these

configurations interact with each other in order to produce a given output. The combination of all of these is the primary reason for the trial-and-error approach to CNN studies.

4.4.2 Learnable Parameters

The learnable parameters of a CNN are those parameters that are updated during the training of the model. Depending on the training strategy followed, these parameters are updated after a single training example is passed through the model (i.e. stochastic gradient descent) or after a mini–batch of training samples are passed through. The way in which these adjustments are made, and when they are made, are governed by the model's hyperparameters.

4.4.2.1 Weights and Bias

It was mentioned earlier that a neuron receives inputs and produces outputs. These inputs and outputs are transfers of information to other parts of the model (i.e. the neuron is connected to other neurons in different parts of the model). Each of these connections has an associate weight which is, conceptually, similar to the strength of the synaptic connection between biological neurons (Rodvold *et al.*, 2001). Each weight within a neural network has a magnitude (i.e. size) and a direction (either positive or negative) that determine the contributions of the input parameters to the output (Olden and Jackson, 2002). Positive weights are analogous to excitatory neurons in biology while negative weights are analogous to inhibitory neurons (Olden and Jackson, 2002; Jain *et al.*, 1996; Lancashire *et al.*, 2009). Input variables that are associated with larger weights contribute more to the output than those associated with smaller weights (Olden and Jackson, 2002). The value and direction of the weights result from the training process (i.e. they are learned) (Chollet and Allaire, 2018; Belciug and El-Darzi, 2010) and therefore the weights of the model can be viewed as "storing" the processing capacity of the model (Gurney, 2010).

There are some important implications from this. First and foremost, since the weights determine the processing capacity of the model, and they are updated through the training process, it means that a sufficient amount of training is required in order for the model to make accurate predictions. Second, it does not require the user to specify in advance which variable within the input is the most important. This is because, in principle, the model will identify the features that allow for the best generalization to be obtained. Given the complexity of image interpretation this self-correcting feature is very useful.

4.4.2.2 Kernel

A kernel is a matrix that is used to convolve a portion of an input image (see §4.4.1.2). Convolutions are mathematical operations that occur in the spatial domain of the image (Russ, 2006) and are a core component of CNNs (Geum *et al.*, 2020). They are used to extract features from

an input image (Palafox *et al.*, 2017) by transforming an image patch (Figure 4.3) to produce a feature map (Geum *et al.*, 2020). An important property of the kernel is that it gets updated during the learning process. This allows the CNN to extract the most useful features from the input image.

An example of this process is given in Figure 4.6. A barchan outline is fed into the CNN and in the first convolution layer it is transformed into a feature map by a kernel. This operation can result in a smaller image (typically a feature map is 2 rows of pixels shorter and two columns of pixels narrower than the original input). However, the hyperparameters of the convolution layer can be configured to retain the dimensions of the input by setting the appropriate padding value. This process continues for all the convolution layers specified with each feature map extracting features from earlier features. This, in principle, is what allows CNNs to extract more abstract shape information from an original input.

4.5 Learning

The concept of learning as applied to a CNN refers to the iterative process whereby the performance of the model is improved by exposing the model to data (Ertel, 2011). It is this that made it possible for machines to address real world problems and to make decisions that appear to be subjective (Goodfellow *et al.*, 2016). Although, technically, the fact that a CNN is a composite mathematical function means that it is not subjective but rather objective. In this section, the strategies associated with learning is briefly discussed and the process of learning, as applied to supervised learning, is elaborated upon. This is necessary information in order to understand how a CNN improves its performance. Within the context of this research, it is also ensure that this performance increase can still be traced back to the improved processing of the barchan image.

4.5.1 Types of Learning

Several different strategies for training a neural network are available (Table 4.6). Supervised learning is a common learning approach (Jordan and Mitchell, 2015; Chollet and Allaire, 2018) and is, essentially, a training approach where the model can determine how correct its classification is (Abdi *et al.*, 2011). By using labelled data (Kattenborn *et al.*, 2021) the neural network is shown what type of response should be used for any given input (Goodfellow *et al.*, 2016). This is achieved via the calculation of the loss function mentioned earlier. In this way, the model learns the appropriate weights that will allow it to map an input to a given output (Chollet and Allaire, 2018), i.e. to place an image into a category. In contrast, unsupervised learning does not make use of labelled data (Jordan and Mitchell, 2015; Goodfellow *et al.*, 2016).

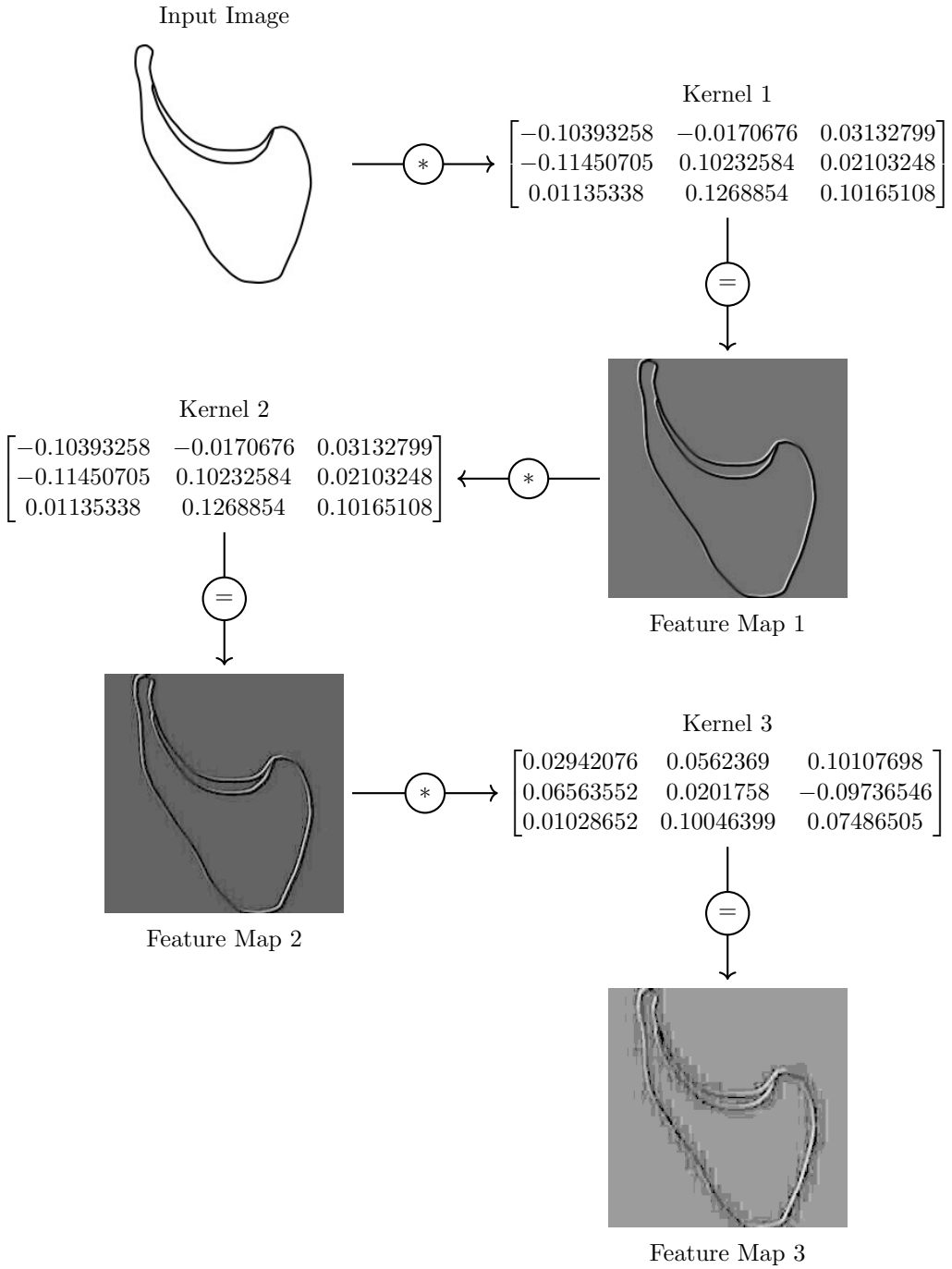


Figure 4.6: An example of several different kernel convolutions applied to a barchan outline. The input image is convolved (represented by the $*$ symbol) with the first kernel to produce a feature map. This feature map is then convolved with the second kernel producing another feature map. Lastly, this second feature map is then convolved with another kernel to produce the final feature map. Convolutions carried out using ImageMagick (https://imagemagick.org/script/index.php) and contrast stretched in Gimp (https://www.gimp.org/) for easier interpretation.

Due to the possibility of a lack of large labelled data sets to train a model (using a supervised learning approach), unsupervised learning was developed as a means to alleviate the dependency on lablled data sets (Zhu *et al.*, 2017). The goal of unsupervised learning is to detect an underlying feature that can be used to place the images into one of several categories (Jordan and Mitchell, 2015; Atkinson and Tatnall, 1997; Goodfellow *et al.*, 2016). While beneficial, it does have the limitation in that these categories do not, necessarily, correspond to those assigned by the community. For example, in the case of barchan research, when tasked with categorising by shape it is not a given that the defined categories will correspond to the $\frac{a}{c}$ classification scheme developed by Finkel (1959). It is also believed that humans use an unsupervised learning approach during the process of classifying natural phenomena (Lamm and Unger, 2011).

Semi-supervised and weakly supervised approaches are methods to combine aspects of both supervised and unsupervised learning. During a semi-supervised approach the model is trained by using a small labelled data set (Kattenborn *et al.*, 2021). More specifically, the model is trained to assign labels to a larger corpus of imagery after it has been trained on the smaller data set. This larger corpus is then used to improve the model's performance. In weakly supervised learning the goal is to use labels of "lower" quality to train a model (Kattenborn *et al.*, 2021). Similarly, weak supervision is also used due to the expense, both in time and money, in obtaining large amounts of training data (Zhou, 2018). Due to the nature of the data used in weakly supervised approaches, which can be inaccurate, inexact, or incomplete (Zhou, 2018), models trained using this approach often show poor performance (Li *et al.*, 2019). Lastly, reinforcement learning works on the principle of a feedback loop between the model and the environment (Goodfellow *et al.*, 2016). It is used in cases where a machine needs to learn to complete a task such as playing a computer game where it learns the actions to take in order to maximise its score on the game (Chollet and Allaire, 2018).

4.5.2 The Learning Process

The learning process of a CNN will be discussed within the context of Figure 4.7. This model takes an image X as input. For this example, the image has only a single channel (i.e it is a grayscale image). Only a single convolutional layer is present whereby a kernel K converts the input into a feature map F. The feature map is then passed on to a fully connected layer for classification into one of two categories.

During the learning process, the model learns the regular patterns that are present within the training data (De Cesarei *et al.*, 2021). In this case, the model would learn those attributes that place the input into one of the two categories (i.e. it assigns a label to the input image). For training to be effective, a label has to be assigned to the input (this occurs during forward

Table 4.6: A description of the main approaches to machine learning.

Type of Learning	Description
Supervised	In supervised learning, models learn a mapping from x to y through exposure to a set of samples (Jordan and Mitchell, 2015; Chollet and Allaire, 2018).
Unsupervised	Used to alleviate the dependence on labelled data sets (Zhu *et al.*, 2017), unsupervised learning attempts to find patterns in the structure of the data (Jordan and Mitchell, 2015).
Semi-supervised	In semi-supervised learning, both labelled and unlabelled examples are used in order to train the model (Goodfellow *et al.*, 2016).
Weakly supervised	In weakly supervised learning, lower quality, compared to the intended use, reference data is used (Kattenborn *et al.*, 2021). Typically used in the absence of precise label information (Li *et al.*, 2019).
Reinforcement	In reinforcement learning, the model receives input from its environment and learns to choose actions that will maximise its reward (Chollet and Allaire, 2018) through the use of feedback loops (Goodfellow *et al.*, 2016).

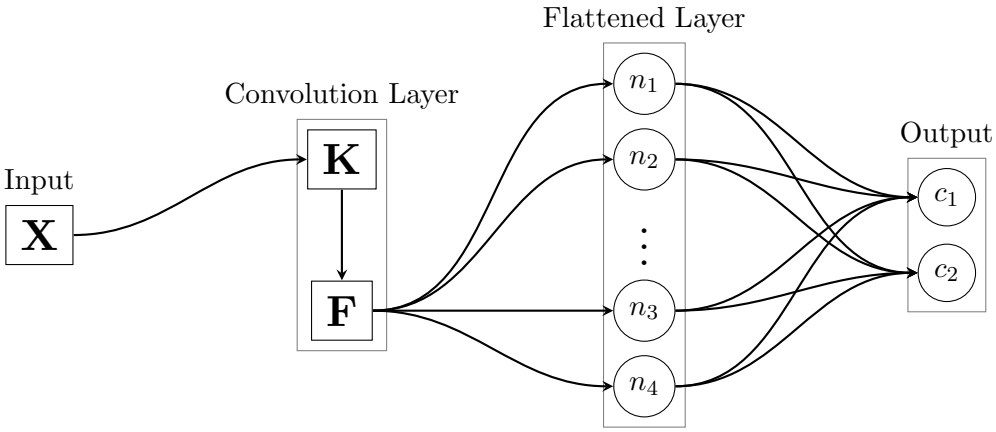


Figure 4.7: A simple CNN with one convolution layer with a single kernel ($\mathbf{K}$) that produces a feature map ($\mathbf{F}$), which is then placed into one of two output categories.

propagation) and the correctness of this label needs to be evaluated by the loss function. Afterwards, the model has to adjust the learnable parameters within the model in such a way as to minimise the error of the model. This occurs during back-propagation.

When a model is created for the first time (i.e. it is initialised) all of the learnable parameters are assigned a random value based on some distribution. After the intialisation is completed, the training process starts (Figure 4.8). For each epoch, a training example is passed through the model (this is known as forward propagation). Since a training example is associated with a label (or value), the difference between the model's output and the expected output can be calculated. This is used to create a loss function for this run of the model. The goal of the training process is to minimise this loss function by adjusting the learnable parameters (which in this example is the kernel K and the weights W) which is carried out during back-propagation. In order to achieve this, the partial derivatives $(\frac{\partial C}{\partial W} \ \frac{\partial C}{\partial K})$ of the learnable parameters with regards to the loss function needs to be determined. These calculated values are used within the optimizer function to determine the direction and magnitude of adjustment that needs to be made to the learnable parameter. After the adjustments have been made, the next training example is used. Since the kernel and weights are now different values, the output of the model is different and a new loss function is calculated. This is repeated until all of the training examples have been used, marking the end of the first epoch. This entire process is then repeated for the amount of epochs specified in the model's hyperparameters. After all the epochs have elapsed, the model is considered trained. The process discussed here relates to a stochastic gradient approach to training where only a single training example is used each time to update the values. Where mini-batch training is used, several training examples are used in the forward propagation process and the average values used for the update process. A more detailed description of this process is provided in Appendix B and Appendix C.

The main contribution of this section is to provide an overview of how a CNN improves its performance. As can be seen, a CNN has the capacity to improve the ability of kernels to extract features that would allow the classification to be as accurate as possible. This is particularly useful in cases such as barchan outline classification since it is not practical to specify that level of detail in advance. Through the learning process, the CNN can therefore identify those features of the outline that best distinguish between the two classes. However, it is also clear that this process can only modify parameters that were initialised. As such, this section describes, briefly, why CNNs are considered to be dependent on initial starting conditions. This provides the basis for the decision to run multiple different versions of the same model (see §5.9) to obtain an average performance.

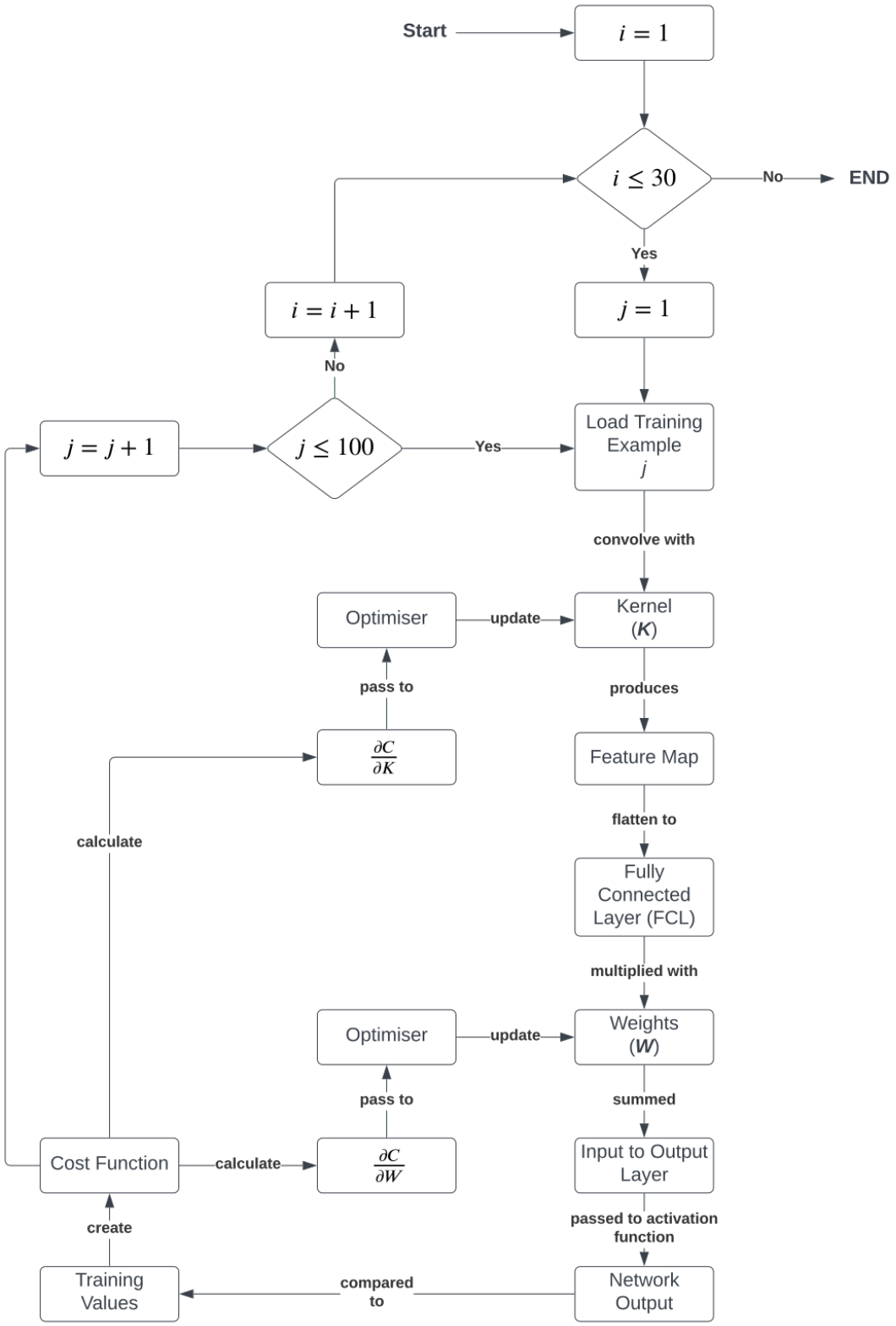


Figure 4.8: A flowchart depicting the training process of the simplified CNN depicted in figure 4.7. For this example, 30 epochs of training (i) was used and 100 training images (j) are available.

4.6 Transfer Learning

Broadly speaking, transfer learning involves using CNN models that were trained on a primary task to perform a secondary task (Azzaoui *et al.*, 2019; de Lima and Marfurt, 2020; Torrey and Shavlik, 2010; Neupane *et al.*, 2021). The rationale is that a model can be trained to extract features on a primary task where an abundant data set is available. During this phase, the learnabale parameters of the model is updated (§4.5) in order to allow it to accurately classify the images of the primary data set. Once this is achieved, it is then possible to leverage this feature extraction capacity to extract features on a secondary classification task (Ball *et al.*, 2017; Huang *et al.*, 2018). In practice, this involves removing the fully connected portion of the model (i.e. the classifier) while retaining the feature extractor (Wan *et al.*, 2018; Wilhelm *et al.*, 2020). The new model would therefore consist of the weights that the model learned and that are stored within the different convolutional layers. Essentially, this refers to the learned kernels which, as discussed previously, are responsible for the extraction of features from the imagery. A new fully connected layer is then added to this feature extractor that is more suited to the task. This is often necessary since the classification categories for the secondary task are likely to be different from those of the primary task. The model is then retrained but, in contrast to the "from scratch" scenario, the weights of the feature extractor (i.e. the kernels in the convolutional layers) are prevented from being updated and it is only the weights of the fully connected layer that are learnable. Therefore, transfer learning is neither a hyperparameter nor a learnable parameter. Rather, it represents a different implementation paradigm.

Despite the "big data" of remote sensing being a reality, suitable data sets upon which a CNN can be trained to interpret remotely sensed imagery is often limited. In this case, suitable data refers not to the image itself, but the existence of image-label pairs that can be used for model training. This can be due to the absence of labelled imagery (Liu *et al.*, 2021), the logistical constraints associated with ground truthing (Kattenborn *et al.*, 2021), or the preference for RGB images which lead to a low representation of multi-spectral imagery (Bhuiyan *et al.*, 2020). Regardless of the reason behind the limited data availability, transfer learning serve as a usable alternative to "from scratch models" for applications were the training data may be limited (Ball *et al.*, 2017; Liu *et al.*, 2021; Kattenborn *et al.*, 2021; Neupane *et al.*, 2021; Wilhelm *et al.*, 2020).

This section describes the principles of transfer learning which is one of the approaches whose performance is evaluated in this study. It is a common method to use when data sets are small, as is the case in this study, and has been frequently applied in remote sensing tasks.

Category	Lower Bound	Upper Bound
Slim	0.125	0.375
Normal	0.376	0.625
Pudgy	0.625	0.875
Fat	0.876	≥ 1

Table 4.7: The classification categories of barchan shape using the $\frac{a}{c}$ ratio (Bourke and Goudie, 2009).

4.7 Why use CNNs?

Given the preceding sections it is reasonable to assume that CNNs are suited to address the research problem. The aim of this research is to determine whether morphological data can be extracted from imagery of barchan outlines. For this to be a reasonable task for a CNN, several conditions need to be met:

1. An image of a barchan outline must contain morphometric information of that barchan.
2. It must be impractical to "hard code[9]" a solution to the problem.
3. The data that is fed into the model must be compatible with the various data processing stages in order to facilitate learning.

From Chapter 2 it is clear that the first criterion is already met. A number of studies have made use of remotely sensed barchan imagery in order to obtain morphometric data (e.g. Bourke and Goudie, 2009; Hamdan *et al.*, 2016; Elbelrhiti *et al.*, 2008). Although it may be stating the obvious, this does show that morphometric data corresponding to real world landforms can be extracted from imagery of the landform provided that the imagery is not distorted. Since an outline of a barchan still contains the information used to determine properties such as shape (see Figure 2.4) it can be concluded that imagery of barchan outlines do contain morphometric data about barchans.

The second condition lies at the heart of Artificial Intelligence (AI) related problems. Although neural networks can be developed to solve any problem, they are less suitable to solving problems that have simpler solutions that can be directly converted to programming. For example, consider the classification of barchans into different shape categories based on their $\frac{a}{c}$ ratio (Table 4.7). Placement inside each of these categories only require that the stoss length of the dune (a) be divided by the distance between the horns (c). Therefore, if these two variables are known, there is no need to develop a neural network for the classification task.

However, in cases where the required information is not available beforehand, such as when images are used as input, it becomes more difficult to use such an approach. This is because the

[9]Here, hard coding is used as an antithesis to learning. It involves a solution that is pre-programmed and therefore not able to adjust to new information in order to improve a process.

program must be configured for any possible combination of inputs. In this case, it needs to be know, in advance, where all the pixels within the image are located that are needed in order to classify the image. Because this is impossible, it is easier to develop a program that can learn the pattern itself (i.e. a neural network).

The last point should be self-evident from the preceding discussion. CNNs are specifically designed to work with images as an input. Therefore, since images of barchan outlines are images, they exist in a format that is suitable for CNNs to interpret. However, it can be argued that other types of neural network can, in principle, handle this information. For example, if the data is converted from a matrix to a vector format, it should be possible to feed the image into a multi-layer perceptron. However, this does result in decreased performance[10] due to the absence of the convolution operations.

[10] Although this is not specifically addressed in this study, earlier versions of the project made use of multi-layer perceptrons that had worse performance than was obtained for CNNs.

Chapter 5

Method

5.1 Chapter Overview

To successfully reach the aim of this project, several aspects need to be addressed (Figure 5.1). It is important that the barchans used are based on the outlines of barchans that are encountered in nature. Since image texture has been identified as a point of bias in the classification process of CNNs, this texture component needs to be removed. The works of Dakir *et al.* (2016) and Azzaoui *et al.* (2019) has shown promising results, however, the results are not yet at a level that matches the requirements for this study. It is also necessary to create several different classification tasks of varying levels of complexity in order to evaluate the diversity of applications to which a CNN is suited. Lastly, several different CNN architectures need to be created within which the influence of some of the hyperparameters that have been discussed can be determined. It is not possible, within this study, to investigate all of the hyperpaprameters along with their possible combinations.

5.2 Site Selection

A barchan dune corridor in the Kunene region in northern Namibia, located between the latitudes of $18.8°S$ and $19°S$, and between the longitudes $12°E$ and $12.5°E$, was chosen as the source of barchan imagery. This is because the region has a large number of isolated barchans and imagery of a resolution sufficient to extract detailed outlines. It therefore meets the requirements et out in Objective 4. The site has an average elevation of 58 m.a.s.l.[1] with a topography that is gently sloping from the north-east to the south-west (Figure 5.2).

[1] Meters above sea level

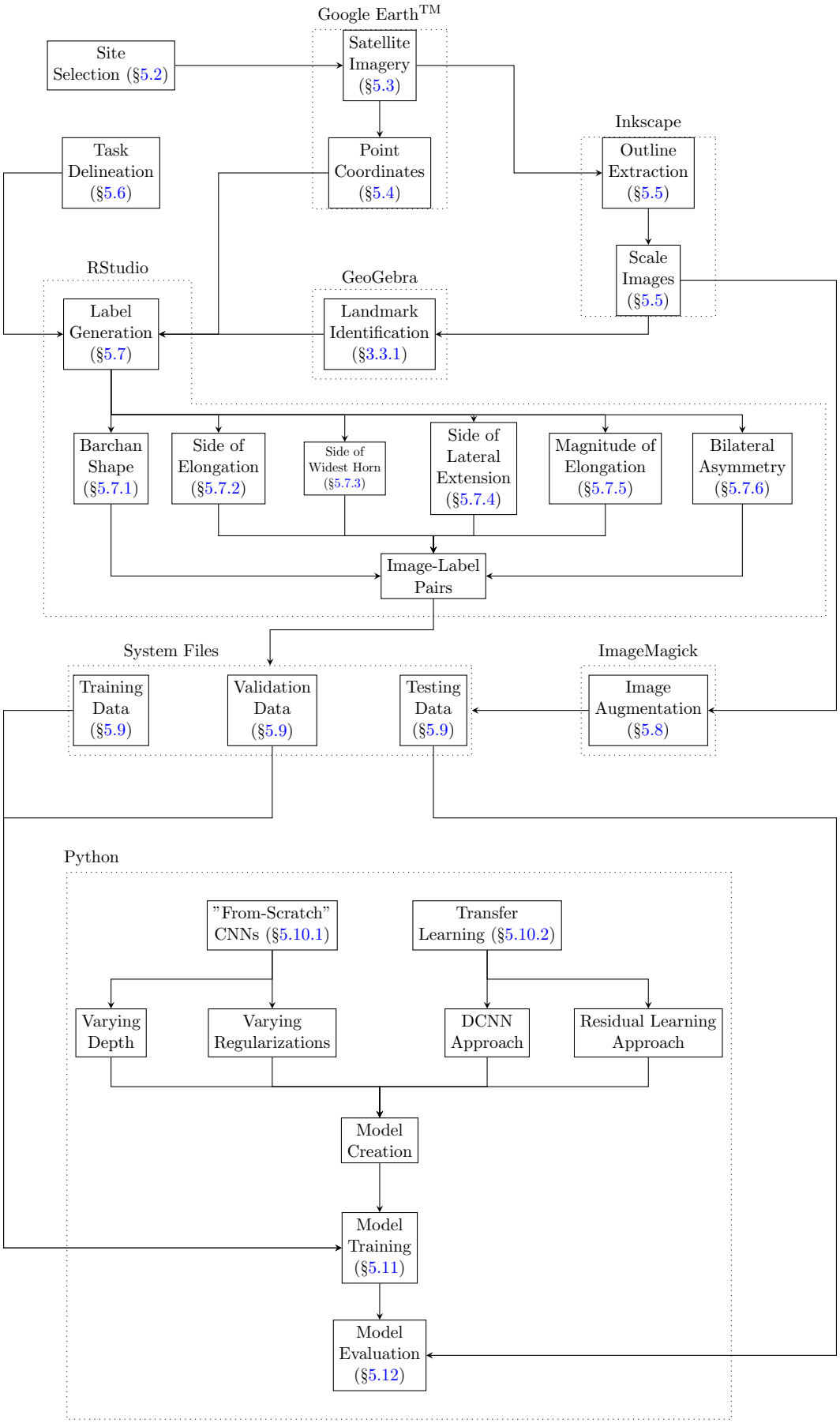


Figure 5.1: Project workflow along with the associated computational environments. The different sections of this chapter will follow this outline and discuss the components in more detail.

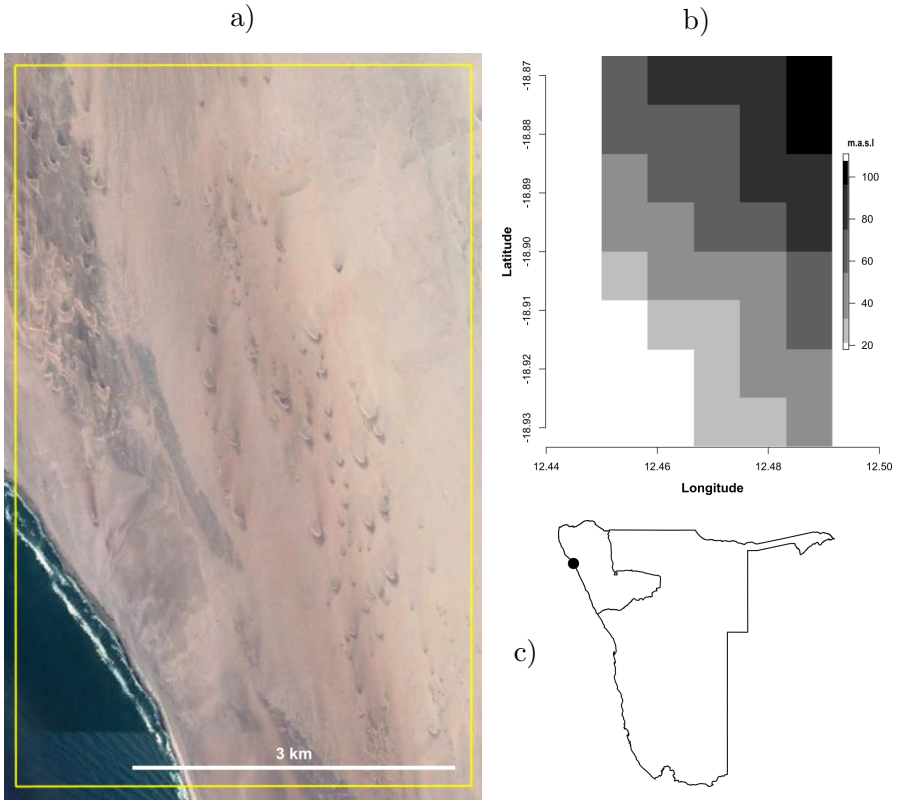


Figure 5.2: a) The barchan dune field that was used in this study (Image from Google Earth™). b) A raster image of the topography (approx. 30m resolution). c) The location of the Kunene dune field in Namibia.

NCEP/NCAR[2] project produces data are consistent with site observations (Bao and Zhang, 2013; Ashkenazy *et al.*, 2012) and have been used in an earlier study on aeolian geomorphology in the region (Barnes, 2001). Therefore, this data can also be used to describe the wind regime of the study site. To ensure that the wind data correspond as closely as possible to the dune configuration at the time the image was taken (12 November 2012), the time period was selected as starting at 12 November 2011 and ending at 12 November 2012. The pressure level for the observations was set at 1000 hPa[3] to ensure readings as close to the surface as possible and the data of four observations per day was chosen to increase the accuracy. This yielded 1465 wind observations for the study period.

The collected wind data was analysed using RStudio. NCEP data is provided in the form of u and v wind components which needed to be transformed into wind vectors. Using the approach of Fryberger (1979) the Drift Potential (DP) (Eq. 5.1) was calculated using wind directions associated with wind speeds between 11 and 40 knots. The variables for this equation are: average wind velocity (V), threshold velocity ($V_t = 12\ knots$), and the percentage occurrence of

[2]National Centers for Environmental Prediction/National Weather Service/NOAA/U.S. Department of Commerce. 1994, updated monthly. NCEP/NCAR Global Reanalysis Products, 1948-continuing. Research Data Archive at NOAA/PSL:/data/gridded/data.ncep.reanalysis.html.

[3]Hectopascals

wind in each direction (t) (Zamani *et al.*, 2019). The drift potential in an area is an indication of the relative amount of sand moved by the surface wind, above a threshold velocity, over a period of time (Fryberger, 1979; Hamdan *et al.*, 2016). The equations provided by Zamani *et al.* (2019) was used to transform the DP data into Resultant Drift Potential (RDP) and Resultant Drift Direction (RDD) values (Eq. 5.2) where θ is the midpoint of each of the orientation classes used to group the wind data. The RDP value represents the magnitude of the drift potential while the RDD value refers to the direction in which the sand will be transported (Hamdan *et al.*, 2016). The ratio, $\frac{RDP}{DP}$ serves as an expression of directional variability with high values reflecting unimodal winds and low values representing more complex wind regimes (Pearce and Walker, 2005).

$$Q \propto V^2(V - V_t)t \tag{Eq. 5.1}$$

$$\begin{aligned} RDP &= \sqrt{(C^2) + (D^2)} \\ RDD &= \arctan(\frac{C}{D}) \\ C &= \sum_{i=1}^{n} (DP_i) \sin\theta_i \\ D &= \sum_{i=1}^{n} (DP_i) \cos\theta_i \end{aligned} \tag{Eq. 5.2}$$

Plotted wind vectors (Figure 5.3) show a mean wind direction that is approximately southerly (179°) which corresponds to the general south-north alignment , i.e. the horns are "pointing" towards the north, of the barchans and is similar to the findings of other studies in Namibia (Barnes, 2001; Bourke and Goudie, 2009; Scheidt and Lancaster, 2013). The region has a DP of 747.7, an RDP of 629.2, and an RDD of 352°. This indicates a high energy wind environment with wind directions that tend to be unimodal (Table 5.1). The RDP value of the study site is higher than that reported at Walvis Bay by Bourke and Goudie (2009) suggesting a more intense sand transport environment. Also, an $\frac{RDP}{DP}$ value of approximately 0.84 suggests a high-energy wind environment with narrow unimodal directional variability (Pearce and Walker, 2005). This value is similar to that calculated for Luderitz ($\frac{RDP}{DP} = 0.85$) in southern Namibia (Tsoar, 2001) and for southern Iran ($\frac{RDP}{DP} = 0.84$) (Moosavi *et al.*, 2014), but less than that calculated for a barchan dune field in Morocco ($\frac{RDP}{DP} = 0.9$) (Elbelrhiti *et al.*, 2008). Approximately 62.2% of the wind readings were of a speed sufficient to result in particle movement (i.e. they were above 11 knots). This indicates that this area is, a very active region in terms of aeolian processes.

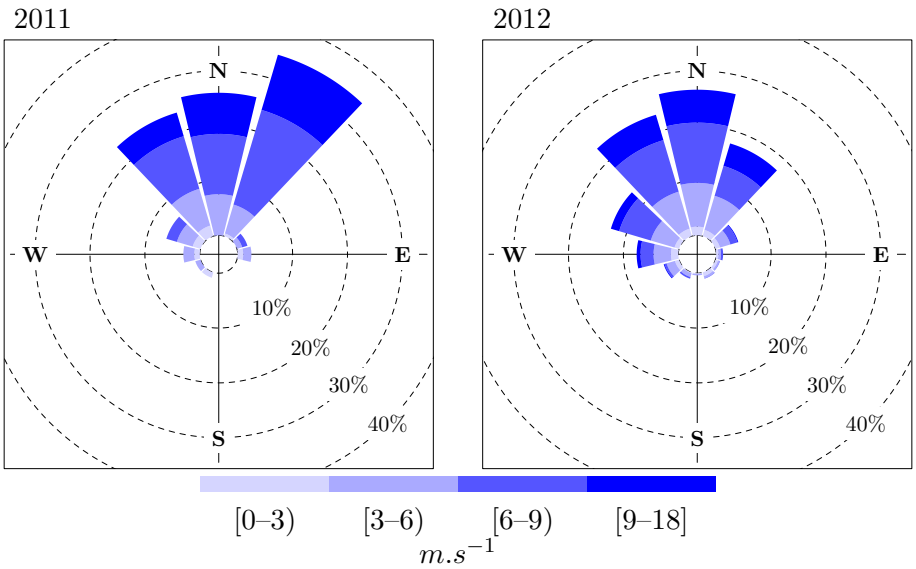


Figure 5.3: Wind vectors for all wind speeds ($n = 1465$) in the study site separated into years. Note that the wind vectors indicate the direction in which the wind is blowing.

Table 5.1: The relationship between DP and $\frac{RDP}{DP}$ in describing the wind regime of a region. Examples of dunes associated with the different environments are also provided (adapted from Pearce and Walker (2005)).

DP	$\frac{\text{RDP}}{\text{DP}}$	Wind Energy	Directional Variability	Dunes
< 200	< 0.3	Low	Complex to obtuse bimodal	Start dunes
200–400	0.3–0.8	Intermediate	Obtuse to acute bimodal	Linear dunes
> 400	> 0.88	High	Wide to narrow unimodal	Barchanoid ridges

This section contributes to the thesis by providing some context for the type of environment from which the barchans were sampled. While not necessary to achieve the aim of this thesis, it is, nevertheless, useful as an aid to describe the current shape of the barchan dunes. The high energy nature of the wind environment indicates that this region is currently very active and that the barchans are actively moving northwards. Additionally, the region has a wide unimodal nature.

5.3 Satellite Imagery

Images of barchans were obtained using Google Earth™ [4]. Although the positional accuracy of Google Earth™ imagery has been questioned (Mohammed *et al.*, 2013; Paredes-Hernández *et al.*, 2013) it is still considered a source of data in barchan research (Bourke and Goudie, 2009; Lorenz *et al.*, 2013; Dakir *et al.*, 2016; Hugenholtz *et al.*, 2012). Only isolated barchans (i.e

[4] The use of Google Earth™ imagery is covered under the ethics application for this project (Ethics Clearance Number: 180000081).

barchans that were not merged nor in a process of merging) were sampled. This is to avoid the subjectivity associated with delineating merged barchans (Hugenholtz *et al.*, 2012).

This section describes the source of the imagery that was used to train the CNN. Key factors to consider is the reliability of the data source. Although the positional accuracy is a concern, it is not considered relevant here since the project does not have a spatial analysis component. As long as it can be confidently claimed that there are no obvious distortions within an image that would alter the shape of the barchan, then the data is considered suitable for use. Given the Google Earth^{TM} data is used frequently by the aeolian research community, it is safe to say that this distortion is not noticeably present.

5.4 KML Data

To describe the dimensions of the barchans, .kml data was collected from each of the sampled dunes (Figure 5.4). It is important to note that this data is only used to describe the size of the barchans that were used in the study and were not used in subsequent analysis. The data from Google Earth^{TM} is in a geographic projection (WGS84) and, since the determination of distances between the points used techniques from Euclidean geometry, it was necessary to convert the points to an appropriate map projection. Since the data was collected in Namibia, the Hartebeeshoek94 projection was selected. The dimensions of the barchans can then be calculated using the following. For stoss length, the Euclidean distance between the toe of the barchan (C) and the crest (D) was used. The lengths of the horns were calculated as the perpendicular distance between the toe of the barchan (C) and the tips of the individual horns (A and B). This does differ from the more conventional descriptions (Sauermann *et al.*, 2000; Wang *et al.*, 2007) who use the bottom of the slip face and the dune crest respectively. However, this method is still adequate to provide an overall description of a barchan's dimensions.

Horn width is defined according to Sauermann *et al.* (2000) as the distance between the horn tips measured perpendicular to the longitudinal axis ($\overline{CF}$). This differs from the interpretation of Norris (1966) who considers it as the Euclidean distance between the tips of the horns. To determine the width of the dune, triangles CAD and CBD were constructed. Using the length of the stoss slope $\overline{CD}$ as a common base. The heights of the two triangles ($\overline{AF}$ and $\overline{BG}$ respectively) can then be summed to calculate the width of the dune. This is achieved by using Heron's formula and the area formula of a triangle[5].

[5]This is discussed in more detail in Section 5.7.1.

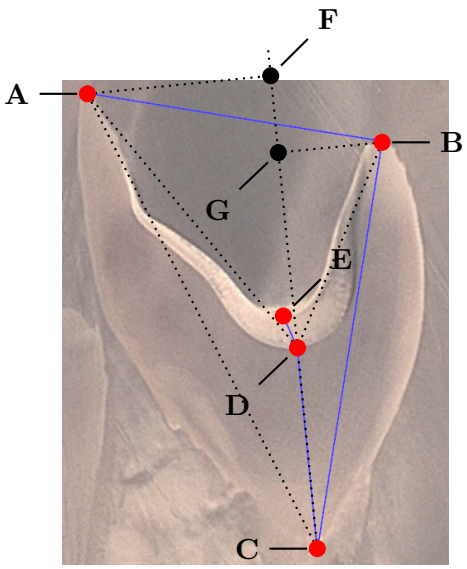


Figure 5.4: Points that were collected from the Google Earth™ data as a .kml file. The path tool (blue) was used to ensure that all of the points (red) can be contained within a single file.

5.5 Outline Extraction and Image Scaling

The RGB images obtained from Google Earth™ contain image texture. This texture information can result from: shape, luminance, surface textures, and background information (Baker *et al.*, 2020). To remove this, the outlines of the barchans were digitised (Zhang *et al.*, 2018) (Figure 5.5). This was carried out in InkScape (https://inkscape.org/) so that the outlines were in scalable vector graphic format (i.e `.svg`). This makes it possible to scale the size of the outline to any value without a loss in detail. Because the dunes are of different sizes, the ability to rescale them allows for the influence of the size of the dune (which in practice would manifest as clustered pixel values), to be effectively eliminated. A key aspect in determining the performance of a CNN is the quality of the images used during training and, ultimately, during evaluation. Resizing raster images can result in additional pixel values being incorporated into the image during the reduction or enlargement process. As a result, an additional variable is introduced within the image that is solely an artefact of the original size of the image. When vector graphics are used, this introduction of additional pixels can be avoided. In order to ensure consistency, all of image data were resized to ensure that the longest axis of the image (in either the x or the y direction) was 224 pixels. The remaining axis was then scaled to 224 pixels to create a 224×224 pixel image. The choice of dimensions was based on the input requirements for the VGG16 and ResNet50 architectures (Bendjillali *et al.*, 2020; LaVezzi *et al.*, 2020). Since models such as VVG16 use RGB images as input, the outlines retained the RGB channels of the image for consistency.

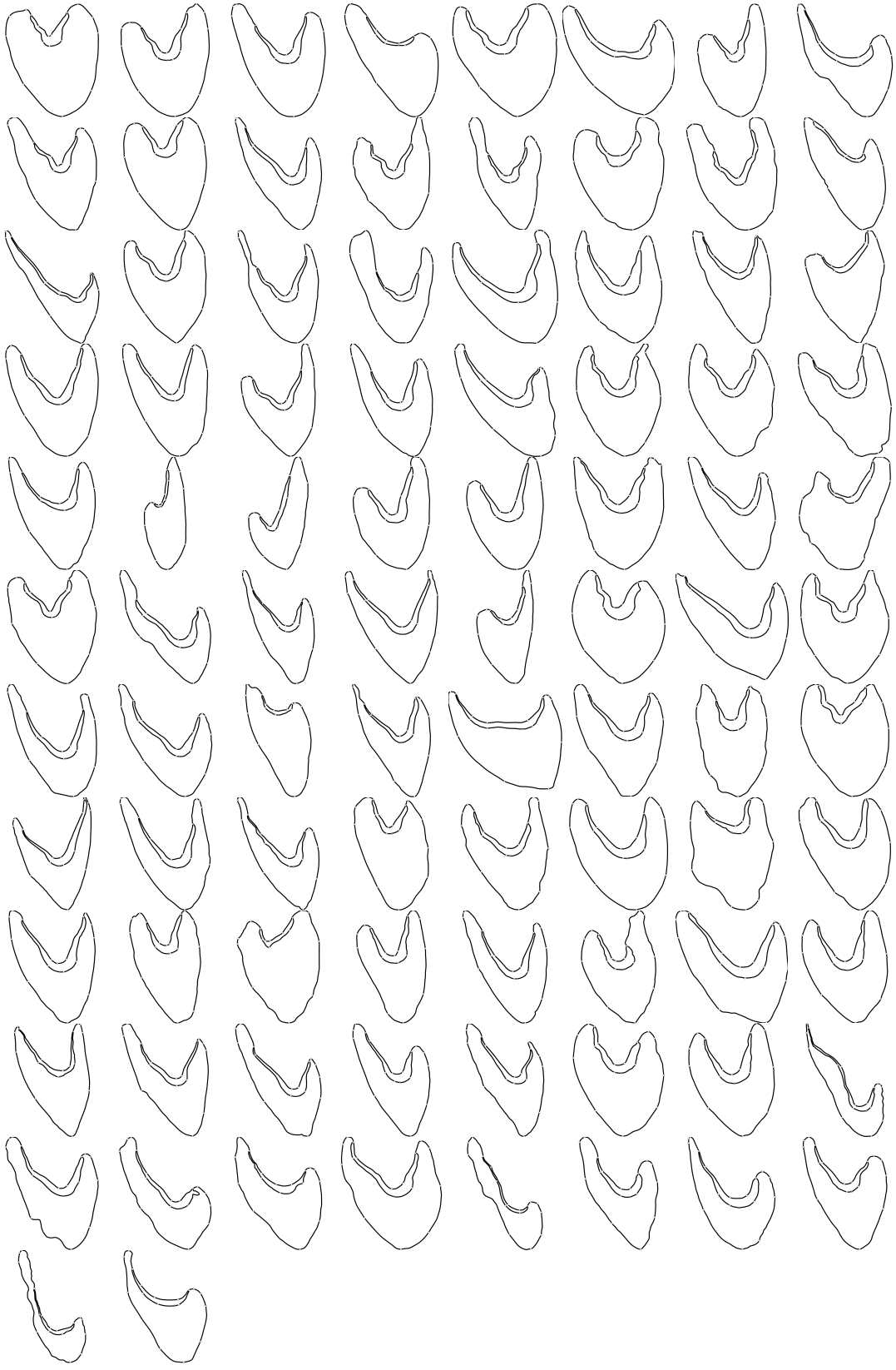


Figure 5.5: The barchan outlines that were used in this study.

A reasonable question to raise is why object identification techniques were not employed here. Object identification, i.e. semantic segmentation, strategies have been successfully used in earlier work to extract barchans (foreground) from their environment (background) (Rubanenko *et al.*, 2021; Azzaoui *et al.*, 2020; Vaz *et al.*, 2015). However, a limitation of this strategy is that it only permits the extraction of the planform shape of the barchan (i.e. its two dimensional footprint on the image). It has not yet been demonstrated that semantic segmentation can distinguish between the slip face and the stoss slope of the barchan. This, essentially, prevents information regarding the brink line to be extracted. Given the importance of the brink line in describing barchan morphology (Figure 2.4) this is a significant shortcoming.

This section explains how the imagery was prepared prior to being used in the training and evaluation of the CNNs (Objective 5). Given the sensitivity of CNNs to image texture (Geirhos *et al.*, 2019), and since it is unknowable in advance how the CNN will make use of this texture information (Baker *et al.*, 2020), it is necessary to remove this so that a baseline examination can be carried out. In other words, it is a means of removing a variable that cannot be satisfactorily controlled for.

5.6 Task Delineation

As stated in the first chapter, the aim of this research is to determine if barchan shape and asymmetry information can automatically be extracted by a CNN from a simplified image of a barchan and, if so, how can the performance of the network be enhanced. For the first part, six different tasks have been developed, each addressing a different aspect of barchan morphometry (Table 5.2). The tasks are discussed in more detail under the relevant subsections while the details surrounding their calculation is provided in Section 5.7. Details on the interpretive value of the morphometric variables[6] associated with each task can be found in Chapter 2.

5.6.1 Classifying Barchan Shape

The $\frac{a}{c}$–ratio is commonly used to describe the shape of barchan dunes (e.g Hamdan *et al.*, 2016; Bourke and Goudie, 2009). Several categories are associated with this variable namely slim, normal, pudgy, and fat. However, the dunes within the sample do not cover the entire spectrum with a frequency that is suitable for model training. Therefore, the median $\frac{a}{c}$ ratio of the data set was used to separate dunes into categories where the $\frac{a}{c}$ ratio is small and cases where it is large. To avoid association with a specific horn elongation, examples from both left and right elongated horns were used in each category (Figure 5.6).

[6] A morphometric variable is a variable such as width and length that is used to describe a specific property of a barchan's shape.

Table 5.2: A summary of the different tasks, along with the classification categories, that the CNN is expected to complete.

Task	Property	Description	Classification Categories
Classifying barchan shape	Shape	This is a frequently used descriptor of barchan shape (e.g. Barnes, 2001; Lorenz *et al.*, 2013) and has been linked to differing migration rates (Hamdan *et al.*, 2016) and predetermining the elongation of the horns (Tsoar and Parteli, 2016). Changes in the $\frac{a}{c}$–ratio can result from changes in the sediment supply to the dune (Parteli *et al.*, 2014).	Large Small
Classifying the side of elongation	Asymmetry	Refers to which of the two horns is elongated relative to the other. Can be used as an indicator of predominantly bimodal wind influence if most of the dunes have the same horn elongated (Tsoar and Parteli, 2016), or collisions predominating if alternating elongations are present (Zhang *et al.*, 2018).	Left Right
Classifying the side of widest horn	Asymmetry	Refers to which horn of the barchan is the widest. This has an influence on the amount of sediment leaving that horn (Elbelrhiti *et al.*, 2008) which can lead to an asymmetric sediment supply on downwind barchans.	Left Right
Classifying the side of lateral extension	Asymmetry	Refers to which side of the barchan is displaced the furthest from the longitudinal axis. Simulations have shown that if barchans occur on an inclined slope, the horn will be laterally extended in the down-slope direction (Parteli *et al.*, 2014). Some observational evidence has also been provided by Finkel (1959).	Left Right
Classifying the magnitude of elongation	Asymmetry	Refers to how much longer the elongated horn is with respect to the shorter horn. Can be used to infer different sediment supply rates across the stoss slope (Parteli *et al.*, 2014) or a collision between two barchans (Bourke, 2010).	Large Small
Classifying bilateral asymmetry	Asymmetry	Refers to the procrustes distance between the barchan and its mirror image. Combines the previous measures into a single continuous variable.	More asymmetrical More symmentrical

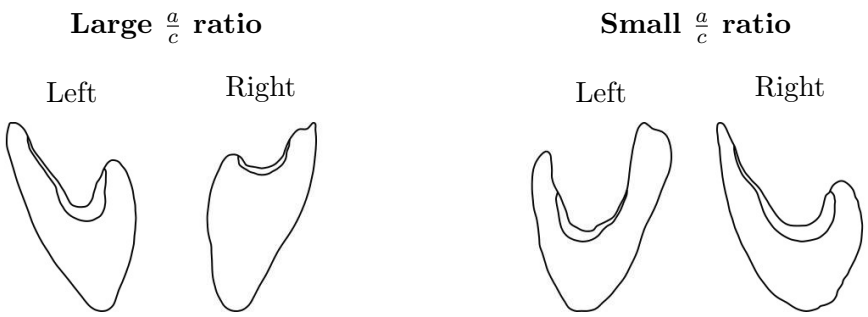


Figure 5.6: Example of imagery used to train a model to categorise barchans based on the $\frac{a}{c}$ ratio. Samples of both left and right elongated dunes were used (but these subcategories did not form part of the classification process).

5.6.2 Side of Elongation

This task requires the model to distinguish between barchans based on which of the horns (left or right) is elongated. Only two categories are, therefore, possible based on the nature of the data (Figure 5.7). The training and evaluation contained examples of barchans where the side of elongation is clearly discernible and examples where it is more subtle. These were, however, not used as subcategories due the small data set. Nevertheless, it does permit conclusions to be drawn regarding the model's performance on detecting subtle differences.

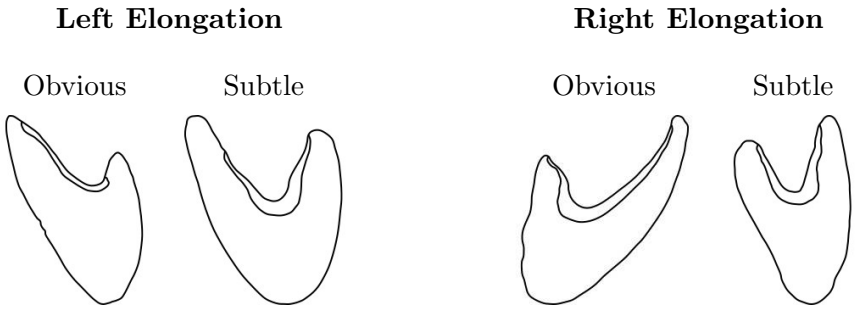


Figure 5.7: Examples of barchans that were used to train and evaluate the neural network's performance on task 1. Note that the categories "obvious" and "subtle" were not used but are placed here for context.

5.6.3 Classifying Side of Widest Horn

The width of each horn of the barchan can be used to determine on which side of the barchan the widest horn occurs. This describes barchan asymmetry according to the criteria presented in Elbelrhiti *et al.* (2008). Similar to the other tasks, two categories were created classifying an individual barchan as having the widest horn on either the left or the right side of the dune (Figure 5.8). Similar to the task on the side of elongation, examples of subtle differences in length (i.e. where there is a small difference in horn width) were combined with examples of larger and more obvious differences in width.

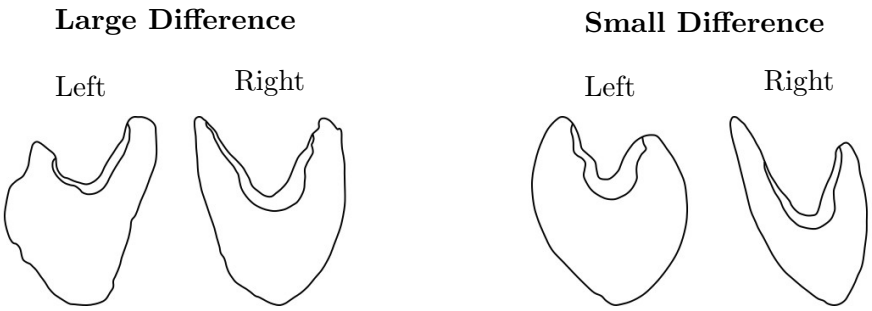


Figure 5.8: Examples of imagery from the side of the widest dune task. The subcategories "large" and "small" were not used but are provided for additional context.

5.6.4 Classifying Side of Lateral Extension

This task is based on the modelling work of Parteli *et al.* (2014) who demonstrated the effect of slope angle on the lateral extension of a barchan horn. In their work, it was possible to generate barchans that had an asymmetry resulting only from the lateral extension. However, no such cases are present in the sample of barchans used in this study. Therefore, the dunes inevitably show a combination of lateral extension and an asymmetry that results from other factors such as horn elongation. Two categories "left" and "right" are used to differentiate between barchans where the lateral extension is on the left and right side respectively (Figure 5.9). The data also consisted of a combination of obvious cases, where the side of lateral extension is clearly visible, and those where the difference is more subtle.

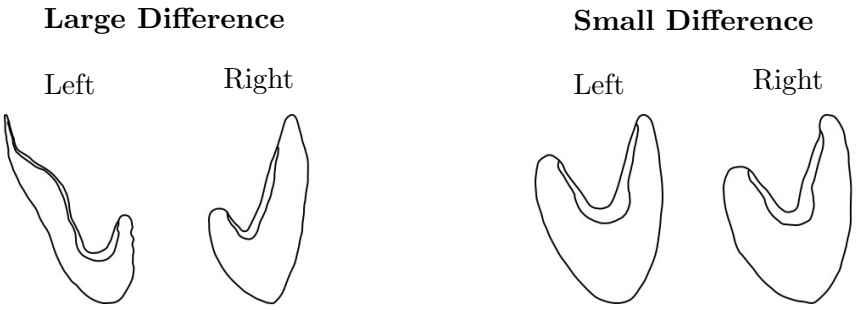


Figure 5.9: Examples of imagery from the side of the widest horn task. The subcategories "large" and "small" were not used but are provided for additional context.

5.6.5 Classifying Differences in Horn Length

To assess the model's capacity to discern the magnitude of horn length differences, two categories were created, "large" and "small". Since there is no literature with explicit categories of the morphometric variable, custom categories were created based on the median value of the horn length differences within the sample. Dunes that had a difference in horn length larger than the median were placed into the "large" category and the other dunes into the "small" category.

For both cases, examples of left and right horn elongation was used (Figure 5.10) in order to prevent the model from associating a specific elongation with the correct classification.

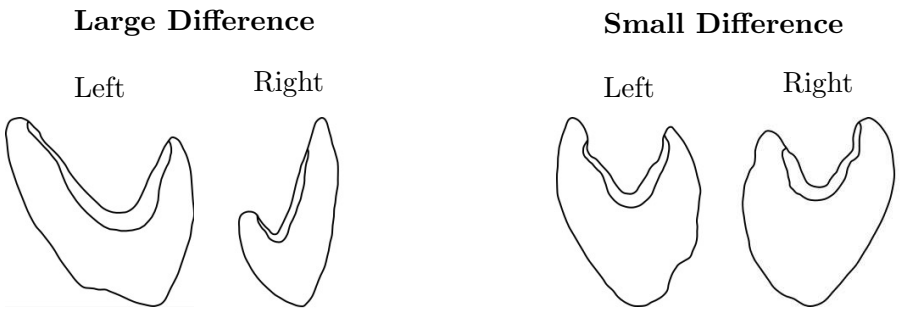


Figure 5.10: Examples of imagery used to train and evaluate the model's capacity to correctly classify the magnitude of difference in horn lengths. The subcategories "left" and "right" were not used but are provided for additional context.

5.6.6 Classifying Bilateral Asymmetry

This task uses a custom description of asymmetry, bilateral asymmetry, that was develop specifically for this project. This method expresses barchan asymmetry as the Procrustes distance between a dune and its mirror image. For a perfectly symmetrical barchan, the Procrustes distance to its mirror will be zero. Deviations from this symmetry is expressed by increasing values of the Procrustes distance. The manner in which this metric is calculated effectively allows for all of the other asymmetry metrics to be incorporated into a single continuous variable. This makes it possible to evaluate whether the model can factor in several different manifestations of asymmetry that act together. Using the median Procrustes distance of the sample, two categories were created with examples of left and right elongated barchans being included to avoid bias in the samples (Figure 5.11).

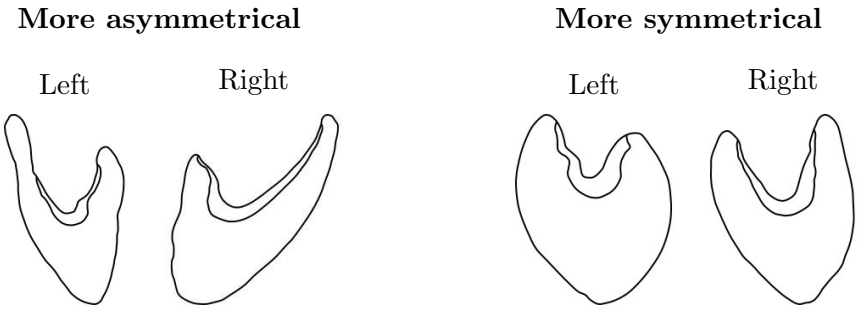


Figure 5.11: Examples of imagery used to train and evaluate models to detect bilateral asymmetry. The "more asymmetrical" category contains dunes that have a procrustes distances higher than the median while the "more symmetrical" category have procrustes distances below the median. The subcategories left and right were not used and are provided for additional context..

This section describes the different tasks that a CNN is expected to complete (Objective 2). These tasks represent a fairly broad overview of the types of morphological data that are used in barchan research, although it is by no means exhaustive.

5.7 Label Generation

Label generation is the process whereby labels are generated for all of the training and testing data. Since the learning paradigm followed here is supervised learning[7], the correct labels of each image is a requirement for training. In this context, the labels refer to the morphometric attributes that are of interest, such as the magnitude of difference in horn lengths, the $\frac{a}{c}$–ratio etc.

5.7.1 Barchan Shape

Two different approaches have been used to calculate the distance between the horns (i.e. the c variable of the $\frac{a}{c}$–ratio calculation). Sauermann *et al.* (2000) expresses this distance as the sum of perpendicular distances between the tip of the horn and longitudinal axis of the dune. In contrast Norris (1966) uses the Euclidean distance between the two horn tips. The method proposed by Sauermann *et al.* (2000) is used since the alternate method is considered to artificially decrease the values for asymmetric barchans thereby creating a bias to the classes with low $\frac{a}{c}$ values.

To calculate the $\frac{a}{c}$ ratio from the landmark data (Figure 5.12), two triangles (ABC and ABD) are constructed using landmarks $6, 18, 2$, and 10 respectively (Figure 3.6). Note that, ass stated earlier, landmark 16 was used in some instances where landmarks 16 and 17 occupied the same position. Using the coordinates of each landmark, the surface area of each triangle can be calculated using Heron's formula (Eq. 5.3), where A is the area of the triangle, a is the length of the line $\overline{AB}$, b is $\overline{BC}$ or $\overline{BD}$, c is $\overline{AC}$ or $\overline{BD}$, and $s = a + b + c$. All distances are calculated using the Euclidean distance formula. Once the area of the triangles are known, the height of each triangle $\overline{CE}$ and $\overline{DF}$ can be calculated using equation (Eq. 5.4). By summing the heights of each triangle, the distance between the horns can be calculated. The median value across the entire data set was used to separate the samples into one of two categories. The median was chosen specifically since it, in most cases, separates the data set into two equal portions.

$$A = \sqrt{s(s-a)(s-b)(s-c)} \qquad \text{(Eq. 5.3)}$$

$$h = \frac{2A}{b} \qquad \text{(Eq. 5.4)}$$

5.7.2 Side of Elongation

The side of elongation is derived from the horn that has experienced the most elongation (Figure 5.13). Two variables are needed to categorise the barchan as being either left or right side

[7] This will be elaborated upon later in the model training section.

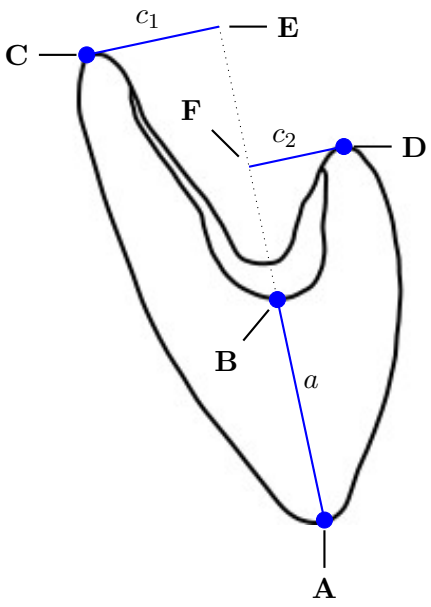


Figure 5.12: The method used to determine the $\frac{a}{c}$–ratio of a barchan ($c = c_1 + c_2$).

elongated: the length of the right horn and the length of the left horn. Here, the length is calculated with reference to landmark 16 (or 18) (which is denoted as B in Figure 5.13). This was done in order to measure the length of the barchan with respect to the brink line. Two triangles BCF and BDE are constructed. The lengths $\overline{CF}$ and $\overline{DE}$ was calculated during the classification of barchan shape label generation (§5.7.1). Note that the coordinates of the points E and F were not calculated. Additionally, the lengths $\overline{BC}$ and $\overline{BD}$ can be calculated using coordinate geometry. Using the Pythagorean theorem, the lengths of $\overline{BF}$ and $\overline{BE}$ can be calculated. These correspond to the lengths of the left (l_H) and right (l_R) horn respectively. After the calculations, the dunes were separated into two groups based on which of the two values, l_H or l_R, is the largest.

5.7.3 Side of Widest Horn

To determine which side of the barchan had the widest horn, the width of both horns were calculated. For the left horn, landmarks 4 and 1 were used, and landmarks 11 and 8 were used for the right horn (Figure 5.14). This is approximately equivalent to the method used by Elbelrhiti *et al.* (2008). These distances were calculated perpendicular to the longitudinal axis of the dune. A key challenge here is that it is not straightforward to construct right–sided triangles to obtain the answer (as was the case for the classifications of barchan shape and side of elongation tasks). An alternative approach was therefore used.

Using the landmarks, the longitudinal axis can be calculated using (Eq. 5.5). In essence, this involves calculating the equation of a line that bisects the angle made between the toe of the barchan and the two horn tips (which corresponds to the geometric construction outlined in

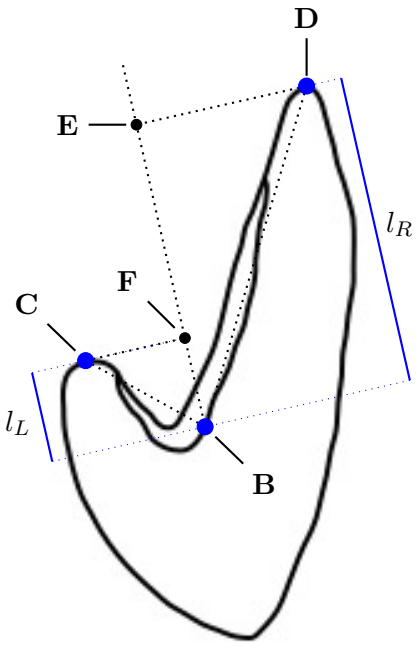


Figure 5.13: The determination of barchan left (l_H) and right (l_R) horn lengths. Measurements made parallel to the longitudinal axis. Landmark B corresponds to either landmark 16 or 18 (See §3.3.1 for further details). The coordinates of the points E and F were not calculated but are indicated here for clarity (see discussion for further details).

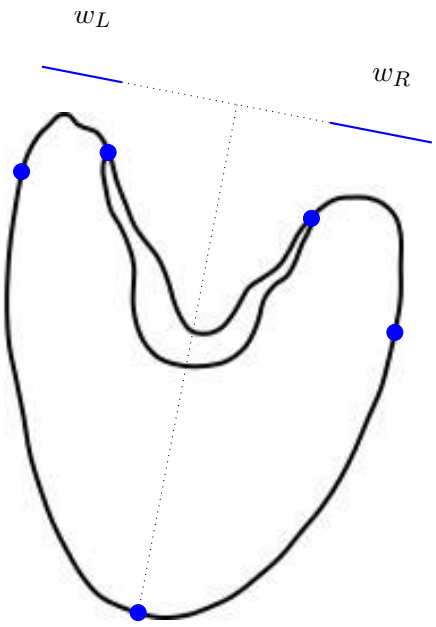


Figure 5.14: The method used to determine the width of the left (w_L) and right (w_R) horns. The width is expressed perpendicular to the longitudinal axis.

§3.3.1). The variables a_1, b_1 and c_1 correspond to the line connecting the toe with the left horn while the variables a_1, b_1 and c_1 represents the line connecting the toe with the right horn tips. The equation of the longitudinal line is then described using the variables a_n, b_n and c_n.

$$\begin{aligned} \frac{a_1x + b_1y + c_1}{\sqrt{a_1^2 + b_1^2}} &= -\frac{a_2x + b_2y + c_2}{\sqrt{a_2^2 + b_2^2}} \\ \therefore a_n &= a_1\left(\sqrt{a_2^2 + b_2^2}\right) + a_2\left(\sqrt{a_1^2 + b_1^2}\right) \\ \therefore b_n &= b_1\left(\sqrt{a_2^2 + b_2^2}\right) + b_2\left(\sqrt{a_1^2 + b_1^2}\right) \\ \therefore c_n &= c_1\left(\sqrt{a_2^2 + b_2^2}\right) + c_2\left(\sqrt{a_1^2 + b_1^2}\right) \end{aligned} \quad \text{(Eq. 5.5)}$$

Once the equation of the longitudinal axis is known, it is possible to calculate the width of the left horn. First, the distance between the longitudinal axis and landmark 4 is calculated using (Eq. 5.6). Second, the distance is calculated between the longitudinal axis and landmark 1. The width of the left horn is then equal to the difference between the first and second calculation. The process is repeated for the right horn using landmarks 11 and 8.

$$Distance = \frac{|a_nx_1 + b_ny_1 + c_n|}{\sqrt{a_n^2 + b_n^2}} \quad \text{(Eq. 5.6)}$$

5.7.4 Side of Lateral Extension

Determining the side of maximum lateral extension (Figure 5.15) followed the same basic process as determining which side had the widest horn. Using the equation of the longitudinal axis, the perpendicular distance to the most lateral points on the barchan (landmarks 4 and 8) are calculated. The side of the longest lateral extension is then the side of the barchan (either left or right) which has the largest value.

5.7.5 Magnitude of Elongation

The magnitude of elongation is calculated as the difference between the left and the right horns. Since the lengths are known (based on the classification of the side of elongation task), this is a simple procedure. Since the process is automated, the absolute value (Eq. 5.7) was used in order to avoid results with a negative value.

$$|x| = \begin{cases} x & \text{if } x \geq 0 \\ -x & \text{if } x < 0 \end{cases} \quad \text{(Eq. 5.7)}$$

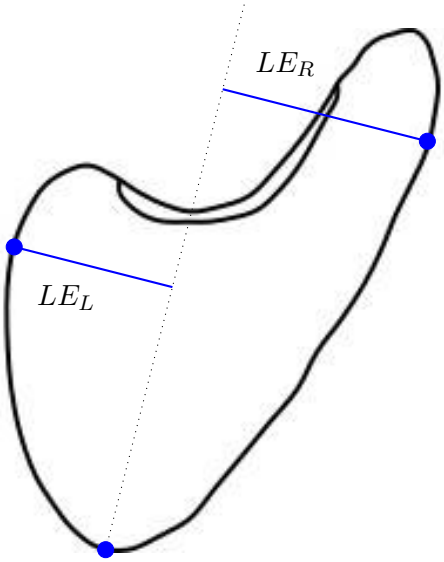


Figure 5.15: The method used to determine which side of the barchan had the most lateral extension (LE_L vs. LE_R). The amount of lateral extension is measured perpendicular to the longitudinal axis.

5.7.6 Bilateral Asymmetry

The extent of bilateral asymmetry is expressed based on the procrustes distance (ρ_p) between a barchan and its mirror image. Since this is a new expression of asymmetry, there are no prior studies that can be used to generate commonly used labels. Therefore, the median procrustes distance is used as a reference point. Dunes that have a ρ_p less than the median are classed as being more symmetrical (due to having lower ρ_p values). Barchans that have a ρ_p larger than the median are classed as more asymmetrical since their ρ_p values are higher. For the bilateral asymmetry task, several dunes had a ρ_p value equal to the median. These ae placed in the same class since there is no objective reasoning that can be applied to separate them into different categories. The result of this is that the training data is slightly skewed in favor of the symmetric barchans (i.e. the number of images in the training data for symmetrical class (64) is larger than that of the asymmetrical class (48)). Although this does make the data imbalanced, albeit marginally, it is not expected to substantially affect the performance of the models.

5.8 Data Augmentation

This section provides a brief overview of the data augmentation process. It was used in this study to increase the size of the data set and thereby improve the ability of the models to classify the imagery. Data augmentation refers to a set of methods that transform the data without altering the nature of the data (Gu *et al.*, 2018). In other words, it is a set of geometric transformations carried out that applies a transformation across the entire image but still retains the subject of the image in a recognisable form. These transformations include: mirroring the image, rotating

the image, translating the image, flipping the image either horizontally or vertically, adjusting the contrast, applying a filter such as sharpening or blurring, applying a shear operation to the image, histogram equalization, and adding Gaussian noise to the image (Nagle-Mcnaughton *et al.*, 2020; Ji *et al.*, 2020; Gu *et al.*, 2018; Maxwell *et al.*, 2020). One of the reasons for using augmentation strategies is to prevent the model from overfitting (Maxwell *et al.*, 2020; Nagle-Mcnaughton *et al.*, 2020; Gao *et al.*, 2021). In other words, it helps the model to generalize better. Another reason, which is used in this study, is to artificially increase the sample size of the images (Castelluccio *et al.*, 2015; Jiang *et al.*, 2018; Zhu *et al.*, 2017; Carbonneau *et al.*, 2020; Huang *et al.*, 2018; Gu *et al.*, 2018).

Training "from scratch" models requires a large data set in order to ensure good performance (Bhuiyan *et al.*, 2020; Goodfellow *et al.*, 2016). Further, in cases where the training data is limited, poor performance can result due to overfitting (Mahdianpari *et al.*, 2018). However, it needs to be mentioned that the ideal sample size is not known and, most likely, will be highly dependent on the classification task and the type of images used during the training process. It is also inevitable that situations can arise where the subject of the study does not have adequate training data available (e.g. Liu *et al.*, 2021). In cases where this occurs, two strategies can be followed. First, data augmentation can be used (as discussed previously). This works by providing additional images that are dissimilar from the other images due to the transformation process. A second approach is using transfer learning (Azzaoui *et al.*, 2020; Wilhelm *et al.*, 2020). In these cases, models that were trained on larger data sets can be used for applications where the data is limited (see §4.6 for more details).

In this study, the image augmentation was limited to the horizontal flipping of the image thereby creating a mirror image. This was in order to ensure that the barchans within the image still retained the same orientation as the original image. This is the side on which an asymmetry is then altered (e.g. if the side of elongation was on the left in the original image it will be on the right in the mirror image), the mirrored images were classified separately from the original image.

5.9 Training, Validation and Testing Data

Following Kattenborn *et al.* (2021) and Chollet and Allaire (2018), the images in this study was separated into three different data sets: training images, validation images, and testing images. Training data, as the name suggests, is used to train the network by supplying it with labelled images. These are then used during backpropagation to update the weights of the connections (see Section 4.5). Validation data is similar in that it is used to assess the accuracy of the model, but the weights are not updated based on the results. The purpose of the validation data is,

Table 5.3: Data split (i.e. the number of images) between training, validation, and testing data for each of the tasks.

Task	Training Data		Validation Data		Testing Data	
	Class 1	Class 2	Class 1	Class 2	Class 1	Class 2
Barchan shape	56	56	24	24	10	10
Side of elongation	56	56	24	24	10	10
Side of widest Horn	56	56	24	24	10	10
Side of lateral Extension	56	56	24	24	10	10
Magnitude of elongation	56	56	24	24	10	10
Bilateral asymmetry	48	64	24	24	10	10

essentially, to provide a measure of the model's ability to generalize. This makes it possible to adjust the parameters of the model without having to use the testing data. Like validation data, the testing data does not result in an update of the model's weights and biases. However, whereas the model has access to the labels of the training data, it does not have access to the labels of the testing data. The testing data allows the performance of the model to be assessed based on its evaluation of data that it has never seen before. It can therefore be regarded as a measure of the model's ability to generalise.

The split of the data for each task is provided in Table 5.3. For all of the tasks, a binary classification was followed which means that there were two categories (§5.6). In order to avoid bias, and to ensure that the number of images in each categorry were equal, the images were split based on their relation to the median value (i.e one category contained images where the morphometric variable in question was below the median and the other category for those above the median). In all cases except one this resulted in a balanced data set. However, for the classification of bilateral asymmetry task, the data set was not equally distributed. In this case, there were several barchans that had the same value as the median. Because there is no objective reason to split these images equally between the two classes, and to prevent the same label from appearing in two different classes, all of the ties were placed together into the same class. Images within each category was randomly split into the three data sets (training, validation, and testing) needed for this study.

5.10 CNN Creation

This section provides an overview of the hyperparameters (see Section 4.4.1) used in the construction of the CNNs used in this study. Two broad categories of CNN are discussed: CNNs that are built "from scratch" and networks that make use of transfer learning. Although the U-Net architecture has been frequently used in geomorphology (Section 4.2.1) it must be pointed out that this technique is not suited to the aim of this study. Because U-Net is a semantic segmentation model, it is, in principle, capable of identifying where a barchan is within an image.

However, aside from identifying where the barchan is located, it is not possible to classify the "type" of barchan present within the image. Since the barchan cannot be classified, it is not possible to extract morphometric data from it.

5.10.1 From Scratch CNNs

Model that are trained "from scratch" offer the largest amount of flexibility to a user. It is possible to adjust all of the hyperparameters to the needs of the application. Given the large amount of hyperparameters and their possible values (§4.4.1), it is not possible to create a set of CNNs based on all possible hyperparameter permutations. Instead, the emphasis is on modifying two sets of hyperparameters that have been shown to play a significant role in model performance, namely model depth and the presence of regularizations. Network depth can, in principle and subject to the availability of computational resources, be any positive integer value. Therefore, for the sake of practicality, it was decided to use models of three depths: 1 layer, 2 layers, and three layers. The motivation is that since the study is a proof of concept, and not a final model, it is more important to collect data on performance enhancing techniques. For the regularizations, only three options are available (L1, L2, and Dropout) and all three were used for each model depth configuration (Objective 6). In other words for each depth, four models were created: one without a regularization, and the remaining three with the three regularizations mentioned. In using this combination, it is possible to determine if CNNs can accurately classify barchans and how the hyperparameters influence the model's performance. Further details on the creation of the "from scratch" models is given below.

5.10.1.1 Model Creation

The architecture of the baseline model is illustrated in figure 5.16. This model consists of a single convolution layer that has 32 3×3 kernels. Therefore, 32 different feature maps are produced (one per kernel). The padding applied to the image prior to the convolution operation ensured that the resulting feature maps had the same dimensions as the input layer. This was to ensure that the barchan's outline was retained in the resulting feature maps. The ReLU activation function was used for the convolution layer and the subsampling was carried out using a 2×2 max-pooling layer with a stride length of 2. The resulting sub-sampled region was then flattened and fully connected to two output neurons, one for each category, that used the softmax activation function.

By adding additional layers, the depth of the model is increased. As mentioned in Chapter 4, it is widely considered that increasing the depth of the model improves its accuracy (Wan *et al.*, 2018; Simonyan and Zisserman, 2015; Shakya *et al.*, 2021; Kattenborn *et al.*, 2021) at

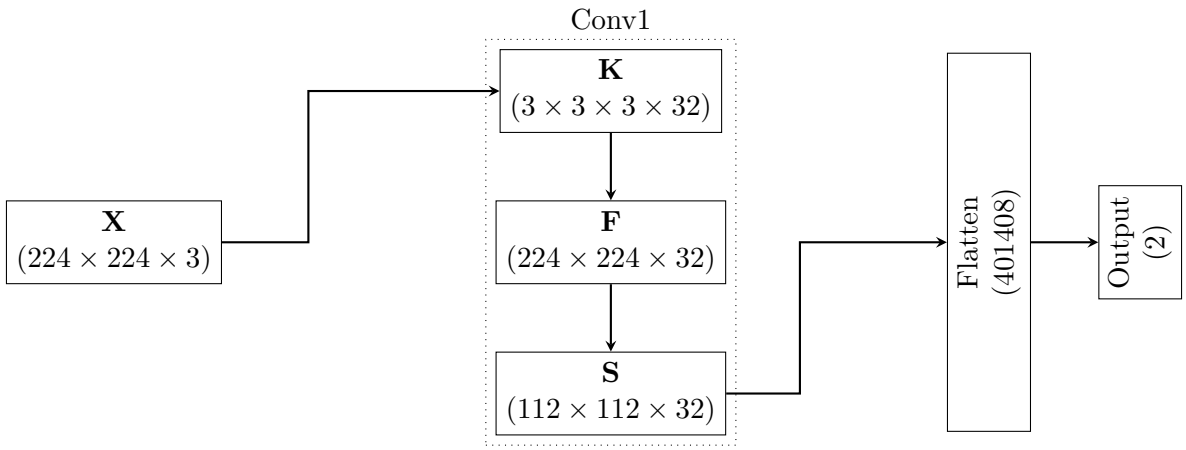


Figure 5.16: The baseline model. The input (**X**) convolves with the kernel (**K**) producing a feature map (**F**) which is sub-sampled by a max-pooling layer creating a sub-sampled feature map (**S**). Afterwards, it is passed to flattening layer before passing to the output layer. The numbers in parenthesis reflect the dimensions of the data along the model.

the expense of increased computational costs (Gu *et al.*, 2018; He and Sun, 2015). Due to the absence of any guidelines on the number of optimal layers, and since overly increasing the layers can result in decreased performance (He *et al.*, 2016; He and Sun, 2015), a conservative approach was followed and three depths were used: 1 layer (which is the baseline), 2 layers (Figure 5.17) and 3 layers (Figure 5.18).

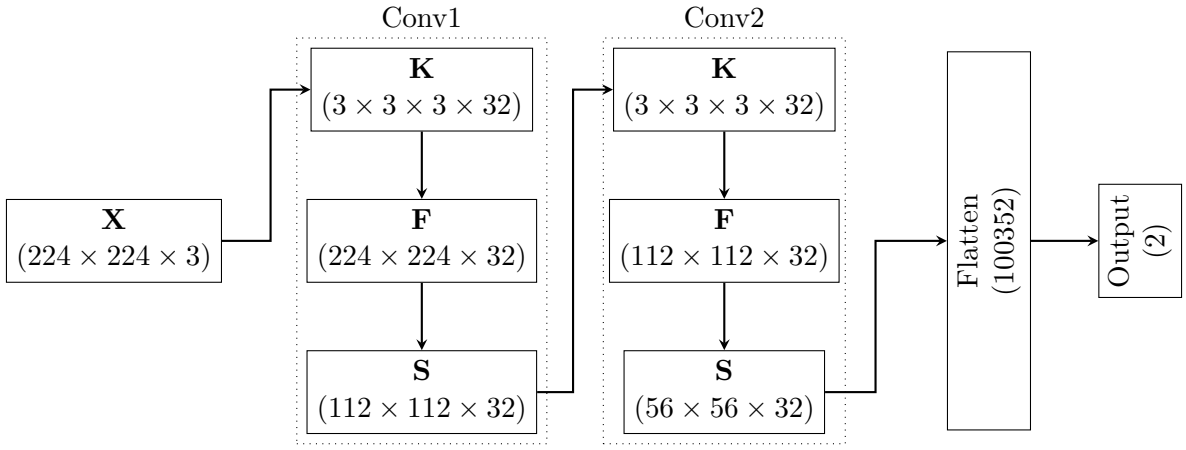


Figure 5.17: The baseline model with an additional convolution layer. The numbers in parenthesis reflect the dimensions of the data along the model.

The L_1 and L_2 regularziers were used as part of the kernel (**K**) optimization within each convolutional layer. The effect of this is that, during the training, as the model is being optimised, the additional term (Eq. 4.8) and (Eq. 4.9) created by the regulizers is added during the update of the kernel only, and not the entire model. As a result, the kernel is "forced" to become more generalized. This prevents a small number of pixels from dominating the feature extraction and, consequently, the image classification (Yosinski *et al.*, 2015). This is beneficial especially when model generalisation is considered. If the kernels are allowed to have only a few

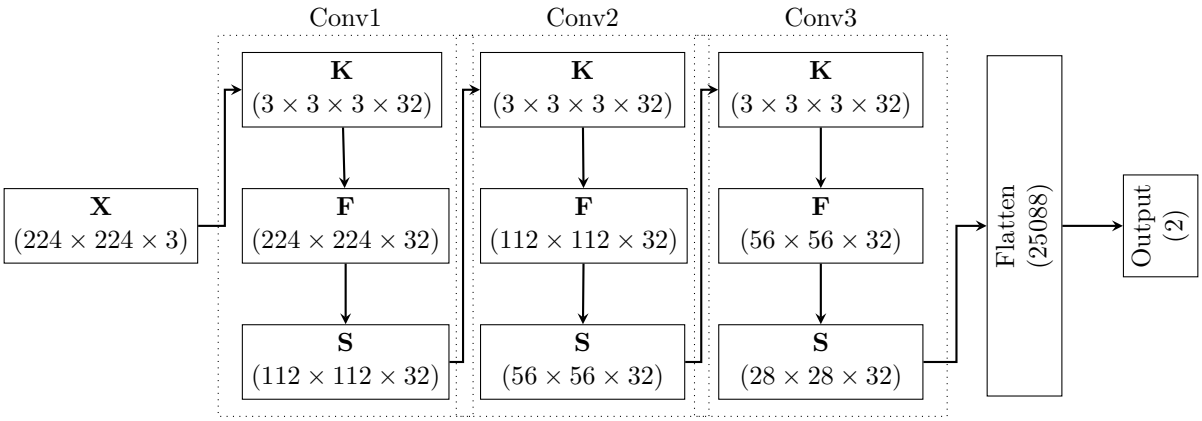


Figure 5.18: The baseline model with two additional convolution layers. The numbers in parenthesis reflect the dimensions of the data along the model.

pixels making the most contribution to the feature extraction process, it may results in missing the detection of other features that may improve its ability to generalize to cases the model was not trained on. The dropout layer is not a layer in the sense of the ones defined thus far. It is a layer that, essentially, serves only to pass data from preceding layers to layers deeper into the model. The key difference is that each neuron within a dropout layer has a certain probability of outputting a value of 0 to the next layer. For all of the "from scratch" models, the dropout layer was placed immediately following the convolution layer with a dropout probability of 0.1. This means that, at any time, 10% of the neurons from the preceding layer would output a value of 0.

5.10.2 Transfer Learning

Models trained from scratch allow the user to specify all of the hyperparameters and allows for the most fine-tuning. However, their performance is dependent on the amount of data that they are trained on (Goodfellow *et al.*, 2016). This can make them especially prone to overfitting (Mahdianpari *et al.*, 2018) when the diversity of imagery to which they are exposed to is low. This can, to an extent, be overcome through the use of transfer learning (de Lima and Marfurt, 2020; Azzaoui *et al.*, 2020; Bhuiyan *et al.*, 2020; Wilhelm *et al.*, 2020).

Transfer learning is a technique that uses the learned feature extraction capacity of a model that was trained on a primary task and applying it to a secondary task (de Lima and Marfurt, 2020; Azzaoui *et al.*, 2020). For this study, the feature extraction capacity of two well known CNN models, VGG16 (Simonyan and Zisserman, 2015) and ResNet50 (He *et al.*, 2016), were used. The choice is based on the popularity of these two models in the CNN literature (Diaz-Pinto *et al.*, 2019; Oo and Oo, 2019; Ilyas *et al.*, 2019; Bendjillali *et al.*, 2020; Mukti and Biswas, 2019; Sun *et al.*, 2019; Agarwal and Mittal, 2019; LaVezzi *et al.*, 2020; Raghu *et al.*, 2020; Mahdianpari *et al.*, 2018; Pradhan and Lee, 2010). These feature extractors were then joined to

a custom classifier in order to classify the outputs from the feature extractor into the categories developed for this study. The classifier was the only part of these pretrained models that was trained. This preserved the feature extraction capacity of the networks and, essentially, only trained the classifier.

5.10.2.1 Model Creation

As mentioned earlier, two pretrained models were used to test the performance of transfer learning:VGG16 and ResNet50. Since VGG16 has a large computaional footprint, it is considered to be ideal for applications where a high amount of accuracy is required, albeit at an increased computational demand (Agarwal and Mittal, 2019). However, ResNet50 is considered to be easier to train (Oo and Oo, 2019).

The structure of the VGG16 model is similar to the "from scratch" models described earlier. The model consists of 5 "blocks" of convolutional layers (with ReLU activation), together a max-pooling layer for each block, and three fully connected layers with a softmax output in the output layer (Pradhan *et al.*, 2020; Mahdianpari *et al.*, 2018). There are several differences between the VGG16 model and the "from scratch" models. First, VGG16 has 13 convolutional layers (Mahdianpari *et al.*, 2018) while the "from scratch" models have a maximum of three. Second, the VGG16 model has three fully connected layers (Pradhan *et al.*, 2020) while the "from scratch" models only have a single fully connected layer. Lastly, the weights of the model are not randomly initialised but are derived from training on the ImageNet database (Chevtchenko *et al.*, 2018). To use the VGG16 model for the barchan classification task, the convolution blocks and their weights were retained and the the fully connected layers were removed. The same fully connected layer configuration that is present on the "from scratch" models is then added to the VGG16. The model is then trained on the same data as the "from scratch" models. It is important to note that the weights of the convolutional blocks were prevented from updating during the training. This is done in order to ensure that the feature extraction ability of the VGG16 model is retained. Only the fully connected layer was allowed to train and have its weights updated. A similar approach was followed by Chevtchenko *et al.* (2018) and LaVezzi *et al.* (2020).

The ResNet50 model was initially developed to solve the vanishing gradient problem (Nagle-Mcnaughton *et al.*, 2020). The vanishing gradient problem occurs during back-propagation in neural networks where the derivative of the activation function results in weight updates that are so small that they are, in effect, no longer trainable (Kattenborn *et al.*, 2021). ResNet50 attempts to prevent this by making use of skip connections (Wang and Anderson, 2018) and residual learning (Weng *et al.*, 2020). In residual learning the output of an earlier layer bypasses

Table 5.4: The training parameters used for the "from scratch" models and the transfer learning models.

Training Parameters	Setting
Optimizer	Adam
Learning rate	0.0001
Loss function	Binary Crossentropy
Batch size	2
Epochs	10

the intermediate convolution operations, and forms part of the input into a layer deeper into the network (Chollet and Allaire, 2018; De Cesarei *et al.*, 2021). ResNet50 contains 49 convolutional layers and one fully connected layer (De Cesarei *et al.*, 2021). The same training approach as for VGG16 was used where the fully connected layer was removed and replaced with the same fully connected layer used in all of the other models. It is important to note that, even though the ResNet50 model only has a single fully connected layer, it was necessary to replace it with a custom one since the number of categories in this study is different from that originally used in the development of ResNet50.

5.11 Model Training

This section provides details on the hyperparameters that were used in the training of the models. A total of 14 models were evaluated for this study. These consist of two models that were pretrained (VGG16 and ResNet50) and 12 models that were created "from scartch". The training parameters for the models are provided in Table 5.4. The Adam optimizer, with a learning rate of 0.0001, was chosen since it is considered to be the most popular optimizer (Kneusel, 2022) and is able to adapt the learning rate during the training process (Bera and Shrivastava, 2020). The learning rate was chosen to be low in order to prevent the loss function from oscillating around a local minimum (§4.4.1.6). In order to allow for better convergence, a batch size of 2 was chosen (§4.4.1.7). This is the same size used by Bhuiyan *et al.* (2020). Because the model only needs to classify an image into two classes, the binary crossentropy loss function was selected. Lastly, since CNNs are initialised with random weights, the initialisation, training, and evaluation stages were repeated 30 times for each model. This makes it possible to store the weights and performance of each iteration. These averages across all iterations of a given model were used to evaluate the model's performance.

5.12 Model Evaluation

Several different metrics were used to evaluate (Objective 7) the performance of the models: accuracy (ACC), F_1-score, and Matthews Correlation Coefficient (MCC). These metrics make use of a confusion matrix (Eq. 5.8), where TP and TN refers to true positive and true negative cases

respectively while FP and FN refer to false positive and negative cases. These designations refer to the agreement between the classifications made by the model and the actual classifications based on the image labels[8]. Therefore, TN values refer to the number of cases where a barchan from class 1 was correctly identified as belonging to class 1 by the CNN. Similarly, TN refers to those cases where a barchan from class 2 was correctly placed into the class 2 category by the CNN. A false positive (FP) occurs when the model classified an image as being in category 2 while it should have been in category 1, and an FN occurs when the model classified the image as category 1 while it should have been category 2.

$$\begin{array}{c} \\ Class\ 1_{(Observations)} \\ Class\ 2_{(Observations)} \end{array} \begin{array}{c} \begin{array}{cc} Class\ 1_{(CNN)} & Class\ 2_{(CNN)} \end{array} \\ \begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix} \end{array} \quad \text{(Eq. 5.8)}$$

ACC is a common metric which expresses the number of predictions that correctly identified both positive and negative instances (Brown, 2018). This metric is frequently used in binary classification tasks (Hossin and Sulaiman, 2015). Here, a high accuracy score represents a situation where the number of images that were correctly classified as belonging to either class is high relative to the number of predictions made by the model. The equation for ACC, following (Hossin and Sulaiman, 2015), is given in equation (Eq. 5.9).

$$ACC = \frac{TP + TN}{TP + FP + TN + FN} \quad \text{(Eq. 5.9)}$$

The F_1-score (Eq. 5.10) (Brown, 2018) represents the harmonic mean between recall (True Positive Rate (TPR)) and precision (Positive Predictive Value (PPV)) (Hossin and Sulaiman, 2015; Maxwell *et al.*, 2020). TPR is the proportion between the TP predictions and all the positive predictions made by the model (Bellows *et al.*, 2011). Precision, also known as the positive prediction value (PPV) (Bellows *et al.*, 2011), is the number of positive predictions made by the model that are true positives (Hossin and Sulaiman, 2015). The equation for the F_1 score, following (Brown, 2018), is given in equation (Eq. 5.10). According to the surface plot of potential F_1-scores (Figure 5.19), both the TPR and the PPV scores of the model need to be high in order to achieve a high F_1–score. This means that a high F_1-score suggests that the model had a large portion of TP predictions and low amounts of FP and FN cases. This follows from equation (Eq. 5.10) where the presence of FN and FP in the denominator of PPV and TPR indicates an inverse relationship. Unfortunately, the surface also suggests that it is not possible to determine from the F_1-score alone where shortcomings occur. In other words, it

[8]For an explanation on how these labels were generated, see Section 5.7.

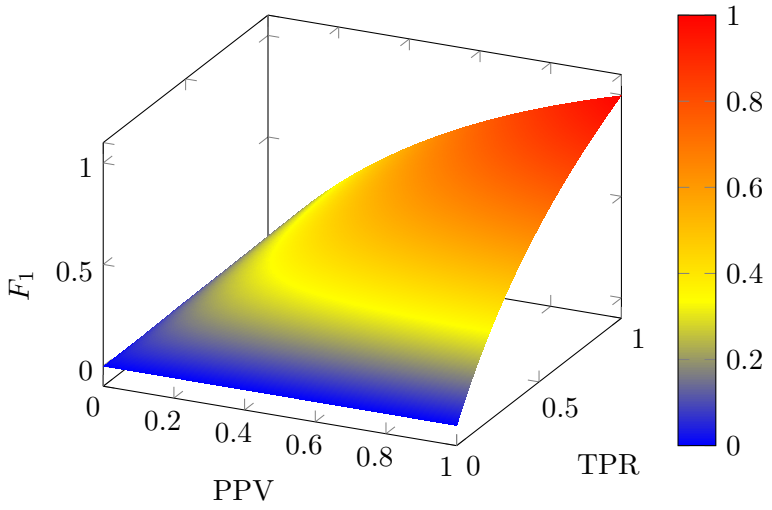


Figure 5.19: The value surface of the F1 metric. In this example, PPV values are plotted along the x-axis, TPR values along the y-axis, and the resultant F1 score along the the z-axis.

can either be from a high false positive rate, a high false negative rate or a combination of both. Notwithstanding this, the F_1-score is still widely used in machine learning (Chicco and Jurman, 2020).

$$F_1 = 2 \times \frac{PPV \times TPR}{PPV + TPR}$$
$$PPV = \frac{TP}{TP + FP} \qquad \text{(Eq. 5.10)}$$
$$TPR = \frac{TP}{TP + FN}$$

Both the ACC and the MCC (Eq. 5.11) use the same variables (Brown, 2018; Hossin and Sulaiman, 2015), but the MCC metric is considered to be more reliable (Chicco and Jurman, 2020). This is because it penalises Type-I (false positive) and Type-II (false negative) errors more than the ACC metric does (Brown, 2018). Essentially, it serves as an indication of how well the actual data correlates with the predicted data with values of +1, zero and -1 indicating perfect agreement, random predictions and total disagreement respectively (Tharwat, 2018). The equation for the MCC metric is given in (Eq. 5.11) (Brown, 2018). A problem with the MCC metric is that, under certain conditions, it cannot be defined due to imbalances in the confusion matrix (Chicco and Jurman, 2020). This happens in cases where both TP and FN are equal to zero causing the denominator to become zero. In cases were this occurred, it was removed from the data and the averages calculated with the adjusted sample size.

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP) \times (TP + FN) \times (TN + FP) \times (TN + FN)}} \qquad \text{(Eq. 5.11)}$$

The performance of the network is assessed based on the following criteria:

1. Does the prediction accuracy exceed those found by (Lloyd *et al.*, 2002) for human operators classifying Level II imagery? Although the classification tasks investigated by Lloyd *et al.* (2002) is not similar to the classification tasks performed here, no other studies are available that can be used. Therefore, this value, which was found to be 70%, was chosen due to the inherent similarity of feature elements within this class to each other. It is therefore somewhat relateable to conditions under which the present classification was carried out.

2. Since the performance of a CNN has a strong stochastic component, what is the probability that:

   (a) Any given instance of the model will exceed the critical accuracy value of 70%.

   (b) Any given instance of the model will achieve an accuracy in excess of 90%.

3. The efficacy of the hyperparameters are decided within the context of the baseline network. Here, several parameters are considered:

   (a) Do the hyperparameters lead to an increase in the classification accuracy of the network?

   (b) Does the addition of the hyperparameters lead to a decrease in the incidence of false positive and false negative classifications?

   (c) Do the inclusion of hyperparameters within the network increase the level of agreement between the actual classification and the predicted classification?

   (d) Given the stochastic nature of CNNs it is expected that there will be a certain degree of variability in the performance of a network given the criteria listed above. Therefore, a final consideration will be the amount of variability in the network's performance and whether the addition of the hyperparameters leads to a more consistent performance[9].

This section provides an overview of the evaluation criteria that was used to evaluate the suitability of the CNN to barchan research. Emphasis is placed on accuracy, absence of false positive and false negative scores, and the correlation between actual categories and model assigned categories. These are compared with what a human operator might achieve. Additionally, the influence of the stochastic nature of CNNs on model performance is assessed based on standard deviation scores and integrals under probability density functions, the latter derived from

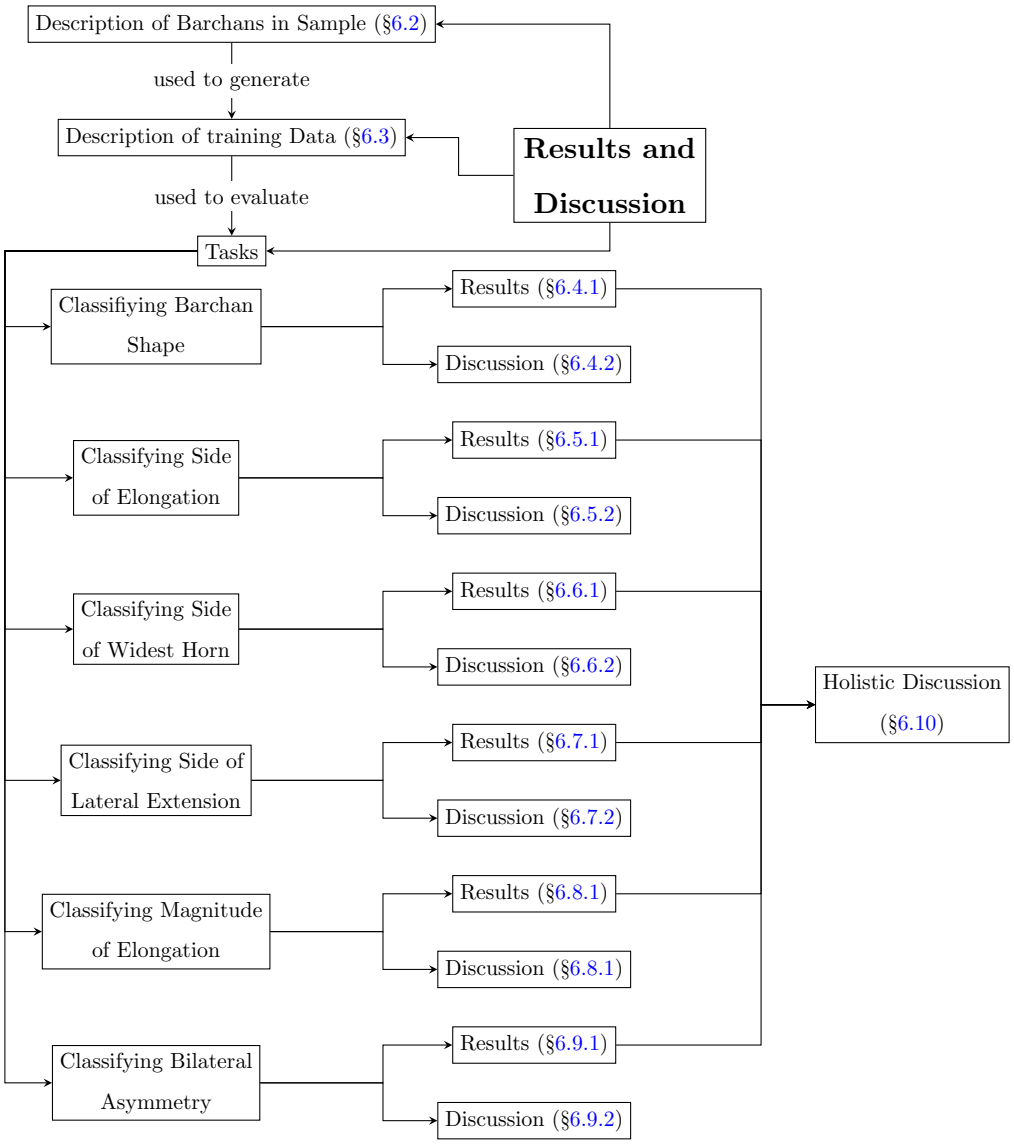
[9]This will manifest by a reduction in the amount of variability.

the scores of each iteration. Lastly, the influence of modifying the hyperparameters, and using transfer learning, on model performance is determined by comparing the different configurations of models to each other.

Chapter 6

Results and Discussion

6.1 Chapter Overview



In order to place the findings of this study into a context, the published performance of models, obtained from the literature, that were applied to geomorphic research are provided in Table 6.1 (see Section 4.2 for an overview of the studies cited). The specific nature of CNNs, in terms of task and input, makes it difficult to directly compare the performance of models between different studies. For example, the model developed by Ji *et al.* (2020) may perform very poorly to the task given to the model developed by Shumack *et al.* (2020). More specifically, the majority of tasks within geomorphology focussed on developing segmentation strategies. Conceptually, this is very different from the classification approach attempted in this study. Nevertheless, it is useful to compare the performance of the models in this study to those developed by other authors in order to place it within a larger context.

Broadly speaking, the performance of a CNN is influenced by two sub-tasks: the extraction relevant features (carried out by the feature extractor), and the correct classification of those features into categories (carried out by the classifier). The former is determined by aspects related to the generation of feature maps. These include the depth of the network, with deeper networks extracting more abstract features[1], and the suitability of the kernel to extract features from a given input. Classification is based on the architecture of the fully connected layer which, in this study, was kept the same for all models. Aside from the stochastic influence of initialisation, the classifier should have the some potential performance capacity for all the models evaluated. Therefore, individual model performance can be considered largely form the ability of the feature extractor to determine unique features for each class.

The remainder of this chapter is structured as follows. A brief description is provided of the barchans that served as the source of imagery for this study. This is relevant because it, in effect, serves to describe the morphometric characteristics, and their distribution within the data set, of the imagery used in this study. Then, the training data is described with a specific focus on how well the balance is within the data, i.e. the ratio between samples in the training, validation, and testing data sets. After this, the results of the models for each task, along with a discussion, is presented in separate sections. Included within the discussion are possible applications of this research to the broader barchan research community (Objective 8). This layout is followed mainly for ease of interpretation since the performance on individual tasks are independent of each other. In support of the discussion, a brief review of the some of the key concepts of CNNs is provided in Table 6.2. The chapter concludes with a holistic discussion on the topic and includes general comments that are applicable across all the tasks.

[1]Abstract features in this context refers to features that are present within features that have already been extracted.

Study	Task	Data	Landform	ACC Value	F_1–score	MCC Value
Baumhoer *et al.* (2019)	D	RS	Iceshelf	–	90–91	–
Carbonneau *et al.* (2020)	C	RS	River	83	–	–
Chen *et al.* (2018)	D	RS	Landslide	61–89	–	–
Du *et al.* (2021)	C	RS	Landslide	58	42	–
Gafurov and Yermolayev (2020)	D	RS	Gully	–	70	–
Gao *et al.* (2021)	D	RS	Yardang	74	85	–
Gao *et al.* (2021)	C	RS	Yardang	76–91	–	–
Ghorbanzadeh *et al.* (2019)	D	RS	Landslide	–	65–88	–
Huang *et al.* (2018)	C	RS	Thermokarst Gully	–	48–79	–
Ji *et al.* (2020)	D	RS	Landslide	95–97	90–94	–
Meena *et al.* (2021)	D	RS	Landslide	–	65–68	0.65–0.69
Palafox *et al.* (2017)	C	RS	Volcanic Rootless Cone/TAR	89–96	–	–
Prakash *et al.* (2021)	D	RS	Landslide	–	60–80	0.57–0.81
Robson *et al.* (2020)	C	RS	Rock Glacier	63–72	–	–
Shumack *et al.* (2020)	D	DM	Linear Dune	2–90	4–70	–
Wilhelm *et al.* (2020)	C	RS	Miscellaneous	–	78–89	–
Witharana *et al.* (2020)	D	RS	Ice-wedge Polygons	–	89–96	–

Table 6.1: The performance of CNNs on classification (C) and detection (D) tasks within geomorphology using Remotely Sensed (RS) or Digital Model (DM) as input to the model. The abbreviation TAR represents transverse aeolian ridges. For the work of Gao *et al.* (2021) and Shumack *et al.* (2020) the F_1 was calculated from the provided PPV and TPR data. The F_1–score in the study by Prakash *et al.* (2020) was extracted from provided data plots. The results of Ji *et al.* (2020) represent the performance of the VGG16 and ResNet50 models on RGB imagery while those of Wilhelm *et al.* (2020) are for their "from scratch" models. Finally, for Carbonneau *et al.* (2020) the results are for the "desk-based" training and the results of Chen *et al.* (2018) are for the areal extent of the landslides.

Table 6.2: A brief description of the key components of a CNN that is relevant to the discussion of model performance.

Concept	Description
Classifier	A broad description that refers to the layers within a CNN, typically the fully-connected layer, that is involved in assigning a classification category to the feature maps received from the feature extractor.
Feature extractor	A braod term that is used to describe the layers within a CNN that are responsible for generating feature maps.
Feature map	An array that results from the convolution of an input with a kernel that emphasizes features, such as lines and curves, that may be present within the input.
Generalization	The ability of a model to successfully apply the rules learned during training, to data that it has not been exposed to before.
Kernel	A matrix of learnable values used to extract features from an input image or an input feature map.
Model complexity	A reference to the number of trainable parameters that any given model has. The more the parameters, the more complex the model.
Regularization	A penalty value that is added to the loss function during the training process with the intent to increase the generilisation potential of the model.

6.2 Barchan description

As a whole, the barchans in the Kunene region have a mean dune width of $46.6m \pm 28.1$ and a mean stoss length of $38.8m \pm 20.04$. Their widths and stoss lengths are significantly ($p < 0.01$) smaller than those reported by Barnes (2001) working further towards the south at Walvis Bay. When compared to barchan dimensions published in other sources (Al-Harthi, 2002; Bailey, 1906; Barnes, 2001; Dong *et al.*, 2000; Douglass, 1909; Embabi, 1982; Finkel, 1959; Hamdan *et al.*, 2016; Jimenez *et al.*, 1999; Long and Sharp, 1964; Norris, 1966; Rempel, 1936; Sagga, 1998; Wang *et al.*, 2007), the barchans of the northern Kunene are relatively small. When the individual barchans within the dune field are considered (Figure 6.1), the size distribution is positively skewed indicating a predominance of smaller barchans within the data set. Although barchans to exhibit allometry (van der Merwe, 2021), this is not expected to negatively influence this study since the delineation of tasks is based on median (§5.6) value rather than a predefined cut-off value.

Frequency plots of the attributes that are considered within each task is provided in Figure 6.2. Most of the barchans ($n = 76$) have an elongated left horn while the remainder ($n = 14$) have an elongated right horn. This contrasts with the findings of Barnes (2001) at Walvis Bay who found no preferential elongation between the horns. Based on the findings of discussion of Tsoar and Parteli (2016) and Bourke (2010), this indicates that the horn elongation of barchans in the Kunene result more from the influence of bimodal winds while those at Walvis Bay result largely from collision. Regardless, all of the dunes in this dune field can be considered to be

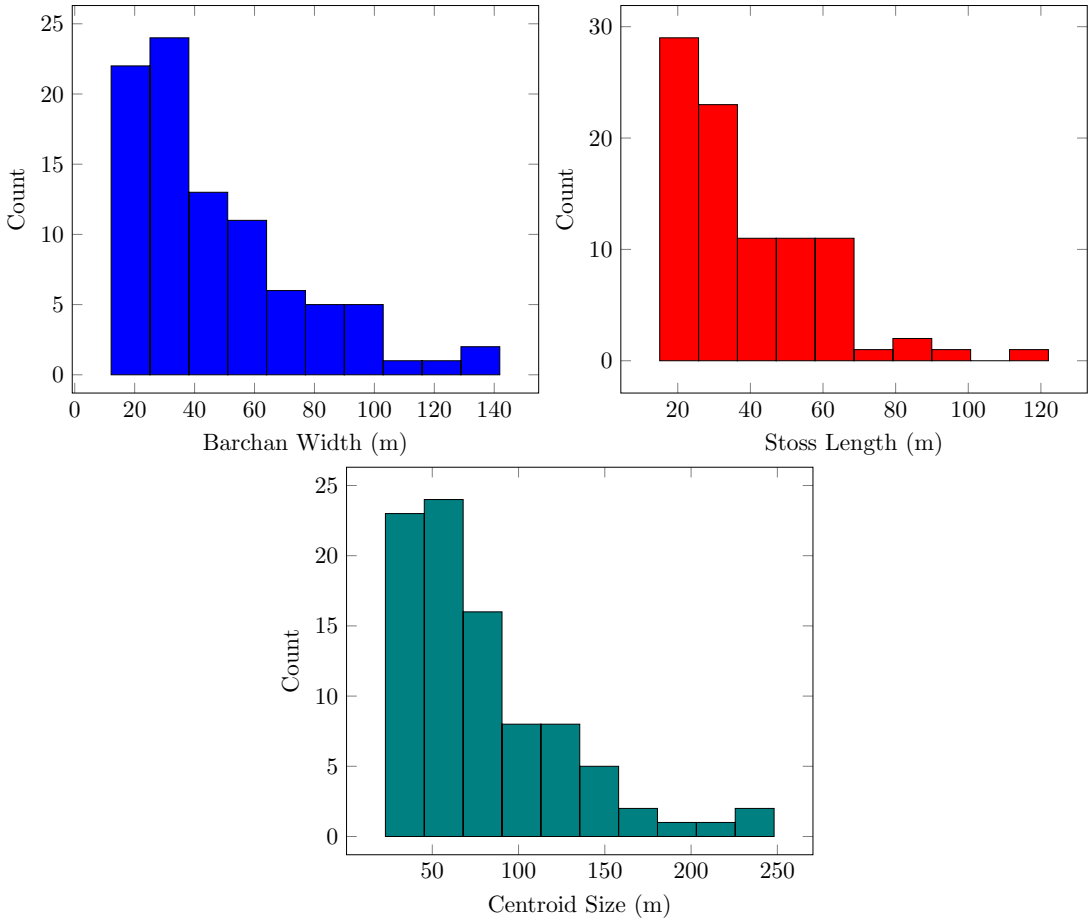


Figure 6.1: Size distributions of barchans based on barchan width, stoss length, and centroid size.

asymmetric based on conventional metrics (e.g. Bourke, 2010). This is to be expected considering that asymmetric barchans predominate in natural settins (Parteli *et al.*, 2014). Similarly, for the majority of barchans ($n = 56$) the left horn was the widest while the right horn was the widest in $n = 34$ cases. The left side of the dune underwent lateral extension in $n = 80$ cases while the right side of the dune was laterally extended in $n = 10$ cases. Given the position of the dunes relative to the downslope direction within the region, this would be expected based on the work of Parteli *et al.* (2014).

The frequency plot representing differences in horn length is positively skewed indicating that the dune field is dominated by barchans with a relatively small difference in horn lengths. Out of the entire barchan population, 43 had an $\frac{a}{c}$ ratio exceeding 0.88, 34 had an $\frac{a}{c}$ ratio between 0.63 and 0.88, and 13 had a ratio between 0.38 and 0.63. The sample is therefore biased more towards dunes where the stoss slope is elongated relative to the distance between the horn tips (i.e. positively skewed). The mean $\frac{a}{c}$ ratio of 0.93 is marginally smaller than that calculated by (Barnes, 2001) working in Walvis Bay. In contrast to the other attributes, the bilateral asymmetry, as indicated by the partial Procrustes distance, shows a more normal distribution

with values ranging from 0.1 to 0.8. This indicates that the dune field contains dunes that have comparatively low levels of asymmetry (i.e. dunes that are quite symmetrical) and dunes that are very asymmetrical. The levels of skewness within the data is not considered to negatively influence the study since the categories used are based on median values (§5.6). However, there is the possibility that the skewness may result in a situation where the *within class diversity* of one category containing data largely from the tail end of the distribution may be higher than the other category. This can, potentially, lead to an imbalance in the data in terms of feature diversity which can negatively impact the performance of a CNN (Nagle-Mcnaughton *et al.*, 2020). In cases where there is a great variety of shapes that are placed in the same category, it becomes challenging for the CNN to find a set of a common features that can be used in a discriminatory fashion. Unfortunately, given the size of the data set, this is not an aspect that can be sufficiently controlled for. However, since this study is a proof of concept and not a final model, the results of the model's output can still be used to assess the suitability of CNNs to barchan research.

The barchans within this region also exhibit allometry (van der Merwe, 2021). This manifests as pronounced differences in the mean shape between barchans from different size clusters (Figure 6.3). Smaller barchans tend to be more symmetrical with horns that are similar in terms of the magnitude of their downwind extent. As the size of the barchans increase, there is an increasing difference in horn lengths with one horn, in this case mostly the left, becoming significantly elongated in the downwind direction.

6.3 Training Data Description

For all cases except Task 4 the datasets are equally balanced. Task 4 has a slight imbalance due to the tied median values discussed in the Method section (§5) which resulted in one class having more images than the other. Therefore, overall, the classes within each task are perfectly balanced. In other words, there are equal amounts of imagery for both of the classes being classified. Therefore, any misclassifications that result cannot be assumed to be a bias towards one class over the other. Rather, the misclassifications have to result from the features learned by the network. It also indicates that the metrics used to evaluate the performance of the networks are more reliable.

6.4 Classifying Barchan Shape

The majority of CNN applications within geomorphology focussed on identifying landforms from remotely sensed imagery using semantic segmentation (e.g. Gafurov and Yermolayev, 2020; Carbonneau *et al.*, 2020) (§4.2.1). While this is useful, it does restrict the range of applications.

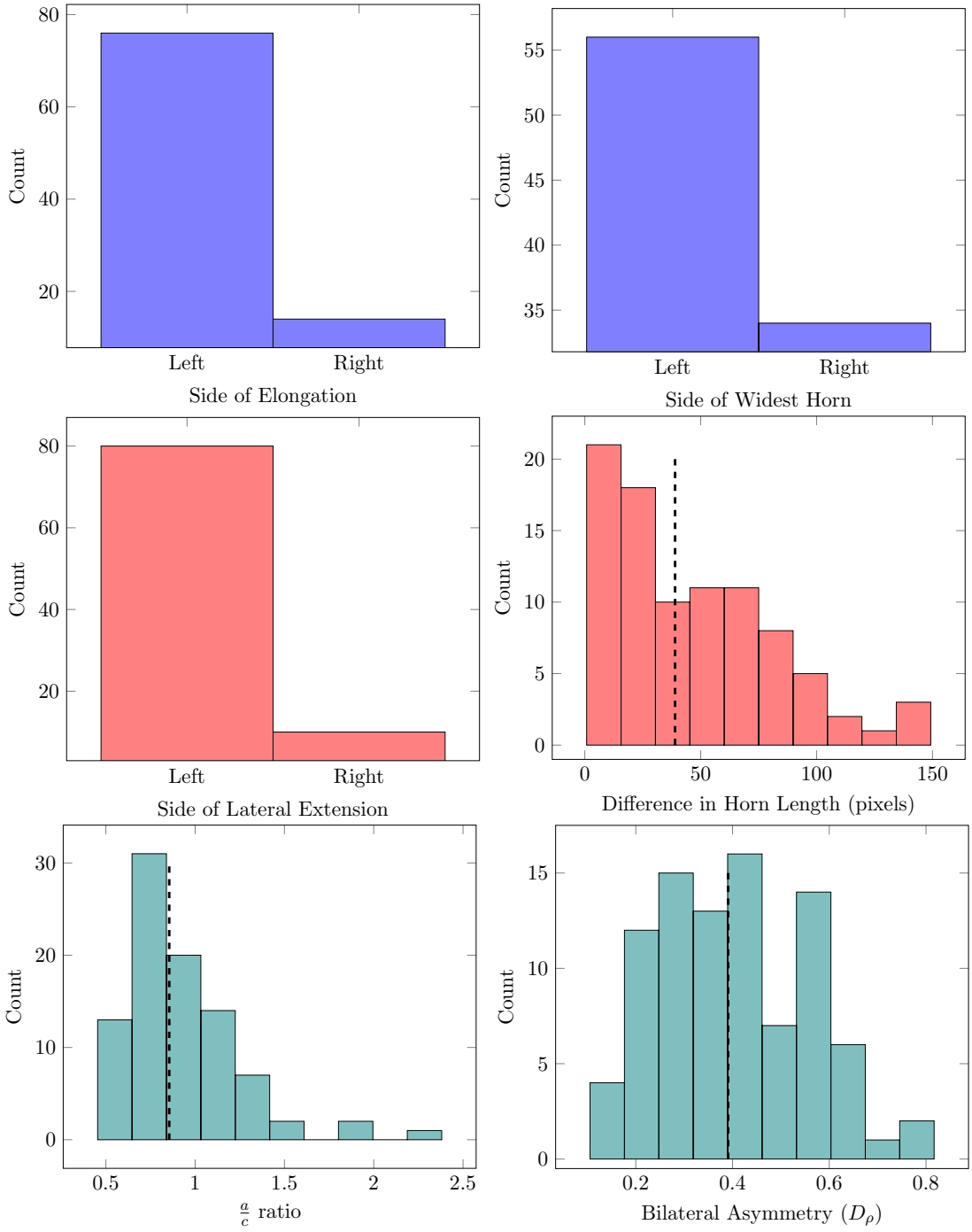


Figure 6.2: Frequency distribution of the barchan variables that form part of each task. The median value, which was used to divided the samples into two classes, is indicated by the dashed line.

Further, it remains to be demonstrated whether CNNs can fulfil a role beyond identification and delineation. Here, it is evaluated whether it is possible to classify barchan outlines into shape classes.

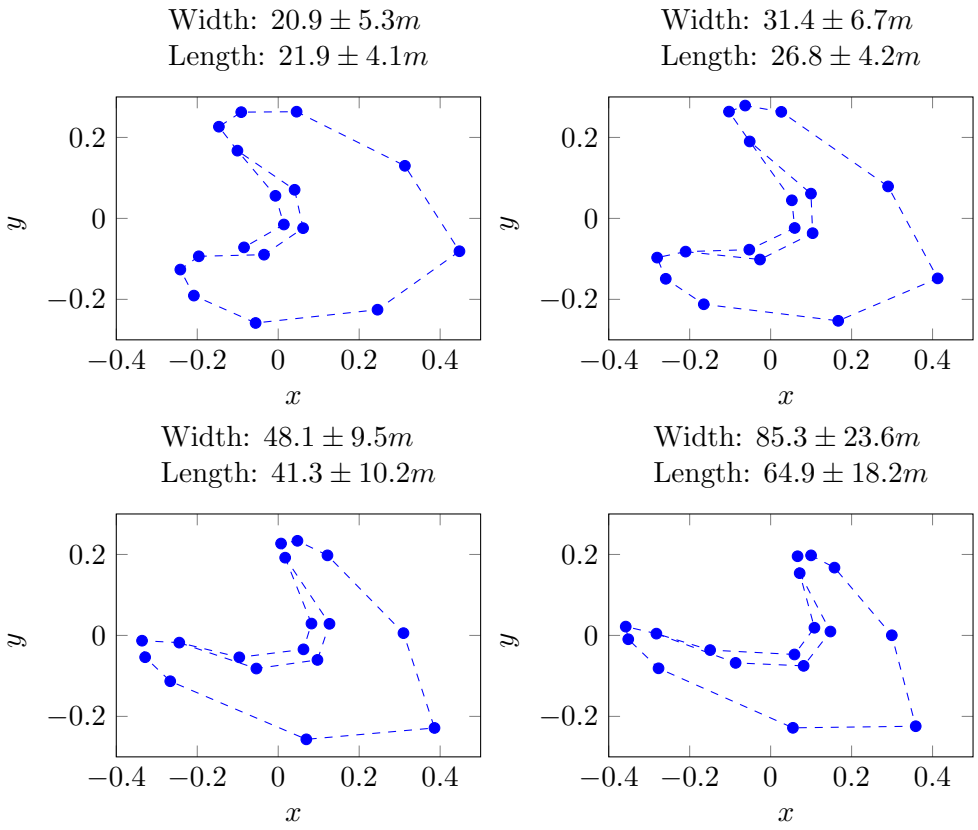


Figure 6.3: The mean shape of four clusters of barchans. Each cluster contains barchans that fall within a specific size range. Clusters were determined using the *classInt* R-package (Bivand and Lewin-Koh, 2016) with quantile breaks. The x and y axis refer to pixel coordinates following Procrustes Superimposition.

6.4.1 Results

6.4.1.1 ACC

The performance of the different models based on the accuracy (ACC) metric is summarised in Table 6.3. The majority of the models performed significantly better than the critical value of 70% ($p < 0.05$). The models that did not achieve this level of performance are the CNN2 model, the baseline model with L1 regularisation, and the CNN3 model with the L2 regularisation.

For the "from scratch" networks it was only possible to significantly increase the accuracy of the model by increasing the depth of the network to three layers and using an L1 regulizer. Both of the transfer learning approaches significantly increased the accuracy when compared to the baseline. The ResNet50 model achieved a higher ACC value than the VGG16 model.

Neither a modification of the hyperparameters nor the use of transfer learning is sufficient to significantly reduce the variability of a CNN on this task. Despite this inherent variability, the probability of an initialized model performing above the minimum accuracy is quite high. The lowest probability is for a baseline model with an L1 regularization. This model only has a 59% chance of achieving an ACC value in excess of 70% upon initialisation. In contrast, a three layer

Table 6.3: Summarised ACC statistics for the classification of the barchan shape task. The variables $\int_c^{100}$ and $\int_d^{100}$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 70\%$) and above the desired ($d = 90\%$). Average values that are bold indicate models that performed significantly above the critical value of 70% ($p < 0.05$), while those with an asterisk (*) had an ACC value that was significantly different from the baseline model ($p < 0.05$).

Model	Average (%)	Median (%)	St. Dev (%)	$\int_c^{100}$	$\int_d^{100}$
Shape Classification (ACC)					
Baseline (B)	**73**	75	6.1	0.71	0
B + 1-layer	72	70	5.3	0.6	0
B + 2-layers	**73**	75	5.7	0.67	0.02
B + L1 Reg	70	75	8.3	0.59	0.01
B + L2 Reg	**73**	75	6.2	0.7	0.01
B + Dropout	**74**	75	6.1	0.73	0.02
B + 1-layer + L1	**72**	70	5.4	0.62	0
B + 1-layer + L2	**73**	73	6.2	0.63	0
B + 1-layer + Dropout	**72**	70	5.4	0.62	0
B + 2-layers + L1	**76***	75	6.8	0.84	0
B + 2-layers + L2	72	73	7.8	0.63	0
B + 2-layers + D	**74**	75	7.5	0.71	0.01
VGG16	**84***	85	5.6	1	0.17
ResNet50	**88***	88	3.1	1	0.29

model with the same regularization has an 83% chance of achieving this level of accuracy. Both of the transfer learning models have a guarantee (i.e. 100%) of performing above this critical value. However, if an ACC value above 90% is desired, the "from scratch" models performed poorly with a maximum likelihood of 2% being achieved. This is considerably lower than the 20% chance that is possible using transfer learning.

6.4.1.2 F_1–score

All of the models had an F_1–score above 70% (Table 6.4). The "from scratch" models' performance is only increased under two conditions: adding another layer to the baseline model, and increasing the depth to three layers and combining it with an L1 regularization. Alternatively, transfer learning can be used which increases the F_1–scores to 83%, in the case of the VGG16 model, and 88% for the ResNet50 model.

No adjustment of the hyperparameters results in a significant decrease in the amount of variability in F_1–scores. Using transfer learning also does not reduce the amount of variability in F_1–scores. There is a wider range of probabilities associated with obtaining F_1–scores above 70% with any given initialisation of the "from-scratch" model. When a two layer model is initialised, there is only a 49% chance that the model would have an F_1–score above 70%. This is increased to 80% for a three layer model using an L1 regularization. When transfer learning is used, the probabilities are increased to 96% for the VGG16 model and 100% for the ResNet50 model.

Table 6.4: Summary of F_1–scores for the classification of the barchan shape task. The variables $\int_c^{100}$ and $\int_d^{100}$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 70\%$) and above the desired ($d = 90\%$). Average values that are bold indicate models that performed significantly above the critical value of 70% ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average (%)	Median (%)	St. Dev (%)	$\int_c^{100}$	$\int_d^{100}$
Shape Classification (F_1–score)					
Baseline (B)	**73**	74	6.6	0.77	0
B + 1-layer	70*	70	6	0.49	0
B + 2-layers	72	73	8	0.62	0.02
B + L1 Reg	71	71	5.3	0.56	0
B + L2 Reg	**74**	75	5.3	0.77	0
B + Dropout	**73**	76	8.9	0.66	0.03
B + 1-layer + L1	71	70	7.5	0.5	0
B + 1-layer + L2	72	70	7.5	0.55	0.01
B + 1-layer + Dropout	71	70	7.1	0.5	0
B + 2-layers + L1	**74***	76	11.8	0.8	0.01
B + 2-layers + L2	71	73	10.0	0.6	0.01
B + 2-layers + D	**72**	74	11.8	0.66	0.03
VGG16	**83***	84	6.7	0.96	0.17
ResNet50	**88***	88	3.3	1	0.28

6.4.1.3 MCC

In all models, the MCC values exceeded 0.5 (Table 6.5) indicating that the models' classification has a strong correlation with the actual classification. For the "from scratch" models, the only hyperparameters that result in a significant improvement in this correlation, is the addition of a dropout layer to the baseline model, and increasing the depth to three layers and adding the L1 regularization. Alternatively, using transfer learning significantly increases the MCC values to 0.74 and 0.79 (for VGG16 and ResNet50 respectively).

There is also a considerable amount of variability in MCC performance, and neither changes in hyperparameters or using transfer learning significantly decreases this variability ($p < 0.05$). Upon initialisation, there is a high probability that any given model will have an MCC value in excess of 0.5. The "worst" performing configuration is the three layer model with an L2 regularization which has a 79% probability of producing weights that allow an MCC value above 0.5. However, using the L2 regularization but reducing the depth to two layers increases this probability to 100%. This is the same level of performance obtained from using transfer learning approaches. However, none of the "from scratch" models can achieve an MCC value above 0.9 while the VGG16 and ResNet50 models have a 4% chance of obtaining this level of performance.

Table 6.5: Summary of MCC values for the classification of the barchan shape task. The variables $\int_c^1$ and $\int_d^1$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 0.5$) and above the desired ($d = 0.9$). Average values that are bold indicate models that performed significantly above the critical value of 0.5 ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^1$	$\int_d^1$
Shape Classification (MCC)					
Baseline (B)	**0.60**	0.62	0.05	0.9	0
B + 1-layer	**0.59**	0.58	0.07	0.87	0
B + 2-layers	**0.60**	0.58	0.06	0.9	0
B + L1 Reg	**0.58**	0.61	0.08	0.81	0
B + L2 Reg	**0.61**	0.61	0.05	0.97	0
B + Dropout	**0.61***	0.62	0.07	0.93	0
B + 1-layer + L1	**0.60**	0.58	0.06	0.98	0
B + 1-layer + L2	**0.61**	0.58	0.06	1	0
B + 1-layer + Dropout	**0.60**	0.58	0.05	1	0
B + 2-layers + L1	**0.63***	0.62	0.08	0.92	0
B + 2-layers + L2	**0.57**	0.59	0.11	0.79	0
B + 2-layers + D	**0.6**	0.62	0.1	0.85	0
VGG16	**0.74***	0.74	0.08	1	0.04
ResNet50	**0.79***	0.78	0.05	1	0.04

6.4.2 Discussion

The majority (79%) of models are able to classify barchans into shape categories with an accuracy that is above the critical value of 70%. Therefore, they are at least able to perform at the same level as a human operator. Given sufficient training, it is therefore possible to create models that classify barchans according to shape categories with a high degree of confidence. This would make it possible to build on the work carried out by Hamdan *et al.* (2016). Since it is possible to detect the position of barchans within an image (see (§4.2.1)) it may be possible to identify the shape of barchans at different positions. Since the date of imagery is known, and the difference in position can be determined through detection, it is possible to determine amount of migration that has occurred. This, in turn, can then be used to determine if there is a correlation between shape and migration speed.

In another study, Bourke and Goudie (2009) studied the varieties of barchan shape (in terms of $\frac{a}{c}$) ratios. The manual annotation of barchan images is a time consuming activity (Azzaoui *et al.*, 2020) that imposes constraints on the extent of the research, in either space or time (Huang *et al.*, 2018). The results show that it is possible to train a CNN to carry out this annotation automatically. Although the current configuration is not accurate enough, it merely represents the first step in developing this technology into a more usable form.

In order for these future applications to be realised the CNN must be able to classify the barchans correctly, have few incidences of misclassifications (i.e. FP and FN cases), and show

a strong correlation with actual classifications. The results show that the majority of "from scratch", and all the transfer models, are accurate enough. The average ACC scores for the "from scratch" models are similar to those obtained by other researchers who have applied CNNs to classify landforms from remotely sensed imagery (e.g. Gao *et al.*, 2021; Shumack *et al.*, 2020; Chen *et al.*, 2018) while being better than the results obtained for other studies (e.g. Du *et al.*, 2021; Robson *et al.*, 2020). However, the ACC scores are lower than those achieved by Ji *et al.* (2020) and Palafox *et al.* (2017). This does not mean that the "from scratch" models cannot achieve this level of performance. The Kernel Density Estimation (KDE) plots of several models have a non-zero integral for the region between 90% and 100%, albeit marginally, indicating that it is at least possible for these models to achieve ACC values that are similar to the ones reported by Ji *et al.* (2020) and Palafox *et al.* (2017). Using transfer learning it is possible, however, to significantly improve the ACC values to similar values as obtained by Carbonneau *et al.* (2020). This indicates that using the feature extraction capacity of a model pre-trained on a different data set can be used to extract barchan shape information provided that the classifier is trained on some relevant data. Similar results can be found in Wilhelm *et al.* (2020).

The F_1–scores show a similar pattern to the ACC values in that model configurations where the ACC values were not significantly above the critical value had F_1–scores that were also not above the critical value. This suggests that the performance of those models may have been influenced by the presence of false positive and false negative classifications. It is interesting to note, however, that a few models had a significant improvement in ACC values while not having a significant improvement in F_1–scores. Why this is the case is not yet known. Although other studies have reached considerably better F_1–scores (e.g Baumhoer *et al.*, 2019; Witharana *et al.*, 2020) the performance of the models on this task is similar to the F_1–scores of other studies using CNNs in geomorphology (Ghorbanzadeh *et al.*, 2019; Huang *et al.*, 2018; Prakash *et al.*, 2021).

For all of the models, the correlation between the classification of the model and that of the actual data set was above 0.5. This indicates that the performance is closer to being perfect than it is to being random. Although it cannot yet be described as being excellent, it does indicate that the classification is deterministic. This is because a random allocation into groups would result in an MCC value of 0 (Tharwat, 2018). Because it is not random it has to be deterministic. As such, it indicates that the model is indeed making use of features in the selection process. Unfortunately, the collected data does not make it possible to determine what those features are but it can be confidently stated that they are there. Compared to other studies, the results

are similar to those achieved by Meena *et al.* (2021) but lower than the highest values achieved by Prakash *et al.* (2021).

There is a more complex relationship between the addition of hyperparameters and the overall performance of the model. The majority of hyperparameter modifications did not significantly improve the performance of the baseline model. This indicates that a very basic model is capable of performing sufficiently in classifying barchan shape. However, the fact that models such as VGG16 and ResNet50, both considerably more complex than the baseline model, resulted in significant improvements indicate that such a conclusion may be premature. It is, unfortunately, difficult to ascribe an explanation to this discrepancy. Both ResNet50 and VGG16 were trained on data sets considerably larger than the ones used here containing more than 1.2 million images (He *et al.*, 2016; Simonyan and Zisserman, 2015). They were also trained for longer time periods. The VGG16 model was trained for 74 epochs (Simonyan and Zisserman, 2015) while the training of ResNet50 was 90 epochs (He *et al.*, 2016). The increased perfromance of VGG16 and ResNet50 can be ascribed to being either the result of improved feature extraction, or better optimisation from the increased number of epochs.

That neither transfer learning nor a modification of hyperparameters reduce performance variability reinforces the view that the performance of a CNN is highly dependent on their initialisation. A similar conclusion (applied to ANNs) was reached by Rodvold *et al.* (2001). This variability means that it is possible to have a model that has ACC values and F_1–scores above 90%, although the probabilities are low. The impact of this is significant. It means that any user will have to initialise a number of different models in order to chance upon a model that performs as desired. From a programming perspective this does not make a large difference, but it does have an impact on system resources. More models take more time to train.

Despite the large stochastic component associated with CNNs, it is clear that they do have the potential to classify barchans, based only on their outlines, into shape categories. Although no data exists to support this, it is unlikely that a human operator will be able to achieve this level of accuracy on visual observation alone.

6.5 Classifying Side of Elongation

6.5.1 Results

6.5.1.1 ACC

All of the "from scratch" models and the transfer learning models have an ACC value above 70% (Table 6.6). The ACC values can be significantly increased by adding an L1 regularization to the baseline model, or by increasing the baseline model's depth to two layers and adding a

dropout layer ($p < 0.05$). However, using a dropout layer and increasing the depth to three layers results in a significant decrease in ACC values ($p < 0.05$) compared to the baseline model. By using transfer learning, it is possible to significantly increase the ACC values above that of the baseline model ($p < 0.05$).

Table 6.6: Summary statistics of ACC values for the classification of the side of elongation task. The variables $\int_c^{100}$ and $\int_d^{100}$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 70\%$) and above the desired ($d = 90\%$). Average values that are bold indicate models that performed significantly above the critical value of 70% ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^{100}$	$\int_d^{100}$
Side of Elongation (ACC)					
Baseline (B)	**79**	75	7.2	0.9	0.08
B + 1-layer	**79**	80	6.07	0.93	0.05
B + 2-layers	**77**	75	6.5	0.85	0.02
B + L1 Reg	**81***	85	6.7	0.93	0.1
B + L2 Reg	**78**	75	7.4	0.83	0.08
B + Dropout	**79**	78	8.9	0.86	0.1
B + 1-layer + L1	**80**	80	4.7	0.96	0.03
B + 1-layer + L2	**80**	80	5.2	0.98	0.04
B + 1-layer + Dropout	**82***	85	6.4	0.95	0.12
B + 2-layers + L1	**81**	83	8.9	0.83	0.17
B + 2-layers + L2	**78**	75	6.0	0.9	0.05
B + 2-layers + D	**76***	75	6.4	0.78	0.02
VGG16	**82***	80	5.2	0.98	0.09
ResNet50	**83***	85	2.4*	1	0

A large amount of variability is present within the ACC scores. This variability can only be significantly reduced by using transfer learning based on the ResNet50 model ($p < 0.05$). Despite this, there is a high probability that any initialisation of the "from scratch" model would produce weights that result in ACC values above 70%. The highest probabilities are associated with transfer learning which have a 98% and 100% chance using the VGG16 and ResNet50 models respectively. Based on the integral of the probability density function, there is also a possibility that the "from scratch" models can achieve accuracies above 90%.

6.5.1.2 F_1–score

The results of the F_1–scores for the side of elongation task are summarised in Table 6.7. All of the "from scratch" models achieved an F_1 score above 70%. Compared with the baseline model, the performance can be increased by increasing the depth to two layers and using either dropout layers or the L1 regularization. Alternatively, using the L1 regularization with a three layer model also significantly increases the F_1–score. Lastly, using transfer learning can also significantly increase the F_1–scores, however, the results are similar to those obtained for modifications to the "from scratch" model.

Table 6.7: Summary statistics of F_1–scores for the classification of the side of elongation task. The variables $\int_c^{100}$ and $\int_d^{100}$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 50\%$) and above the desired ($d = 90\%$). Average values that are bold indicate models that performed significantly above the critical value of 70% ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^{100}$	$\int_d^{100}$
Side of Elongation (F_1–score)					
Baseline (B)	**76**	72	11.4	0.71	0.11
B + 1-layer	**79**	80	6.1	0.93	0.05
B + 2-layers	**77**	75	6.5*	0.85	0.02
B + L1 Reg	**81***	85	6.7	0.93	0.1
B + L2 Reg	**78**	75	7.4	0.83	0.08
B + Dropout	**79**	78	8.9	0.86	0.1
B + 1-layer + L1	**80***	80	4.7	0.96	0.03
B + 1-layer + L2	**80**	80	5.2	0.98	0.04
B + 1-layer + Dropout	**82***	85	6.4	0.95	0.12
B + 2-layers + L1	**81***	83	8.9	0.83	0.17
B + 2-layers + L2	**78**	75	6.0	0.9	0.05
B + 2-layers + D	**76**	75	6.4	0.78	0.02
VGG16	**82***	80	5.2*	0.98	0.09
ResNet50	**83***	85	2.4*	1	0

The only hyperparameters that can significantly decrease the amount of variability in F_1–scores is to increase the depth of the baseline model to three layers without incorporating a regularization ($p < 0.05$). Alternatively, transfer learning can be used based on either the VGG16 or ResNet50 model. The probabilities associated within a random initialisation of a "from scratch" model achieving an F_1–score above 70% are quite high. It ranges from 71% for the baseline model to 98% for a two layer model with the L2 regularization. This is comparable to the performance of the transfer learning models. It is interesting to note that the "from scratch" models tend to have a better chance of achieving F_1–scores above 90% than the transfer learning models.

6.5.1.3 MCC

For the side of elongation task, all of the evaluated models had an MCC value above 0.5 (Table 6.8). The performance of the baseline can be significantly improved ($p < 0.05$) using several strategies: adding the L1 regularization to the baseline model, adding both an L1 regularization and increasing the depth to three layers, increasing the depth to two layers and including dropout layers, or using transfer learning with either the VGG16 or ResNet50 models.

Improvements in variability is, however, considerably more difficult to obtain. Only transfer learning based on ResNet50 show a significant reduction in the amount of variability ($p < 0.05$). The probabilities of the "from scratch" models to achieve correlations above 0.7 are exceptionally high with several models having probabilities of 100% making it similar to that of the transfer

Table 6.8: Summary statistics of MCC values for the classification of the side of elongation task. The variables $\int_c^1$ and $\int_d^1$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 0.5$) and above the desired ($d = 0.9$). Average values that are bold indicate models that performed significantly above the critical value of 0.5 ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^1$	$\int_d^1$
Side of Elongation (MCC)					
Baseline (B)	**0.65**	0.61	0.11	0.93	0
B + 1-layer	**0.67**	0.67	0.08	1	0
B + 2-layers	**0.64**	0.62	0.09	0.91	0
B + L1 Reg	**0.69***	0.73	0.1	0.95	0
B + L2 Reg	**0.64**	0.61	0.11	0.85	0.03
B + Dropout	**0.67**	0.67	0.11	0.95	0.01
B + 1-layer + L1	**0.67**	0.67	0.06	1	0
B + 1-layer + L2	**0.68**	0.67	0.07	1	0
B + 1-layer + Dropout	**0.71***	0.74	0.09	1	0.02
B + 2-layers + L1	**0.70***	0.73	0.10	0.98	0.02
B + 2-layers + L2	**0.65**	0.62	0.08	0.98	0
B + 2-layers + D	**0.62***	0.61	0.09	0.91	0
VGG16	**0.71***	0.68	0.07	1	0.02
ResNet50	**0.72***	0.74	0.03*	1	0

learning models. It is also possible for the "from scratch" models to achieve correlations above 0.9, and even exceed the performance of the transfer learning models.

6.5.2 Discussion

It is investigated whether CNNs are able to classify barchans into categories based on which side of the barchan (left or right) contains the elongated horn. The calculated ACC values show that all of the models are able to successfully accomplish this task with an accuracy that is significantly above the critical value of 70%. The performance of the models is similar to those obtained by Gao *et al.* (2021) who developed a model to classify yardangs. It exceeds the ACC values obtained by Du *et al.* (2021) and Robson *et al.* (2020), but failed to reach the accuracies achieved by Ji *et al.* (2020) and Palafox *et al.* (2017).

Although it is commonly held that increasing the depth of the model can result in improvements in accuracy (e.g Shakya *et al.*, 2021) the results show that this relationship is more complex. Increasing the depth on its own does not significantly improve accuracy. This agrees with the findings of Carbonneau *et al.* (2020). However, increasing the depth to two layers and adding a dropout does. But, increasing the depth to three layers while retaining dropout layers results in a significant decrease in performance. Why this is the case is unknown, but it does provide additional support for the idea that a simple "rule book" for CNNs cannot be developed.

Although the use of transfer learning resulted in a significant improvement over the baseline model, it is only marginally so and not significantly different from two hyperparameter modifica-

tions to the baseline (adding just the L1 regularisation or adding an additional layer along with a dropout layer). This is particularly interesting. Since the classifier was identical in all cases, it suggests that the feature extractor of these layers performed similarly. While reinforcing the already mentioned lack of a guide to CNNs, it also hints at the possibility that the success of a model's architecture is also dependent on the task at hand. This echoes the view of Kattenborn *et al.* (2021). Also, it indicates that the use of transfer learning is not necessarily always needed.

All of the models had an F_1 score above the critical value indicating that the number of false positive and false negative classifications are relatively low. The models outperformed those developed by others (Du *et al.*, 2021; Gafurov and Yermolayev, 2020; Meena *et al.*, 2021; Shumack *et al.*, 2020). While not achieving the performance of Witharana *et al.* (2020), the results are still comparable to other studies of CNNs within geomorphology.

Several different approaches can be followed to reduce the incidence of false positive and negative results. These can either be in the form of using transfer learning or adjusting the hyperparpameters. Similar to the ACC values, there are adjustments that can be made to only the hyperparameters that will result in similar levels of performance as can be achieved using transfer learning. This reinforces that the feature extraction capacity of the "from scratch" models is capable of identifying features that can be considered unique to each class. What these features are is unknown, but future work can investigate them by examining the generated feature maps. However, it must be emphasized that it is not always the case that a human would use the same distinguishing characteristics as the machine when making a classification. This is the case irrespective of whether the classification, as far as humans are concerned, follows a strict protocol. Whether or not this is ultimately an issue is a topic for another study and is not addressed here.

There are also further complexities that arise when examining the F_1–scores. For example, the baseline model has a lower average accuracy than the two layer model without a regularization. However, it has a higher probability of initialising to a model that can have F_1–scores in excess of 90%. While this is partly explained by the variability of the former being larger than the latter, it still suggests the possibility that the relationship between features extracted and classification accuracy is more complex.

The high MCC values indicate that the classification assigned by the models correspond strongly to the actual classifications that would be assigned through more time consuming analysis. It is considerably higher than the accuracy achieved by Meena *et al.* (2021) and within the range of values obtained by Prakash *et al.* (2021). It is almost guaranteed that any given initialisation of a model will achieve an accuracy in excess of 0.5, indicating that it will be closer

to a perfect approximation than a random assignment. This also means that there is some underlying feature, although it is unknown what this is, that can be used by a CNN to classify the side of elongation.

As with the other metrics, the MCC values also indicate a complex relationship between hyperparameters and model performance. Adding the L1 regularization increases the performance in the baseline model and the three layer model, but not the two layer model. The complex relationship indicates that the influence of hyperparameter modification on performance is still unlikely to be predictable. It is also interesting to note that even though the use of transfer learning leads to a significant improvement compared to the baseline, the overall performance is similar to performances reachable by just modifying the hyperparameters.

As mentioned in Chapter 2, there are two principle models that are believed to determine which of the horns on a barchan gets elongated. CNNs have shown that it is possible to automatically extract data on which one of the barchan horns is elongated. This makes it possible to build on the work of Lv *et al.* (2016) who speculated that the angle between the bidirectional winds determines which of the two models are in effect. Specifically, when combined with wind vector data from NCEP/NCAR Reanalysis platform, the automatic extraction of side of elongation data will allow for larger data sets to be collected for more in depth statistical analysis.

In the case of Martian aeolian activity, the rapid extraction of side of elongation data can support investigations into processes that were in effect during Mars' distant past. In cases where the side of elongation for the majority of barchans is the same, it can be inferred that the morphology within the dune field resulted largely from the influence of bimodal winds (Sagga, 1998; Engel *et al.*, 2018; Lv *et al.*, 2016; Courrech du Pont, 2015; Hobbs *et al.*, 2010; Boulghobra, 2016; Norris, 1966). However, if the side alternates between different dunes, it is likely that dune collisions are largely involved (Bourke, 2010; Tsoar and Parteli, 2016; Zhang *et al.*, 2018). This will give new insights into the paleoclimatology of the planet.

6.6 Classifying Side of Widest Horn

6.6.1 Results

6.6.1.1 ACC

None of the models, either "from scratch" or transfer, achieved an ACC value above 70% (Table 6.9). Although several different modifications to the hyperparameters resulted in significant improvements in ACC over the baseline model, these improvements are not sufficient to exceed the critical value. The use of transfer learning did not result in a significant improvement in

ACC. Unsurprisingly, the low ACC values translate to the probabilities of for an initialised model to exceed the critical level. The majority of models cannot achieve an ACC values above 70% (i.e. the probability of this is 0). The exceptions are: a two layer model with the L2 regularization, and three layer model with either the L1 regularization or dropout layers. These models have a chance of 1%, 6%, and 2% respectively. Given this, it is also unsurprising that no model can exceed the 90% benchmark.

Table 6.9: Summary statistics of ACC values for the classification of the side of widest horn task. The variables $\int_c^{100}$ and $\int_d^{100}$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 70\%$) and above the desired ($d = 90\%$). Average values that are bold indicate models that performed significantly above the critical value of 70% ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^{100}$	$\int_d^{100}$
Side of Widest Horn (ACC)					
Baseline (B)	**45**	45	5.2	0	0
B + 1-layer	**49***	50	6.9	0	0
B + 2-layers	**49***	50	7.6*	0	0
B + L1 Reg	**45**	45	5.6	0	0
B + L2 Reg	**45**	45	4.9	0	0
B + Dropout	**46**	45	4.4	0	0
B + 1-layer + L1	**48***	50	6.2	0	0
B + 1-layer + L2	**47**	45	7.6*	0.01	0
B + 1-layer + Dropout	**48***	45	5.5	0	0
B + 2-layers + L1	**54***	55	9.8*	0.06	0
B + 2-layers + L2	**53***	50	7.3	0	0
B + 2-layers + D	**51***	50	7.8*	0.02	0
VGG16	**45**	45	10.8*	0	0
ResNet50	**44**	45	5.7	0	0

The variability in performance is also high, although the values are still similar to those found for other tasks in which the performance is better. For the "from scratch" models the variability can be decreased by increasing the depth of the baseline to three layers, by adding the either the L1 regularization or dropout layers to this 3 layer model, or using a two layer model with the L2 regularization. When using transfer learning, only the use of VGG16 significantly improves the variability ($p < 0.05$).

6.6.1.2 F_1–score

None of the models achieved a high F_1–score on this task (Table 6.10). The F_1–score can be significantly improved ($p < 0.05$) by increasing the depth of the model to two layers (without regularization) or to three layers with an L1 regularization. However, neither of these modifications increase the F_1–score above the critical value. It is interesting to note that the use of transfer learning does not significantly increase the F_1–score.

Table 6.10: Summary statistics of F_1–scores for the classification of the side of widest horn task. The variables $\int_c^{100}$ and $\int_d^{100}$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 50\%$) and above the desired ($d = 90\%$). Average values that are bold indicate models that performed significantly above the critical value of 70% ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^{100}$	$\int_d^{100}$
Side of Widest Horn (F_1–score)					
Baseline (B)	**46**	48	13.5	0.02	0
B + 1-layer	**51***	52	9.5	0	0
B + 2-layers	**46**	47	15.7	0.05	0
B + L1 Reg	**42**	48	17.2	0.04	0
B + L2 Reg	**42**	48	15.1	0	0
B + Dropout	**42**	42	14.9	0.04	0
B + 1-layer + L1	**50**	53	11.8	0.05	0
B + 1-layer + L2	**44**	44	15.0	0.04	0
B + 1-layer + Dropout	**47**	49	11.3	0.02	0
B + 2-layers + L1	**52***	54	12.6	0.08	0
B + 2-layers + L2	**51**	52	15.1	0.08	0
B + 2-layers + D	**49**	51	17.0	0.12	0
VGG16	**41**	43	13.1	0	0
ResNet50	**35***	36	12.1	0	0

The variability within the F_1–score is high when compared to other tasks such as the side of elongation. The low average score is also reflected in the probabilities of exceeding the critical value upon initialisation. Only the three layer model with dropout layers has a probability of above 10% for exceeding this critical value. None of the models, "from scratch" or transfer, has the potential to exceed a 90% F_1–score.

6.6.1.3 MCC

The same trend that was observed for ACC and F_1–scores is also present in the MCC values with none of the models exceeding the critical value of 0.5 (Table 6.11). Several different adjustments can be made in order to increase the MCC values. First, the depth of the baseline model can be increased to two layers using either the L2 regularization or including dropout layers. Alternatively, the baseline's depth can be increased to 3 layers and the L1 regularization incorporated. In contrast to the ACC and F_1 metrics, using either the VGG16 or the ResNet50 transfer learning models results in a significant improvement in MCC values ($p < 0.05$).

As to be expected there is also a considerable mount of variability present in the MCC values that result from the initialisation of individual models. Although none of the models had, on average, an MCC value above the critical value of 0.5, any individual run of any model had a chance of achieving an MCC value above 0.5. Using the VGG16 model in transfer learning means that there is a 56% chance of any individual model performing above this critical value. None of the individual models could achieve an MCC value above 0.9. Also, no configuration

Table 6.11: Summary statistics of MCC values for the classification of the side of widest horn task. The variables $\int_c^1$ and $\int_d^1$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 0.5$) and above the desired ($d = 0.9$). Average values that are bold indicate models that performed significantly above the critical value of 0.5 ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^1$	$\int_d^1$
Side of Widest Horn (MCC)					
Baseline (B)	0.46	0.49	0.08	0.39	0
B + 1-layer	0.49*	0.5	0.03	0.44	0
B + 2-layers	0.46	0.49	0.09	0.4	0
B + L1 Reg	**0.43**	0.46	0.09	0.25	0
B + L2 Reg	**0.46**	0.48	0.05	0.26	0
B + Dropout	**0.45**	0.46	0.07	0.2	0
B + 1-layer + L1	**0.45**	0.48	0.07	0.29	0
B + 1-layer + L2	0.48*	0.48	0.06	0.37	0
B + 1-layer + Dropout	**0.47***	0.48	0.06	0.37	0
B + 2-layers + L1	0.49*	0.50	0.05	0.45	0
B + 2-layers + L2	**0.44**	0.48	0.08	0.3	0
B + 2-layers + D	**0.45**	0.48	0.07	0.3	0
VGG16	0.52*	0.51	0.05	0.56	0
ResNet50	0.48*	0.48	0.05	0.36	0

of hyperparameters nor the use of transfer learning could significantly reduce the amount of variability within the model.

6.6.2 Discussion

The goal of this task is to investigate whether the feature extraction and classification ability of a CNN is able to accurately determine which one of the barchan horns is the widest. The width of the horn plays an important role in dune field sediment dynamics since it influences how much sediment is lost from the horns (Elbelrhiti *et al.*, 2008). Given the influence of this on barchan asymmetry (Parteli *et al.*, 2014; Lv *et al.*, 2016), being able to extract this information can be beneficial in better understanding barchan dune field interactions and asymmetry.

Unfortunately, all of the CNN models performed poorly on this task. It is noteworthy that the transfer models VGG16 and ResNet50 performed worse than a number of "from scratch" models. Since these transfer models performed well on other tasks, it suggests that, inherently, they have the capacity to extract usable features and to classify them into the appropriate categories. Their poor performance can be attributed to a shortcoming in either of these aspects (i.e feature extraction or classification). Given the performance of transfer learning on the other tasks, it suggests that the features needed to discern between the different categories are insufficient. Given the data, it is not possible to determine which of these, or the weights of their combined influence, is responsible for the poor performance. However, it is possible that the features needed for distinguishing between the two different cases are "overpowered" by the

noise resulting from the other features. In effect, there is not enough data in order for the model to separate distinguishing features from irrelevant features. The poor performance of VGG16 and ResNet50 suggest that the solution to this problem does not lie in increasing the depth or using residual connections. Rather, it is more likely that improvements can be achieved by increasing the number of training samples.

Unsurprisingly, this also means that it is not possible for any of the models to reach an ACC value above 90%. Although it is possible to significantly increase the ACC values through either a modification of the hyperparameters or the use of transfer learning, none of these strategies improve the ACC to above the critical value. Compared with other models, the models performed better than the ones developed by Shumack *et al.* (2020) who had an accuracy of 2% in some cases. While Shumack *et al.* (2020) attributed the accuracy deficit to image resolutions, this cannot be the case here owing to the image preparation prior to training and evaluation (§5.5).

The low F_1–scores also indicate that there is a relatively high amount of false positive and false negative classifications. The few options available to improve upon the baseline's performance are also insufficient to improve the performance beyond the critical value. A high incidence in false positives and negatives suggests that the model is not able to distinguish features from the training data that can be generalised to other instances that the model has not seen before. This can either indicate that suitable features could not be identified, that extracted features could not be effectively classified, or some combination of both.

Although the overall performance of the models are poor, the MCC values are close to the critical value of 0.5 with the VGG16 transfer learning approach exceeding this value (although not significantly). This indicates that despite the poor classification performance, the predictions of the models are still not random. It is therefore possible that the features extracted by the model are not discriminatory enough in order to accurately separate the images into their respective categories.

A potential explanation for the poor performance may be the nature of the task itself. Compared to the other tasks, this task focussed on a comparatively small region of the barchan. As a result, there is a large amount of image information that is, in effect, irrelevant for accurate classification. Unfortunately, there is insufficient data available to determine with any certainty whether or not this is the case.

6.7 Classifying Side of Lateral Extension

6.7.1 Results

6.7.1.1 ACC

In contrast to the results obtained for the side of the widest horn task, the ACC values of all the models exceeded the critical value of 70% (Table 6.12). However, it is curious to note that no adjustment to the hyperparameters could significantly increase the performance of the baseline model. Also, only when the ResNet50 model is used during transfer learning is the ACC value significantly increased ($p < 0.05$).

Table 6.12: Summary ACC value statistics for the classification of the side of lateral extension task. The variables $\int_c^{100}$ and $\int_d^{100}$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 70\%$) and above the desired ($d = 90\%$). Average values that are bold indicate models that performed significantly above the critical value of 70% ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^{100}$	$\int_d^{100}$
Side of Lateral Extension – (ACC)					
Baseline (B)	**88**	90	8.0	0.9	0.39
B + 1-layer	**86**	85	6.1	0.97	0.3
B + 2-layers	**86**	88	6.1	1	0.32
B + L1 Reg	**84***	83	9.6	0.79	0.24
B + L2 Reg	**85**	90	10.5	0.84	0.33
B + Dropout	**85**	90	8.0	0.9	0.31
B + 1-layer + L1	**84***	85	8.1	0.92	0.25
B + 1-layer + L2	**87**	90	5.2*	0.99	0.36
B + 1-layer + Dropout	**87**	85	6.5	0.98	0.32
B + 2-layers + L1	**89**	90	4.4*	1	0.45
B + 2-layers + L2	**87**	88	5.5	0.99	0.35
B + 2-layers + D	**87**	90	7.2	0.94	0.38
VGG16	**89**	90	5.1*	0.98	0.4
ResNet50	**92***	95	4.1*	1	0.73

The variability in the ACC values is similar to those of the other tasks discussed so far. Any given initialisation of a model has a high probability (79% - 100%) of achieving an ACC score above the critical value. There is also a high probability that a model can achieve ACC values above 90% with transfer learning based on the ResNet50 model having a 73% chance of reaching this level of performance. The amount of variability can be significantly improved ($p < 0.05$) by increasing the depth of the "from scratch" model to two or three layers and using the L2 and L1 regularizations respectively. It is also possible to significantly reduce the variability by using transfer learning ($p < 0.05$).

6.7.1.2 F_1–score

The F_1–scores for this task are summarised in Table 6.13. All of the models achieved an F_1–score above 70%. For the "from scratch" models, this can only be improved by increasing the depth of the network to three layers and using the L1 regularization. Alternatively, transfer learning (based on either VGG16 or ResNet50) can be used.

Table 6.13: Summary F_1–score statistics for the classification of the side of lateral extension task. The variables $\int_c^{100}$ and $\int_d^{100}$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 50\%$) and above the desired ($d = 90\%$). Average values that are bold indicate models that performed significantly above the critical value of 70% ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^{100}$	$\int_d^{100}$
Side of Lateral Extension (F_1–score)					
Baseline (B)	**86**	90	9.6	0.86	0.4
B + 1-layer	**86**	86	6.2*	0.97	0.27
B + 2-layers	**85**	87	7.9	0.93	0.3
B + L1 Reg	**82**	82	11.1	0.82	0.32
B + L2 Reg	**86**	89	9.5	0.87	0.42
B + Dropout	**82**	89	11.2	0.78	0.31
B + 1-layer + L1	**84**	84	7.4	0.98	0.26
B + 1-layer + L2	**86**	89	6.3	1	0.32
B + 1-layer + Dropout	**85**	86	8.9	0.93	0.29
B + 2-layers + L1	**89***	89	4.5*	1	0.41
B + 2-layers + L2	**87**	88	5.4*	1	0.3
B + 2-layers + D	**86**	89	7.8	0.95	0.37
VGG16	**88***	89	5.1*	0.98	0.36
ResNet50	**92***	95	3.9*	1	0.72

Although the influence of initial conditions is still visible in the variability of the models' performance, any initialisation of any model architecture has a high probability (78% to 100%) of having an F_1–score above 70%. There is also a high probability that the F_1–score can exceed 90% with transfer learning based on the ResNet50 model having the highest likelihood. Two strategies can be followed to reduce the amount of variability in F_1–scores. First, the depth of the "from scratch" model can either be increased to two layers (without using regularizations), or it can be increased to three layers using either the L1 or the L2 regularization. The second option is to use transfer learning with either the VGG16 or the ResNet50 model.

6.7.1.3 MCC

All of the models had an MCC value above the critical value of 0.5 (Table 6.14). For the "from scratch" models, the MCC values can be significantly increased ($p < 0.05$) only by increasing the depth to three layers and using the L1 regularization. Alternatively, transfer learning can be used based on either the VGG16 or the ResNet50 models.

Table 6.14: Summary MCC value statistics for the classification of the side of lateral extension task. The variables $\int_c^1$ and $\int_d^1$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 0.5$) and above the desired ($d = 0.9$). Average values that are bold indicate models that performed significantly above the critical value of 0.5 ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^1$	$\int_d^1$
Side of Lateral Extension (MCC)					
Baseline (B)	**0.78**	0.82	0.13	1	0.25
B + 1-layer	**0.76**	0.74	0.09	1	0.08
B + 2-layers	**0.76**	0.78	0.1	1	0.08
B + L1 Reg	**0.73***	0.70	0.15	0.95	0.19
B + L2 Reg	**0.77**	0.82	0.13	1	0.18
B + Dropout	**0.74**	0.82	0.13	0.98	0.11
B + 1-layer + L1	**0.73**	0.74	0.12	0.96	0.08
B + 1-layer + L2	**0.78**	0.82	0.08*	1	0.08
B + 1-layer + Dropout	**0.76**	0.74	0.10	0.98	0.12
B + 2-layers + L1	**0.81***	0.82	0.07*	1	0.14
B + 2-layers + L2	**0.78**	0.78	0.09	1	0.11
B + 2-layers + D	**0.78**	0.82	0.10	1	0.14
VGG16	**0.80***	0.82	0.08*	0.98	0.11
ResNet50	**0.86***	0.90	0.06*	1	0.34

Despite the variability in performance, any given initialisation of the model has a probability of more than 95% in achieving an MCC score above the critical value. The probabilities of achieving MCC values above 0.9 is also considerably higher than for the other tasks with transfer learning based on ResNet50 (34%). The amount of variability can, however, be significantly reduced ($p < 0.05$) by: increasing the depth to two layers and using the L2 regularization, increasing the depth to three layers using the L1 regularization, or using transfer learning based on either the VGG16 or ResNet50 models.

6.7.2 Discussion

It is investigated whether it is possible for a CNN to detect which side of the barchan experiences the most lateral extension. In other words, on which side of the barchan is the most lateral point furthest removed from the longitudinal axis. Based on the ACC values, CNNs are easily able to make this determination. All of the models performed significantly better than 70% with all models having an ACC value in excess of 84%. It therefore exceeds the performance of several earlier models (Carbonneau *et al.*, 2020; Du *et al.*, 2021; Robson *et al.*, 2020) and had a similar classification accuracy as achieved by Gao *et al.* (2021) for yardangs. From all tasks this one had the most consistent improvement with the L1 regularization leading to improved performance when used with either the baseline model or the two layer model. Only the transfer learning used with ResNet50 had a significant improvement in performance. This reinforces some of the earlier conclusions that it is not necessarily always needed to resort to transfer learning.

Overall the $F-1$–scores are very high exceeding those of Huang *et al.* (2018), Du *et al.* (2021), Meena *et al.* (2021), Gafurov and Yermolayev (2020), and Shumack *et al.* (2020). It is similar to the values by Gao *et al.* (2021) and Wilhelm *et al.* (2020) and less than those obtained by Witharana *et al.* (2020) and Baumhoer *et al.* (2019). These values indicate that false positive and false negative classifications occur infrequently.

Just as with the ACC values and the F_1–scores, the MCC values are also high for this task indicating a large amount of agreement between the classification of the model and the actual classification. It reinforces the reliability of the model on this task, although more refinements are necessary to increase the correlation further. It is almost a certainty that any given initialisation of a model will be able to reach an MCC value of over 0.5.

The modelling work carried out by Parteli *et al.* (2014) suggests that the lateral extension of the barchans can be caused by their location on an inclined slope. Finkel (1959) also made mention of this possibility in describing the observed asymmetry of barchans in Peru. However, there is still not a lot of published material on this topic. Using CNN models such as the ones developed here, and combining it with local digital elevation models, can provide sufficient data in order statistically evaluate this proposed relationship.

6.8 Classifying Magnitude of Elongation

6.8.1 Results

6.8.1.1 ACC

The results of the magnitude of elongation task (Table 6.15) shows that any of the models evaluated can achieve an ACC value above 70%. There are also several different strategies that can be followed to significantly improve on this performance ($p < 0.05$). For the "from scratch" models, improvements can be achieved by increasing the depth of the model to either two or three layers (either with or without any of the regularizations evaluated). The only exception being the combination of three layers with the L2 regularization. It is also possible to significantly increase the ACC values by using transfer learning.

The high average ACC values also translates into increased probabilities of achieving the critical value on any given initialisation of a model. The lowest probabilities are obtained when using the baseline model combined with the L1 regularization, which results in only a 74% chance. However, this can be increased by using: a two layer model without regularization, a two layer model with the L1 regularization, a three layer model with dropout layers, or transfer learning using ResNet50. The model configurations that significantly reduced the amount of variability in ACC values are any of the two layer models (i.e. a two layer model with or

Table 6.15: Summary of ACC values for the classification of the magnitude of elongation task. The variables $\int_c^{100}$ and $\int_d^{100}$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 70\%$) and above the desired ($d = 90\%$). Average values that are bold indicate models that performed significantly above the critical value of 70% ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^{100}$	$\int_d^{100}$
Magnitude of Elongation – (ACC)					
Baseline (B)	**77**	80	9.8	0.76	0.09
B + 1-layer	**82***	80	5.8*	0.98	0.11
B + 2-layers	**82***	80	6.2	0.95	0.14
B + L1 Reg	**76**	75	8.4	0.76	0.03
B + L2 Reg	**77**	75	7.8	0.74	0.07
B + Dropout	**78**	80	8.6	0.82	0.11
B + 1-layer + L1	**83***	85	6.4*	0.93	0.16
B + 1-layer + L2	**82***	80	5.1*	0.98	0.07
B + 1-layer + Dropout	**81***	80	5.8*	0.95	0.07
B + 2-layers + L1	**80***	80	7.6	0.85	0.07
B + 2-layers + L2	**82**	**85**	7.4	0.92	0.17
B + 2-layers + D	**84***	85	5.8*	0.98	0.17
VGG16	**80***	80	6.3*	0.92	0.06
ResNet50	**82***	83	4.7*	0.98	0.06

without regularization), a three layer model with dropout layers, or any of the transfer learning models.

6.8.1.2 F_1–score

Similar to the ACC values, all of the models exceeded the critical F_1–score for this task (Table 6.16). It is also possible to significantly improve on these values ($p < 0.05$) using the same strategies that are used to improve ACC values. The exception is that only a transfer learning approach based on ResNet50 will result in an improvement.

When considering the probabilities of any given initialisation exceeding the critical value, a broad range of values can be obtained. The baseline model had a probability of 53% while the highest probability (99%) is found using a two layer model with an L1 regularization. It is interesting to note that several of the "from scratch" configurations exceeds the performance of transfer learning models in this regard. Similarly, several "from scratch" models also have higher probabilities of achieving F_1–scores above 90% compared to the transfer learning approaches. Only one strategy, namely using transfer learning with ResNet50, can significantly reduce the amount of variability in model performance on this metric ($p < 0.05$).

Table 6.16: Summary of MCC values for the classification of the magnitude of elongation task. The variables $\int_c^{100}$ and $\int_d^{100}$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 50\%$) and above the desired ($d = 90\%$). Average values that are bold indicate models that performed significantly above the critical value of 70% ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^{100}$	$\int_d^{100}$
Magnitude of Elongation (F_1–score)					
Baseline (B)	**75**	78	17	0.53	0.14
B + 1-layer	**81***	82	7.2	0.91	0.1
B + 2-layers	**82***	83	6.7	0.96	0.12
B + L1 Reg	**75**	78	12.8	0.57	0.13
B + L2 Reg	**78**	78	7.2	0.88	0.03
B + Dropout	**77**	81	12.9	0.68	0.15
B + 1-layer + L1	**84***	84	4.9*	0.99	0.11
B + 1-layer + L2	**82***	82	5.9*	0.96	0.07
B + 1-layer + Dropout	**81***	82	6.6*	0.93	0.07
B + 2-layers + L1	**78**	78	10.3	0.79	0.08
B + 2-layers + L2	**82***	84	7.4	0.94	0.14
B + 2-layers + D	**84***	86	6.9	0.94	0.17
VGG16	**78**	80	7.6	0.85	0.05
ResNet50	**80***	82	6.2*	0.92	0.06

6.8.1.3 MCC

The results (Table 6.17) show that all of the models are capable of achieving MCC values above the critical value (0.5). It is also possible to increase the level of correlation using the same strategies used to improve on the ACC values.

Any given initialisation of the models evaluated has a high probability (≥82%) of having an MCC value higher or equal to 0.5. The highest probability (100%) is achieved using the following models: a two layer model without regularization, three layer model without regularization, a two layer model with L2 regularization, a three layer model with dropout layers, transfer learning using VGG16, and transfer learning using ResNet50. However, it is only possible to achieve a correlation higher than 0.9 using transfer learning based on the VGG16 model. It is also possible to reduce the amount of variability using the same approaches as for the ACC values, with the exception that a three layer model without a regularization also results in a significant decrease in variability ($p < 0.05$).

6.8.2 Discussion

It is evaluated whether a CNN can extract sufficient information from an image of a barchan outline to accurately determine the magnitude of elongation (i.e. the absolute difference in length between the two horns). Since all of the CNN models had an average ACC value above the critical value of 70%, it can be concluded that this is indeed possible. Despite the stochastic nature of

Table 6.17: Summary of MCC values for the classification of the magnitude of elongation task. The variables $\int_c^1$ and $\int_d^1$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 0.5$) and above the desired ($d = 0.9$). Average values that are bold indicate models that performed significantly above the critical value of 0.5 ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^1$	$\int_d^1$
Magnitude of Elongation (MCC)					
Baseline (B)	**0.63**	0.67	0.16	0.82	0
B + 1-layer	**0.70***	0.68	0.08*	1	0
B + 2-layers	**0.70***	0.67	0.08*	1	0
B + L1 Reg	**0.61**	0.61	0.13	0.84	0
B + L2 Reg	**0.63**	0.61	0.11	0.85	0
B + Dropout	**0.65**	0.67	0.12	0.87	0
B + 1-layer + L1	**0.72***	0.74	0.09*	0.94	0
B + 1-layer + L2	**0.70***	0.68	0.07*	1	0
B + 1-layer + Dropout	**0.69***	0.68	0.09*	0.97	0
B + 2-layers + L1	**0.68***	0.67	0.10	0.94	0
B + 2-layers + L2	**0.71***	0.74	0.09	0.96	0
B + 2-layers + D	**0.73***	0.74	0.08*	1	0
VGG16	**0.68***	0.68	0.08*	1	0.02
ResNet50	**0.71***	0.71	0.06*	1	0

the initialisation of a model, there is a high probability ($\geq 70\%$) that any given initialisation will have an ACC value above the critical value. This indicates that the high average score is unlikely to result from the influence of a few outlier models that had an anomalously high performance. Although it is small, there is also the possibility that an initialised model can achieve ACC values in excess of 90%. Here, it is interesting to note that several "from scratch" models have a greater chance of achieving this level of performance than the transfer learning models. This is yet more support that resorting to transfer learning models may not always be necessary. The ACC values obtained on this task is higher than those obtained by Du *et al.* (2021) and are within the range of values obtained by several other authors (Robson *et al.*, 2020; Gao *et al.*, 2021; Chen *et al.*, 2018).

The high F_1–scores associated with this task also indicate that there are relatively few false positive and false negative classifications. This corroborates the high ACC values mentioned earlier. There is also an admittedly small, but nonetheless possible, chance that a given CNN initialisation can have an F_1–score in excess of 90%. This would put the model in the same performance range as the models developed by Baumhoer *et al.* (2019) and Witharana *et al.* (2020). For the majority of cases, the same strategies used to improve the ACC values can be used to increase the F_1–scores. The exception to this is the use of transfer learning based on the VGG16 model. Why this may be the case is unknown. Models with fewer convolution layers

can result in significant improvements as well as models with more (e.g. ResNet50). The latter may, however, benefit from the presence of residual connections (Khan *et al.*, 2020).

There is also a high correlation between the classification proposed by the model and the actual classification of the images (based on the MCC values). It is almost guaranteed that any random initialisation of the model can be trained to reach this level of performance, but it is not possible for a "from scratch" model trained under the conditions used in this study to achieve MCC values in excess of 0.9 on this task. An interesting result is that all of the regularisations evaluated require at least a depth of two layers in order to be effective. The MCC values achieved on this task exceeds those achieved by Meena *et al.* (2021) and are within the upper range obtained by Prakash *et al.* (2021).

It is therefore possible for CNNs to be used in studies investigating the relationship between barchan morphology and wind regime. Simulations carried out by Parteli *et al.* (2014) show that there is a positive relationship between the duration of secondary winds and the magnitude of elongation of a barchan horn. This same interpretive value is used by Bourke (2010). By combining automatic morphological data extraction with derived meteorological observations, it is possible to evaluate the level of agreement between the simulations of Parteli *et al.* (2014) and observations.

6.9 Classifying Bilateral Asymmetry

6.9.1 Results

6.9.1.1 ACC

A summary of the ACC values for each model is provided in Table 6.18. The majority of models achieved an average ACC value above he critical value (70%). The only exceptions are the baseline model and, interestingly, all of the three layer models that had some form of regularization included. Improvements on the performance of the baseline model can be achieved by: adding either the L1 or the L2 regularization to the baseline model, increasing the depth to two layers with, or without, regularizations, or using transfer learning based on either the VGG16 or ResNet50 models.

There is also a relatively high probability of any given initialisation having an ACC value equal or higher than the critical value. Modifications to the baseline model can increase the probability from 71% to 86% when increasing the depth to two layers and using the L1 regularization. However, it can also decrease from 71% to 55% when using a three layer model with dropout layers. There is also a small chance (<10%) that some initialisations can have ACC values above 90%. However, when transfer learning is used this can increase to 10% and 35%

Table 6.18: Summary of ACC values for the classification of bilateral asymmetry task. The variables $\int_c^{100}$ and $\int_d^{100}$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 70\%$) and above the desired ($d = 90\%$). Average values that are bold indicate models that performed significantly above the critical value of 70% ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^{100}$	$\int_d^{100}$
Bilateral Asymmetry – (ACC)					
Baseline (B)	72	75	7.6	0.71	0
B + 1-layer	**74***	75	5.3	0.78	0
B + 2-layers	**73**	75	8.1	0.68	0.02
B + L1 Reg	**75***	75	7.6	0.76	0.01
B + L2 Reg	**75***	75	9.1	0.72	0.06
B + Dropout	**73**	75	6.3	0.7	0
B + 1-layer + L1	**75***	75	4.0	0.86	0
B + 1-layer + L2	**75***	75	7.6	0.76	0
B + 1-layer + Dropout	**76***	80	7.9	0.78	0.03
B + 2-layers + L1	73	75	8.8	0.63	0.01
B + 2-layers + L2	72	75	8.0	0.62	0.01
B + 2-layers + D	71	70	8.2	0.55	0.02
VGG16	**83***	85	5.3	0.98	0.1
ResNet50	**88***	90	4.7	1	0.35

for VGG16 and ResNet50 respectively. The amount of variability associated with the baseline model cannot be improved upon by modifying the hyperparameters or by using transfer learning.

6.9.1.2 F_1–score

The majority of models achieved an F_1–score above 70% (Table 6.19) with the only exceptions being the baseline model and the three layer model with an L2 regularization. The strategies followed for increasing the ACC values can also be used to increase the F_1–scores, with the addition of increasing the depth to three layers (without regularizations) or adding a dropout layer to the baseline model.

There is also a wide range of probabilities associated with any given initialisation having an F_1–score larger or equal to 70%. This ranges from 52% for a three layer network with the L2 regularization, to 100% when using transfer learning based on the ResNet50 model. Most of the models have at least a small chance that any given initialisation can have an F_1–score above 90%, with the exceptions being a baseline model with an L2 regularization, or a two layer model with an L1 regularization. The variability cannot be significantly improved upon using transfer learning. However, it is possible to significantly improve the variability ($p < 0.05$) by adding the L2 regulizer to the baseline model, or increasing the baseline model's depth to two layers and adding the L1 regulizer.

Table 6.19: Summary of F_1–scores for the classification of bilateral asymmetry task. The variables $\int_c^{100}$ and $\int_d^{100}$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 50\%$) and above the desired ($d = 90\%$). Average values that are bold indicate models that performed significantly above the critical value of 70% ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^{100}$	$\int_d^{100}$
Bilateral Asymmetry (F_1–score)					
Baseline (B)	69	73	13.9	0.57	0.03
B + 1-layer	**74***	76	8.1	0.67	0.04
B + 2-layers	**74***	76	8.9	0.73	0.03
B + L1 Reg	**72***	78	13.6	0.6	0.1
B + L2 Reg	**77***	78	6.0*	0.85	0
B + Dropout	71*	74	12.1	0.53	0.09
B + 1-layer + L1	**75***	76	5.3*	0.83	0
B + 1-layer + L2	**72***	77	14.8	0.59	0.1
B + 1-layer + Dropout	**73***	77	14.4	0.62	0.11
B + 2-layers + L1	71	75	13.0	0.61	0.06
B + 2-layers + L2	69	70	12.4	0.52	0.02
B + 2-layers + D	70	72	11.0	0.55	0.03
VGG16	**82***	83	6.1	0.96	0.07
ResNet50	**86***	89	6.1	1	0.28

6.9.1.3 MCC

All of the models achieved an MCC value equal to or above the critical value of 0.5 (Table 6.20). The level of correlation can be significantly improved by: adding either the L1 or L2 regularization to the baseline model, increasing the depth to two layers with or without any of the regularizations, or increasing the depth to three layers without using regularizations. The level of correlation can also be significantly increased by using transfer learning.

Based on the results, there is a high probability that any given initialisation (81% to 100%) will have an MCC value greater than or equal to 0.5. However, if a level of correlation above 0.9 is desired, it can only be achieved by using transfer learning with ResNet50 having the highest (8%) probability. The amount of variability can only be significantly reduced by increasing the depth of the baseline model to two layers and adding an L1 regularisation.

6.9.2 Discussion

Barchan asymmetry is admittedly a complex phenomenon and it is unlikely that shape deformation of any barchan will be limited to just elongation or widening. It is therefore evaluated whether it is possible for a CNN to identify the combined effect of different types of asymmetry. This is evidenced in the Procrustes distance between a barchan and its mirror image.

The high ACC values confirm that it is possible to achieve this goal. The majority of models ($\approx 71\%$) are capable of achieving ACC values significantly above 70%. Since these models were

Table 6.20: Summary of MCC values for the classification of bilateral asymmetry task. The variables $\int_c^1$ and $\int_d^1$ indicate the probability (based on $n = 30$ models) of any one instance of the model achieving a performance above the minimum ($c = 0.5$) and above the desired ($d = 0.9$). Average values that are bold indicate models that performed significantly above the critical value of 0.5 ($p < 0.05$), while those with an asterisk (*) had a score that was significantly different from the baseline model ($p < 0.05$).

Model	Average	Median	St. Dev	$\int_c^1$	$\int_d^1$
Bilateral Asymmetry (MCC)					
Baseline (B)	**0.57**	0.61	0.11	0.81	0
B + 1-layer	**0.61***	0.62	0.06	0.97	0
B + 2-layers	**0.61***	0.61	0.08	0.92	0
B + L1 Reg	**0.61***	0.62	0.1	0.84	0
B + L2 Reg	**0.64***	0.62	0.05	1	0
B + Dropout	**0.58**	0.61	0.10	0.87	0
B + 1-layer + L1	**0.62***	0.62	0.05*	1	0
B + 1-layer + L2	**0.61***	0.61	0.11	0.9	0
B + 1-layer + Dropout	**0.62***	0.66	0.10	0.87	0
B + 2-layers + L1	**0.60**	0.61	0.09	0.85	0
B + 2-layers + L2	**0.58**	0.58	0.09	0.8	0
B + 2-layers + D	**0.58**	0.57	0.08	0.82	0
VGG16	**0.72***	0.74	0.07	1	0.02
ResNet50	**0.78***	0.82	0.08	1	0.08

trained on a relatively small dataset, and given that the performance of CNNs improve with increased amounts of training data, it is possible that increasing the amount of training data can lead to increased performance. The classification accuracy of the barchans is similar to the classification accuracy achieved by Gao *et al.* (2021). It is also possible to improve the ACC values by either modifying the hyperparameters or by using transfer learning. It is, however, interesting to note that increasing the depth from two layers to three layers does not result in a significant improvement over the baseline, despite the two layer network being a significant improvement on the model.

Although the F_1–scores are generally lower than those for the magnitude of elongation task, the majority of models (64%) have scores significantly above 70%. This indicates that the majority of models have a relatively low incidence of false positives and negatives. Any configuration of hyperparameters, with the exception of those where regularizations are added to a three layer network, is able to increase the F_1–scores. In this case, the three layer network without regularization can improve performance, but the addition of any regularization decreases the performance associated with the additional layers. What complicates this matter further is that the use of transfer learning based on either the VGG16 or the ResNet50 models improves the F_1–score. The VGG16 architecture, which has more convolutional layers than any "from scratch" model evaluated here, was trained using the L2 and dropout regularization (Simonyan and Zisserman, 2015). However, in the case of the "from scratch" models the addition of the

L2 regularization or the use of dropout layers did not improve the F_1–score. This is yet further evidence that a universally applicable strategy for improving model performance may not be obtainable.

The high MCC values indicate a generally strong correlation between the classification performed by the model and the actual classification. As with some of the other tasks, it is almost guaranteed that the correlation will be closer to a perfect correspondence than to a random assignment. It can therefore be concluded that the model does in fact learn features that aid in the classifications, although it is not known what these features are. Improvements in model performance is similar to those for improving the ACC values and the F_1–scores, but not identical. This can signify a more complex relationship between the metrics in addition to the complex performance modifications associated with the models.

This high level of performance on the bilateral asymmetry task indicates that it is possible to extract the combined influence of several different asymmetry mechanisms on a dune. However, it is not possible, at this stage, to determine the relative contribution of each separate mechanism of asymmetry to the overall asymmetry.

6.10 Holistic Discussion

6.10.1 Hyperparameters

The biggest challenge involved in evaluating and understanding the performance of a CNN is the so-called "black–box" nature of these models. In a "black–box" model it is difficult to evaluate the contribution of all the variables to the output of the model (Olden and Jackson, 2002; Wang *et al.*, 2020). Therefore, in the absence of *post–hoc* analysis, it is unknown whether the features used by humans are similar to the features used by the model during classification. Table 6.21 to 6.23 contain a summary of the main strategies that can be followed in order to enhance the performance of the different metrics. This data was derived by combining the average performance of the models for all tasks. Aside from the benefit of transfer learning strategies, there is a considerable amount of variability in performance boosting strategies. The remainder of this section will provide a higher level overview of the suitability of CNNs to geomorphology.

When viewed in its entirety, improvements in the ACC values can be obtained following several different strategies. It is clear that, in general, using transfer learning based on the ResNet50 model is the best option to improve the ACC values. This agrees with de Lima and Marfurt (2020) who also found that transfer learning approaches outperform "from scratch" models. Although this strategy is not guaranteed to result in an improvement, there is, nevertheless, an 83% chance that a model will have a higher ACC value, compared to the baseline,

Table 6.21: A summary of improvement strategies that can be applied to the baseline model in order to increase the ACC values. The antecedent column represents the modifications to the baseline model and the consequent the desired change in the model. The column $P(Y|X)$ represents the probability of improving the model's performance given that the antecedent conditions are present.

Antecedent (X)	**Consequent (Y)**	**P(Y|X)**
ResNet50	Improvement	0.83
VGG16	Improvement	0.67
+ 1 layer, + L1	Improvement	0.67
+ 1 layer, + Dropout	Improvement	0.67
+ L1	Improvement	0.56
+ 1 layer	Improvement	0.54
+ 2 layers, + L1	Improvement	0.50
+ 2 layers, + Dropout	Improvement	0.50
+ Dropout	Improvement	0.39
+ 2 layers	Improvement	0.38
+ 1 layer, + L2	Improvement	0.33
+ L2	Improvement	0.22
+ 2 layers, + L2	Improvement	0.17

when ResNet50 is used. Other studies have also found that ResNet50 results in better performance (Ji *et al.*, 2020; Mahdianpari *et al.*, 2018). These findings differ from those of Wilhelm *et al.* (2020) and Agarwal and Mittal (2019) who found that VGG16 performed better. Why this may be the case is unclear. ResNet50 contains 48 convolutional layers compared to the 13 that are present within VGG16. More convolution layers, in principle, means more abstract features can be extracted. Also, more layers have been associated with improved performance (Kattenborn *et al.*, 2021). Therefore, it may simply be the case that ResNet50 has a superior feature extraction capacity compared to VGG16. Additionally, residual network models such as ResNet50 learn features differently from VGG16 due to the presence of skip connections (Khan *et al.*, 2020). Since this directly alters the feature maps, it may be the case that this modification provides ResNet50 with an additional performance enhancement that is not available to VGG16. The transfer learning models are also the best options for improving the F_1–scores and the MCC values.

Aside from using transfer learning, the most beneficial strategy to follow is increasing the model depth and combining it with some form of regularization. In some sense this differs from the views of He and Sun (2015) and He *et al.* (2016) who found that increasing depth results in poorer performance. The reason why it disagrees only to an extent is that, in the case of improving ACC values, the use of three layers and an L1 regularization is less likely to lead to a performance gain than using a two layer network with a L1 regularization. What complicates the matter further is that the baseline model with L1 regularization performed better than the three layer model with L1 but worse than the two layer model with L1. When considering metrics

Table 6.22: A summary of improvement strategies that can be applied to the baseline model in order to increase the F_1–scores. The antecedent column represents the modifications to the baseline model and the consequent the desired change in the model. The column $P(Y|X)$ represents the probability of improving the model's performance given that the antecedent conditions are present.

Antecedent (X)	**Consequent (Y)**	**P(Y|X)**
ResNet50	Improvement	1.00
VGG16	Improvement	0.67
+ 2 layers, + L1	Improvement	0.67
+ L1	Improvement	0.50
+ 1 layer	Improvement	0.50
+ 1 layer, + L1	Improvement	0.50
+ 1 layer, + Dropout	Improvement	0.50
+ 2 layers	Improvement	0.33
+ 1 layer, + L2	Improvement	0.33
+ Dropout	Improvement	0.28
+ L2	Improvement	0.22
+ 2 layers, + L2	Improvement	0.17
+ 2 layers, + Dropout,	Improvement	0.17

such as F_1–scores and MCC values, the pattern at first supports the view shared by others that increasing the depth results in increased performance (Kattenborn *et al.*, 2021; Shakya *et al.*, 2021; Simonyan and Zisserman, 2015; Wan *et al.*, 2018; Ciresan *et al.*, 2011). But the relationship is more complex than that. The best, in terms of having the highest likelihood of success, hyperparameter modification for both F_1–scores and MCC values is the three layer model with L1 regularization. If model depth was a simple linear relationship, it would be expected that a two layer model with L1 would perform slightly worse while the baseline model with L1 would perform the worst out of the three. This progression is not, however, observed and the baseline model with L1 had a higher chance of improving performance than the two layer model with L1.

In the absence of regularizations, i.e. when only working with model depth, it is clearly the case that additional layers do not guarantee improvement over the baseline model. For all three metrics, the two layer model had a higher probability of improving performance than the three layer model. This agrees with He and Sun (2015) and He *et al.* (2016). It also provides support for the observation by Du *et al.* (2021) that more complex models may perform worse than simpler models when supplied with limited training data.

The use of the L2 regularization had a consistent pattern across all metrics. Across all tasks the progression of likelihood of performance increase, from highest to lowest, involving the L2 regularization is: two layers with L2, baseline with L2, and three layers with L2. However, the likelihood of L2 regularization increasing model performance is generally lower than the use of

Table 6.23: A summary of improvement strategies that can be applied to the baseline model in order to increase the MCC values. The antecedent column represents the modifications to the baseline model and the consequent the desired change in the model. The column $P(Y|X)$ represents the probability of improving the model's performance given that the antecedent conditions are present.

Antecedent (X)	**Consequent (Y)**	**P(Y|X)**
ResNet50	Improvement	1.00
VGG16	Improvement	1.00
+ 2 layers, + L1	Improvement	0.83
+ 1 layer, + Dropout	Improvement	0.67
+ L1	Improvement	0.56
+ 1 layer	Improvement	0.50
+ 1 layer, + L2	Improvement	0.50
+ 2 layers	Improvement	0.42
+ Dropout	Improvement	0.39
+ 1 layer, + L1	Improvement	0.33
+ 2 layers, + Dropout	Improvement	0.33
+ L2	Improvement	0.28
+ 2 layers, + L2	Improvement	0.17

L1. This suggests that the penalty value of the L2 regularization, when applied to the kernel, is too severe and prevents the convolution operation from extracting usable features.

The use of dropout layers also show a complex relationship. The combination of an additional layer and two convolutional layers are associated with a higher probability of success than a three layer model with three dropout layers or the baseline layer with a single dropout layer. Aside from the ACC metric, the baseline model with a single dropout layer had a higher probability of improved performance than the three layer model with three dropout layers. The purpose of dropout layers is to prevent overfitting of the data (Shakya *et al.*, 2021; Chevtchenko *et al.*, 2018). The fact that the use of dropout layers are not consistently associated with model performance enhancement (i.e. the probability of improving performance is high) suggests that the "from scratch" models tend to not suffer from overfitting. Additionally, the results show that the findings of Shakya *et al.* (2021) that dropout layers can degrade performance cannot be taken as a universal rule across all CNN applications.

6.10.2 CNNs and Shape

The literature is divided on the capacity of CNNs to recognise objects based on their shape. Some view CNNs as able to recognise and classify objects based on their shape (e.g. Kubilius *et al.*, 2016; Kattenborn *et al.*, 2021), while others consider CNNs to be more dependent on the texture of the image (e.g. Geirhos *et al.*, 2019; Baker *et al.*, 2020). Additionally, in cases where it is recognised that shape contributes to classification performance, there is the view that this is limited to the identification of local shape (e.g Baker *et al.*, 2020, 2018). A midway is argued

by (Chollet and Allaire, 2018) who states that the convolution layers detect local shape patterns and the fully connected layers use this information to detect global patterns.

The results presented in this study support the view that CNNs can classify objects with high accuracy when texture elements are removed. With the exception of the side of widest horn task, the landform classification accuracy of the models evaluated here are similar to those achieved by Gao *et al.* (2021) for yardangs. A key difference is that Gao *et al.* (2021) used satellite imagery to make the classification, which contains a lot of textural information within the pixels. That using only outlines delivers similar levels of performance is further evidence that CNNs can evalute the shape of objects in the absence of abundant texture information.

6.10.3 CNNs and Landform Classification

Classifying remotely sensed images involves categorizing the image into pre-defined classes based on the contents of the image (De Cesarei *et al.*, 2021; Cheng *et al.*, 2018). However, it is believed that humans and CNNs approach this task differently. For humans, this classification is based largely on the object's shape (Baker *et al.*, 2020; Kubilius *et al.*, 2016). There is somewhat of a divided view on the approach followed by CNNs. Baker *et al.* (2020) argues that, for a CNN, image texture is at the very least an equally strong cue for image classification. This is supported by the works of others who found that alterations of image texture can lead to misclassifications (Carrara *et al.*, 2018; Nguyen *et al.*, 2015). Others, such as Kattenborn *et al.* (2021), state that CNNs are designed to learn the shape elements, such as corners and edges, along with textural information that best describe the class of interest. However, the larger spatial context of these edges may not be learnable (Baker *et al.*, 2018, 2020). A third view offered by Kubilius *et al.* (2016) is that CNNs can recognize objects when other information such as texture is removed. Although it was not investigated how classification accuracies change depending on whether image texture was retained or removed, the fact that high classification accuracies were achieved seems to support the contention that CNNs can successfully classify an image based only on outline information. This makes it possible to combine this work with the work of others on barchan boundary extraction (Dakir *et al.*, 2016; Azzaoui *et al.*, 2019) to create a workflow that can classify barchans from imagery. In doing so, the issues raised by Carrara *et al.* (2018) and Geirhos *et al.* (2019) on the influence of texture on classification performance can be avoided.

Gao *et al.* (2021) found that landforms that have a more diverse range of shapes and sizes lead to more false positive and false negative classifications. There is insufficient data to conclude why the performance of the models developed here differs from that developed by Gao *et al.* (2021). Their work used satellite images where multiple yardangs are present, while the images

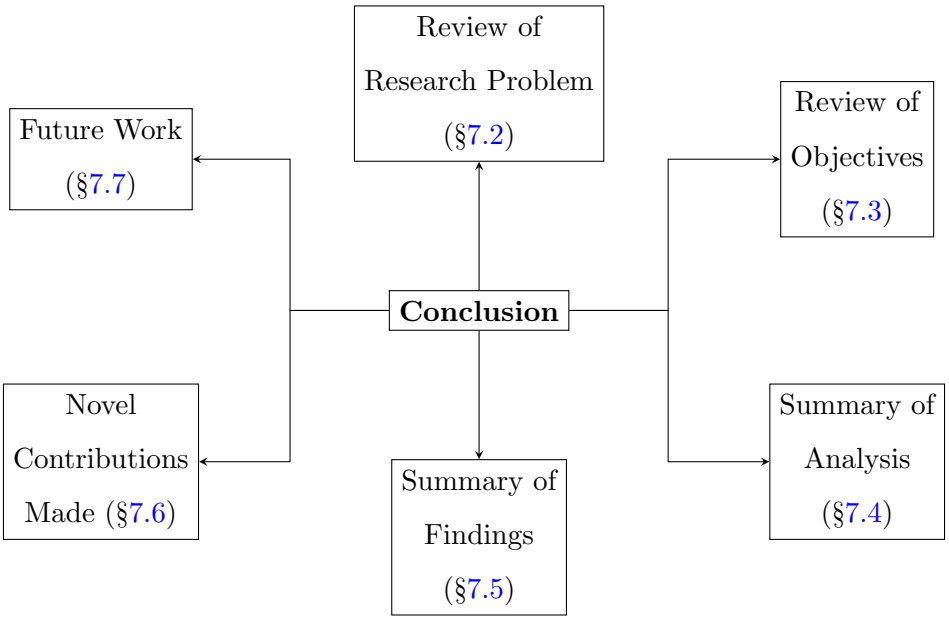
used in this study contains individual barchan outlines. Gao *et al.* (2021) speculates that the diverse size and morphology of whalebacks (a subclass of yardang) may have contributed to an increase in false positive and false negative classifications. Since size was corrected in the present study, it cannot be used as a potential explanation. It is more likely that the similarity of the barchans to each other contributed to incorrect classifications. More specifically, it is argued that there is an inverse relationship between model accuracy, and the proportion that the unique feature constitutes out of the total features extracted from the image.

The findings of this study agrees with the statement by Maggiori *et al.* (2017) that CNNs are capable of detecting shape information. The image input is, however, different from that typically used in remote sensing. However, the work of Azzaoui *et al.* (2019) has demonstrated that it is possible to extract barchan outlines from more "conventional" imagery. Therefore, if the method developed here is added as a second step, it can be used to extract the morphometric data from remotely sensed imagery. However, a considerable amount of additional work still needs to be carried out in order to improve the performance.

Chapter 7

Conclusion

7.1 Chapter Overview



7.2 Review of Problem Statement

The rapid accumulation of remotely sensed data requires novel data extraction methods so that it can be effectively used to study barchan dunes and dune field processes. Fortunately, this increase in data availability occurred at the same time as a prolific increase in the accessibility and sophistication of Artificial Intelligence (AI) technologies. A specific type of AI, Convolutional Neural Networks (CNNs), holds particular promise due its proven ability to accurately classify imagery. CNNs have been successfully applied to the identification of landforms within remotely sensed imagery, however, there are some unanswered questions. First, are CNNs capable of extracting morphometric data from images of barchans? By morphometric features, it is meant those properties of a barchan that are defined based on the shape and dimensions

of the barchan. Second, if a CNN can successfully extract morphometric data, how can the hyperparameters of the CNN be modified in order to improve performance? Model performance refers to the ability of the model, based on one or more metrics, to perform the task required of it. In this instance, this is a classification task. Hyperparameters refer to a collection of configurations that are programmed into the CNN model that have an influence on its capacity to learn from known examples in order to carry out the task required of it. In order to fully utilise the remotely sensed big data, both of these questions need to be addressed.

7.3 Review of Objectives

The objectives of the study, along with a brief description of how they were met, are given below.

1. **Collate data on the type of morphometric data that is typically used within the literature.**

   This was completed as part of the literature review (§2.2.2). One of the outputs of this objective is a summary of the different measurements carried out on barchan dunes, with the goal of describing their shape, which has until now been absent from the barchan literature.

2. **Develop tasks that cover a wide range of barchan morphological investigations with an emphasis on those that rely on barchan shape and asymmetry.**

   This objective was completed in Chapters 3 and 5. A notable output of this objective is the development of a novel method that allows for the combined interaction between different manifestations of asymmetry (such as longitudinal elongation and lateral extension) to be quantitatively expressed. This formed part of the identification of the extent of the bilateral asymmetry task. The other identified tasks include: describing barchan shape (based on the $\frac{a}{c}$ ratio), identifying which side of the barchan contains the elongated horn, identifying the side of the barchan that is laterally extended, identifying which side of the barchan contains the widest horn, and classifying the magnitude of horn elongation.

3. **Identify the various hyperparameters that are known to significantly influence the performance of CNNs.**

   A survey of the different hyperparpameters that exert an influence on the performance of CNNs are provided in Section 4.4.1. Although all hyperparameters have an influence on model performance, the depth of the model and the presence of regularizations can be considered to have an especially large influence.

4. **Select a suitable site that meets the following criteria: it must have high resolution imagery so that small barchans can be clearly delineated and it must contain isolated barchans.**

   A suitable site was identified in the Kunene region of northern Namibia. A description of this study is provided in Section 5.2. Imagery from this site dated 12 November 2012 contains images of individual barchans that have a resolution sufficient for accurate outline extraction.

5. **Prepare the imagery obtained from the site in order to remove the potential textural bias and to meet the input requirements of the CNNs.**

   By manually extracting the outlines (§5.5) it is possible to remove the texture data from the image while retaining the morphological components of the barchan. Since this is carried out in a vector graphics software, it is also possible to scale the images to uniform size without an associated loss of image quality. This ensures that the classification carried out by the CNN is not biased either in terms of the image texture data or the size of the barchan within the image.

6. **Develop and train several different CNN architectures that encompass the "from scratch" and transfer learning approaches. The former consists of variations in model architecture using the hyperparameters identified in objective 3.**

   A total of 14 CNN models are evaluated (§5.10). Twelve of these form part of the "from-scratch" and represent different combinations of depth and use of regularizations. Two of the models, VGG16 and ResNet50, are used as part of the transfer learning approach and represent commonly used models within the CNN literature.

7. **Evaluate the performance of each model against a defined set of criteria.**

   The models are evaluated using several different metrics (§5.12) that are commonly used in the literature: accuracy (ACC), F_1–score, and Matthews Correlation Coefficient (MCC). The ACC metric identifies the number of class 1 and class 2 images that were correctly classified. The F_1–score provides a means to account for false positive and false negative classifications. Lastly, the MCC value is an indication of the extent to which the models' classifications correlate with the actual classifications. A broad overview of the criteria (see §5.12) used to evaluate the performance of the models are:

   (a) An ACC and F_1–score above 70%, and an MCC value above 0.5.

(b) Does a modification to hyperparameters improve a model's performance in terms of ACC, F_1–score, and MCC?

(c) The probability of any given instantiation of the model achieving above the critical value (70% for ACC and F_1–scores, and 0.5 for MCC) and above a target value (90% for ACC and F_1–scores, and 0.9 for MCC).

8. **Interpret the results and make recommendations regarding the use of CNNs in barchan research.**

   This is covered extensively in Chapter 6 and also summarised in Section 7.5.

7.4 Summary of Analysis

To determine whether a CNN can extract morphometric features, a case study was carried out using a sample of barchans from the Kunene region in northern Namibia. Two approaches to CNN use are investigated: developing a network "from scratch" and training it on a custom data set (in this case barchan imagery); and using transfer learning where a model that was trained on an unrelated data set is used. Both of these strategies have advantages and disadvantages. The "from scratch" approach affords maximum flexibility in configuring the hyperparameters and, in principle, creating a more specialized CNN. In transfer learning, the ability to modify the hyperparmeters is sacrificed in favour of a model with a proven history of good image classification with a feature extractor that was trained using resources well beyond most users.

To avoid the CNN from classifying images based purely on pixel properties, which may not even be visible to humans, the outlines of the barchans were manually extracted and the images scaled to a uniform size. Given all possible permutations of hyperparameters it is not possible to systematically evaluate each combination. Instead, for the from scratch CNNs, emphasis was placed on modifying the depth of the model and adding regularizations such as the L1 and L2 regularizations, and by incorporating dropout layers. Increasing the depth is considered to result in an improved ability to extract more complex features from an object (where a feature refers to lines, curves, etc.) which can, in principle, enable the network to detect features that are specific to certain types of barchans. Regularizations and dropout layers attempt to increase the generalization ability of a CNN which prevents it from "memorizing" the images it was trained on. This allows it to generalize to cases that it has not been exposed to during training by looking for similarities between the "new" image and the images it was trained on.

Since the goal is to determine if the barchan research community can benefit from the use of CNNs, six classification tasks were developed that represent common uses of barchan shape and asymmetry currently present in the literature. The first involves classifying barchans based

on their shape which is here taken as the $\frac{a}{c}$–ratio. This value is often used to describe the shape of barchans (e.g. Barnes, 2001) and has been linked to barchan migration (Hamdan *et al.*, 2016) and changes in this ratio has been attributed to sediment dynamics (Parteli *et al.*, 2014). The second task requires the CNN to classify barchans based on the side of the dune where the elongated horn is located. Horn elongation can provide insights into large scale controls on barchan asymmetry such as, but not limited to, bimodal winds (e.g Courrech du Pont, 2015) and barchan collisions (e.g. Tsoar and Parteli, 2016). A third task investigates whether the CNN can determine on which side the widest horn is located. Since the width of the horn determines the amount of sediment lost at that horn (Elbelrhiti *et al.*, 2008), this morphemetric variable is useful for investigating the sediment transfer from barchans to downwind sinks. For the fourth task, the CNN needs to be able to classify barchans based on the side where the most lateral extension occurs. This morphometric property has been linked to the influence of additional downslope forces that have an influence on the shape of the barchan (Parteli *et al.*, 2014). The fifth task evaluates whether the CNN can classify barchans based on the magnitude of elongation of one of the horns (i.e. the difference in horn lengths). The elongation of one horn has been linked to the duration of oblique winds (Bourke, 2010), and to the influence of asymmetric sediment supply to the upwind portion of the dune (Parteli *et al.*, 2014). For the sixth task, it is determined whether the CNN can distinguish between the combined effect of several different forms of asymmetry, such as the combined effect of horn elongation, lateral extension, and changes in horn width. No method is available within the barchan literature to investigate this in a way that allows for easy classification. Therefore, a novel technique was developed based on geometric morphometrics.

Three different performance metrics are used: accuracy (ACC), F_1–score, and Mathew's Correlation Coefficient (MCC). The ACC metric quantifies the ability of the model to correctly place images into the correct category. It therefore only focuses on which classifications are correct. The F_1–score takes into consideration incorrect classifications as well and is expressed as the harmonic mean between model recall and precision. Increased amounts of incorrect predictions will, therefore, lower the F_1–score. Using the MCC metric, it is possible to determine the extent to which the model's classification correlates to the actual classification. MCC values range from -1 indicating total disagreement between the model's classifications and the actual classifications, to 0 indicating random classification on the part of the model, to $+1$ which indicates complete agreement.

7.5 Summary of Key Findings

The results show that CNNs are capable of completing the majority of tasks with a high degree of accuracy. The accuracy (ACC) ranges from 44% to 92%, the F_1–score from 35% to 92%, and the MCC values from 0.43 to 0.86. CNNs are capable of classifing the barchans with a high degree of accuracy in the majority of tasks. This indicates that, when provided with suitable imagery, CNNs have the capacity to serve as a time–saving addition to barchan research. However, the CNNs performed very poorly on the classification tasks based on horn widths. This can, potentially, be improved through the use of additional training images.

For all tasks the use of transfer learning has the highest probability of improving classification performance. This indicates that it is possible to use models that were pre–trained on imagery, unrelated to the task at hand, to classify barchans. However, this only refers to the feature extractor portion of the dune. In order to successfully use transfer learning, it is required that the fully-connected layer be modified to the number of categories required and, importantly, that this layer be trained to correctly use the features extracted for the desired classification.

The "from scratch" models are also able to classify the barchans into the correct classes. In some instances, their performance equals that of the pre–trained model. This indicates that it is not, necessarily, required to make use of transfer learning. Based on the results it is not possible to recommend a specific combination of hyperparameters that would result in a performance increase. The combinations of hyperparameters are very variable and the results suggest that the performance of the CNN is strongly dependent on the properties of the model when it is initialised.

Based on the findings it can be concluded that there is definitely potential in developing CNNs as a tool for barchan research. It is therefore possible to use remote sensing big data to address key areas of research related to barchans. Aside from aiding in the research upon which the tasks are based, other benefits are also possible. For example, by carrying out the automatic extraction of barchans for regions where meteorological data is available, the interpretation value of barchan shape, or asymmetry, as descriptors of climates can be improved. This has a direct benefit on the interpretation of the environments of other planets such as Mars. Another area of research is a better understanding of barchan processes based on an extensive catalogue of barchan shapes (as recommended by Bourke and Goudie (2009)). A sufficiently trained automated extraction system effectively reduces the constraints associated with time and space to one of data availability. As such, it is possible to increase both the spatial and the temporal scale of barchan research.

7.6 Contributions to Aeolian Geomorphology

This work makes a significant contribution to barchan research in the following ways:

1. While other work has focused on the ability of CNNs to either detect or extract barchans from imagery, this work is original in being the first to determine whether it is possible to extract usable morphometric data from imagery. This is a novel application within the field of aeolian research and, to a large extent, the discipline of geomorphology.

2. Currently, there is no method available to evaluate the combined influence of several different types of asymmetry and express it as a single variable. This study developed such new method. It is based on geometric morphometrics which makes it possible to describe either barchan shape (van der Merwe, 2021) or barchan asymmetry. Although the field of geometric morphometrics is not new, this is the first time it has been applied within this context to barchan dunes

3. It also presents a summary of the variety of morphologic parameters that have been recorded on barchans by researchers in the field. Aside from being of interest to the curious researcher, and a handy reference for method sources, it shows that there are a number of different morphometric variables that can be extracted from such a seemingly simple shape.

7.7 Future Work

Another beneficial outcome of this research is that it identified future areas of research:

1. First, the lack of improvement in performance with the addition of a dropout layer, which contradicts the findings of other researchers working on CNNs, may be the result of position within the network. In the present configuration, these layers are restricted to the feature extractor. This, in effect, exerts an influence over the type of features present within the feature map which, ultimately, influence the classification. This raises the question regarding the effects of placing the dropout layer within the classifier. By analogy, this is similar to shifting focus from the ability to recognise features towards classifying images based on the features at hand.

2. Another area of research deals directly with the issues raised on training data. The effect of data set sizes is taken here as self evident. However, a more useful approach would be to look at the amount of variability in images that make up the training data and relate this to model performance. In this study, the data set was not altered during different

runs of the training sequence. But can changes to the training data itself, in terms of the composition, affect the effectiveness of the training process?

3. An investigation of the activity maps of CNNs involved in barchan classification can be beneficial. It will assist in determining where within the image most of the "influential" input neurons are located. This will help identify the regions on the barchan that the CNN uses to make its classification.

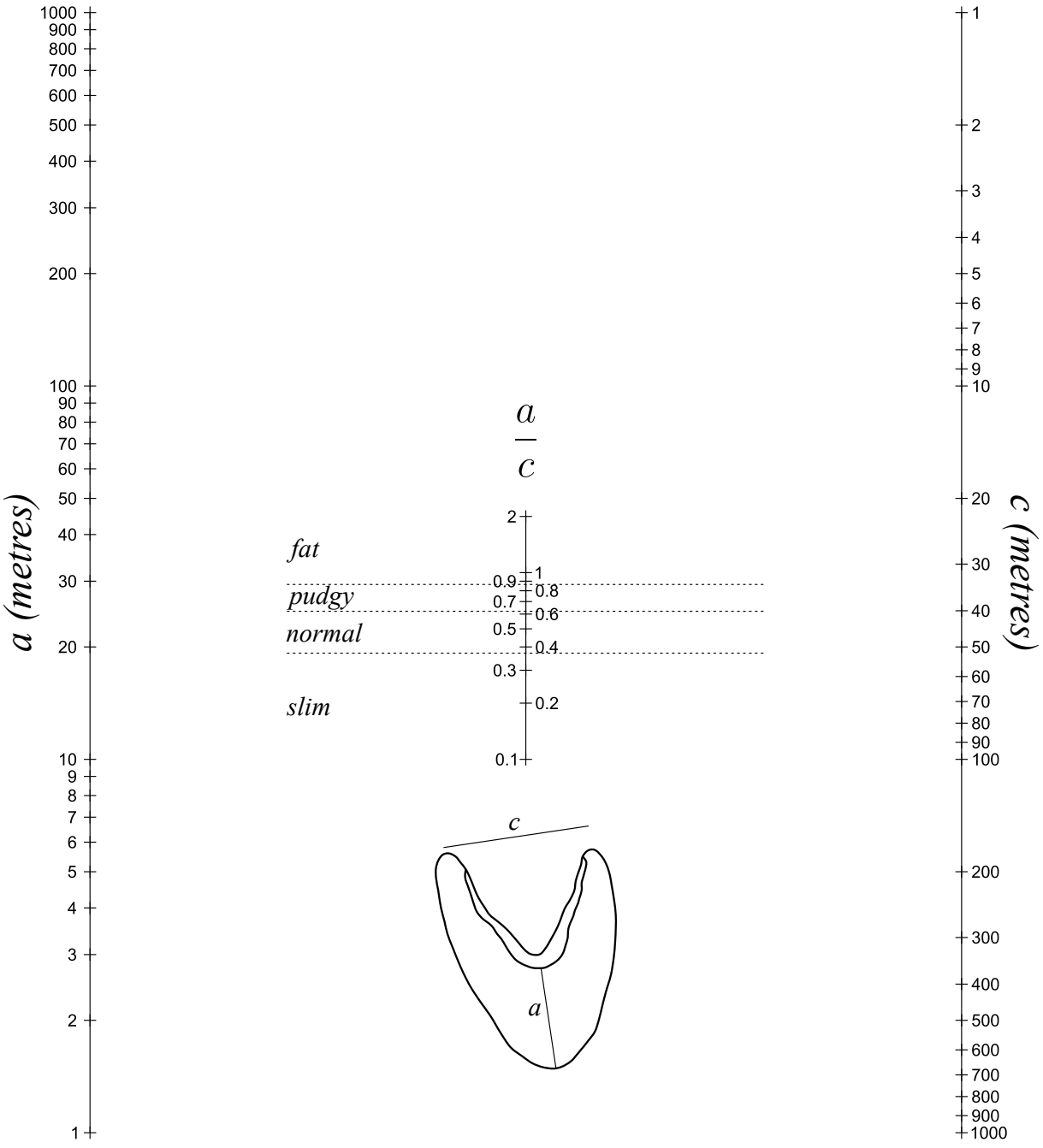
4. The parameters of this study required that the most simplistic use scenario be evaluated. Therefore, the images were all scaled to occupy the entire input size. However, future work should consider the extent to which the size of the barchan within the input window affects the performance of the CNN.

5. Although a surrogate for human classification accuracy was used, it would be beneficial to carry out a study using the same data to determine human performance on this task. This will aid in placing the performance of CNNs into a more relevant context.

6. Future work should also focus on the misclassifications. In this study, the output was taken only as being in one of two classes. However, since the data is actually a set of probabilities of an image belonging to each class, it is possible to relate the misclassified images to the level of "uncertainty" that the network had during the classification. A benefit of this is that it allows images where the CNN was uncertain (i.e. the assigned probabilities to the two classes are similar) to be investigated. This may yield insights into how the CNN interprets shape.

Appendix A

Barchan Shape Nomogram



Appendix B

ANN Learning Example

Consider the simple neural network depicted in Figure B.1. This is a simple neural network where there is a single input (x) that feeds to a single neuron ($a_1^{(0)}$) with a weight of w_1. The bias (b) for $a_1^{(0)}$ is kept constant at 1 and the activation function is the ReLU function (Eq. B.1) to produce a single output ($\hat{y}$). The bias term is not associated with a weight for this example. This ANN can then be presented as a computational graph (Figure B.2). The loss function for this network is Mean Square Error (MSE) and because only one input is received it can be represented as in (Eq. B.2).

$$\sigma(z) = \begin{cases} 0 & : z \leq 0 \\ z & : z > 0 \end{cases} \tag{Eq. B.1}$$

$$C = (y - \hat{y})^2 \tag{Eq. B.2}$$

B.1 Forward propagation

Given $x = 5$,$w = 3$, $b = 1$, and the output of the ANN should be $y = 8$ the forward propagation for this ANN involves the following:

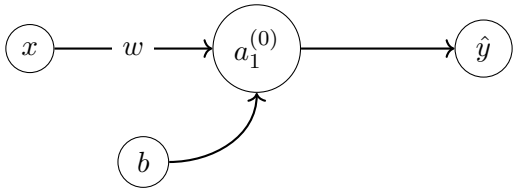


Figure B.1: A simple ANN consisting of one input, one node, and a single output. For the purpose of this example the bias is kept constant at 1 and the activation function is sigmoid.

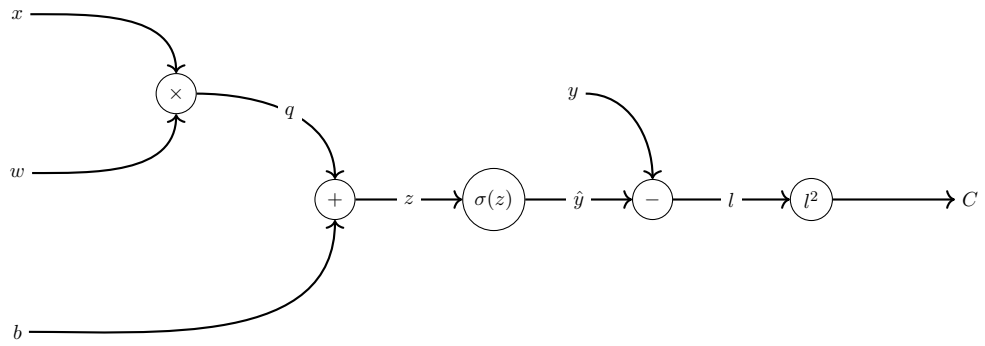


Figure B.2: The computation graph for the neuron described in Figure B.1 along with intermediate variables. The activation function is indicated by $\sigma()$.

$$q = wx = 15$$
$$z = q + b = 16$$
$$\hat{y} = \sigma(z) = 16$$

The output of this network is therefore 16 and the target value is 8. During back-propagation, this difference is used to calculate an adjustment to the parameter w.

B.2 Back-propagation

First the value of the cost function is calculated:

$$l = y - \hat{y} = -8$$
$$C = l^2 = 64$$

To determine how the network should be updated the partial derivatives of each variable with respect to the cost function needs to be determined:

$$\frac{\partial C}{\partial l} = 2l = -16 \qquad \frac{\partial l}{\partial \hat{y}} = -1$$
$$\frac{\partial \hat{y}}{\partial z} = 1 \qquad \frac{\partial z}{\partial q} = 1$$
$$\frac{\partial q}{\partial w} = 5$$

The chain rule can then be used to determine the partial derivative of the cost function with regards to the parameter w:

$$\begin{aligned}\frac{\partial C}{\partial w} &= \frac{\partial C}{\partial l} \times \frac{\partial l}{\partial \hat{y}} \times \frac{\partial \hat{y}}{\partial z} \times \frac{\partial z}{\partial q} \times \frac{\partial q}{\partial w} \\ &= 80\end{aligned}$$

Given a learning rate of $\eta = 0.01$ the parameter w is updated using the gradient descent algorithm[1]:

$$\begin{aligned}w_{new} &= w_{old} - \eta \frac{\partial C}{\partial w} \\ &= 3 - (0.01 \times 80) \\ &= 2.2\end{aligned}$$

[1]Only a single step of the algorithm is used as illustration.

On the next forward pass the values become:

$$q = wx = 11$$
$$z = q + b = 12$$
$$\hat{y} = \sigma(z) = 12$$
$$l = y - \hat{y} = -4$$
$$C = l^2 = 16$$

This shows that the w parameter is approaching the value needed in order for the network to make accurate predictions.

Appendix C

CNN Learning Example

Consider the CNN in Figure C.1 that represents a network that takes as input **X** a 1 band image (i.e. a grayscale) of 4×4 pixels. This network creates a single feature map (**F**) using a single kernel, **K**, which is then flattened into a fully connected network with a single output ($\hat{y}$) using the sigmoid activation function (Eq. C.1). The goal is for the network to classify the input image as one that has a line that is diagonal from the top left to the bottom right. Here, the category associated with this type of feature is 0 (i.e. $y = 0$) Because the sigmoid function is used, $\hat{y}$ will be a real value between 0 and 1.

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{Eq. C.1}$$

C.1 Forward propogation

Given an input with a line diagonally down from left to right, and with values $[0, 255]$, the input x can be described using the matrix in (Eq. C.2).

$$\mathbf{X} = \begin{bmatrix} 255 & 0 & 0 & 0 \\ 0 & 255 & 0 & 0 \\ 0 & 0 & 255 & 0 \\ 0 & 0 & 0 & 255 \end{bmatrix} \tag{Eq. C.2}$$

Upon initialisation, the kernel is random. Taking as an example random integers between [0,3] being assigned to the kernel, an example kernel can be represented by the matrix in (Eq. C.3).

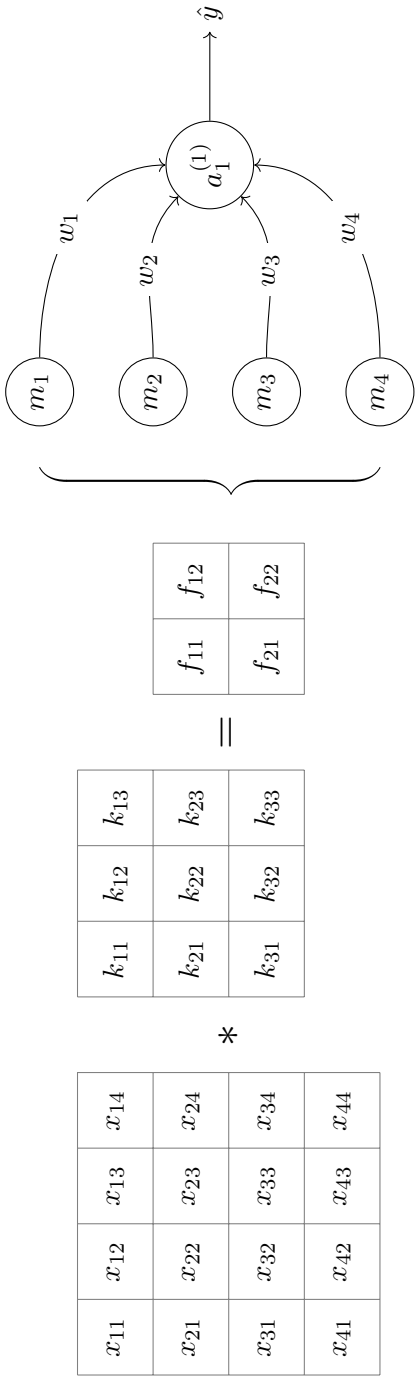


Figure C.1: A simplified CNN. A 4×4 input $\mathbf{X}$ is convolved by kernel $\mathbf{K}$ (i.e. $\mathbf{X} * \mathbf{K}$) to produce a feature map $\mathbf{F}$.

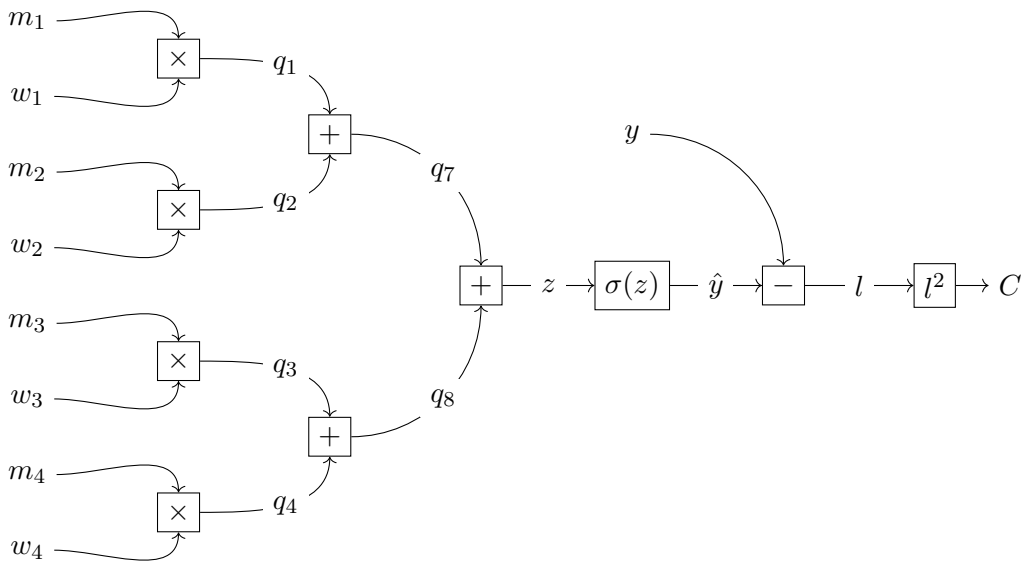


Figure C.2: Computation graph for the fully connected portion of the CNN in figure C.1.

$$\mathbf{K} = \begin{bmatrix} 0 & 0 & 1 \\ 3 & 1 & 1 \\ 2 & 1 & 1 \end{bmatrix} \qquad \text{(Eq. C.3)}$$

For each cell in the feature map, the convolution calculation is:

$$m_1 = F_{11} = (x_{11}k_{11}) + (x_{12}k_{12}) + (x_{13}k_{13}) + (x_{21}k_{21}) + (x_{22}k_{22}) + (x_{23}k_{23}) + (x_{31}k_{31}) + (x_{32}k_{32}) + (x_{33}k_{33})$$

$$m_2 = F_{21} = (x_{21}k_{11}) + (x_{22}k_{12}) + (x_{23}k_{13}) + (x_{31}k_{21}) + (x_{32}k_{22}) + (x_{33}k_{23}) + (x_{41}k_{31}) + (x_{42}k_{32}) + (x_{43}k_{33})$$

$$m_3 = F_{12} = (x_{12}k_{11}) + (x_{13}k_{12}) + (x_{14}k_{13}) + (x_{22}k_{21}) + (x_{23}k_{22}) + (x_{24}k_{23}) + (x_{32}k_{31}) + (x_{33}k_{32}) + (x_{34}k_{33})$$

$$m_4 = F_{21} = (x_{22}k_{11}) + (x_{23}k_{12}) + (x_{24}k_{13}) + (x_{32}k_{21}) + (x_{33}k_{22}) + (x_{34}k_{23}) + (x_{42}k_{31}) + (x_{43}k_{32}) + (x_{44}k_{33})$$

The feature map ($\mathbf{F}$) that results from the convolution of the input with the kernel ($\mathbf{X} * \mathbf{K}$) is given by (Eq. C.4).

$$\mathbf{F} = \begin{bmatrix} 510 & 1020 \\ 255 & 510 \end{bmatrix} \tag{Eq. C.4}$$

In this example the values of the flattened layer is derived from the feature map such that:

$$F_{11} = m_1 = 510 \qquad F_{12} = m_2 = 1020$$
$$F_{21} = m_3 = 255 \qquad F_{22} = m_4 = 510$$

Assigning a random value over the interval $[0, 1]$ to each of the weights the following values are obtained:

$$w_1 = 0.64 \qquad w_2 = 0.56$$
$$w_3 = 0.53 \qquad w_4 = 0.97$$

The prediction $\hat{y}$ then becomes[1]:

$$\begin{aligned} \hat{y} &= \sigma(\sum_{i=1}^{4}(m_i w_i)) \\ &= \sigma(1527.45) \\ &= 1 \end{aligned}$$

where σ is the sigmoid activation function.

C.2 Back-propagation

This process is conceptually the same as the back-propagation for an ANN, but more variables are involved due to the presence of the kernels and feature maps. For this example, the mean square error loss function will is used (Eq. C.5) and the actual value (y) is zero.

$$\begin{aligned} C &= (y - \hat{y})^2 \\ &= 1 \end{aligned} \tag{Eq. C.5}$$

[1]For the sake of simplicity the bias values are omitted.

Creating intermediate variables in order to simplify the calculations:

$$\hat{y} = \sigma(z)$$
$$z = \sum_{i=1}^{4}(m_i w_i)$$

The partial derivatives that determine how the loss function (C) is influenced by the learnable parameters in the kernel (i.e. each of the 9 values in the kernel) can then be presented as follows[2]:

$$\frac{\partial C}{\partial K_{11}} = \left(\frac{\partial m_1}{\partial K_{11}} \times \frac{\partial z}{\partial m_1} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_2}{\partial K_{11}} \times \frac{\partial z}{\partial m_2} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_3}{\partial K_{11}} \times \frac{\partial z}{\partial m_3} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_4}{\partial K_{11}} \times \frac{\partial z}{\partial m_4} \times \frac{\partial C}{\partial z}\right)$$

$$\frac{\partial C}{\partial K_{12}} = \left(\frac{\partial m_1}{\partial K_{12}} \times \frac{\partial z}{\partial m_1} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_2}{\partial K_{12}} \times \frac{\partial z}{\partial m_2} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_3}{\partial K_{12}} \times \frac{\partial z}{\partial m_3} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_4}{\partial K_{12}} \times \frac{\partial z}{\partial m_4} \times \frac{\partial C}{\partial z}\right)$$

$$\frac{\partial C}{\partial K_{13}} = \left(\frac{\partial m_1}{\partial K_{13}} \times \frac{\partial z}{\partial m_1} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_2}{\partial K_{13}} \times \frac{\partial z}{\partial m_2} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_3}{\partial K_{13}} \times \frac{\partial z}{\partial m_3} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_4}{\partial K_{13}} \times \frac{\partial z}{\partial m_4} \times \frac{\partial C}{\partial z}\right)$$

$$\frac{\partial C}{\partial K_{21}} = \left(\frac{\partial m_1}{\partial K_{21}} \times \frac{\partial z}{\partial m_1} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_2}{\partial K_{21}} \times \frac{\partial z}{\partial m_2} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_3}{\partial K_{21}} \times \frac{\partial z}{\partial m_3} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_4}{\partial K_{21}} \times \frac{\partial z}{\partial m_4} \times \frac{\partial C}{\partial z}\right)$$

$$\frac{\partial C}{\partial K_{22}} = \left(\frac{\partial m_1}{\partial K_{22}} \times \frac{\partial z}{\partial m_1} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_2}{\partial K_{22}} \times \frac{\partial z}{\partial m_2} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_3}{\partial K_{22}} \times \frac{\partial z}{\partial m_3} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_4}{\partial K_{22}} \times \frac{\partial z}{\partial m_4} \times \frac{\partial C}{\partial z}\right)$$

[2]For the sake of simplicity the weights are taken to be fixed

$$\frac{\partial C}{\partial K_{23}} = \left(\frac{\partial m_1}{\partial K_{23}} \times \frac{\partial z}{\partial m_1} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_2}{\partial K_{23}} \times \frac{\partial z}{\partial m_2} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_3}{\partial K_{23}} \times \frac{\partial z}{\partial m_3} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_4}{\partial K_{23}} \times \frac{\partial z}{\partial m_4} \times \frac{\partial C}{\partial z}\right)$$

$$\frac{\partial C}{\partial K_{31}} = \left(\frac{\partial m_1}{\partial K_{31}} \times \frac{\partial z}{\partial m_1} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_2}{\partial K_{31}} \times \frac{\partial z}{\partial m_2} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_3}{\partial K_{31}} \times \frac{\partial z}{\partial m_3} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_4}{\partial K_{31}} \times \frac{\partial z}{\partial m_4} \times \frac{\partial C}{\partial z}\right)$$

$$\frac{\partial C}{\partial K_{32}} = \left(\frac{\partial m_1}{\partial K_{32}} \times \frac{\partial z}{\partial m_1} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_2}{\partial K_{32}} \times \frac{\partial z}{\partial m_2} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_3}{\partial K_{32}} \times \frac{\partial z}{\partial m_3} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_4}{\partial K_{32}} \times \frac{\partial z}{\partial m_4} \times \frac{\partial C}{\partial z}\right)$$

$$\frac{\partial C}{\partial K_{33}} = \left(\frac{\partial m_1}{\partial K_{33}} \times \frac{\partial z}{\partial m_1} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_2}{\partial K_{33}} \times \frac{\partial z}{\partial m_2} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_3}{\partial K_{33}} \times \frac{\partial z}{\partial m_3} \times \frac{\partial C}{\partial z}\right) + \left(\frac{\partial m_4}{\partial K_{33}} \times \frac{\partial z}{\partial m_4} \times \frac{\partial C}{\partial z}\right)$$

Glossary

Abductive Reasoning One of three types of reasoning, the others being deductive and inductive, where reasoning proceeds from the observation to a speculated cause.

Allometry Refers to the relationship between the change in shape of a landfrom as its size changes.

Architecture Here, the term architecture refers to the number components of the CNN model, their properties, and how they transfer information through the model.

Convolution The sum of the element-wise multiplications between an input matrix and a kernel.

Convolutional Neural Network A neural network that is characterised by the presence of convolution operations.

Equifinality A debated concept within geomorphology which argues that the same landfrom may result from different processes that stem form different starting conditions.

Features A geometric property of an image such as edges or curves. Can be combined into more "abstract" geometries, such as peaks, half-circles, in deeper layers within a model.

Genesis Within the context of this study, genesis refers to the development of a barchan from a loalised sediment accumulation.

Hyperparameters Those parameters of a Neural Network that determine the learning strategy that the model will follow, how information is processed within the model, and what transformations will be applied to the inputs. Hyperparameters remain fixed after the model is initialised.

Isometry When the shape of a landform does not change as its size changes.

Learnable Parameters Those parameters within a neural network, such as weights and kernel values, that are updated during the training of the model. The learnable parameters remain fixed once the training process has been completed.

Machine Learning Algorithms that allow software to improve its performance on a task based on some metric.

Neural Network A collection of individual processing units that attempt to reproduce the information processing capacity of a human brain.

Neuron An information processing unit consisting of several parts: connections that trasnfer information to the neuron from earlier neurons in the network, a summing function that combines the weights of the conections with the outputs of the earlier neurons, an activation function which converts the linear summing function into a nonlinear variable, connections that transfer the output of the neuron to deeper layers in the network.

Object Based Image Analysis A GIS technique that partions a raster image into different classes based on the data contained within the pixels.

Optimizer A function that determines how the learnable parameters within a neural network are updated during the training process.

Overfitting A situation that arises when the model is trained excessively and becomes so fitted to the trainng data that it loses its generalization capabilities.

Regulizer An additional numeric value that is added to the loss function during the training process in order to reduce the generalization error of the model. The procesess is collectively referred to as regularization.

Semantic Segmentation A classification approach whereby every pixel within an image is assigned to a category based on its value.

Vanishing Gradient Problem A situation that occurs during the trainng of a model where the update to a parameter is negligibly small to the extent that any additional update becomes negligibly small.

References

Abdi, H., Valentin, D. and Edelman, B. (2011) *Qunatitative applications in the social sciences: Neural Networks*. Thousand Oaks: SAGE Publications, Inc. doi:10.4135/9781412985277.

Abdu, A. (1976) The crescentic dunes of northern Sudan. *East African Geographical Review*, 14(June), 61–71.

Abu-Zeid, M.M., Baghdady, A.R. and El-Etr, H.A. (2001) Textural attributes, mineralogy and provenance of sand dune fields in the greater Al Ain area, United Arab Emirates. *Journal of Arid Environments*, 48(4), 475–499. doi:10.1006/jare.2000.0776.

Adams, D.C. and Otárola-Castillo, E. (2013) Geomorph: An R package for the collection and analysis of geometric morphometric shape data. *Methods in Ecology and Evolution*, 4(4), 393–399. doi:10.1111/2041-210X.12035.

Adams, D.C., Rohlf, F.J. and Slice, D.E. (2004) Geometric morphometrics: Ten years of progress following the 'revolution'. *Italian Journal of Zoology*, 71(1), 5–16. doi:10.1080/11250000409356545.

Agarwal, T. and Mittal, H. (2019) Performance Comparison of Deep Neural Networks on Image Datasets. In: *2019 Twelfth International Conference on Contemporary Computing (IC3)*. IEEE, pp. 1–6. doi:10.1109/IC3.2019.8844924.

Al-Dousari, A.M., Al-Enezi, A.K. and Al-Awadhi, J. (2015) Textural variations within different representative types of dune sediments in Kuwait. *Arabian Journal of Geosciences*, 1(1), 17–31. doi:10.1007/s12517-008-0002-4.

Al-Dousari, A.M. and Pye, K. (2005) Mapping and monitoring of dunes in Northwestern Kuwait. *Kuwait Journal of Science and Engineering*, 32(2), 119–134.

Al-Harthi, A.A. (2002) Geohazard assessment of sand dunes between Jeddah and Al-Lith, western Saudi Arabia. *Environmental Geology*, 42(4), 360–369. doi:10.1007/s00254-001-0501-z.

Alvarez, C.A. and Franklin, E.M. (2018) Role of Transverse Displacements in the Formation of Subaqueous Barchan Dunes. *Physical Review Letters*, 121(16), 164503. doi: 10.1103/PhysRevLett.121.164503.

Andreotti, B., Claudin, P. and Douady, S. (2002) Selection of dune shapes and velocities part 1: Dynamics of sand, wind and barchans. *European Physical Journal B*, 28(3), 321–339. doi:10.1140/epjb/e2002-00236-4.

Armi, L. and Fekri-Ershad, S. (2019) Texture image analysis and texture classification methods - A review. *International Online Journal of Image Processing and Pattern Recognition*, 2(1), 1–29.

Ashkenazy, Y., Yizhaq, H. and Tsoar, H. (2012) Sand dune mobility under climate change in the Kalahari and Australian deserts. *Climatic Change*, 112(3-4), 901–923. doi:10.1007/s10584-011-0264-9.

Atkinson, P.M. and Tatnall, A.R.L. (1997) Introduction Neural networks in remote sensing. *International Journal of Remote Sensing*, 18(4), 699–709. doi:10.1080/014311697218700.

Aydda, A., Althuwaynee, O.F. and Pokharel, B. (2020) An easy method for barchan dunes automatic extraction from multispectral satellite data. *IOP Conference Series: Earth and Environmental Science*, 419(1). doi:10.1088/1755-1315/419/1/012015.

Azzaoui, M.A., Adnani, M., El Belrhiti, H., Chaouki, I.E. and Masmoudi, L. (2019) Detection of crescent sand dunes contours in satellite images using an active shape model with a cascade classifier. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 42(4/W12), 17–24. doi:10.5194/isprs-archives-XLII-4-W12-17-2019.

Azzaoui, M.A., Masmoudi, L., El Belrhiti, H. and Chaouki, I.E. (2020) Barchan sand dunes collisions detection in high resolution satellite images based on image clustering and transfer learning. *International Journal of Advanced Computer Science and Applications*, 11(1), 376–383. doi:10.14569/ijacsa.2020.0110147.

Baddock, M.C., Nield, J.M. and Wiggs, G.F. (2018) Early-stage aeolian protodunes: Bedform development and sand transport dynamics. *Earth Surface Processes and Landforms*, 43(1), 339–346. doi:10.1002/esp.4242.

Bagnold, R.A. (1931) Journeys in the Libyan Desert 1929 and 1930. *The Geographical Journal*, 78(1), 13–33.

Bagnold, R.A. (1933) A further journey through the Libyan Desert. *The Geographical Journal*, 82(2), 103–126.

Bagnold, R.A. (1951) Sand formations in southern Arabia. *The Geographical Journal*, 117(1), 78–86.

Bagnold, R.A. (1954) *The physics of blown sand and desert dunes*. Mineola: Dover Publications.

Bailey, S.I. (1906) The sand dunes of the Desert of Islay. In: *Annals of the Astronomical Observatory of Harvard College*, volume XXXIX. Observatory of Harvard College.

Baker, N., Lu, H., Erlikhman, G. and Kellman, P.J. (2018) Deep convolutional networks do not classify based on global object shape. *PLoS Computational Biology*, 14(12), 1–43. doi: 10.1371/journal.pcbi.1006613.

Baker, N., Lu, H., Erlikhman, G. and Kellman, P.J. (2020) Local features and global shape information in object classification by deep convolutional neural networks. *Vision Research*, 172(March), 46–61. doi:10.1016/j.visres.2020.04.003.

Ball, J.E., Anderson, D.T. and Chan, C.S. (2017) Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community. *Journal of Applied Remote Sensing*, 11(04), 1. doi:10.1117/1.jrs.11.042609.

Bandeira, L., Marques, J.S., Saraiva, J. and Pina, P. (2013) Advances in automated detection of sand dunes on Mars. *Earth Surface Processes and Landforms*, 38(3), 275–283. doi:10.1002/esp.3323.

Bao, X. and Zhang, F. (2013) Evaluation of NCEP-CFSR, NCEP-NCAR, ERA-Interim, and ERA-40 reanalysis datasets against independent sounding observations over the Tibetan Plateau. *Journal of Climate*, 26(1), 206–214. doi:10.1175/JCLI-D-12-00056.1.

Barnes, J. (2001) Barchan dunes on the Kuiseb River Delta, Namibia. *South African Geographical Journal*, 83(3), 283–292. doi:10.1080/03736245.2001.9713747.

Bauer, B.O. (2013) Fundamentals of Aeolian Sediment Transport: Boundary-Layer Processes. In: *Treatise on Geomorphology*, volume 11, chapter 2. Elsevier, pp. 7–22. doi: 10.1016/B978-0-12-374739-6.00295-5.

Baumhoer, C.A., Dietz, A.J., Kneisel, C. and Kuenzer, C. (2019) Automated extraction of antarctic glacier and ice shelf fronts from Sentinel-1 imagery using deep learning. *Remote Sensing*, 11(21), 1–22. doi:10.3390/rs11212529.

Beadnell, H.J.L. (1910) The sand-dunes of the Libyan Desert. Their origin, form, and rate of movement, considered in relation to the geological and meteorological conditions of the region. *The Geographical Journal*, 35(4), 379. doi:10.2307/1777018.

Belciug, S. and El-Darzi, E. (2010) A partially connected neural network-based approach with application to breast cancer detection and recurrence. *2010 IEEE International Conference on Intelligent Systems, IS 2010 - Proceedings*, 191–196. doi:10.1109/IS.2010.5548358.

Bellows, B., Bhandari, A., Ibrahim, M. and Sandhu, J.S. (2011) Peering into the Black Box. *Information Communication Technologies*, (January), 38–43. doi:10.4018/9781599049496.ch067.

Bendjillali, R.I., Beladgham, M., Merit, K. and Taleb-Ahmed, A. (2020) Illumination-robust face recognition based on deep convolutional neural networks architectures. *Indonesian Journal of Electrical Engineering and Computer Science*, 18(2), 1015–1027. doi:10.11591/ijeecs.v18.i2.pp1015-1027.

Benuwa, B.B., Zhan, Y.Z., Ghansah, B., Wornyo, D.K. and Banaseka Kataka, F. (2016) A Review of Deep Machine Learning. *International Journal of Engineering Research in Africa*, 24, 124–136. doi:10.4028/www.scientific.net/JERA.24.124.

Bera, S. and Shrivastava, V.K. (2020) Analysis of various optimizers on deep convolutional neural network model in the application of hyperspectral remote sensing image classification. *International Journal of Remote Sensing*, 41(7), 2664–2683. doi:10.1080/01431161.2019.1694725.

Bhuiyan, M.A.E., Witharana, C., Liljedahl, A.K., Jones, B.M., Daanen, R., Epstein, H.E., Kent, K., Griffin, C.G. and Agnew, A. (2020) Understanding the effects of optimal combination of spectral bands on deep learning model predictions: A case study based on permafrost tundra landform mapping using high resolution multispectral satellite imagery. *Journal of Imaging*, 6(9). doi:10.3390/JIMAGING6090097.

Bishop, M.A. (2001) Seasonal variation of crescentic dune morphology and morphometry, Strzelecki-Simpson Desert, Australia. *Earth Surface Processes and Landforms*, 26(7), 783–791. doi:10.1002/esp.223.

Bishop, M.A. (2007) Point pattern analysis of north polar crescentic dunes, Mars: A geography of dune self-organization. *Icarus*, 191(1), 151–157. doi:10.1016/j.icarus.2007.04.027.

Bishop, S.R., Momiji, H., Carretero-González, R. and Warren, A. (2002) Modelling desert dune fields based on discrete dynamics. *Discrete Dynamics in Nature and Society*, 7(1), 7–17. doi:10.1080/10260220290013462.

Bivand, R. and Lewin-Koh, N. (2016) Maptools: Tools for reading and handling spatial objects (R package version 0.8-39) [computer software].

Borówka, R.K. (1980) Present day dune processes and dune morphology on the Łeba Barrier, Polish coast of the Baltic. *Geografiska Annaler. Series A, Physical Geography*, 62(1/2), 75. doi:10.2307/520454.

Boulghobra, N. (2016) Climatic data and satellite imagery for assessing the aeolian sand deposit and barchan migration, as a major risk sources in the region of In-Salah (Central Algerian Sahara). *Arabian Journal of Geosciences*, 9(6), 450. doi:10.1007/s12517-016-2491-x.

Boulghobra, N. and Dridi, H. (2016) Fine resolution imagery and gis for investigating the morphological characteristics, and migration rate of barchan dunes in the erg sidi moussa dunefield near in-salah (Algeria). *Geographia Technica*, 11(2), 14–21. doi:10.21163/GT_2016.112.02.

Boulila, W., Sellami, M., Driss, M., Al-Sarem, M., Safaei, M. and Ghaleb, F.A. (2021) RS-DCNN: A novel distributed convolutional-neural-networks based-approach for big remote-sensing image classification. *Computers and Electronics in Agriculture*, 182(January), 106014. doi:10.1016/j.compag.2021.106014.

Bourke, M.C. (2010) Barchan dune asymmetry: Observations from Mars and Earth. *Icarus*, 205(1), 183–197. doi:10.1016/j.icarus.2009.08.023.

Bourke, M.C., Balme, M., Beyer, R.A., Williams, K.K. and Zimbelman, J. (2006) A comparison of methods used to estimate the height of sand dunes on Mars. *Geomorphology*, 81(3-4), 440–452. doi:10.1016/j.geomorph.2006.04.023.

Bourke, M.C., Ewing, R.C., Finnegan, D. and McGowan, H.A. (2009) Sand dune movement in the Victoria Valley, Antarctica. *Geomorphology*, 109(3-4), 148–160. doi:10.1016/j.geomorph.2009.02.028.

Bourke, M.C. and Goudie, A.S. (2009) Varieties of barchan form in the Namib Desert and on Mars. *Aeolian Research*, 1(1-2), 45–54. doi:10.1016/j.aeolia.2009.05.002.

Breed, C.S. (1977) Terrestrial analogs of the Hellespontus Dunes, Mars. *Icarus*, 30, 326–340. doi:10.1016/0019-1035(77)90168-3.

Breed, C.S. and Grow, T. (1979) Morphology and distribution of dunes in sand seas observed by remote sensing. In: E.D. McKee (Ed.) *A Study of Global Sand Seas: Geological Survey Professional Paper 1052*, chapter J. Washington: United States Government Printing Office, pp. 253 – 302.

Brown, J.B. (2018) Classifiers and their Metrics Quantified. *Molecular Informatics*, 37(1), 1–11. doi:10.1002/minf.201700127.

Bui, V.H., Bui, M.D. and Rutschmann, P. (2019) Combination of discrete element method and artificial neural network for predicting porosity of gravel-bed river. *Water (Switzerland)*, 11(7). doi:10.3390/w11071461.

Burrough, S.L., Thomas, D.S.G., Bailey, R.M. and Davies, L. (2012) From landform to process: Morphology and formation of lake-bed barchan dunes, Makgadikgadi, Botswana. *Geomorphology*, 161-162, 1–14. doi:10.1016/j.geomorph.2012.03.027.

Cacchione, D.A., Field, M.E., Drake, D.E. and Tate, G.B. (1987) Crescentic dunes on the inner continental shelf off northern California. *Geology*, 15(12), 1134–1137. doi:10.1130/0091-7613(1987)15⟨1134:CDOTIC⟩2.0.CO;2.

Cady, F. (2017) *The Data Science Handbook*. Hoboken, New Jersey: John Wiley & Sons, Inc. doi:10.1002/9781119092919.

Carbonneau, P.E., Belletti, B., Micotti, M., Lastoria, B., Casaioli, M., Mariani, S., Marchetti, G. and Bizzi, S. (2020) UAV-based training for fully fuzzy classification of Sentinel-2 fluvial scenes. *Earth Surface Processes and Landforms*, 45(13), 3120–3140. doi:10.1002/esp.4955.

Cardinale, M., Komatsu, G., Silvestro, S. and Tirsch, D. (2012) The influence of local topography for wind direction on Mars: Two examples of dune fields in crater basins. *Earth Surface Processes and Landforms*, 37(13), 1437–1443. doi:10.1002/esp.3289.

Carrara, F., Falchi, F., Caldelli, R., Amato, G. and Becarelli, R. (2018) Adversarial image detection in deep neural networks. *Multimedia Tools and Applications*, 1–21. doi:10.1007/s11042-018-5853-4.

Castelluccio, M., Poggi, G., Sansone, C. and Verdoliva, L. (2015) Land Use Classification in Remote Sensing Images by Convolutional Neural Networks, 1–11. doi:10.1021/cr8002505.

Chen, Z., Zhang, Y., Ouyang, C., Zhang, F. and Ma, J. (2018) Automated landslides detection for mountain cities using multi-temporal remote sensing imagery. *Sensors (Switzerland)*, 18(3). doi:10.3390/s18030821.

Cheng, G., Yang, C., Yao, X., Guo, L. and Han, J. (2018) When Deep Learning Meets Metric Learning: Remote Sensing Image Scene Classification via Learning Discriminative CNNs. *IEEE Transactions on Geoscience and Remote Sensing*, 56(5), 2811–2821. doi:10.1109/TGRS.2017.2783902.

Chevtchenko, S.F., Vale, R.F., Macario, V. and Cordeiro, F.R. (2018) A convolutional neural network with feature fusion for real-time hand posture recognition. *Applied Soft Computing Journal*, 73, 748–766. doi:10.1016/j.asoc.2018.09.010.

Chicco, D. and Jurman, G. (2020) The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, 21(1), 1–13. doi:10.1186/s12864-019-6413-7.

Chojnacki, M., Burr, D.M., Moersch, J.E. and Michaels, T.I. (2011) Orbital observations of contemporary dune activity in Endeavor crater, Meridiani Planum, Mars. *Journal of Geophysical Research E: Planets*, 116(4), 1–20. doi:10.1029/2010JE003675.

Chollet, F. and Allaire, J. (2018) *Deep learning with R*. Shelter Island: Manning.

Ciresan, D., Meier, U., Masci, J. and Schmidhuber, J. (2011) A committee of neural networks for traffic sign classification. In: *The 2011 International Joint Conference on Neural Networks*, volume 1. IEEE, pp. 1918–1921. doi:10.1109/IJCNN.2011.6033458.

Claes, P., Daniels, K., Walters, M., Clement, J., Vandermeulen, D. and Suetens, P. (2012) Dysmorphometrics: the modelling of morphological abnormalities. *Theoretical Biology and Medical Modelling*, 9(1), 5. doi:10.1186/1742-4682-9-5.

Clarke, G.K.C., Berthier, E., Schoof, C.G. and Jarosch, A.H. (2009) Neural networks applied to estimating subglacial topography and glacier volume. *Journal of Climate*, 22(8), 2146–2160. doi:10.1175/2008JCLI2572.1.

Clevert, D.A., Unterthiner, T. and Hochreiter, S. (2016) Fast and accurate deep network learning by exponential linear units (ELUs). *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*, 1–14.

Cornish, V. (1897) On the formation of sand-dunes. *The Geographical Journal*, 9(3), 278–302. doi:10.2307/1774940.

Courrech du Pont, S. (2015) Dune morphodynamics. *Comptes Rendus Physique*, 16(1), 118–138. doi:http://dx.doi.org/10.1016/j.crhy.2015.02.002.

Dakir, D., Rhinane, H., Saddiqi, O., El Arabi, E. and Baidder, L. (2016) Automatic Extraction of Dunes From Google Earth Images New Approach To Study the Dunes Migration in the Laâyoune City of Morocco. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W1(October), 53–59. doi:10.5194/isprs-archives-XLII-2-W1-53-2016.

Daniell, J.J. and Hughes, M. (2007) The morphology of barchan-shaped sand banks from western Torres Strait, northern Australia. *Sedimentary Geology*, 202(4), 638–652. doi:10.1016/j.sedgeo.2007.07.007.

De Cesarei, A., Cavicchi, S., Cristadoro, G. and Lippi, M. (2021) Do Humans and Deep Convolutional Neural Networks Use Visual Information Similarly for the Categorization of Natural Scenes? *Cognitive Science*, 45(6). doi:10.1111/cogs.13009.

de Lima, R.P. and Marfurt, K. (2020) Convolutional neural network for remote-sensing scene classification: Transfer learning analysis. *Remote Sensing*, 12(1). doi:10.3390/rs12010086.

Diaz-Pinto, A., Morales, S., Naranjo, V., Köhler, T., Mossi, J.M. and Navea, A. (2019) CNNs for automatic glaucoma assessment using fundus images: An extensive validation. *BioMedical Engineering Online*, 18(1), 1–19. doi:10.1186/s12938-019-0649-y.

Dong, Z., Wang, X. and Chen, G. (2000) Monitoring sand dune advance in the Taklimakan Desert. *Geomorphology*, 35(3-4), 219–231. doi:10.1016/S0169-555X(00)00039-8.

Douglass, A.E. (1909) The crescentic dunes of Peru. *Appalachia*, 12(1), 34 – 45.

Dryden, I.L. and Mardia, K.V. (1993) Multivariate shape analysis. *Sankhya: The Indian Journal of Statistics, Series A (1961-2002)*, 55(3), 460–480.

Dryden, I.L. and Mardia, K.V. (2016) *Statistical Shape Analysis, with Applications in R*. Wiley Series in Probability and Statistics, 2nd edition. Chichester, UK: John Wiley & Sons, Ltd. doi:10.1002/9781119072492.

Du, B., Zhao, Z., Hu, X., Wu, G., Han, L., Sun, L. and Gao, Q. (2021) Landslide susceptibility prediction based on image semantic segmentation. *Computers and Geosciences*, 155(November 2020), 104860. doi:10.1016/j.cageo.2021.104860.

Durán, O., Parteli, E.J.R. and Herrmann, H.J. (2010) A continuous model for sand dunes: Review, new developments and application to barchan dunes and barchan dune fields. *Earth Surface Processes and Landforms*, 35(13), 1591–1600. doi:10.1002/esp.2070.

Durán, O., Schwämmle, V., Lind, P.G. and Herrmann, H.J. (2009) The dune size distribution and scaling relations of barchan dune fields. *Granular Matter*, 11, 7–11. doi:10.1007/s10035-008-0120-4.

El belrhiti, H. and Douady, S. (2011) Equilibrium versus disequilibrium of barchan dunes. *Geomorphology*, 125(4), 558–568. doi:10.1016/j.geomorph.2010.10.025.

Elbelrhiti, H. (2012) Initiation and early development of barchan dunes: A case study of the Moroccan Atlantic Sahara desert. *Geomorphology*, 138(1), 181–188. doi:10.1016/j.geomorph.2011.08.033.

Elbelrhiti, H., Andreotti, B. and Claudin, P. (2008) Barchan dune corridors: Field characterization and investigation of control parameters. *Journal of Geophysical Research: Earth Surface*, 113(2), 1–21. doi:10.1029/2007JF000767.

Elliott, D.L. (1993) A Better Activation Function for Artificial Neural Networks. ISR Technical Report TR 93-8. Technical report, Institute for Systems Research, University of Maryland.

Elsayed, N., Maida, A.S. and Bayoumi, M. (2018) Empirical activation function effects on unsupervised convolutional LSTM learning. *Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI*, 2018-Novem, 336–343. doi:10.1109/ICTAI.2018.00060.

Embabi, N.S. (1982) Barchans of the Kharga Depression. In: F. El-Baz and T.A. Maxwell (Eds.) *Desert Landforms of Southwest Egypt: A Basis for Comparison with Mars*, chapter 11. Washington, DC: NASA Scientific and Technical Information Branch, p. 372.

Embabi, N.S. and Ashour, M.M. (1993) Barchan dunes in Qatar. *Journal of Arid Environments*, 25(1), 49–69. doi:10.1006/jare.1993.1042.

Engel, M., Boesl, F. and Brückner, H. (2018) Migration of barchan dunes in qatar–controls of the shamal, teleconnections, sea-level changes and human impact. *Geosciences (Switzerland)*, 8(7). doi:10.3390/geosciences8070240.

Engelbrecht, A.P. (2007) *Computational intelligence: An introduction*. 2nd edition. Chichester, UK: John Wiley & Sons, Ltd. doi:10.1002/9780470512517.

Ertam, F. (2017) Data classification with deep learning using tensorflow. *2nd International Conference on Computer Science and Engineering, UBMK 2017*, 755–758. doi:10.1109/UBMK.2017.8093521.

Ertel, W. (2011) *Introduction to Artificial Intelligence*. Undergraduate Topics in Computer Science. London: Springer London. doi:10.1007/978-0-85729-299-5.

Ewing, R.C., Hayes, A.G. and Lucas, A. (2014) Sand dune patterns on Titan controlled by long-term climate cycles. *Nature Geoscience*, 8(1), 15–19. doi:10.1038/ngeo2323.

Farzad, A., Mashayekhi, H. and Hassanpour, H. (2019) A comparative performance analysis of different activation functions in LSTM networks for classification. *Neural Computing and Applications*, 31(7), 2507–2521. doi:10.1007/s00521-017-3210-6.

Finkel, H.J. (1959) The barchans of southern Peru. *The Journal of Geology*, 67(6), 614–647. doi:10.1086/626622.

Floreano, D. and Mattiussi, C. (2008) *Bio-inspired artificial intelligence: Theories, methods, and technologies*. Cambridge: MIT Press.

Forouzan, B. and Mosharraf, F. (2012) *Foundations of computer science*. 2nd edition. Hampshire: Cengage Learning.

Forsyth, T.D. (1876) On the buried cities in the shifting sands of the great desert of Gobi. *Proceedings of the Royal Geographical Society of London*, 21(1), 27–46. doi:10.2307/1799910.

Franklin, E.M. and Charru, F. (2011) Subaqueous barchan dunes in turbulent shear flow. Part 1. Dune motion. *Journal of Fluid Mechanics*, 675, 199–222. doi:10.1017/S0022112011000139.

Fryberger, S.G. (1979) Dune forms and wind regime. In: E.D. McKee (Ed.) *A Study of Global Sand Seas: Geological Survey Professional Paper 1052*, chapter F. Washington: United States Government Printing Office, p. 439.

Gafurov, A.M. and Yermolayev, O.P. (2020) Automatic Gully Detection: Neural Networks and Computer Vision. *Remote Sensing*, 12(11). doi:10.3390/rs12111743.

Gao, B., Chen, N., Blaschke, T., Wu, C.Q., Chen, J., Xu, Y., Yang, X. and Du, Z. (2021) Automated characterization of yardangs using deep convolutional neural networks. *Remote Sensing*, 13(4), 1–19. doi:10.3390/rs13040733.

Gao, X., Narteau, C., Rozier, O. and du Pont, S.C. (2015) Phase diagrams of dune shape and orientation depending on sand availability. *Scientific reports*, 5, 14677. doi:10.1038/srep14677.

Gardin, E., Allemand, P., Quantin, C., Silvestro, S. and Delacourt, C. (2012) Dune fields on Mars: Recorders of a climate change? *Planetary and Space Science*, 60(1), 314–321. doi:10.1016/j.pss.2011.10.004.

Gay, S.P. (1999) Observations regarding the movement of barchan sand dunes in the Nazca to Tanaca area of southern Peru. *Geomorphology*, 27, 279–293. doi:10.1016/S0169-555X(98)00084-1.

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F.A. and Brendel, W. (2019) ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In: *International Conference on Learning Representations*. pp. 1 – 22.

Geum, Y.H., Rathie, A.K. and Kim, H. (2020) Matrix expression of convolution and its generalized continuous form. *Symmetry*, 12(11), 1–13. doi:10.3390/sym12111791.

Ghorbanzadeh, O., Blaschke, T., Gholamnia, K., Meena, S.R., Tiede, D. and Aryal, J. (2019) Evaluation of different machine learning methods and deep-learning convolutional neural networks for landslide detection. *Remote Sensing*, 11(2). doi:10.3390/rs11020196.

Gómez-Ortiz, D., Martín-Crespo, T., Rodríguez, I., Sánchez, M.J. and Montoya, I. (2009) The internal structure of modern barchan dunes of the Ebro River Delta (Spain) from ground penetrating radar. *Journal of Applied Geophysics*, 68(2), 159–170. doi:10.1016/j.jappgeo.2008.11.007.

Goodfellow, I., Bengio, Y. and Courville, A. (2016) *Deep Learning*. Cambridge, Massachusetts: MIT Press.

Goudie, A.S. (2020) Global barchans: A distributional analysis. *Aeolian Research*, 44(February). doi:10.1016/j.aeolia.2020.100591.

Grove, A.T. (1969) Landforms and climatic change in the Kalahari and Ngamiland. *The Geographical Journal*, 135(2), 191. doi:10.2307/1796824.

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J. and Chen, T. (2018) Recent advances in convolutional neural networks. *Pattern Recognition*, 77, 354–377. doi:https://doi.org/10.1016/j.patcog.2017.10.013.

Gundogdu, O., Egrioglu, E., Aladag, C.H. and Yolcu, U. (2016) Multiplicative neuron model artificial neural network based on Gaussian activation function. *Neural Computing and Applications*, 27(4), 927–935. doi:10.1007/s00521-015-1908-x.

Güner, H.A.A., Yüksel, Y. and Çevik, E.Ö. (2013) Longshore sediment transport—Field data and estimations using neural networks, numerical model, and empirical models. *Journal of Coastal Research*, 287, 311–324. doi:10.2112/JCOASTRES-D-11-00074.1.

Gurney, K. (2010) Neural networks for perceptual processing: From simulation tools to theories. In: C. Tosh and G. Ruxton (Eds.) *Modelling Perception with Artificial Neural Networks*, chapter 1. Cambridge: Cambridge University Press, pp. 7–34. doi:10.1017/CBO9780511779145.002.

Hagan, M., Demuth, H. and Beale, M. (1996) *Neural network design*. Boston: PWS Publishing Company.

Haines-Young, R. and Petch, J. (1983) Multiple working hypotheses: equifinality and the study of landforms. *Transactions of the Institute of British Geographers New Series*, 8(4), 458–466.

Hamdan, M.A., Refaat, A.A. and Abdel Wahed, M. (2016) Morphologic characteristics and migration rate assessment of barchan dunes in the Southeastern Western Desert of Egypt. *Geomorphology*, 257, 57–74. doi:10.1016/j.geomorph.2015.12.026.

Hao, D., Peng, J., Wang, Y., Liu, J., Zhou, X. and Zheng, D. (2019) Evaluation of convolutional neural network for recognizing uterine contractions with electrohysterogram. *Computers in Biology and Medicine*, 113(August), 103394. doi:10.1016/j.compbiomed.2019.103394.

Haykin, S. (1994) *Neural networks: A comprehensive foundation*. Upper Saddle River: Prentice Hall.

Haykin, S. (2009) *neural networks and learning machines*. 3rd edition. Upper Saddle River: Pearson.

Hayward, R.K., Mullins, K.F., Fenton, L.K., Hare, T.M., Titus, T.N., Bourke, M.C., Colaprete, A. and Christensen, P.R. (2007) Mars global digital dune database and initial science results. *Journal of Geophysical Research E: Planets*, 112(11), 1–17. doi:10.1029/2007JE002943.

He, K., Gkioxari, G., Dollár, P. and Girshick, R. (2020) Mask R-CNN. 42(2), 386–397. doi:10.1109/TPAMI.2018.2844175.

He, K. and Sun, J. (2015) Convolutional neural networks at constrained time cost. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 5353–5360. doi:10.1109/CVPR.2015.7299173.

He, K., Zhang, X., Ren, S. and Sun, J. (2016) Deep residual learning for image recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-Decem, 770–778. doi:10.1109/CVPR.2016.90.

Hebb, D. (2002) *The Organization of Behavior : A Neuropsychological Theory*. New York, NY: Taylor & Francis.

Herrmann, H.J., Sauermann, G. and Schwämmle, V. (2005) The morphology of dunes. *Physica A: Statistical Mechanics and its Applications*, 358, 30–38. doi:10.1016/j.physa.2005.06.004.

Hersen, P. (2004) On the crescentic shape of barchan dunes. *The European Physical Journal B - Condensed Matter*, 37(4), 507–514. doi:10.1140/epjb/e2004-00087-y.

Hersen, P., Andersen, K.H., Elbelrhiti, H., Andreotti, B., Claudin, P. and Douady, S. (2004) Corridors of barchan dunes: Stability and size selection. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 69(1 Pt 1), 011304. doi:10.1103/PhysRevE.69.011304.

Hersen, P., Douady, S. and Andreotti, B. (2002) Relevant length scale of barchan dunes. *Physical Review Letters*, 89(26), 264301. doi:10.1103/PhysRevLett.89.264301.

Hesp, P.A. and Hastings, K. (1998) Width, height and slope relationships and aerodynamic maintenance of barchans. *Geomorphology*, 22(2), 193–204. doi:10.1016/S0169-555X(97)00070-6.

Hesse, R. (2009) Do swarms of migrating barchan dunes record paleoenvironmental changes? - A case study spanning the middle to late Holocene in the Pampa de Jaguay, southern Peru. *Geomorphology*, 104(3-4), 185–190. doi:10.1016/j.geomorph.2008.08.006.

Hjort, J. and Marmion, M. (2008) Effects of sample size on the accuracy of geomorphological models. *Geomorphology*, 102(3-4), 341–350. doi:10.1016/j.geomorph.2008.04.006.

Hobbs, S.W., Paull, D.J. and Bourke, M.C. (2010) Aeolian processes and dune morphology in Gale Crater. *Icarus*, 210(1), 102–115. doi:10.1016/j.icarus.2010.06.006.

Hossin, M. and Sulaiman, M.N. (2015) A Review on Evaluation Metrics for Data Classification Evaluations. *International Journal of Data Mining & Knowledge Management Process*, 5(2), 01–11. doi:10.5121/ijdkp.2015.5201.

Huang, L., Liu, L., Jiang, L. and Zhang, T. (2018) Automatic mapping of thermokarst landforms from remote sensing images using deep learning: A case study in the northeastern Tibetan Plateau. *Remote Sensing*, 10(12). doi:10.3390/rs10122067.

Hugenholtz, C.H. and Barchyn, T.E. (2012) Real barchan dune collisions and ejections. *Geophysical Research Letters*, 39(2), 1–5. doi:10.1029/2011GL050299.

Hugenholtz, C.H., Levin, N., Barchyn, T.E. and Baddock, M.C. (2012) Remote sensing and spatial analysis of aeolian sand dunes: A review and outlook. *Earth-Science Reviews*, 111(3-4), 319–334. doi:10.1016/j.earscirev.2011.11.006.

Igonin, D.M., Kolganov, P.A. and Tiumentsev, Y.V. (2020) Choosing Hyperparameter Values of the Convolution Neural Network When Solving the Problem of Semantic Segmentation of Images Obtained by Remote Sensing of the Earth's Surface. *Optical Memory and Neural Networks (Information Optics)*, 29(4), 317–329. doi:10.3103/S1060992X20040086.

Ilyas, B.R., Mohammed, B., Khaled, M. and Miloud, K. (2019) Enhanced Face Recognition System Based on Deep CNN. *Proceedings - 2019 6th International Conference on Image and Signal Processing and their Applications, ISPA 2019*, 15–20. doi:10.1109/ISPA48434.2019.8966797.

Inkpen, R. (2005) *Science, philosophy and physical geography*. London: Routledge.

Inman, D.L., Ewing, G.C. and Corliss, J.B. (1966) Coastal sand dunes of Guerrero Negro, Baja California, Mexico. *Geological Society of America Bulletin*, 77(8), 787. doi:10.1130/0016-7606(1966)77[787:CSDOGN]2.0.CO;2.

Jain, A., Jianchang Mao and Mohiuddin, K. (1996) Artificial neural networks: a tutorial. *Computer*, 29(3), 31–44. doi:10.1109/2.485891.

Jensen, R.J. (2003) The conundrum of morphometrics. *Taxon*, 52(4), 663–671.

Ji, S., Yu, D., Shen, C., Li, W. and Xu, Q. (2020) Landslide detection from an open satellite imagery and digital elevation model dataset using attention boosted convolutional neural networks. *Landslides*, 17(6), 1337–1352. doi:10.1007/s10346-020-01353-2.

Jiang, J., Zhang, Z., Dong, Q. and Ni, F. (2018) Characterization and identification of asphalt mixtures based on Convolutional Neural Network methods using X-ray scanning images. *Construction and Building Materials*, 174, 72–80. doi:10.1016/j.conbuildmat.2018.04.083.

Jimenez, J.A., Maia, L.P., Serra, J. and Morais, J. (1999) Aeolian dune migration along the Ceará coast, north-eastern Brazil. *Sedimentology*, 46(4), 689–701. doi:10.1046/j.1365-3091.1999.00240.x.

Jinsakul, N., Tsai, C.F., Tsai, C.E. and Wu, P. (2019) Enhancement of deep learning in image classification performance using Xception with the swish activation function for colorectal polyp preliminary screening. *Mathematics*, 7(12). doi:10.3390/MATH7121170.

Jordan, M.I. and Mitchell, T.M. (2015) Machine learning: Trends, perspectives, and prospects. *Science (New York, N.Y.)*, 349(6245), 255–260. doi:10.1126/science.aaa8415.

Kalita, S. and Biswas, M. (2019) Improved Convolutional Neural Networks for Hyperspectral Image Classification. *Advances in Intelligent Systems and Computing*, 740, 397–410. doi:10.1007/978-981-13-1280-9_37.

Kattenborn, T., Leitloff, J., Schiefer, F. and Hinz, S. (2021) Review on Convolutional Neural Networks (CNN) in vegetation remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*, 173(January), 24–49. doi:10.1016/j.isprsjprs.2020.12.010.

Kendall, D.G. (1977) The diffusion of shape. *Advances in Applied Probability*, 9(3), 428 – 430. doi:10.2307/1426091.

Keskar, N.S., Nocedal, J., Tang, P.T.P., Mudigere, D. and Smelyanskiy, M. (2017) On large-batch training for deep learning: Generalization gap and sharp minima. *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*, 1–16.

Khan, H.A., Jue, W., Mushtaq, M. and Mushtaq, M.U. (2020) Brain tumor classification in MRI image using convolutional neural network. *Mathematical Biosciences and Engineering*, 17(5), 6203–6216. doi:10.3934/MBE.2020328.

Klingenberg, C.P. (2010) Evolution and development of shape: Integrating quantitative approaches. *Nature reviews. Genetics*, 11(9), 623–635. doi:10.1038/nrg2829.

Klingenberg, C.P. (2013) Visualizations in geometric morphometrics: How to read and how to make graphs showing shape changes. *Hystrix*, 24(July), 15–24. doi:10.4404/hystrix-24.1-7691.

Klingenberg, C.P. and McIntyre, G.S. (1998) Geometric morphometrics of developmental instability: Analyzing patterns of fluctuating asymmetry with procrustes methods. *Evolution*, 52(5), 1363 – 1375. doi:10.2307/2411306.

Kneusel, D.T. (2022) *Math for deep learning: What you need to know to understand neiral networks*. San Francisco: No Starch Press.

Kocurek, G., Townsley, M. and Yeh, K. (1992) Dune and dune-field development on Padre Island, Texas, with Implications for interdune deposition and water-table-controlled accumulation. *SEPM Journal of Sedimentary Research*, Vol. 62(4), 622–635. doi:10.1306/D4267974-2B26-11D7-8648000102C1865D.

Kubilius, J., Bracci, S. and Op de Beeck, H.P. (2016) Deep Neural Networks as a Computational Model for Human Shape Sensitivity. *PLoS Computational Biology*, 12(4), 1–26. doi:10.1371/journal.pcbi.1004896.

Łabuz, T.A., Grunewald, R., Bobykina, V., Chubarenko, B., Česnulevičius, A., Bautrenas, A., Orkunaite, R. and Tõnisson, H. (2018) Coastal dunes of the Baltic Sea shores: A review. *Quaestiones Geographicae*, 37(1), 47–71. doi:10.2478/quageo-2018-0005.

Lai, D., Tian, W. and Chen, L. (2019) Improving Classification with Semi-Supervised and Fine-grained Learning. *Pattern Recognition*, 88, 547–556. doi:S0031320318304230.

Lakshmanan, S. (2005) Statistical Methods for Image Segmentation. *Handbook of Image and Video Processing*, (January), 443–453. doi:10.1016/B978-012119792-6/50090-5.

Lamm, E. and Unger, R. (2011) *Biological computation*. Boca Raton, Florida: CRC Press.

Lancashire, L.J., Lemetre, C. and Ball, G.R. (2009) An introduction to artificial neural networks in bioinformatics - Application to complex microarray and mass spectrometry datasets in cancer studies. *Briefings in Bioinformatics*, 10(3), 315–329. doi:10.1093/bib/bbp012.

Lancaster, N. (1996) Field studies of sand patch initiation processes on the northern margin of the Namib Sand Sea. *Earth Surface Processes and Landforms*, 21(10), 947–954. doi:10.1002/(SICI)1096-9837(199610)21:10⟨947::AID-ESP634⟩3.0.CO;2-7.

Lancaster, N. (2014) Dune systems of the Namib Desert – a spatial and temporal perspective. *Transactions of the Royal Society of South Africa*, 69(3), 133–137. doi:10.1080/0035919X.2014.924085.

LaVezzi, G., Margapuri, V., Stewart, R. and Wagner, D. (2020) Bombus Species Image Classification. *arXiv*.

Lecun, Y., Bengio, Y. and Hinton, G. (2015) Deep learning. *Nature*, 521(7553), 436–444. doi:10.1038/nature14539.

Lessar, M.P. (1885) The Kara-Kum, or desert of Turkomania. *Proceedings of the Royal Geographical Society and Monthly Record of Geography*, 7(4), 231–238. doi:10.2307/1800466.

Lewin, J. (2015) The lexicon of geomorphology. *Earth Surface Processes and Landforms*, 41(1), 5–15. doi:10.1002/esp.3733.

Li, H., Xu, Z., Taylor, G., Studer, C. and Goldstein, T. (2018) Visualizing the loss landscape of neural nets. *Advances in Neural Information Processing Systems*, 2018-Decem(NeurIPS 2018), 6389–6399.

Li, W. and Hsu, C.Y. (2020) Automated terrain feature identification from remote sensing imagery: a deep learning approach. *International Journal of Geographical Information Science*, 34(4), 637–660. doi:10.1080/13658816.2018.1542697.

Li, Y.F., Guo, L.Z. and Zhou, Z.H. (2019) Towards Safe Weakly Supervised Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(1), 1–1. doi:10.1109/TPAMI.2019.2922396.

Liu, D., Li, J. and Fan, F. (2021) Classification of landslides on the southeastern Tibet Plateau based on transfer learning and limited labelled datasets. *Remote Sensing Letters*, 12(3), 286–295. doi:10.1080/2150704X.2021.1890263.

Livingstone, I., Wiggs, G.F.S. and Weaver, C.M. (2007) Geomorphology of desert sand dunes: A review of recent progress. *Earth-Science Reviews*, 80(3-4), 239–257. doi:10.1016/j.earscirev.2006.09.004.

Lloyd, R., Hodgson, M.E. and Stokes, A. (2002) Visual categorization with aerial photographs. *Annals of the Association of American Geographers*, 92(2), 241–266. doi:10.1111/1467-8306.00289.

Long, J.T. and Sharp, R.P. (1964) Barchan-dune movement in Imperial Valley, California. *Geological Society of America Bulletin*, 75(1), 149–156. doi:10.1130/0016-7606(1964)75.

Lonsdale, P. and Malfait, B. (1974) Abyssal dunes of foraminiferal sand on the Carnegie Ridge. *Bulletin of the Geological Society of America*, 85(11), 1697–1712. doi:10.1130/0016-7606(1974)85⟨1697:ADOFSO⟩2.0.CO;2.

Lorenz, R.D., Gasmi, N., Radebaugh, J., Barnes, J.W. and Ori, G.G. (2013) Dunes on planet Tatooine: Observation of barchan migration at the Star Wars film set in Tunisia. *Geomorphology*, 201, 264–271. doi:10.1016/j.geomorph.2013.06.026.

Lu, L. (2020) Dying ReLU and Initialization: Theory and Numerical Examples. *Communications in Computational Physics*, 28(5), 1671–1706. doi:10.4208/cicp.OA-2020-0165.

Lv, P., Dong, Z., Narteau, C. and Rozier, O. (2016) Morphodynamic mechanisms for the formation of asymmetric barchans: improvement of the Bagnold and Tsoar models. *Environmental Earth Sciences*, 75(3), 259. doi:10.1007/s12665-015-5083-2.

Ma, X., Yan, J. and Fan, F. (2014) Morphology of submarine barchans and sediment transport in barchans fields off the Dongfang coast in Beibu Gulf. *Geomorphology*, 213, 213–224. doi:10.1016/j.geomorph.2014.01.010.

MacDougal, D.T. and Sykes, G. (1907) The desert basins of the Colorado Delta. *Bulletin of the American Geographical Society*, 39(12), 705—-729. doi:10.2307/198572.

MacMillan, R.A. and Shary, P.A. (2009) Landforms and landform elements in geomorphometry. *Developments in Soil Science*, 33(C), 227–254. doi:10.1016/S0166-2481(08)00009-3.

Maggiori, E., Tarabalka, Y., Charpiat, G. and Alliez, P. (2017) Convolutional neural networks for large-scale remote-sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(2), 645–657. doi:10.1109/TGRS.2016.2612821.

Maghsoudi, M., Lorestani, G., Shahriar, A. and Hosainzadeh, A. (2010) Relationships between effective parameters in barchans' shape (Case study : South-west of Maranjab) [predatory journal]. *American-Eurasian Journal of Agriculture & Environmental Science*, 8(3), 245–252.

Mahdianpari, M., Salehi, B., Rezaee, M., Mohammadimanesh, F. and Zhang, Y. (2018) Very deep convolutional neural networks for complex land cover mapping using multispectral remote sensing imagery. *Remote Sensing*, 10(7). doi:10.3390/rs10071119.

Márquez, E.J., Cabeen, R., Woods, R.P. and Houle, D. (2012) The measurement of local variation in shape. *Evolutionary Biology*, 39, 419–439. doi:10.1007/s11692-012-9159-6.

Maxwell, A.E., Pourmohammadi, P. and Poyner, J.D. (2020) Mapping the topographic features of mining-related valley fills using mask R-CNN deep learning and digital elevation data. *Remote Sensing*, 12(3). doi:10.3390/rs12030547.

McCulloch, D. and Janda, R. (1964) Subaqueous river channel barchan dunes. *Journal of Sedimentary Petrology*, 34(3), 694–694. doi:10.1163/157005863X00357.

McMahon, H. (1906) Recent survey and exploration in Seistan. *The Geographical Journal*, 28(3), 209–228. doi:10.2307/1776729.

Meena, S.R., Ghorbanzadeh, O., van Westen, C.J., Nachappa, T.G., Blaschke, T., Singh, R.P. and Sarkar, R. (2021) Rapid mapping of landslides in the Western Ghats (India) triggered by 2018 extreme monsoon rainfall using a deep learning approach. *Landslides*, 18(5), 1937–1950. doi:10.1007/s10346-020-01602-4.

Melton, F.A. (1940) A tentative classification of sand dunes its application to dune history in the southern high plains. *The Journal of Geology*, 48(2), 113–174. doi:10.1086/624871.

Minár, J. and Evans, I.S. (2008) Elementary forms for land surface segmentation: The theoretical basis of terrain analysis and geomorphological mapping. *Geomorphology*, 95(3-4), 236–259. doi:10.1016/j.geomorph.2007.06.003.

Mitteroecker, P. and Gunz, P. (2009) Advances in geometric morphometrics. *Evolutionary Biology*, 36(2), 235–247. doi:10.1007/s11692-009-9055-x.

Mitteroecker, P., Gunz, P., Windhager, S. and Schaefer, K. (2013) A brief review of shape, form, and allometry in geometric morphometrics, with applications to human facial morphology. *Hystrix*, 24(1), 59–66. doi:10.4404/hystrix-24.1-6369.

Mohammed, N.Z., Ghazi, A. and Mustafa, H.E. (2013) Positional accuracy testing of Google Earth. *International Journal of Multidisciplinary Sciences and Engineering*, 4(6), 6–9.

Momiji, H., Nishimori, H. and Bishop, S.R. (2002) On the shape and migration speed of a proto-dune. *Earth Surface Processes and Landforms*, 27(12), 1335–1338. doi:10.1002/esp.410.

Moosavi, V., Moradi, H., Shamsi, S.R.F. and Shirmohammadi, B. (2014) Assessment of the planimetric morphology of barchan dunes. *Catena*, 120, 12–19. doi:10.1016/j.catena.2014.03.017.

Morgan, E.D. (1881) Steppe routes from Karshi to the Amu-Daria. *Proceedings of the Royal Geographical Society and Monthly Record of Geography*, 3(12), 723–731.

Mukti, I.Z. and Biswas, D. (2019) Transfer Learning Based Plant Diseases Detection Using ResNet50. *2019 4th International Conference on Electrical Information and Communication Technology, EICT 2019*, (December), 20–22. doi:10.1109/EICT48899.2019.9068805.

Nagle-Mcnaughton, T., McClanahan, T. and Scuderi, L. (2020) Planet: A neural network for detecting transverse aeolian ridges on mars. *Remote Sensing*, 12(21), 1–15. doi:10.3390/rs12213607.

Nai, Y., Li-lan, S., Lin, W., Jie, Y., Shi-yi, C. and Wei-lu, H. (2016) A method for building 3D models of barchan dunes. *Geomorphology*, 253, 181–188. doi:10.1016/j.geomorph.2015.10.011.

Negnevitsky, M. (2011) *Artificial intelligence: A guide to intelligent systems*. 3rd edition. Harlow: Addison Wesley.

Neupane, B., Horanont, T. and Aryal, J. (2021) Deep learning-based semantic segmentation of urban features in satellite images: A review and meta-analysis. *Remote Sensing*, 13(4), 1–41. doi:10.3390/rs13040808.

Nguyen, A., Yosinski, J. and Clune, J. (2015) Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June. IEEE, pp. 427–436. doi:10.1109/CVPR.2015.7298640.

Niepert, M., Ahmed, M. and Kutzkov, K. (2016) Learning Convolutional Neural Networks for Graphs. 1. doi:10.5465/amj.2011.0336.

Norris, R.M. (1966) Barchan dunes of Imperial Valley, California. *The Journal of Geology*, 74(3), 292–306. doi:10.1086/627164.

Norris, R.M. and Norris, K.S. (1961) Algodones dunes of southern California. *Geological Society of America Bulletin*, 7(April), 605–620.

Olden, J.D. and Jackson, D.A. (2002) Illuminating the "black box": A randomization approach for understanding variable contributions in artificial neural networks. *Ecological Modelling*, 154(1-2), 135–150. doi:10.1016/S0304-3800(02)00064-9.

Oo, S.L.M. and Oo, A.N. (2019) Child Face Recognition with Deep Learning. *2019 International Conference on Advanced Information Technologies, ICAIT 2019*, 155–160. doi:10.1109/AITC.2019.8921152.

Palafox, L.F., Hamilton, C.W., Scheidt, S.P. and Alvarez, A.M. (2017) Automated detection of geological landforms on Mars using Convolutional Neural Networks. *Computers and Geosciences*, 101(January), 48–56. doi:10.1016/j.cageo.2016.12.015.

Paredes-Hernández, C.U., Salinas-Castillo, W.E., Guevara-Cortina, F. and Martínez-Becerra, X. (2013) Horizontal positional accuracy of Google Earth's imagery over rural areas: a study case in Tamaulipas, Mexico. *Boletim de Ciências Geodésicas*, 19(4), 588–601. doi:10.1590/S1982-21702013000400005.

Parteli, E. and Pöschel, T. (2017) Barchan dunes on Pluto? *EPJ Web of Conferences*, 140, 8–11. doi:10.1051/epjconf/201714014010.

Parteli, E.J.R., Durán, O., Bourke, M.C., Tsoar, H., Pöschel, T. and Herrmann, H. (2014) Origins of barchan dune asymmetry: Insights from numerical simulations. *Aeolian Research*, 12, 121–133. doi:10.1016/j.aeolia.2013.12.002.

Parteli, E.J.R., Durán, O. and Herrmann, H.J. (2007) Minimal size of a barchan dune. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 75(1 Pt 1), 011301. doi:10.1103/PhysRevE.75.011301.

Parteli, E.J.R. and Herrmann, H.J. (2007) Dune formation on the present Mars. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 76(4), 1–16. doi:10.1103/PhysRevE.76.041307.

Parteli, E.J.R., Schatz, V. and Herrmann, H.J. (2005) Barchan dunes on Mars and on Earth. In: *Proceedings of Powders and Grains 2005*. Balkema, pp. 959–962.

Pearce, K.I. and Walker, I.J. (2005) Frequency and magnitude biases in the 'Fryberger' model, with implications for characterizing geomorphically effective winds. *Geomorphology*, 68(1-2), 39–55. doi:10.1016/j.geomorph.2004.09.030.

Pelletier, J.D. (2013) Deviations from self-similarity in barchan form and flux: The case of the Salton Sea dunes, California. *Journal of Geophysical Research: Earth Surface*, 118(4), 2406–2420. doi:10.1002/2013JF002867.

Pelletier, J.D., Sherman, D.J., Ellis, J.T., Farrell, E.J., Jackson, N.L., Li, B., Nordstrom, K.F., Maia, L.P. and Omidyeganeh, M. (2015) Dynamics of sediment storage and release on aeolian dune slip faces: A field study in Jericoacoara, Brazil. *Journal of Geophysical Research: Earth Surface*, 120(9), 1911–1934. doi:10.1002/2015JF003636.

Planche, B. and Andres, E. (2019) *Hands-on computer vision with TensorFlow 2: Leverage deep learning to create powerful image processing apps with TensorFlow 2.0 and Keras*. Birmingham: Packt Publishing.

Pradhan, A., Dey, B.K., Nath, R.K. and Sarma, B. (2020) Transfer Learning based Classification of Diabetic Retinopathy Stages. *2020 International Conference on Computational Performance Evaluation, ComPE 2020*, 813–819. doi:10.1109/ComPE49325.2020.9200092.

Pradhan, B. and Lee, S. (2010) Delineation of landslide hazard areas on Penang Island, Malaysia, by using frequency ratio, logistic regression, and artificial neural network models. *Environmental Earth Sciences*, 60(5), 1037–1054. doi:10.1007/s12665-009-0245-8.

Prakash, N., Manconi, A. and Loew, S. (2020) Mapping landslides on EO data: Performance of deep learning models vs. Traditional machine learning models. *Remote Sensing*, 12(3). doi:10.3390/rs12030346.

Prakash, N., Manconi, A. and Loew, S. (2021) A new strategy to map landslides with a generalized convolutional neural network. *Scientific Reports*, 11(1), 1–16. doi:10.1038/s41598-021-89015-8.

Qian, G., Yang, Z., Tian, M., Dong, Z., Liang, A. and Xing, X. (2021) From dome dune to barchan dune: Airflow structure changes measured with particle image velocimetry in a wind tunnel. *Geomorphology*, 382, 107681. doi:10.1016/j.geomorph.2021.107681.

Qiumei, Z., Dan, T. and Fenghua, W. (2019) Improved Convolutional Neural Network Based on Fast Exponentially Linear Unit Activation Function. *IEEE Access*, 7, 151359–151367. doi:10.1109/ACCESS.2019.2948112.

Radebaugh, J., Lorenz, R., Farr, T., Paillou, P., Savage, C. and Spencer, C. (2010) Linear dunes on Titan and Earth: Initial remote sensing comparisons. *Geomorphology*, 121(1-2), 122–132. doi:10.1016/j.geomorph.2009.02.022.

Raghu, S., Sriraam, N., Temel, Y., Rao, S.V. and Kubben, P.L. (2020) EEG based multi-class seizure type classification using convolutional neural network and transfer learning. *Neural Networks*, 124, 202–212. doi:10.1016/j.neunet.2020.01.017.

Raschka, S. and Mirjalili, V. (2019) *Python Machine Learning*. 3rd edition. Birmingham: Packt Publishing.

Ravichandiran, S. (2019) *Hands-on deep learning algorithms with python*. Packt Publishing.

Rempel, P.J. (1936) The crescentic Dunes of the Salton Sea and their relation to the vegetation. *Ecology*, 17(3), 347–358. doi:10.2307/1931839.

Richards, K. (1996) Samples and cases: generalisation and explanation in geomorphology. In: B.L. Rhoads and C.E. Thorn (Eds.) *The Scientific Nature of Geomorphology: Proceedings of the 27th Binghampton Symposium in Geomorphology held 27-29 September 1996*, September 1996. Chichester: John Wiley & Sons Ltd., pp. 171–190.

Richtsmeier, J.T., Burke Deleon, V. and Lele, S.R. (2002) The promise of geometric morphometrics. *American Journal of Physical Anthropology*, 119(S35), 63–91. doi:10.1002/ajpa.10174.

Robson, B.A., Bolch, T., MacDonell, S., Hölbling, D., Rastner, P. and Schaffer, N. (2020) Automated detection of rock glaciers using deep learning and object-based image analysis. *Remote Sensing of Environment*, 250(February). doi:10.1016/j.rse.2020.112033.

Rodvold, D.M., McLeod, D.G., Brandt, J.M., Snow, P.B. and Murphy, G.P. (2001) Introduction to artificial neural networks for physicians: Taking the lid off the black box. *Prostate*, 46(1), 39–44. doi:10.1002/1097-0045(200101)46:1⟨39::AID-PROS1006⟩3.0.CO;2-M.

Rohlf, F.J. (2000) Statistical power comparisons among alternative morphometric methods. *American Journal of Physical Anthropology*, 111(4), 463–478. doi:10.1002/(SICI)1096-8644(200004)111:4⟨463::AID-AJPA3⟩3.0.CO;2-B.

Rojas, R. (1996) *Neural networks: A sytematic introduction*. Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-61068-4.

Ronneberger, O., Fischer, P. and Brox, T. (2015) U-net: Convolutional networks for biomedical image segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9351, 234–241. doi:10.1007/978-3-319-24574-4_28.

Ros, J., Evin, A., Bouby, L. and Ruas, M.P. (2014) Geometric morphometric analysis of grain shape and the identification of two-rowed barley (Hordeum vulgare subsp. distichum L.) in southern France. *Journal of Archaeological Science*, 41, 568–575. doi:10.1016/j.jas.2013.09.015.

Rubanenko, L., Perez-Lopez, S., Schull, J. and Lapotre, M.G. (2021) Automatic Detection and Segmentation of Barchan Dunes on Mars and Earth Using a Convolutional Neural Network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14, 9364–9371. doi:10.1109/JSTARS.2021.3109900.

Russ, J.C. (2006) *The Image Processing Handbook*. 5th edition. Boca Raton: CRC Press. doi:10.1201/9780203881095.

Sagga, A. (1998) Barchan dunes of Wadi Khulays, western region of Saudi Arabia: Geomorphology and sedimentology relationships. *Journal of King Abdulaziz University-Earth Sciences*, 10(1), 105–114. doi:10.4197/Ear.10-1.7.

Sauermann, G., Andrade, J.S., Maia, L.P., Costa, U.M.S., Araújo, A.D. and Herrmann, H.J. (2003) Wind velocity and sand transport on a barchan dune. *Geomorphology*, 54, 245–255. doi:10.1016/S0169-555X(02)00359-8.

Sauermann, G., Rognon, P., Poliakov, A. and Herrmann, H.J. (2000) The shape of the barchan dunes of Southern Morocco. *Geomorphology*, 36(1-2), 47–62. doi:10.1016/S0169-555X(00)00047-7.

Schatz, V., Tsoar, H., Edgett, K.S., Parteli, E.J.R. and Herrmann, H.J. (2006) Evidence for indurated sand dunes in the Martian north polar region. *Journal of Geophysical Research E: Planets*, 111(4). doi:10.1029/2005JE002514.

Scheidt, S.P. and Lancaster, N. (2013) The application of COSI-Corr to determine dune system dynamics in the southern Namib Desert using ASTER data. *Earth Surface Processes and Landforms*, 38(February), 1004–1019. doi:10.1002/esp.3383.

Shaf, A., Ali, T., Farooq, W., Javaid, S., Draz, U. and Yasin, S. (2018) Two Classes Classification Using Different Optimizers in Convolutional Neural Network. *Proceedings of the 21st International Multi Topic Conference, INMIC 2018*. doi:10.1109/INMIC.2018.8595534.

Shakya, A., Biswas, M. and Pal, M. (2021) Parametric study of convolutional neural network based remote sensing image classification. *International Journal of Remote Sensing*, 42(7), 2663–2685. doi:10.1080/01431161.2020.1857877.

Shumack, S., Hesse, P. and Farebrother, W. (2020) Deep learning for dune pattern mapping with the AW3D30 global surface model. *Earth Surface Processes and Landforms*, 45(11), 2417–2431. doi:10.1002/esp.4888.

Silvestro, S., Di Achille, G. and Ori, G.G. (2010) Dune morphology, sand transport pathways and possible source areas in east Thaumasia Region (Mars). *Geomorphology*, 121(1-2), 84–97. doi:10.1016/j.geomorph.2009.07.019.

Simmons, F.S. (1956) A note on Pur-Pur Dune, Virú Valley, Peru. *The Journal of Geology*, 64(5), 517–521.

Simonyan, K. and Zisserman, A. (2015) Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1–14.

Sinha, S., Singh, T.N., Singh, V.K. and Verma, A.K. (2010) Epoch determination for neural network by self-organized map (SOM). *Computational Geosciences*, 14(1), 199–206. doi:10.1007/s10596-009-9143-0.

Slice, D.E. (2007) Geometric morphometrics. *Annual Review of Anthropology*, 36(1), 261–281. doi:10.1146/annurev.anthro.34.081804.120613.

Strauss, R.E. and Bookstein, F.L. (1982) The truss: Body form reconstructions in morphometrics. *Systematic Zoology*, 31(2), 113–135. doi:10.2307/2413032.

Suliman, A. and Zhang, Y. (2015) A Review on Back-Propagation Neural Networks in the Application of Remote Sensing Image Classification. *Journal of Earth Science and Engineering*, 5, 52–65. doi:10.17265/2159-581X/2015.

Sun, Y., Huang, J., Ao, Z., Lao, D. and Xin, Q. (2019) Deep learning approaches for the mapping of tree species diversity in a tropical wetland using airborne LiDAR and high-spatial-resolution remote sensing images. *Forests*, 10(11). doi:10.3390/F10111047.

Tajbakhsh, N., Shin, J.Y., Gurudu, S.R., Hurst, R.T., Kendall, C.B., Gotway, M.B. and Liang, J. (2016) Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning? *IEEE Transactions on Medical Imaging*, 35(5), 1299–1312. doi:10.1109/TMI.2016.2535302.

Thakkar, A. and Lohiya, R. (2021) Analyzing fusion of regularization techniques in the deep learning-based intrusion detection system. *International Journal of Intelligent Systems*, (July), 1–49. doi:10.1002/int.22590.

Tharwat, A. (2018) Classification assessment methods. *Applied Computing and Informatics*. doi:10.1016/j.aci.2018.08.003.

Todd, B.J. (2005) Morphology and composition of submarine barchan dunes on the Scotian Shelf, Canadian Atlantic margin. *Geomorphology*, 67(3-4), 487–500. doi:10.1016/j.geomorph.2004.11.016.

Tombe, R. and Viriri, S. (2020) Effective Processing of Convolutional Neural Networks for Computer Vision: A Tutorial and Survey. *IETE Technical Review (Institution of Electronics and Telecommunication Engineers, India)*, 0(0), 1–14. doi:10.1080/02564602.2020.1823252.

Torrey, L. and Shavlik, J. (2010) Transfer learning. In: E.S. Olivas, J.D.M. Guerrero, M.M. Sober, J.R.M. Benedito and A.J.S. Lopez (Eds.) *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*. Hershey, PA: IGI global, pp. 242–264.

Tran, D.T., Iosifidis, A. and Gabbouj, M. (2018) Improving efficiency in convolutional neural networks with multilinear filters. *Neural Networks*, 105, 328–339. doi:10.1016/j.neunet.2018.05.017.

Traore, B.B., Kamsu-Foguem, B. and Tangara, F. (2018) Deep convolution neural network for image recognition. *Ecological Informatics*, 48(September), 257–268. doi:10.1016/j.ecoinf.2018.10.002.

Tripathi, G.C., Rawat, M. and Rawat, K. (2019) Swish Activation Based Deep Neural Network Predistorter for RF-PA. *IEEE Region 10 Annual International Conference, Proceedings/-TENCON*, 2019-Octob, 1239–1242. doi:10.1109/TENCON.2019.8929500.

Tsoar, H. (1984) The formation of seif dunes from barchans - a discussion. *Zeitschrift Fur Geomorphologie*, 28(JANUARY 1984), 99–103.

Tsoar, H. (2001) Types of aeolian sand dunes. In: N.J. Balmforth and A. Provenzale (Eds.) *Geomorphological Fluid Mechanics*, chapter 17. Berlin: Springer-Verlag, pp. 403–429.

Tsoar, H. and Blumberg, D.G. (2002) Formation of parabolic dunes from barchan and transverse dunes along Israel's Mediterranean coast. *Earth Surface Processes and Landforms*, 27(11), 1147–1161. doi:10.1002/esp.417.

Tsoar, H. and Parteli, E.J. (2016) Bidirectional winds, barchan dune asymmetry and formation of seif dunes from barchans: a discussion. *Environmental Earth Sciences*, 75(18), 1–10. doi: 10.1007/s12665-016-6040-4.

van der Merwe, B.J. (2021) The relationship between barchan size and barchan morphology: a case study from Northern Namibia. *South African Geographical Journal*, 103(1), 119 – 138. doi:10.1080/03736245.2021.1876753.

Vaz, D.A., Sarmento, P.T., Barata, M.T., Fenton, L.K. and Michaels, T.I. (2015) Object-based Dune Analysis: Automated dune mapping and pattern characterization for Ganges Chasma and Gale crater, Mars. *Geomorphology*, 250, 128–139. doi:10.1016/j.geomorph.2015.08.021.

Velasco, L.C.P., Granados, A.R.B., Ortega, J.M.A. and Pagtalunan, K.V.D. (2018) Performance analysis of artificial neural networks training algorithms and transfer functions for medium-term water consumption forecasting. *International Journal of Advanced Computer Science and Applications*, 9(4), 109–116. doi:10.14569/IJACSA.2018.090419.

Viscosi, V. and Cardini, A. (2011) Leaf morphology, taxonomy and geometric morphometrics: a simplified protocol for beginners. *PloS one*, 6(10), e25630. doi:10.1371/journal.pone.0025630.

Von Tschudi, J.J. (1847) *Travels in Peru, during the years 1838-1842 on the coast, in the Sierra, across the Cordilleras and the Andes, into primeval forests.* London: Bradbury and Evans.

Wadhawan, S. (1996) Textural attributes of recent aeolian deposits in different sub-basins of the Thar Desert, India. *Journal of Arid Environments*, 32(1), 59–74. doi:10.1006/jare.1996.0006.

Wan, S., Liang, Y. and Zhang, Y. (2018) Deep convolutional neural networks for diabetic retinopathy detection by image classification. *Computers and Electrical Engineering*, 72, 274–282. doi:10.1016/j.compeleceng.2018.07.042.

Wang, B., Ma, R., Kuang, J. and Zhang, Y. (2020) How Decisions Are Made in Brains: Unpack 'Black Box' of CNN with Ms. Pac-Man Video Game. *IEEE Access*, 8, 142446–142458. doi: 10.1109/ACCESS.2020.3013645.

Wang, C. and Anderson, W. (2018) Large-eddy simulation of turbulent flow over spanwise-offset barchan dunes: Interdune vortex stretching drives asymmetric erosion. *Physical Review E*, 98(3), 1–15. doi:10.1103/PhysRevE.98.033112.

Wang, Y., Fang, Z. and Hong, H. (2019) Comparison of convolutional neural networks for landslide susceptibility mapping in Yanshan County, China. *Science of the Total Environment*, 666, 975–993. doi:10.1016/j.scitotenv.2019.02.263.

Wang, Z.T., Tao, S.C., Xie, Y.W. and Dong, G.H. (2007) Barchans of Minqin: Morphometry. *Geomorphology*, 89(3-4), 405–411. doi:10.1016/j.geomorph.2006.12.014.

Webber, A.J. and Hunda, B.R. (2007) Quantitatively comparing morphological trends to environment in the fossil record (Cincinnatian series; Upper Ordovician). *Evolution*, 61(6), 1455–1465. doi:10.1111/j.1558-5646.2007.00123.x.

Webster, M. and Sheets, H.D. (2010) A practical introduction to landmark-based geometric morphometrics. *Quantitative methods in paleobiology*, 16, 168 –188.

Weng, L., Wang, L., Xia, M., Shen, H., Liu, J. and Xu, Y. (2020) Desert classification based on a multi-scale residual network with an attention mechanism. *Geosciences Journal*. doi: 10.1007/s12303-020-0022-y.

Wiggs, G. (2013) Dune morphology and dynamics. In: J.F. Shroder (Ed.) *Treatise on Geomorphology*, volume 11, chapter 11. Elsevier, pp. 201–218. doi:10.1016/B978-0-12-374739-6.00304-3.

Wiggs, G.F., Livingstone, I. and Warren, A. (1996) The role of streamline curvature in sand dune dynamics: evidence from field and wind tunnel measurements. *Geomorphology*, 17(1-3), 29–46. doi:10.1016/0169-555X(95)00093-K.

Wilhelm, T., Geis, M., Püttschneider, J., Sievernich, T., Weber, T., Wohlfarth, K. and Wöhler, C. (2020) DoMars16k: A diverse dataset for weakly supervised geomorphologic analysis on mars. *Remote Sensing*, 12(23), 1–38. doi:10.3390/rs12233981.

Wippermann, F. and Gross, G. (1986) The wind-induced shaping and migration of an isolated dune: A numerical experiment. *Boundary-Layer Meteorology*, 36, 319–334. doi:10.1007/s13398-014-0173-7.2.

Witharana, C., Bhuiyan, M.A.E. and Liljedahl, A.K. (2020) Big Imagery and High Performance Computing As Resources To Understand Changing Arctic Polygonal Tundra. *ISPRS - Inter-*

national Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLIV-M-2-2(November), 111–116. doi:10.5194/isprs-archives-xliv-m-2-2020-111-2020.

Worman, S.L., Murray, A.B., Littlewood, R., Andreotti, B. and Claudin, P. (2013) Modeling emergent large-scale structures of barchan dune fields. *Geology*, 41, 1059–1062. doi:10.1130/G34482.1.

Yosinski, J., Clune, J., Nguyen, A., Fuchs, T. and Lipson, H. (2015) Understanding Neural Networks Through Deep Visualization. In: *Deep Learning Workshop, 31st International Conference on Machine Learning*. Lille, France.

Zamani, S., Mahmoodabadi, M., Yazdanpanah, N. and Farpoor, M.H. (2019) Meteorological application of wind speed and direction linked to remote sensing images for the modelling of sand drift potential and dune morphology. *Meteorological Applications*, (October 2019), 1–16. doi:10.1002/met.1851.

Zeiler, M.D. and Fergus, R. (2014) Visualizing and Understanding Convolutional Networks. In: *Analytical Chemistry Research*, ICLR. pp. 818–833. doi:10.1007/978-3-319-10590-1_53.

Zelditch, M.L., Swiderski, D.L., Sheets, H.D. and Fink, W.L. (2004) *Geometric Morphometrics for Biologists*. 2nd edition. London: Academic Press.

Zhang, C., Recht, B., Bengio, S., Hardt, M. and Vinyals, O. (2017) Understanding deep learning requires rethinking generalization. *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*, 107–116. doi:10.1145/3446776.

Zhang, D., Yang, X., Rozier, O. and Narteau, C. (2014) Mean sediment residence time in barchan dunes. *Journal of Geophysical Research: Earth Surface*, 119(3), 451–463. doi:10.1002/2013JF002833.

Zhang, Z., Dong, Z., Hu, G. and Parteli, E. (2018) Migration and Morphology of Asymmetric Barchans in the Central Hexi Corridor of Northwest China. *Geosciences*, 8(6), 204. doi:10.3390/geosciences8060204.

Zheng, Q., Yang, M., Tian, X., Wang, X. and Wang, D. (2020) Rethinking the role of activation functions in deep convolutional neural networks for image classification. *Engineering Letters*, 28(1), 80–92.

Zhou, Z.H. (2018) A brief introduction to weakly supervised learning. *National Science Review*, 5(1), 44–53. doi:10.1093/nsr/nwx106.

Zhu, X.X., Tuia, D., Mou, L., Xia, G.S., Zhang, L., Xu, F. and Fraundorfer, F. (2017) Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources. *IEEE Geoscience and Remote Sensing Magazine*, 5(4), 8–36. doi:10.1109/MGRS.2017.2762307.