Contents

[Abstract 3](#_Toc157498655)

[1.Introduction 4](#_Toc157498656)

[1.1 Problem Statement 4](#_Toc157498657)

[1.2 The Context of the Study 4](#_Toc157498658)

[1.3 Aims, Objectives and the Research Questions/Hypothesis 4](#_Toc157498659)

[1.4 Personal Motivation 5](#_Toc157498660)

[2 Literature Review 7](#_Toc157498661)

[2.1 Image Recognition & areas of application 7](#_Toc157498662)

[2.3 Convolutional Neural Networks and Their Components 9](#_Toc157498663)

[2.3.1 CNN Structure 9](#_Toc157498664)

[2.3.2 Advantages and disadvantages of CNN 10](#_Toc157498665)

[2.3.3 Existing CNN model architectures 12](#_Toc157498666)

[2.3.4 Transfer Learning with CNN 13](#_Toc157498667)

[2.4 Animal Image Classification using CNN 14](#_Toc157498668)

[2.5 Past Research 14](#_Toc157498669)

[3.Methodology 19](#_Toc157498670)

[3.1 Research Objective 19](#_Toc157498671)

[3.1.1 Custom Dataset 19](#_Toc157498672)

[3.1.2 CNN Architectures Used 19](#_Toc157498673)

[3.1.3 Different Parameters and Augmentations 20](#_Toc157498674)

[3.1.4 Train CNN models 20](#_Toc157498675)

[3.1.5 Evaluate and Compare the accuracy results 20](#_Toc157498676)

[3.2 Dataset Creation and Pre-processing 22](#_Toc157498677)

[3.3 CNN Models Training 22](#_Toc157498678)

[3.3.1 ResNet50 Architecture 22](#_Toc157498679)

[3.3.2 VGG16 Architecture 23](#_Toc157498680)

[3.3.3 AlexNet Architecture 24](#_Toc157498681)

[3.4 Horse Breed Classification 25](#_Toc157498682)

[3.5 Accuracy Metrics Used 25](#_Toc157498683)

[References 27](#_Toc157498684)

List Of Abbreviations

MCAST Malta College of Arts, Science & Technology

CNN Convolutional Neural Network

GBIF Global Biodiversity Information Facility

PCA Principal Component Analysis

LDA Linear Discriminant Analysis

LBPH Linear Binary Patterns Histograms

ReLu Rectifier Linear Unit

TP True Positive

TN True Negative

FP False Positive

FN False Negative

ACC Accuracy

PPV Precision

TPR Recall

DARPA Defense Advanced Research Projects Agency

SIFT Scale-Invariant Feature Transform

SURF Speeded Up Robust Features

HOG Histogram of Oriented Gradients

# Abstract

TO BE EDITED AT THE END OF THE RESEARCH

Computer Vision algorithms are always on the increase with people trying to create better algorithms and fine-tuning to create models which are more precise. Lately, it can be seen from recent studies that the interest is growing in researchers to create models which are able to identify between one horse breed and another. The study being proposed offers a solution to identify the horse breeds from a mixture of different images given to the model developed. Solutions to identify horse breeds were previously implemented with the aim of finding a final solution to this issue, however, up to this date there is still a gap in such an area. My idea to this problem suggests a procedure which is split in three phases to detect and identify the horse breed by making use of only images which were found on the net. In this study I am aiming, to answer 3 questions which arise when implementing a similar model, what kind of dataset is required when undergoing such a study, how can a model be developed to accurately detect horses and their breeds and lastly what the performance of the model is when compared to other studies making use of computer vision techniques. This technique uses Transfer Learning to detect horses. In Phase 1 of the study images are downloaded to create the dataset, later in the second step of phase 1 the data is augmented and split into train and validation. In addition, in phase 2 of the study the model is trained, and the prediction of the horse breeds given in the inferencing folder are printed. In phase 3 of the study Evaluation takes place and the data outputted from the model is analysed, furthermore, in this stage important metrices such as accuracy, precision, recall and F1 Score are calculated, and compared to similar studies using computer vision. It was determined that the system created is able to detect and identify horses, and the results were similar to other procedures which makes use of Computer Vision Technologies.

Index Terms—MCAST, IICT, Computer Vision, Image Classification, Horses, Paper

# 1.Introduction

## 1.1 Problem Statement

Image recognition is a subfield of Computer Vision which has been a popular area of research for a number of decades already with various applications, such as augmented reality, driver assistance and self-driven cars, medical diagnostics and health monitoring, security surveillance, etc. Image recognition provides various benefits, namely efficiency and real time assistance in decision making.

Computer vision technologies have lately been evolving with promising results in multiple areas [1]. Animal Image Recognition is relevant area of research which can be beneficial for various reasons, it can contribute to the conservation of wildlife by population monitoring and detecting illegal activity such as poaching or habitat destruction. Furthermore, it might help in Ecological research including behavioural studies making it easier for ecologists to undertake such research about biodiversity. Image Recognition such algorithms might come in handy during search and rescue operations which helps civil protection units worldwide identify missing pets or wildlife during natural disasters.

This thesis will be addressing horse breed identification, which usually requires the services of an expert in the area and is quite challenging for people who are not familiar with equestrian sport or giving a home to some horses. Therefore, being able to distinguish and compare between different horse breeds makes life easier for enthusiasts who are interested in such area. Horse breed image classification poses some challenges one of which is visual similarity between breeds, which potentially makes it difficult for computer vision algorithms to identify them correctly.

## 1.2 The Context of the Study

The goal of this research is to provide an effective system that uses machine learning to automatically identify horse breeds. This would make it easier for people who were not born into this activity to recognize or confirm their opinion on the horse breed faster simply by providing an image in the model and getting a clear breed suggestion. As mentioned, previous, this kind of system can be utilized not just in horse breed identification, but also in cow recognition to help determine which cattle is which from a facial image.

## 1.3 Aims, Objectives and the Research Questions/Hypothesis

The hypothesis for this study is that “It is possible to correctly identify horse breeds using computer vision models, such as Convolutional Neural Networks by training a model on images of distinct horse breeds and anticipating the photos provided to the model”.

Following the hypothesis, the following research questions were offered:

1. Which Deep Learning model is the most accurate and how can a model be fine-tuned to accurately detect horse and their breeds?
2. How does the model perform in comparison to previous research that employ computer vision techniques?
3. What type of dataset is needed to improve the accuracy further?

## 1.4 Personal Motivation

Computer Vision Image processing field has been my personal educational interest for a number of years. At the same time, active participation in various fields of equestrian sports over the past years has resulted in personal inspiration for the implementation of computer vision algorithms to assist in such areas. People practising in such sports may be struggling to identify different horse breeds which are used for different purposes in different sports. Not so many models were proposed and implemented for the horse breed recognition specifically. Therefore, this research could provide various benefits as this can help makes equestrian enthusiasts communicate with each other and explain themselves more comfortably.

The composition of this study is based on different chapters. In the below structure, chapter 2 will delve into the history and characteristics of Image Classification, CNN architectures and past research in the animal recognition systems. Further, the methodology chapter 3 will provide details related to the creation and pre-processing of the dataset and its description together with the models to be used, the approach for results evaluation and the accuracy metrics to be used. This section will give a full rationale of the selected research approach as well as the limits of such methods. The 4th chapter will focus on data gathered and results presentation using graphs and metrics mentioned in chapter 3. Finally, Chapter 5 will discuss the extracted results, study limitations, and recommendations for further research.

## 

# 2 Literature Review

Artificial Intelligence (AI) is a very broad field which beginning dates to the 1940s with the achievements of the first electronic computers. AI refers to development of intelligent systems which have the ability to perform tasks that normally require human, therefore mimicking human intelligence.

A long-time pioneer in research of AI, DARPA managed to group many different approaches into two main groups which are Handcrafted Knowledge and Machine Learning algorithms. Handcrafted Knowledge systems are as old as the first electronic computers in the form of “if given x input, provide y output”, at the time such programs seemed as smart.

Machine learning algorithms allow machines to learn from existing data and further refine the accuracy and performance, with the aim of performing tasks such as computer vision classification, pattern recognition, decision making. This has gained attention during the last decades due to the advances obtained in deep learning. During these decades various approaches to predict classes of animal images were tested. Some of these approaches made use of colour descriptors. Furthermore, the Bag-Of-Visual-Words (BOW) has been applied to different fields in a quest to improve recognition performance. BOW involves the extraction of features and keeping them in a codebook using an algorithm such as K-means clustering, local constrained linear coding for pooling clusters and spectral clustering [2].

## 2.1 Image Recognition & areas of application

Computer Vision (CV) is a field of Artificial Intelligence which gives computers and other systems opportunity to extract information from digital images and videos and be able to give intelligent and valuable suggestion to the user depending on the data extracted. CV is commonly used for object recognition, image classification, facial recognition and video tracking in different sectors such as diagnostics, self-driven cars, etc. In order for Computer Vision to work it needs to be given a big amount of data on which it can run analysis of the given data over and over until it points out distinctions between the data given and ultimately it is able to recognize images.

Up till few decades ago humans had an advantage since they had a lifetime of gathering visual information, its analysis and learning to see if objects are moving or static, or how far they are, etc. However, at present, Computer Vision algorithms can train systems to perform such tasks. Moreover, these systems can be trained on much more complex patterns in a shorter amount of time since they use cameras, vast datasets, and variety of powerful algorithms. Such systems are capable of performing thousands of processes a minute which can easily achieve greater performances than humans.

Thus, various computer vision algorithms have been used and suggested by various researchers some of them will be mentioned below. Scale-Invariant Feature Transform algorithm was described in a paper published by David Lowe in 1999. This algorithm was used to identify and match local features such as blobs and corners in images [3]. Another algorithm used in past years was Speeded Up Robust Features which is used for feature detection in images.

Furthermore in 2001 Paul Viola and Michael Jones created the Viola-Jones algorithm which was specifically used for face detection. This specific algorithm uses the technique “integral image” which matches the features of typical human faces. In addition, it uses “cascading classifiers” which are a group of features, which makes predictions weather a human face is present in the given image. This algorithm is widely used in security systems and photo tagging [4].

In addition, to the previously mentioned another algorithm is Histogram of Oriented Gradients (HOG) which is especially used for object detection by encoding edge directions within an image is worth mentioning. In HOG an image is divided into small cells 8x8 pixels and then a histogram of gradient orientations is computed for each cell. Later on, in the process the histograms for each cell are concatenated to create a vector for the entire image [5].

One of the more recent and successful developments in the area of image recognition were technologies, such as deep learning and Convolutional Neural Networks. “Convolutional Neural Network (CNN) is a class of deep neural networks, most commonly applied to analyse visual data. CNNs are specialized in processing structured grid data, such as images, and have proven highly effective in various computer vision tasks.” Deep learning represents a subset of Machine learning algorithms that use multilayer neural networks. Convolutional Neural Networks have developed within the deep learning area as the most successful and accurate tool for a variety of Image Recognition tasks. Training the neural network involves an iterative process of optimising or updating its weights in such a way that the network learn to classify or perform specific task more accurately. The end goal of a trained neural network is to be able to make correct predictions on new, unseen data.

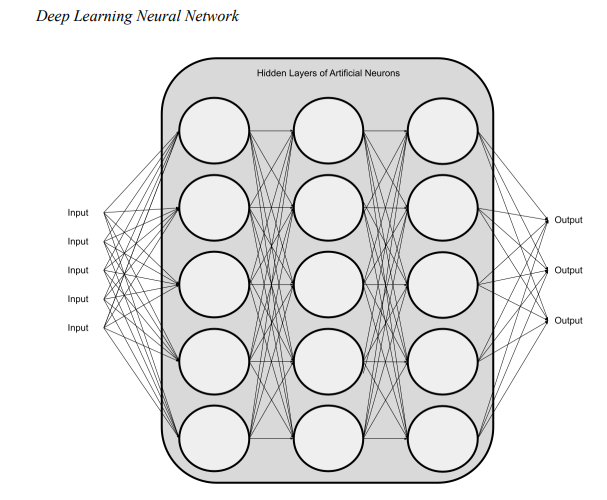


Figure 2.1 Neural Network

Image from: [Assessing Convolutional Neural Network Animal Classification Models for Practical Applications in Wildlife Conservation (sjsu.edu)](https://scholarworks.sjsu.edu/cgi/viewcontent.cgi?article=8731&context=etd_theses)

The steps to train a neural network involve the following:

* Initialization where the network is initialized with random weights.
* Followed by forward propagation were input data is fed through the network, where each layer performing a calculation and passes the result to the next layer.
* Loss Calculation where a loss is calculated meaning how far off the network’s prediction is from the actual target.
* Calculation of Gradient of the loss with respect to weighted.
* Updating Weights where the network weights are adjusted so that loss is reduced. This process is repeated multiple times (epochs)
* Validation is performed where the performance of the network is evaluated on validation dataset to ensure that the network is not overfitting.
* Last step involved is testing of the network.

## 2.3 Convolutional Neural Networks and Their Components

### 2.3.1 CNN Structure

CNN is a neural network designed for processing structured grid data, such as images. It is very powerful in computer vision tasks such as image recognition, image segmentation and object detection. CNN makes use of a specialized architecture that includes 3 different types of layers to learn hierarchical patterns:

A diagram of a process

Description automatically generated

Figure 2.2: CNN structure and layers [6].

1. **Convolutional Layers** - are the fundamental building blocks of the CNNs. These are responsible for handling convolution operations on the input data such as applying filters or kernels to detect local patterns and features.
2. **Pooling Layers** - are used for down sampling the spatial dimensions of the input volume which leads to a reduced amount of computation in the network. Common methods are max pooling or average pooling, taking the maximum or average value, respectively, from a group of neighbouring pixels.
3. **Fully Connected Layers** - they handle the connection of a neuron in a layer to the next layer, allowing the network to learn global patterns and relationships. Such a layers are used towards the end of the network to make predictions based on the learned features.

### 2.3.2 Advantages and disadvantages of CNN

Convolutional Neural Networks have various advantages, particularly in the field of computer vision, but they also have several drawbacks.

Advantages:

1. **Hierarchical Feature Learning** - They can learn hierarchical representations of features from the input data. Convolutional layers capture low-level features and as you move deeper into the network, higher-level features and abstractions are learned.
2. **Translation Invariance** - CNNs offer translation invariance due to the use of convolutional and pooling layers. This implies that the network can detect patterns regardless of where they are in the input space, making it ideal for applications such as picture recognition.
3. **Parameter Sharing** - CNNs reduce the total number of parameters in the network by using shared weights through convolutional filters. This results in a more effective use of memory and makes learning translation-invariant characteristics easier.
4. **Spatial Hierarchy** - Data’s spatial hierarchies are naturally captured by CNNs. Local patterns are captured by convolutional layers, and significant information is retained while spatial dimensions are reduced by pooling layers.
5. **State-of-the-art Performance in Image Recognition** - In a variety of computer vision tasks, CNNs have proven to function at the cutting edge, particularly in image classification. They have proven their efficacy in large-scale picture identification through their performance in contests like ImageNet.

Disadvantages:

1. **Computational Cost** - The process of training deep CNNs can be time-consuming and computationally costly, particularly when there are many parameters involved. As the need for bigger, more intricate models grows, this problem gets worse.
2. **Need for Large Datasets** - CNNs most of the time require large amounts of labelled data, obtaining and preparing such datasets can be challenging.
3. **Overfitting** - When dealing with limited data overfitting might occur, as a result techniques such as dropout and regularization are often employed to mitigate this issue.
4. **Interpretability** - It can be difficult to comprehend how particular judgements are made by CNNs. It is challenging to understand how the network makes decisions because of the intricate, hierarchical structure of the learned features.
5. **Transferability** - Although CNNs can be pre-trained on vast datasets for general tasks, it might not always be easy to apply this knowledge to particular areas. It can be required to fine-tune on smaller, domain-specific datasets.  
     
   <https://tejasmohanayyar.github.io/animals-classification-cnn>

CNNs can be further fine-tuned in order to better adapt it to a new task. This involves taking a pre-trained CNN architecture which has been trained on large datasets such as ImageNet. Furthermore, the top layers need to be removed as they are specific to the original task and needs to be changed to the new task at hand. Further fine tuning can be obtained by modifying the hyper parameters such as learning rate, dropout or regularization strength.

### 2.3.3 Existing CNN model architectures

Various CNN model architectures have been suggested and used over the years. Each architecture has its strengths and weaknesses, and the choice depends on the specific requirements of the task and the available resources. Some of them will be discussed further, mentioning their structure, advantages and disadvantages.

The original ResNet-34, which had 34 weighted layers, made a breakthrough in CNN architecture by overcoming the vanishing gradient problem using shortcut connections. These connections allow the network to bypass specific levels, converting a conventional network into a residual network. ResNet employs fewer filters, resulting in a less complex structure. ResNet adheres to two design principles: keeping the same number of filters in each layer based on the output feature map size and doubling the number of filters when the number of filters when the feature map size is halved to maintain temporal complexity per layer [7].

The VGG network, specifically VGG-16, uses relatively small 3x3 convolutional filters in its design, which consists of 13 convolutional layers and 3 fully connected layers. Designed for 224 x 224 pixel input pictures, the convolutional layers use 3x3 receptive fields and 11 filters, followed by Rectified Linear Unit activation for fast training. The fixed 1-pixel stride ensures spatial resolution. Hidden layers consistently employ ReLU instead of Local Response Normalization to save memory and training time without improving overall accuracy. VGG-16 has three fully connected layers, two with 4096 channels each and one with 1000 channels, which represent ImageNet competition classes [8].

The LeNet Architecture is divided into three layers: input, hidden, and output. The hidden layers include two pooling layers and three convolutional layers. The output layer, also known as the fully connected layer, contains neurons that represent the many animal types being studied. Figure 3 shows a block schematic of the LeNet architecture. The test iterations for RUG-Goat, Snake, and Wild-Anim datasets are set at 78, 41, and 80. The advantage of LeNet architecture is that it is relatively simple architecture, making it easy to understand and implement. On the other hand, since this architecture is simple it might present limitations to more complex problems [9].

A diagram of a pool

Description automatically generated

Figure 2.3. Block diagram illustration of the LeNet architecture

MobileNet uses depthwise separable convolutions, with two fundamental layers: depthwise and pointwise. Depthwise convolution is the process of filtering data without creating additional features. Pointwise convolution, which generates additional features, was merged. The combination of two layers was named depthwise separable convolution. The model applied depthwise convolutions to each input channel, followed by 1x1 convolution (pointwise) to construct a linear combination of the output from the depthwise layer. Following each convolution, we applied Batch Normalization (BN) and Rectified Linear Unit (ReLU). Figure 1 depicts the steps of depthwise and pointwise convolution [10].

### 2.3.4 Transfer Learning with CNN

Transfer learning is the process of using a previously trained model as a starting point for a new task or domain. The goal is to take the information gained by the pre-trained model on a large dataset. This allows us to profit from the broad features and patterns learned by the pre-trained model while conserving time and processing resources [11].

Transfer learning works with a number of sequential layers in the model:

* First layer is known as the **input layer** which accepts the input image data, this layer’s dimensions correspond to the size of the input images.
* After the input layer comes the **pre-training layer**, which is a layer sub divided into other pooling layers of the pre-trained model which is often trained on a large dataset. Such layers have learned a variety of image recognition tasks which are ideal for a variety of image recognition tasks.
* These layers are followed by **dense layers** which are additional convolution in order to further fine tune the base model.
* In addition, a **flattening layer** exists in order to convert the multi-dimensional data into a vector format that can be fed into the subsequent fully connected layers.
* A **fully connected layer** is used which makes the predictions based on all the learned features during the training.
* The last layer in Transfer Learning is the **Output layer**. For multi-class classification tasks, the output layer typically utilizes a softmax activation function, crating probability distributions over the classes.

A diagram of layers of paper

Description automatically generated

Figure 2.4 Transfer Learning Layers

## 2.4 Animal Image Classification using CNN

Image Classification was used in 2017 by Atabay which used a custom dataset in Transfer Learning architectures. Which was followed by the study performed by Wang, Qin, Hou and Gong on a dataset of cattle in order for the model to be able to identify if a cow is the same in different images or if it is different. In 2018 a study by Shahram Taheri and Onsen Toygar uses multiple architectures on a given animal dataset. These researchers used VGG16 and AlexNet architectures in order to perform image classification. In this study AlexNet obtained 89.91% accuracy , VGG16 obtained 92.84% accuracy, however, when fine-tuned AlexNet obtained 91.06% accuracy and VGG16 obtained 94.39% accuracy [11]. Furthermore, animal image classification was used on marine species in a study conducted in 2019.

In a study conducted by Atabay (2017), the author made use of a custom-made dataset since there was a lack of ready-made datasets about horse breeds. Moreover, the horse breeds used were very well known and can be easily distinguished by the human eye. The six horse breeds used to create the dataset were: the Akhal-Teke, American Paint, Belgian, Fjord, Shetland Pony, and Gypsy.

For his research Atabay made use of a CNN architecture that has been pre-trained on the ImageNet dataset. Furthermore, Atabay made use of the Keras framework based on Tensorflow to do the implementations. Moreover, he made use of 5 different CNN’s which were: VGG16, VGG19, InceptionV3, ResNet50 and Xception. The results obtained by Atabay, including the average classification accuracy, number of epochs used in training and the time to reach 99.5% accuracy on training are shown in Table 2.1 below.

A table with numbers and letters

Description automatically generated

Table 2.1: CNN’s test results, 2017 Atabay study

The results shown in Table 2.1 display the superiority of ResNet50 in terms of classification accuracy. However, it takes a very long time to reach the 99.5% accuracy on the training set. On the other hand, VGG models are also considerable considering their speed and accuracy. They can reach above the 99.5% accuracy in less than 2 days and still the test results are quite good. Atabay points out that the results obtained in this study can be used for future studies of Image Classification. Moreover, he says that better tools can be provided for the purpose of horse breed image classification. (Cite: Atabay 2017 study)

In a separate study by Hansen et al (2019) a dataset of 63,364 specimen images from 291 species was used. After that for each specimen the images were split into 3 groups are the training (50%), validation (20%) and testing group (30%). This study like the one by Atabay made use of the TensorFlow scripts for the training of an InceptionV3 model. This study did not augment any of the images that are fed into the model, however, optimisation of the model was noticed when training the model with learning rates of 0.5, 0.3, 0.1, 0.45, 0.01, 0.001 and 0.0001. During this study the following were calculated from all the test images: True positives (TP), False positives (FP), True negative (TN) and False negative (FN) which were later used to evaluate classification precision, classification recall, classification accuracy, true positive rate, true negative rate, and balanced classification accuracy.

Results were extracted from the model performance in which the model managed to correctly predict 9,949 (51.9%) from 19,164 images to the correct species. However, when extracting the genus names of predictions 74.9% were predicted to the correct genus which can be seen in Table 2.2.

A screenshot of a graph

Description automatically generated

Table 2.2: Test Results of 2019 Hansen study

The confusion Matrix shows that it is common for species to be confused with others in the same genus. It was noted that convolutional neural networks can provide ecologists, conservationists, and museum curators the opportunity to correctly predict the consequences of environmental changes for different organisms.

In addition, in 2018 in a study by Horn et al a dataset was constructed using the Global Biodiversity Information Facility (GBIF) archive from which 13 super classes were selected after setting a criterion that every taxa must have at least 20 observations from 20 different users. Furthermore, this required for these taxa to be split into train, validation, and test splits, from which 40% of the observers were to be in the test split and remaining 60% to be used in training and validation splits. As a result, the final image splits contained 579,184 training images, 182 707 test images and 95 986 validation images.

Similar to the studies of Atabay (2017) and Hansen (2019) this study used state of the art deep network architectures such as ResNets, InceptionV3, Inception ResNet V2 and MobileNet. Random cropping with aspect ratio augmentation was performed during training. Training batches of size 32 were formed by consistently sampling from all available training photos of insects rather than uniformly sampling from classes. All networks were fine-tuned using ImageNet pretrained weights with a learning rate of 0.0045, decaying exponentially by 0.94 per 4 epochs, using RMSProp optimization with momentum and decay both set to 0.9. The picture size used for training and testing was 299 x 299, with a single centred crop during testing. In Table 2.3 on can see that Inception ResNet V2 performs better than Inception V3. Furthermore, when Squeeze-and-Excitation are used it further improves the performance. This technique consists of two operations which are squeeze which involves global information gathering by reducing the spatial dimensions of the input feature maps to single channel representation. In addition, excitation aims to capture channel wise dependencies and relationships, this involves two fully connected layers acting as an attention mechanism [12].

A table with numbers and symbols

Description automatically generated

Table 2.3: Classification Results for different CNNs trained, 2018 Horn study

This study outlines that Inception ResNet V2 is the best performing model since it achieves a crop-top 1 accuracy of 80.2% and a crop-top 5 of 95.21%. The results obtained by this model Inception ResNet V2 can be seen in table 2.4 showing that it still struggles with some classes one of which is Reptilia.

A table with numbers and text

Description automatically generated

Table 2.4: Super-class accuracy level for the best performing model Inception ResNet V2.

Lastly this study suggest that future research is still needed to improve on this. Further annotations such as sex and life stage attributes, habitat tags and pixel level labels for the super classes that were difficult to annotate should be investigated [12].

In a study by Trnovsky et al. uses Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Linear Binary Patterns Histograms (LBPH) and a CNN were implemented in MATLAB and Python programming language. In this study a dataset with 5 main classes was created holding 100 images for each of the classes (wolf, fox, bear, hog, and deer) having 500 images in total. In addition, some of the images has variation in scale. The proposed CNN consisted of 8 stages first, all the input images were resized to 32 x 32 pixels. Later, a 2D CNN layer and a Rectifier linear unit (ReLU) were used. In the third layer a 2 x 2 Kernel was used with a dropped probability to 0.25 to reduce overfitting. In the fourth stage a 2D CNN with 32 feature maps was used. In the fifth stage the researchers made use of MaxPooling layer, then dropout function was set to 0.25 and as the last layer a dense layer with 5 classes and SoftMax activation was used.

A diagram of a cat

Description automatically generated

Figure 2.3: Layers of the proposed CNN.

This study results about recognition accuracy can be seen in Table 2.5. This shows that the proposed CNN by the researchers exceeded all other models tested. The obtained results are divided into 6 main parts. Each part was tested with a ratio (test: training), part A(10:90), B(20:80), C(30:70), D(40:60), E(50:50) and F(60:40). As a result, the best recognition accuracy of 98% was achieved when the researchers made use of part A using 10% test data and 90% training data.

A table with numbers and letters

Description automatically generated

Table 2.5: Recognition results obtained using proposed CNN

Furthermore, a confusion matrix (Table 2.6) was extracted during the study showing how each column was predicted when using the proposed CNN, the green diagonal in the confusion matrix represents how many times the model correctly predicted that specific animal. After taking into consideration the correct predictions of the model again the number of images used in the dataset, shows that the model has a recognition accuracy of 94.2%.

A table with numbers and a number in green and purple

Description automatically generated

Table 2.6: Confusion matrix

For future works the researchers plan to test more complex algorithm and testing the proposed CNN against other algorithms such as deep learning. Furthermore, the plan is to make use of bigger datasets. In addition, the researchers suggest that this method be used on other animal databases.

Wang, Qin, Hou and Gong employed Transfer Learning in their work, where they used a phone camera to capture 36 cattle data using video to construct a dataset. This study, like Atabay’s, employs Transfer Learning to compensate for a paucity of data required by the neural network. In this study, it was discovered that VGGFace has the highest performance when fine-tuned, 93 percent accuracy and 74 positive recalls, however when using VGGFace without fine-tuning, the positive recall is 50 percent. According to both research, Transfer Learning is one of the most significant methods to utilize in picture classification in order to leverage the expertise gained in prior trials to complete a given job in less time and with more accuracy.

After taking into consideration several methods to perform image classification, it has been decided to test and compare the performance of ResNet50, VGG16 and AlexNet architectures with our custom dataset as these appear to be the best options for image recognition on animal datasets.

# 3.Methodology

This chapter provides a detailed description of all the research methodology that will be implemented for the extraction and analysis of horse breed visual data.

## 3.1 Research Objective

The aim of this study is to create a model which is capable of classifying images of different horse breeds. The system is expected to accurately classify each image within the custom- made test set.

The approach to the study will contain several stages:

1. Construct a custom dataset consisting of images of 5 different horse breeds.
2. Test various deep learning architectures to be used in determining horse species-specific recognition.
3. Experiment with different parameters and augmentation of the images in order to improve accuracy.
4. Train the CNN models with the best set of parameters.
5. Evaluate and compare the accuracy results of best models and parameters.

The pipeline of the study in Figure 3.1 specifies various stages of our methodology, which outline how such objectives are to be attended.

### 3.1.1 Train CNN models

After choosing the best set of parameters for each model, each model is trained in order to extract the train-loss line graph to check if overfitting is occurring. In addition, the confusion matrix is extracted right after the training. Furthermore, the models trained are saved so that after the training is done there won’t be the need to train the models again, one can just uses them.

### 3.1.2 Evaluate and Compare the accuracy results

After working the accuracy metrics for all the models than each metric is evaluated against each other to determine which model is the best in performing the task at hand with the given dataset. The metrics to be evaluated are Accuracy, Precision, Recall and F1 Score. Then it will be determined which model is best for the classification by checking the F1 Score which indicates the overall performance of the model, meaning that the model can effectively identify positive cases while minimizing false positives and false negatives.

A screenshot of a computer

Description automatically generated

Figure 3.1: Research Pipeline

## 3.2 Dataset Creation and Pre-processing

The dataset used for this study consisted of images which were found from the internet. Images were taken on different backgrounds from different angles with horses facing in various directions.

In order to create the custom dataset needed various images were obtained through the internet search, each showing one of the 5 different horse breeds selected:

* American Bashkir Curly,
* American Paint,
* Friesian,
* Shetland Pony,
* Belgian Draught.

A dataset was constructed for the implementation part of this study consists of 500 images in total, 100 for each breed. The images selected had to be taken from different angles and with different resolutions, ideally avoiding those in which it is difficult to distinguish between one horse breed and the other.

Furthermore, images were augmented before the training the model. The images were rescaled, rotated at a range of 40, width\_shift, height\_shift, zoomed, changed the brightness between the range of 0.8 and 1.2, lastly, they are horizontally flipped.

The images acquired were placed in a folder to be used by the detector during the prediction process during the research. A directory structure named ’data’ was created to be used by the ResNet50 model. Another internal folder ’Horse-photos’ was created containing another 5 sub-folders one for each breed being investigated in this study, one named ’AmericanBashkirCurly’, another ’AmericanPaintHorse’, the third folder named ‘BelgianDraught’, the fourth folder named ‘Friesian’ and lastly ‘ShetlandPony’.

In addition, each internal folder holds a hundred images each which later 70 percent were supposed to be used for training set and the other 30 percent for validation set.

## 3.3 CNN Models Training

For this study 3 different CNN architectures were tested with our custom dataset with various parameters to see if a model can accurately predict the horse breed. Namely ReseNet50, VGG16, AlexNet architectures. Later on in the study the architectures were evaluated and compared against each other to select which of the 3 architectures performs best for the task at hand.

### 3.3.1 ResNet50 Architecture

For the system being developed during this study the ResNet50 model was used as the “base model” to train the system to identify between different horse breeds accurately. The architecture of such a model can be seen in Figure 3.2 [6]. ResNet50 consists of 50 layers.

As can be seen in Figure 3.2 the model consists of different groups which are identified by different colours. Furthermore, the curve lines show in the figure represents identity blocks which are used to identify that previous layer were used in the following layers [6].

The model starts with 64 filters with a kernel size of 7x7 which is then followed by maxpooling layers of size 3x3. As can be seen with the grey colour the first three layers are identical followed by 3 identical groups. After all the blocks are handled there will be 38 connected layers which will be responsible to handle the classification task which we will work on.

A diagram of a computer

Description automatically generated with medium confidence

Figure 3.2: ResNet50 Architecture

### 3.3.2 VGG16 Architecture

Second architecture that was implemented for the aim of this study is VGG16 seen in Figure 3.3 which was previously used in other studies of Image recognition. The VGG16 model consists of 16 layers. This architecture consists of two divisions which are the feature extractor which consists of VGG blocks and the classifier consist of FC and output layer. The input of such architecture receives a 224 x 224 image. In the convolution layers VGG makes use of the smallest possible receptive data of 3 x 3.

This layer is followed by ReLU activation which is a function that provides a matching output for positive inputs and outputs zero for negative inputs. Furthermore, a convolution stride of 1 pixel was set to preserve the spatial resolution after convolution. In addition, a pooling layer follows the convolutional layers in order to reduce the dimensionality and the number of parameters of the feature maps created. Lastly before activating softmax layer comes 3 fully connected layers the first two include 4096 channels and the third layer has 1000 channels [13].

A diagram of a box with numbers and a line of cubes

Description automatically generated with medium confidence

Figure 3.3: VGG16 architecture [13]

### 3.3.3 AlexNet Architecture

In addition, to the previous two architectures AlexNet seen in Figure 3.4 was tested as well with my dataset in order to obtain the results. AlexNet is an architecture that consists of 5 convolution layers, 3 max-pooling layers, 2 Normalized layers, 2 fully connected layers and 1 SoftMax layer. Each convolution layer consists of a convolution filter and a non-linear activation function called “ReLU”. Such layers are followed by pooling layers which are used to perform the max-pooling function. This function is in charge of accumulating features from maps generated by convolving a filter over an image [14].

The input image to such architecture is 227 x 227 x 3.

A diagram of a diagram of a number

Description automatically generated with medium confidence

Figure 3.4: AlexNet architecture

## 3.4 Finetuning Parameters and Image Augmentation

When it comes to augmentation rescaling was performed on the images in order to normalize the pixel values. Rotation of 40 was performed followed by width shift and height shift of 0.2. The images were augmented further by changing shear range and the zoom. Further on, brightness was manipulated from 0.8 to 1.2. Before starting the model training horizontal flip was set to true.

## 3.5 Horse Breed Classification

The ResNet50 technique was used to recognize and categorize the horses in the photos using the model created for this investigation. The model employed brand fresh photos that the model had never seen before and were placed in the inferencing folder. After this procedure is performed successfully, the predictions are displayed on the screen, indicating how the model categorized each image by displaying the name of the class anticipated.

A black and white screen with numbers

Description automatically generated

Figure 3.5 Predictions done by the model

The VGG16 technique was used on my dataset as well as in order to be able to compare the results obtained from the first model with this one using the same performance metrics that were used on the ResNet50 model.

## 3.6 Accuracy Metrics Used

Upon obtaining the results of classification and comparing them with the real class, the confusion matrix will be created. A confusion matrix uses four basic characteristics which are:

* **True Positive (TP),** this occurs when the observation of the model is predicted to be positive, and it is positive in the image being classified.
* **False Positive (FP),** a FP occur when the observation made by the model is predicted to be a positive and, it is negative.
* **True Negative (TN),** a TN occur when the observation made by the model is predicted to be a negative and, it is negative.
* **False Negative (FN),** a FN occur when the observation made by the model is predicted to be a negative and, it is a positive.

At this stage of the study, the basic characteristics extracted from the confusion matrix namely are TP, FP, TN and FN, will be used in equations in order to be able to calculate important metrices to determine the accuaracy of the model when it comes to classification of the wanted horse breeds. The metrices to be calculated are Accuracy (ACC), Precision (PPV), Recall (TPR) and F1 Score:

1. **Accuracy (ACC)**

The accuracy metric measures the correct predictions made by the model against the total number of predictions made by the model. High accuracy score indicates that the model implemented is making correct predictions during most of the time, however, a low accuracy score indicates that the model is making incorrect predictions and hence needs more fine-tuning. The equation of accuracy is as follows: Accuracy = (TP+TN)/(TP+TN+FP+FN)

1. **Precision (PPV)**

This metric is used to evaluate the accuracy of the positive predictions made by the model. Furthermore, this measures the proportion of the true positive predictions out of all the positive predictions made. This metric helps to assess the model’s ability to avoid making false positive errors. This is measured by the following equation: Precision = (TP)/ (TP + FP)

1. **Recall (TPR)**

Recall which is also known as the true positive rate is an important metric used to measure the ability of a model to capture all instances of a particular class within the given dataset. A high value shows that the model is able to identify most of the actual positive instances, minimising the false negatives. Furthermore, increasing the recall value affects the precision of a model. This metric is calculated by this equation: Recall = (TP)/ (TP + TN)

1. **F1 Score**

F1 Score is calculated when the harmonic means between precision and recall need to be considered simultaneously. Furthermore, it is a balanced measure of a model’s accuracy. The F1 Score is a valuable metric when one needs to assess the overall accuracy of a classification model. F1 Score will be calculated using this equation: F1 Score = (2 x Precision x Recall)/ (Precision + Recall)

# References

|  |  |
| --- | --- |
| [1] | G. A, “A concise, practical, and readable overview of Artificial Intelligence,” *Understanding AI Technology,* pp. 5-7, 2020. |
| [2] | E. O, *Deep Learning for Animal Recognition,* University of Groningen, 2019, p. 13. |
| [3] | T. Lindeberg, “Scale Invariant Feature Transform,” *Scholarpedia,* p. 1, 2015. |
| [4] | P. V and M. J, “Rapid Object Detection using a Boosted Cascade of Simple,” in *ACCEPTED CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION 2001*, 2001. |
| [5] | N. D and B. T, “Histograms of Oriented Gradients for Human Detection,” Montbonnot, 2005. |
| [6] | E. J. R. Van Hiep P, “A High‐Accuracy Model Average Ensemble of,” *applied sciences,* vol. 9, no. 4500, p. 3, 2019. |
| [7] | Datagen, “ResNet-50: The Basics and a Quick Tutorial,” Datgen, [Online]. Available: https://datagen.tech/guides/computer-vision/resnet-50/#. [Accessed 30 January 2024]. |
| [8] | S. Bangar, “VGG-Net Architecture Explained,” Medium, 28 June 2022. [Online]. Available: https://medium.com/@siddheshb008/vgg-net-architecture-explained-71179310050f. [Accessed 30 January 2024]. |
| [9] | W. Karsens, *Animal Recognition Using Different Deep Convolutional Neural Networks,* Groningen: University Of Groningen. |
| [10] | W. S, W. W and P. A, “Convolutional Neural Networks Using MobileNet,” Prince of Songkla University , Songkhla. |
| [11] | S. T and O. T, “Animal classification using facial images with score-level fusion,” *IET Computer Vision,* vol. 12, no. 5, 2018. |
| [12] | G. V. Horn, O. M. Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona and S. Belongie , “The iNaturalist Species Classification and Detection Dataset,” in *IEEE conference on computer vision and pattern recognition*, 2018. |
| [13] | datagen, “Understanding VGG16: Concepts, Architecture, and Performance,” datagen, [Online]. Available: https://datagen.tech/guides/computer-vision/vgg16/#. [Accessed 11 January 2024]. |
| [14] | S. B, “AlexNet Architecture Explained,” Medium, 24 June 2022. [Online]. Available: https://medium.com/@siddheshb008/alexnet-architecture-explained-b6240c528bd5. [Accessed 11 January 2024]. |
| [15] | J. R, S. D, R. G and A. F, “Cornell University,” 9 May 2016. [Online]. Available: https://arxiv.org/pdf/1506.02640.pdf. [Accessed 13 November 2023]. |