成都理工大學中英合作办学
CDUT Sino-British Collaborative Education

OXFORD
BROOKES
UNIVERSITY

UNDERGRADUATE PROJECT REPORT

Project Title:	**Dog Breed Classification using Convolutional Neural Network**
Surname:	**Fanlin**
First Name:	**Zhou**
Student Number:	**201918020209**
Supervisor Name:	**Dr. Grace Ugochi Nneji**
Module Code:	**CHC 6096**
Module Name:	**Project**
Date Submitted:	**May 5, 2023**

Chengdu University of Technology Oxford Brookes College

Chengdu University of Technology

BSc (Single Honours) Degree Project

Programme Name: Software Engineering

Module No.: CHC 6096

Surname: Fanlin

First Name: Zhou

Project Title: Dog Breed Classification using Convolutional Neural Network

Student No.: 201918020209

Supervisor: Dr. Grace Ugochi Nneji

2ND Supervisor: Mr. Joojo Walker Martin

Date submitted: 5th May, 2023

A report submitted as part of the requirements for the degree of BSc (Hons) in Software Engineering

At

Chengdu University of Technology Oxford Brookes College

Declaration

Student Conduct Regulations:

Please ensure you are familiar with the regulations in relation to Academic Integrity. The University takes this issue very seriously and students have been expelled or had their degrees withheld for cheating in assessment. It is important that students having difficulties with their work should seek help from their tutors rather than be tempted to use unfair means to gain marks. Students should not risk losing their degree and undermining all the work they have done towards it. You are expected to have familiarised yourself with these regulations.

https://www.brookes.ac.uk/regulations/current/appeals-complaints-and-conduct/c1-1/

Guidance on the correct use of references can be found on www.brookes.ac.uk/services/library, and also in a handout in the Library.

The full regulations may be accessed online at https://www.brookes.ac.uk/students/sirt/student-conduct/

If you do not understand what any of these terms mean, you should ask your Project Supervisor to clarify them for you.

I declare that I have read and understood Regulations C1.1.4 of the Regulations governing Academic Misconduct, and that the work I submit is fully in accordance with them.

Signature Zhou Fanlin Date 29/4/2023

REGULATIONS GOVERNING THE DEPOSIT AND USE OF OXFORD BROOKES UNIVERSITY MODULAR PROGRAMME PROJECTS AND DISSERTATIONS

Copies of projects/dissertations, submitted in fulfilment of Modular Programme requirements and achieving marks of 60% or above, shall normally be kept by the Oxford Brookes University Library.

I agree that this dissertation may be available for reading and photocopying in accordance with the Regulations governing the use of the Oxford Brookes University Library.

Signature Zhou Fanlin Date 29/4/2023

Acknowledgment

I would like to express my thanks to all the people who have supported me in completing this project. First of all, I would like to thank my supervisor Grace Ugochi Nneji for providing me with valuable feedback and guidance throughout the writing process, the staff in OBU and my project module leader Joojo Walker Martin for his teaching and advice. Secondly, I would like to thank Oxford Brookes University and Chengdu University of Technology for their cooperation.

I would like to thank my family for their unwavering support and encouragement, which played an important role in my project success. I am also grateful to my classmates and friends for their encouragement and support. I will always be grateful.

Table of Contents

LISTS OF FIGURE

CHPATER 1:

CHPATER 3:

CHAPTER 4:

CHPATER 5:

LIST OF TABLE

CHPATER 2:

CHPATER 3:

CHPATER 4:

CHAPTER 5:

Abstract

Dog breed classification is a popular and important problem in the field of computer vision. In this project, this project explored the effectiveness of three popular deep learning models, VGG16, ResNet50, and Xception, for classifying dog breeds using dog datasets.

The dataset contains more than 10,000 images belonging to 120 different dog breeds and the task is to correctly identify the breed of a given dogs image. To achieve this, the project preprocessed the data by applying various data enhancement techniques to increase the diversity of the training data. More so, the project applied transfer learning, taking advantage of the pre-training weights of three different models.

After training the models, the project evaluated their performance on a validation set and selected the best-performing models for testing on a separate test set. The project used a variety of performance metrics, such as accuracy, precision and recall rates, to assess the model's effectiveness in accurately identifying breeds. The results show that all of the three models achieve a high level of accuracy, precision, recall with Xception lightly better than the other two models. The results highlight the effectiveness of deep learning in solving complex classification of dog breed.

The project will be of vital use to animal shelters or rescues, veterinarians, dog trainers or behaviorists, and dog lovers for recognizing varieties of dog breeds. They will be better able to identify the characteristics of the dog according to the project so that they can better understand the dog breed.

Keywords: *Dog Breed, Convolutional Neural Network, Image Processing, transfer learning*

Abbreviations

GUI: Graphical User interface

GPU: Graphics processing unit

CNN: Convolutional Neural Network

FCI: The Federation Cynologique Internationale

SVM: Support Vector Machine

SIFT: Scale Invariant Feature Transform

PCA: principal component analysis

ML: machine learning

IEEE: Institute of Electrical and Electronics Engineers

Glossary

Image classification: Image classification is the process of categorizing and labeling groups of pixels or vectors within an image based on specific rules.

training accuracy: training accuracy is usually the accuracy you get if you apply the model on the training data, while testing accuracy is the accuracy for the testing data.

validation accuracy: the accuracy you calculate on the data set you do not use for training.

Transfer learning: Is a noun in machine learning that refers to the influence of one type of learning on another type of learning, or the influence of learned experiences on the completion of other activities.

Pre-trained model: Pre-trained models are machine learning models that are trained, developed and made available by other developers.

ImageNet: ImageNet is an image database organized according to the WordNet hierarchy (currently only the nouns), in which each node of the hierarchy is depicted by hundreds and thousands of images.

ResNet: is a specific type of convolutional neural network (CNN).

Framework: A framework can also be seen as a template.

Xception: Xception is a deep convolutional neural network architecture that involves Depthwise Separable Convolutions.

Chapter 1 Introduction

1.1 Background

Dogs have been an integral part of human history for thousands of years, serving as loyal companions, protectors and working animals [1]. They are kept for a variety of purposes, such as hunting, herding, guarding, and even as support animals. Today, dogs are still widely used for these purposes, but they also play an important role in the society as family pets and emotional support animals. Traditionally, the diversity of dog breeds is mainly due to genetic diversity, which results in differences in appearance, temperament and behavior. However, as the number of breeds increases and different breeds interbreed, it becomes increasingly challenging to distinguish between many different types of dogs. This makes the classification of dog breeds a complex task, and by 2022, the number of FCI-approved breeds has reached 356 [2]. Therefore, how to quickly and accurately identify dog breeds is a challenging problem.

To address this challenge, experts have developed various image classification methods to help identify different dog breeds. J. Liu et al. [3] proposed a dog breed classification method with local localization to classify dog breeds in 2012. After the local localization method was proposed, X. Wang et al. developed a Grassmann manifold method for dog breed classification in 2014 on the basis of the establishment of [3]. However, these methods can be time-consuming and are only effective under certain conditions, with limited accuracy. Additionally, the cost of preprocessing and the need for fully labeled datasets can be prohibitive. The researchers applied traditional computer image classification methods (pre-processing + feature extraction + classifier, such as SIFT + AdaBoost, SIFT+SVM) to dog breed classification. In addition, a pet classification method based on DPM and word bag model [5], a dog breed classification method based on local location [6], a dog breed classification method based on manifold space [7] and other classification methods are proposed. However, these methods have many problems in cost, accuracy and difficulty of application, which can not meet the needs of practical application.

Overall, breed classification remains a complex and challenging task. Therefore, the development or improvement of dog breed identification methods to achieve fast, accurate and low-cost dog breed identification is the main demand of the current dog breed identification problem. But the development of new machine learning and computer vision technologies promises to improve the accuracy and efficiency of breed identification. As understanding of the genetic and behavioral diversity of dogs continues to grow, it is important to use better and different machine learning models to improve dog breed classification.

1.2 What is Convolutional Neural Network

This project will apply convolutional neural networks. Here's what a convolutional neural network is. Convolutional neural network (CNN) is a multi-layer perceptron and depth model for visual image analysis. It's a great learning algorithm to understand the content of images. And has a great role in image segmentation and classification tasks. It is understood that CNN has been used for MNIST handwritten data recognition, ImageNet, license plate recognition, as well as face recognition and many other computer vision image recognition.

In convolutional neural networks, there are three important operations and six important layers. They are local receptive field, weight sharing, pooling layer. And the input layer, the convolution layer, the pooling layer, the activation layer, the full connection layer, and the output layer.

This project is aimed at 120 dog breeds classification. CNN can be well used in image recognition. The convolutional structure can reduce the memory occupied by the deep network. When the amount of data is too large, it can be costly and inefficient. In addition, images can lose their original character when converted to digital. CNN is similar to the visual way. It preserves the image features very well. Therefore, using convolutional neural network becomes the best choice for dog breed classification.

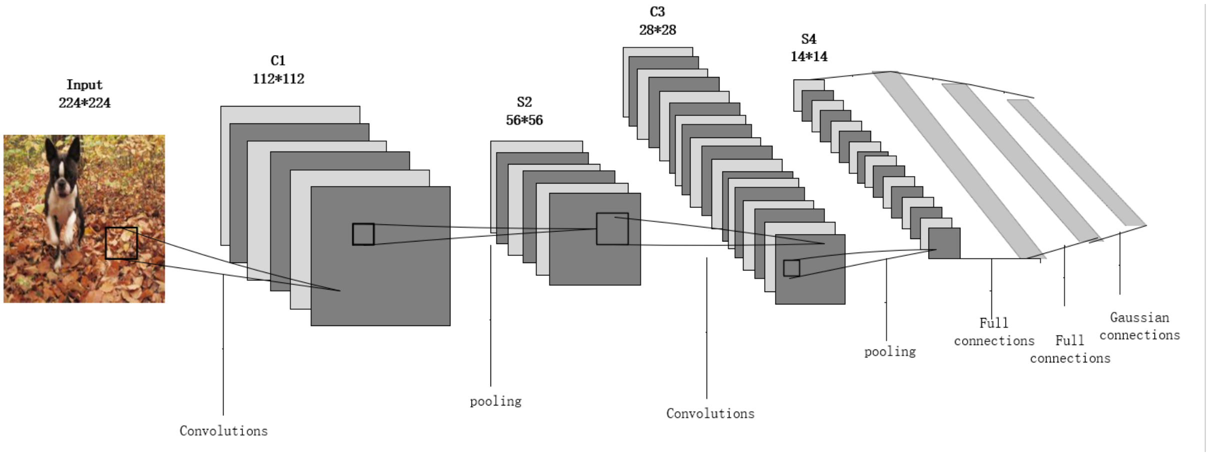

Figure 1-1: Convolutional neural network model diagram

1.2.1 Input Layer

Input layer is used for data input. When you input a 224jpg image, the input layer reads a 224*224*3 matrix, with 3 being its depth (R, G, B).

1.2.2 Conv Layer

In CNN, the convolutional layer is the most important existence. The convolution layer consists of a set of filters with three imensional structure. The filter extracts features from the input data. And then it does the convolution on these inputs. In the calculation, there is a small window consisting of convolution nuclei that can slide at regular intervals. And then, the corresponding entry is multiplied by the entry in the convolution kernel. The results of these multiplications are then summed to produce an output value, which is placed in the appropriate position in the output matrix. The process of sliding the kernel over the input matrix is repeated at all locations in the matrix, producing a new matrix called the output feature map.

The convolutional layer can learn to detect various features in the input data by adjusting the kernel weight during training. The number and size of cores used in the Conv layer can be adjusted to allow the layer to extract different types of features.

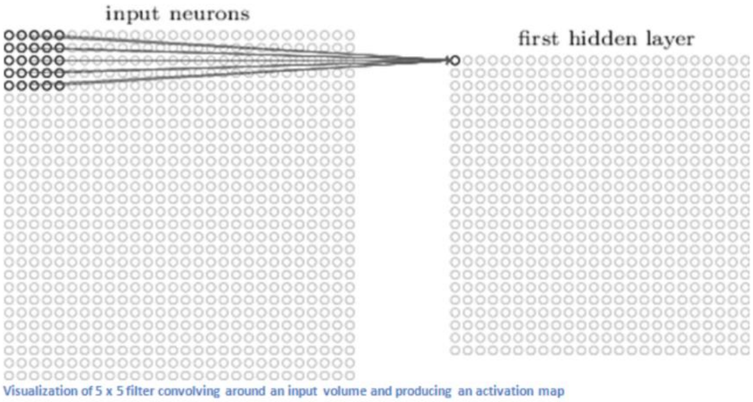
input neurons
first hidden layer
Visualization of 5 x 5 filter convolving around an input volume and producing an activation map

Figure 1-2: The Conv layer [27]

1.2.3 FC Layer

The output layer in CNN is responsible for producing the final output of the model. Before reaching the output layer, the input data goes through several layers of convolution, excitation, and pooling to extract high-quality characteristics. After these operations, the fully connected layer can be added to further refine the extracted features. However, having too many neurons in the FC layer can lead to overfitting, so the dropout technique can be used to randomly remove some neurons during training. Local normalization (LRN) and data promotion can also be used to increase the robustness of the model.

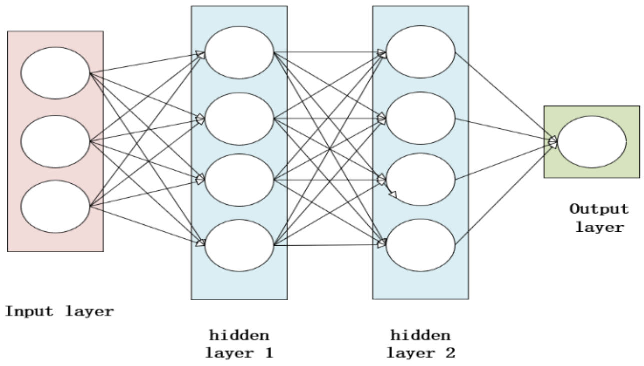

Figure 1-3: FC layer

1.2.4 Pooling Layer

The subsampling is carried out, and the feature map is sparsely processed to reduce the amount of data computation.

Pooling, also known as undersampling or downsampling. By downsampling the feature maps generated by the convolution layer, the pooling layer can compress the amount of data processed and reduce the number of parameters required by the model. Another benefit of using pooling layers is that it can help prevent overfitting by reducing the spatial resolution of feature maps, making them more robust to changes in the input data. In addition, pooling layers can help improve the model's fault tolerance, making it more invariant to small changes in inputs.

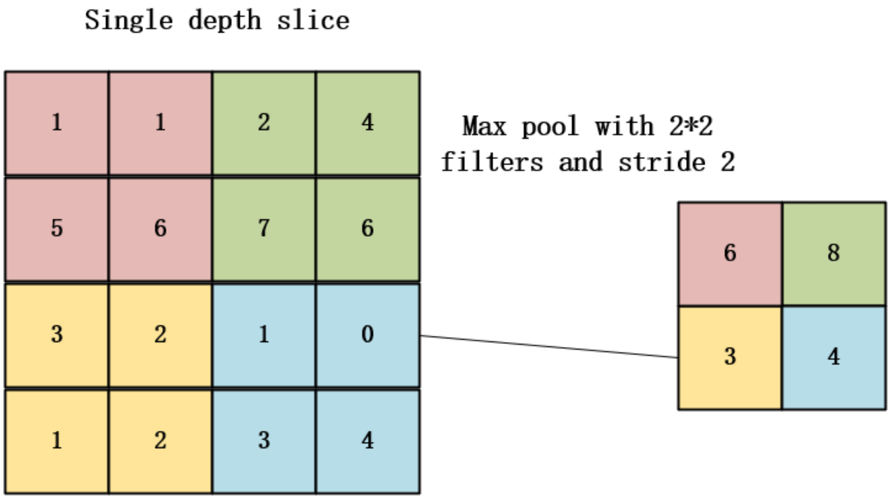

Figure 1-4 : Pooling layer

1.2.5 Loss

Loss and accuracy are two important indicators in deep learning. The loss demonstrates the performance of the model. And accuracy demonstrates the accuracy of the model. The loss displays the difference between the training results of the model and the actual results obtained. Generally speaking, it aims to minimize the loss of the model as much as possible.

In depth learning, loss function needs to be calculated. The loss function in the machine learning model measures the inconsistency or degree of inconsistency between the predicted output and the actual output. It is a non negative real-valued function, usually expressed as L (Y, f (x)). The goal of the model is to minimize the value of the loss function, which indicates how the model performs. In the above function, Y represents the actual output, and f (x) represents the predicted output value for the model of input x.

Loss function is a key part of economic risk function and an important part of structural risk function. Most importantly, it is the most critical part of the deep learning model. When calculating the loss function and showing it in picture form. In the case of the breed classification project, it can clearly see the errors and expectations generated by the model on the training set.

1.2.6 Function

Convolution calculation method:

bs:batchsize, in_c: Input channel, h/w: input size, k: convolution kernel, p:padding, s: step size, out_c: output channel

Conv calculation equations (1)-(3):

$$\text{IN: bs*in\_c*h*w*} \tag{1}$$

$$\text{Conv: k, p, s, out\_c} \tag{2}$$

$$\text{OUT: bs*out\_c*}\left(\frac{h-k+2*p}{s}+1\right) * \left(\frac{w-k+2*p}{s}+1\right) \tag{3}$$

1.3 Aim

In the dog classification project, the goal is to compare three different learning models VGG-16, ResNet50, and Xception based on the transfer learning approach. From these, an accurate and efficient machine learning model can be selected to classify the images of a given dog into the corresponding breed category.

1.4 Objectives

The objectives are as follow:

a. Understanding the background of dog breed classification and selecting appropriate deep learning techniques for the task. Transfer learning with a pre-trained framework will be used to perform classification, enabling the project to leverage existing knowledge and expertise in the field.

b. Identifying a suitable dataset for dog breed classification that includes training, validation, and test sets, as well as the size of images for ImageNet. This dataset will be used to train and evaluate the classification model.

c. The pretrained model will be used as a fixed image feature extractor, allowing the model to learn from the pre--existing feature maps and further refine them to improve classification performance.

d. A global averaging pooling layer and fully connected layers will be added to the model for the classification of dog breeds. These layers will allow the model to make predictions about the breed of the dog in the input image.

e. The quality of the model is judged by using different indicators such as accuracy, loss value, accuracy, and recall rate.

f. Fine-tune the pre-trained model by adjusting the hyperparameters, such as learning rate, number of epochs, and batch size, to improve the model's performance.

g. In order to obtain better data, the project can adopt corresponding data augmentation techniques to achieve it. Classifying the dataset and resizing the images have become the tasks to be completed in this project. In order to achieve better results.

h. By displaying the loss and accuracy through images, it is possible to clearly see the changes that the model undergoes during training.

i. Investigate the model's performance using different pre-trained models, such as VGG16, ResNet50, and XInception, to compare their performance.

1.5 Project Overview

This project is based on convolutional neural network about deep learning, using the pre-training model in transfer learning to classify dog breeds.

1.5.1 Scope

This design has realized the classification of 120 common dog breeds. In view of the problems of high time and cost in literature [3-7], the reason why dog breed identification is commonly used in the actual environment is fully considered. And the accuracy of image classification. The convolutional neural network is used as the design basis for dog breed classification.

The significant of this project are as follows:

a. By using an existing model in migration learning, you can save a lot of time and resources that would otherwise be spent building a new model from scratch. In addition, existing models may already have been trained on large data sets, which can lead to better performance and more accurate results.

b. Data set preprocessing and enhancement are key steps to optimize the performance of machine learning models. By analyzing and modifying the data set, you can ensure that the model is trained on high-quality data. Preprocessing techniques can lead to more robust and accurate models that can be better generalized to new data.

c. CNN can be used to obtain higher accuracy and efficiency for dog breed classification. The convolutional layer can reduce data memory. And can be widely used.

d. This project is based on dog breed classification, so it is easy to identify different kinds of dogs. If faced with a few breeds and unfamiliar dogs, it is easy to track them.

e. After continuous training and use of the model. This project could provide an opportunity to explore the potential of deep learning for a variety of applications beyond breed classification.

By applying transfer learning, the pre-training model can be adapted to recognize the unique features and characteristics of different dog breeds without requiring the same amount of data and computation as a model trained from scratch. The pre-training model is fine-tuned on the new data set to improve its performance. In the context of the classification of 120 dog breeds, transfer learning can leverage what is learned from pre-training models on large data sets such as ImageNet to improve the efficiency and accuracy of the classification task.

1.5.2 Audience

The audience of dog breed classification can follow:

a. Children: Children often have difficulty identifying different breeds of dogs. This project can provide educational resources to help children identify and learn various dog breeds, improve their cognitive abilities and understanding of the animal world.
b. Government: The government can benefit from this project as it can serve as a source of information for different breeds of dogs. This is particularly useful for animal management authorities and organizations involved in animal welfare and protection.
c. Dog enthusiasts: There are many people who love dogs and want to help them. But they don't know the breed of dog. This project can help them identify the breeds of dogs they encounter and provide better care and management for their dogs.
d. Doctor: When a dog gets sick, it can provide doctors with knowledge about different dogs and help them better understand them.
e. Public: This project may arouse public interest, as they may be curious about how dog breeds and deep learning can be applied to animal classification. It can also raise awareness of the importance of responsible dog keeping and protecting endangered species.

To sum up, the breed classification program has a wide range of potential stakeholders, including children, government agencies and dog lovers. By providing valuable educational and information resources, the project can make a positive contribution to society and improve the understanding of the animal world.

Chapter 2 Background Review

Dog breed classification using deep learning has been an active area of research in recent years. Researchers have explored various approaches to improve the accuracy of classification models. Raduly et al. [8] used fine-tuning of Convolutional Neural Networks (CNNs) to achieve high accuracy in classifying dog breeds from the Stanford dog dataset. Similarly, Lai et al. [10,11] used transfer learning on CNNs to achieve an accuracy rate of 86.63% on the same dataset.Other researchers have explored combining different features and techniques to improve classification accuracy. Kumar et al. [12] used Principal Component Analysis (PCA) to analyze 8,351 dog images and achieved an accuracy rate of 90%. Jain et al. [13] used a convolutional neural network, specifically the Res-Net model, to achieve an accuracy rate of 84.578%. Some researchers have used pre-trained models and transfer learning to achieve high accuracy rates. Borwarnginn et al. [14] achieved an accuracy rate of 89.92% for 133 dog breeds by using a pre-trained CNN. Dabrowski and Michalik [16] used transfer learning to achieve a 70% accuracy rate by retraining an image classification model. Vrbani et al. [17] used a trained model to classify X-ray images with an entire accuracy rate of 94.76%.

In addition to the above approaches, Jiongxin et al. [15] developed geometric and appearance models of dog breeds and their facial parts based on samples to achieve 67% accuracy on 133 dog breeds with 8,351 images. Massinee et al. [18] The method (crude classification and PCA

fine classification) was used on 700 facial images of 35 dog breeds with an accuracy of 93%. It is better than PCA based classifier without coarse classification. Khosla et al. [19] created a Stanford Dog dataset where they used the SIFT descriptor to classify Stanford dogs with 22% accuracy. Among them, SVM (selective pooled vector) model training method can achieve 52% accuracy in Stanford Dog data set [20]. It encodes descriptors (perhaps images, audio, etc.) using vectors, and then compares the differences between these vectors to determine if they belong to the same category. Similar descriptors will have similar vector representations, while descriptors between different categories will have large vector differences. The method involving the Nordic region [21] achieved some imperfect accuracy (47% accuracy) based on the same data set as above, creating a model that doesn't have to worry about images of different sizes by using pattern detection units and image descriptors. In this way, there will be no deviation due to the difference in image size.

2.1 Transfer Learning

Generally speaking, transfer learning is the ability to learn new knowledge by applying existing knowledge. The main task of transfer learning is to use previously learned knowledge and experience to help solve new tasks, and to reduce the difficulty and time of learning by finding and utilizing the common points between new tasks and existing knowledge. This is called transfer learning. Migration learning solves problems as diverse as data obsolescence and data labeling abundance and allows machine learning to be used beyond tasks and domains. And it can increase productivity by reducing the time it takes to implement new projects. [22]

2.2 VGG-16

Uno et al. [23] used the fusion of AlexNet and VGG-16 models to study the classification of fine-grained dogs, and the accuracy of VGG-16 increased from 81.2% to 84.8%. Wang et al. [24] combined the processed binary image with VGG16 in defect identification and classification of polyethylene gas pipeline, and the accuracy rate was as high as 97%. A mock-up of the VGG-16 is shown below.

2.3 ResNet50

Kumar et al. [25] used ResNet50 to achieve fine-grained image classification accuracy of 87.89% for dog breeds. Raduly et al. determined dog breeds in a given image, using two different models for training and evaluation on the Stanford Dog dataset. The Inception-ResNet-v2 training model has an accuracy of 93.65% [8].

2.4 Xception

Wang et al. [26] used the Inception-ResNet v2 and Xception models to classify 21 dog and cat species. In this model ,the training accuracy of 99.49% and effective accuracy of 99.21%.

In conclusion, the application of convolutional neural networks has the potential to improve the accuracy and generalization ability of dog breed classification model. This is because CNN can learn and capture more complex and abstract features from images compared with traditional methods. In addition, deep learning techniques such as transfer learning allow models to leverage knowledge from pre-trained networks and improve performance on both new and unseen data.

Researchers	Their Model	Performance Metrics
Kumar et al. [12]	ResNet + PCA	Accuracy = 90%
Jain et al. [13]	Res-Net	Accuracy = 84.578%
Borwarnginn et al. [14]	Pretrained	Accuracy = 88.92%
Jiongxin et al. [15]	Not find	Accuracy = 67%
Dbrowski and Michalik [16]	Re-trained	Accuracy = 70%
Vrbani et al. [17]	Pretrained	Accuracy = 94.76%
Chanvichitkul, M. [18]	PCA	Accuracy = 93%
Uno et al. [23]	VGG-16	Accuracy = 84.8%
Wang et al. [24]	VGG-16	Accuracy = 97%
Kumar et al. [25]	ResNet50	Accuracy = 87.89%

Wang et al. [26]	Inception-ResNet v2 and Xception	Accuracy = 94.49%

Table 2-1: Summary of existing methods for dog breed classification

Chapter 3 Methodology

3.1 Approach

In the design and implementation, dog breed classification will be implemented using a pre-trained model. Pretraining is a technique commonly used in natural language processing (NLP) to train large language models, such as GPT-3 and BERT, on large amounts of textual data.

The structure of the pre-training model usually involves a deep neural network architecture composed of multiple layers of neurons. The architecture may vary from model to model, but it usually includes an encoder and a decoder. The encoder receives an input tag sequence (usually a word or subword) and generates a higher-dimensional representation of that sequence. The decoder then takes the output from the encoder and generates a prediction for the next tag in the sequence.

Once the pre-training is complete, the pre-trained model is fine-tuned for the specific task. Fine-tuning trains the model on a smaller data set specific to the task at hand. In the process of tuning, the pre-trained model is further trained to make its parameters adapt to specific tasks, so as to obtain a highly accurate and efficient NLP model.

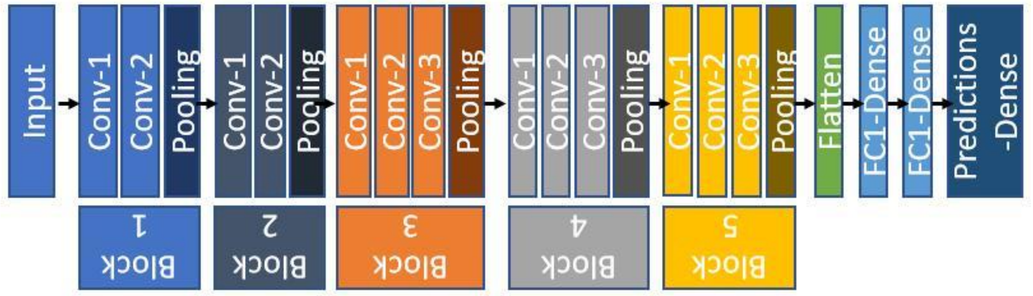

Figure 3- 1: The pre-trained architecture for this project [9].

3.1.1 Dataset:

The dog breed classification dataset on Kaggle is a collection of tagged images belonging to 120 different breeds of dogs. The data set consists of approximately 10,000 images from the training set, 8,000 images from the test set, and an additional sample submission file. The images are in JPG format and vary in size, with the largest being 4,000 pixels wide.

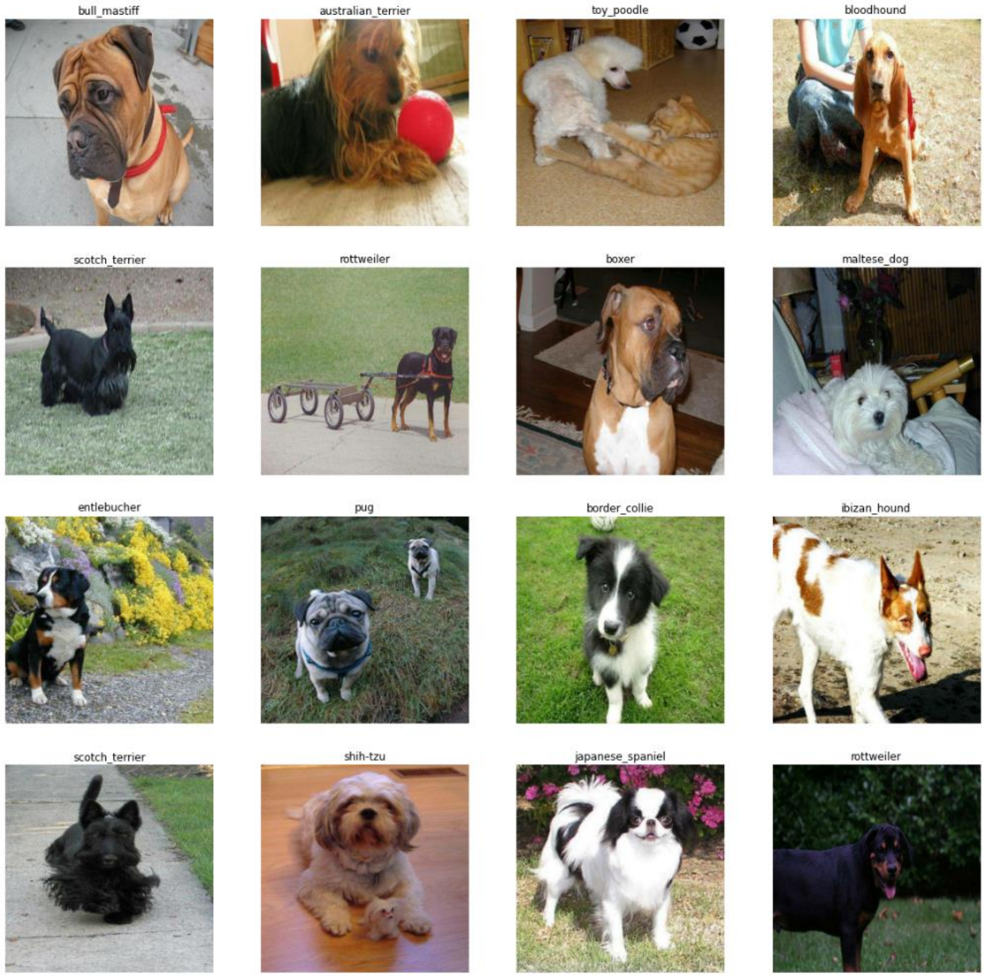

Figure 3- 2: Dog breed datasets

Total datasets	The number of taining	The number of testing	Image size (length * breadth * channel)
20581	10222	10357	224*224*3

Table 3- 1: The information of dataset

Data processing: Before building the model, the dataset was also preprocessed. On the one hand, it can help understand the quality of data. On the other hand, it can prevent data redundancy. The data preprocessing method used in this project, image size adjustment, is the process of changing the image size. This can be achieved by increasing or decreasing the number of pixels in the image, which in turn affects the overall size and quality of the image. When adjusting the image size, it is important to maintain the aspect ratio to avoid image distortion. Aspect ratio is the relationship between the width and height of an image. This project uniformly adjusts the data size of the dataset to 224 * 224 * 3. This has a great impact on the project.

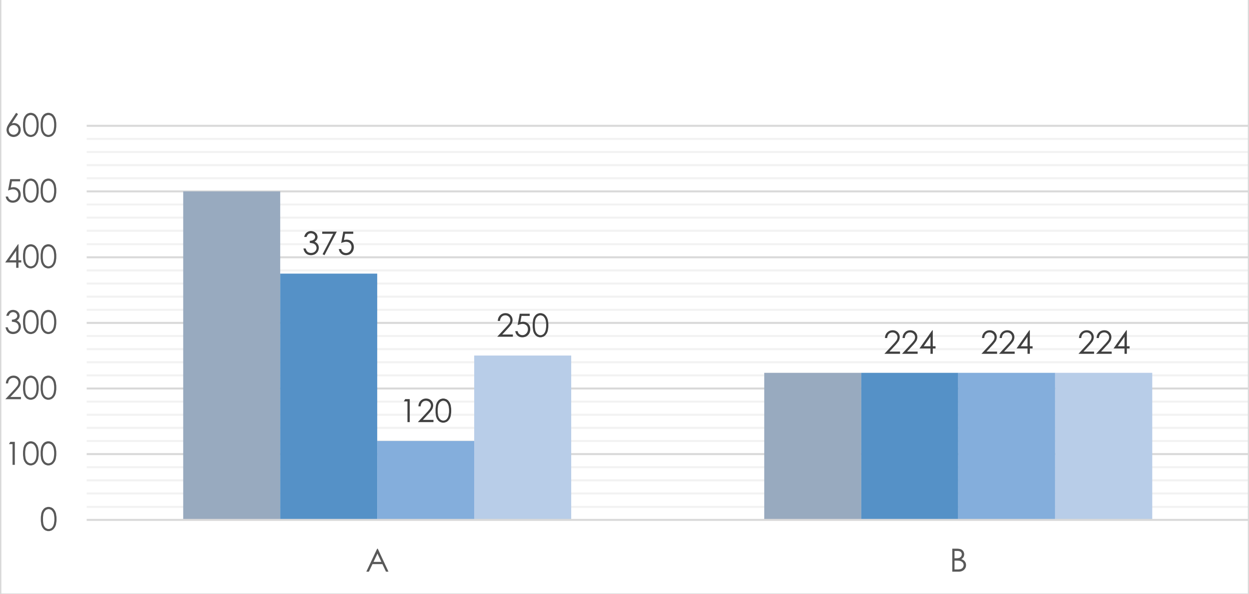

Table 3- 2: Data processing-resizing

3.1.2 Model

a. VGG-16 : VGG16 is a CNN structure. It is a deep learning model widely used for image classification and classification tasks. The VGG16 structure consists of 16 layers, including 13 convolutional layers and 3 fully connected layers. Each Conv layer uses a small 3x3 filter with a step length of 1. The number of filters increases as the network deepens. The maximum pooling layer is used after every two or three convolution layers to reduce the space size of the feature map.The FC layers at the end of the network are used for classification, and they take the high-level features extracted by the convolutional layer as input. The architecture uses ReLU activation functions in all layers except the output layer, which uses softmax to generate probability values for each class.VGG16 is known for its simplicity and high precision in image classification tasks, and it has been used as a baseline model for many computer vision applications. The model architecture of VGG16 is shown below.

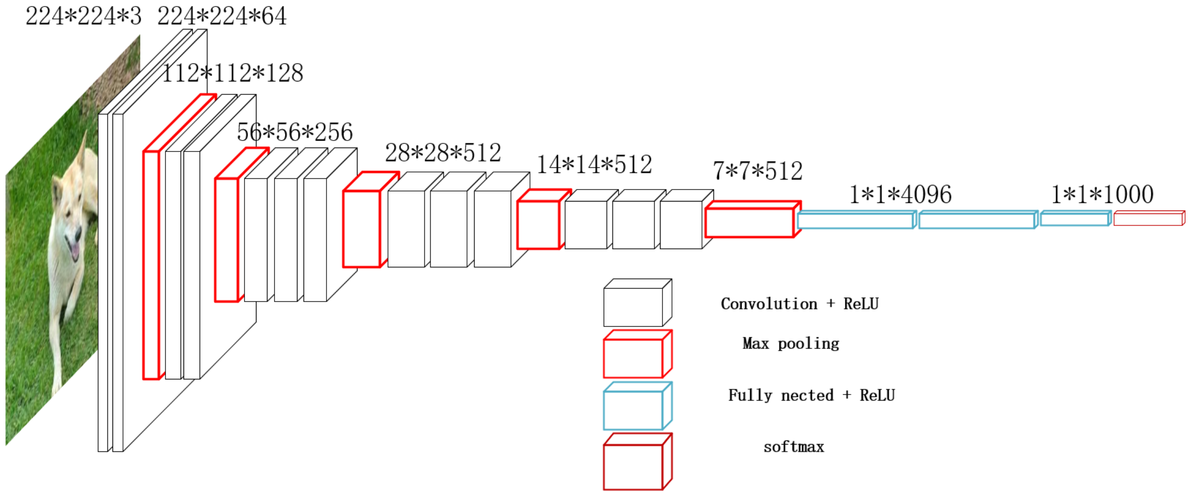

Figure 3- 3：VGG-16 model Frame

b. ResNet50

ResNet50 is a CNN construction in deep learning. ResNet50 consists of 50 layers, including convolution layer, pooling layer, full connection layer and identity shortcut connection. The identity shortcut connection allows the network to learn the residual function and makes it easier to train very deep neural networks. The architecture also includes batch normalization and ReLU activation functions. ResNet50 has been widely used for a variety of computer vision tasks such as image classification, object detection and segmentation. The model of ResNet is shown below:

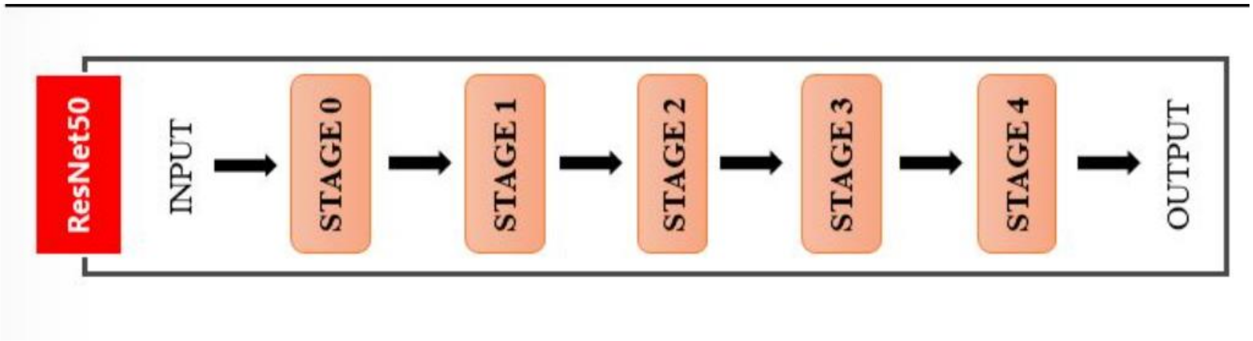

Figure 3- 4: The model of ResNet50

c. Xception

Xception (Extreme Inception) is a convolutional neural network (CNN) construction, which is a kind of Inception architecture. Xception aims to address the computational cost of the original Inception structure. Xception replaces the traditional Inception module with a deep separable convolution module consisting of a deep convolution layer and a point convolution layer. The module reduces the number of parameters and calculations required while maintaining the accuracy of the network. In addition, Xception uses a linear stack of depth-separable convolution modules, rather than the multi-branch architecture used in Inception. Deep separable convolution allows better information flow through the network while reducing the computational complexity of the model. At the same time, it also applied it in a number of areas including image classification.

The essence of Xception is equivalent to an existing separable convolution in terms of depth. It consists of a spatial convolution performed separately for each channel in terms of depth, followed by a point-by-point convolution (a 1 by 1 convolution across channels). The structure diagram for Xception is shown below.

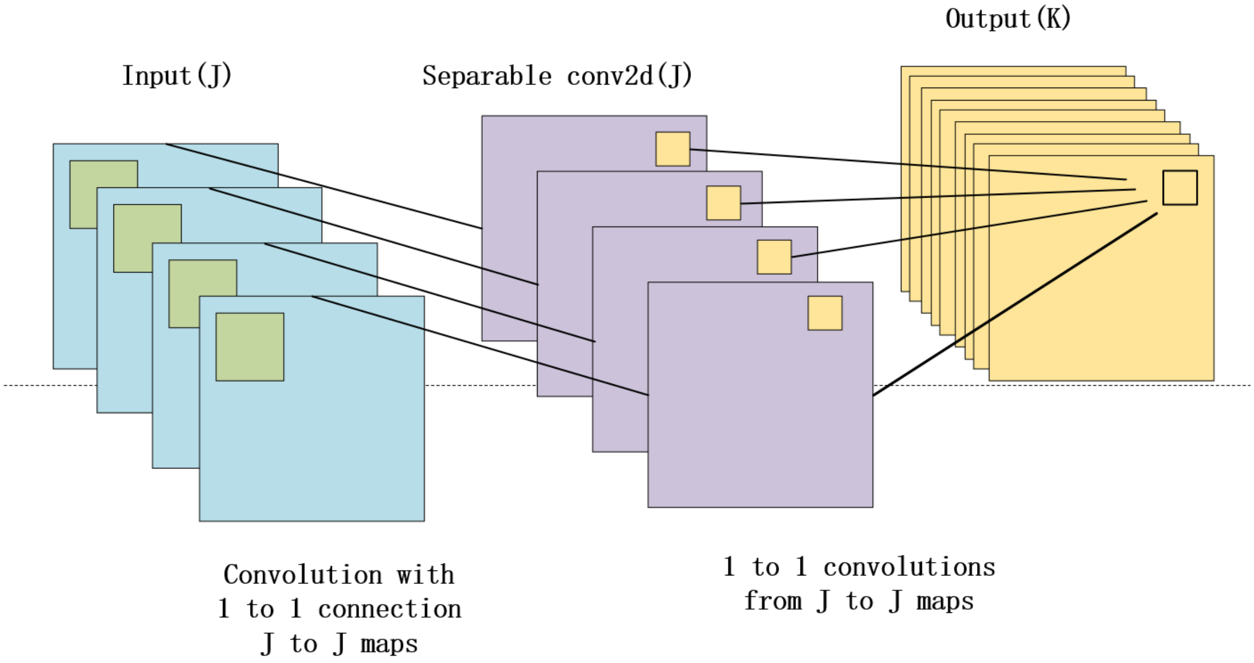

Figure 3- 5 : The model of Xception

3.2 Technology

The system configuration of the project is as follows: The project runs on an Intel Core 15 8th-generation processor. It has 8GB of RAM and 256GB of hard drive. The operating system is Windows 10 64-bit. Software requirements for this project include the TensorFlow framework and the Python programming language. In addition, it also requires pre-trained models such as Xception, VGG-16 and ResNet50. The system also needs to install Numpy, Matplotlib, Keras and other libraries. This project is also available for Win 11 64-bit. And use the CPU. But CPU performance is not as good as GPU performance. The configuration of hardware and software can meet the requirements of this project. And can well complete the model training and data results processing.

Part name	Model number
GPU	Intel Core 15 8th Gen
Memory	8GB
Disk space	256GB
System	Win 10 64-bits
Framework	TensorFlow
Language	Python
Pre-trained model:	VGG-16, ResNet50, Xception
Libraries	Numpy, matplotlib, keras

Table 3- 3 : The technology requirements

3.3 Project Version Management

In the context of breed classification, detection and evaluation can be divided into two parts. The first part involves generating loss and accuracy maps for training and test data. These charts provide insight into how the model learns and generalizes new data. The second part of the test and evaluation involves evaluating the model's diagnostic performance in identifying breeds.

In order to evaluate the diagnostic performance of the model, several evaluation criteria are used as indicators. These include accuracy (ACC), precision (PRC), recall (recall), sensitivity (SEN), and specificity. ACC refers to the proportion of correctly classified samples to the total number of samples. PRC is the proportion of true positive samples to all predicted positive samples, and recall is the proportion of true positive samples to all actual positive samples. SEN is true positive rate, that is, the proportion of correctly classified positive samples; Specificity is the true negative rate, that is, the proportion of correctly classified negative samples.

By using these metrics to evaluate the model's performance, the project can gain a deeper understanding of its strengths and weaknesses. This information can be used to optimize the architecture of the model and fine-tune its parameters to improve the accuracy of the diagnosis. The metrics are explained below.

True Positive (TP): indicates that the number of positive classes is predicted to be positive

True Negative (TN): predicts the number of negative classes as negative

False Positive (FP): Predict the number of negative classes to positive classes (Type I error)

False Negative (false negative, FN): predicts the number of positive classes as negative classes → misses (Type II error)

	Positive	Negative
True	True Positive (TP)	True Negative (TN)
False	False Positive (FP)	False Negative (FN)

Table 3- 4: Evaluate the diagnostic performance of the model

The equations (1)-(5) for evaluating performance is shown below:

$$ACC = \frac{\mathrm{TP} + \mathrm{TN}}{TP + FN + FP + TN} \quad (1)$$

$$Sensitivity = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \quad (2)$$

$$Specificity = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \quad (3)$$

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$

$$Recall = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \qquad (5)$$

3.4 Project delivery

The following shows the resulting code links for the final training of the VGG16, ResNet50, and Xception models:

https://github.com/L5S3Catherine/dog-breed-classification

Chapter 4 Results

This section presents a detailed description of the project results. The project evaluated the data separately using three different models.

In order to achieve better test results, this project adopted data processing to enhance the data set in advance before training the model. And then you use the preconditioning model, the trained model, and then you train on top of that. This allows for better accuracy.

In the training process, the three models use different parameters for processing. batch size and epochs parameter values suitable for three models are set.

After training the results of the VGG16 model, it was found that the model was not suitable for the data set used by the project. VGG16 is suitable for large data sets. However, this project data set cannot meet this requirement. Therefore, data processing and final training results are not ideal. All that remains is to train the data using the ResNet50 and Xception models. Among these models, ResNet50 has obtained better test results. The training results will be shown below.

The accuracy of VGG16 model is too low. The project also fine-tuned the VGG16 model to further improve its performance. Retrain the VGG16 model. Data sets are trained using unified data sets. The accuracy of VGG16 model after fine-tuning is 92%, indicating that the fine-tuning process successfully improves the diagnostic performance of the model.

4.1 Parameters setting for three models

To implement the approach, this project uses the VGG16, ResNet50, and Xception models from the keras and rensorflow repositories. VGG16 was fine-tuned using the Stanford breed data set. This project was fine-tuned using training and validation data sets. The number of fully connected layers is redefined in terms of the number of varieties.

The training parameters are shown in the chart. For the VGG16 training, this project used 20 epochs and 16 batch sizes. The ResNet50 model was trained with an epoch of 50 and a batch size of 32. The size of 15epoch and 32batch was used for Xception. This project uses stochastic gradient Descent (SGD) as the optimizer on VGG16 and Xception. And VGG16 and ResNet50 learning rate to le-4 as the standard. Xception is set to 0.1.

Parameters	VGG-16 Values	ResNet50 Values	Xception Values
Input shape	224*224*3	224*224*3	224*224*3
Activation function	ReLU, softmax	ReLU, softmax	ReLU, softmax
Epochs	20	50	15
Training set of images	10222	10222	10222
Testing set of images	10357	10357	10357
Learning Rate	Le-4	Le-4	0.1
Output	11	10	10
Batch size	16	32	32

Table 4- 1: Layers of CNN Parameters Setting

4.2 Evaluations of VGG16 model

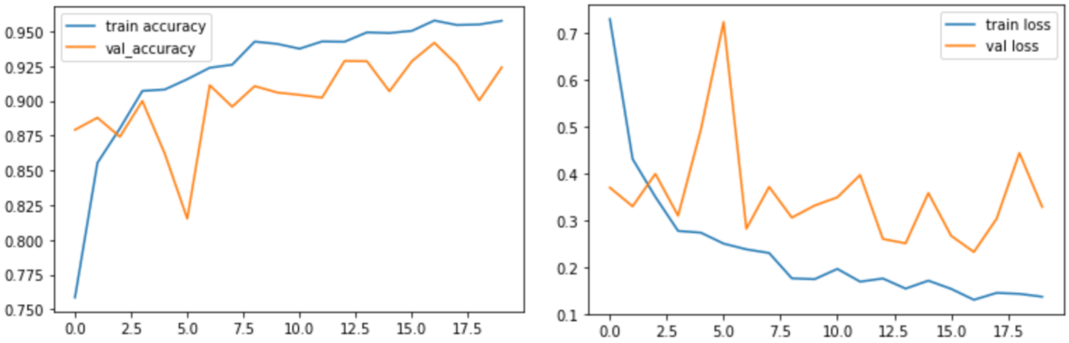

Figure 4- 1: Accuracy and train loss graph for VGG16 neural network

From Figure 4-1, it is ev=ident that the VGG16 neural network has been evaluated based on its performance with respect to accuracy and loss. The first image in the graph shows the relationship between accuracy values and epoch numbers, which reflects the overall ability of the model to correctly predict the class labels of the input data. The maximum accuracy achieved by the model after 20 epochs, including both training and validation accuracy, is 99.17%.

The second image in the graph displays the loss value of the training data as well as the validation data against the epoch number. This helps to assess how well the model is able to minimize the error between its predictions and the actual labels of the input data. From the evaluation of the graph, it is observed that with the increase in epoch number, the accuracy of the model also increases, indicating an improvement in its performance. Additionally, the loss value decreases as well, suggesting that the model is becoming better at making accurate predictions.

By applying transfer learning on ImageNet, VGG16 model is used. It is a model with 16 layers, of which 13 are convolutional layers and 3 are full connection layers. The architecture and parameters of the VGG16 neural network model are shown in Table 4-2. The table displays the input image size for each layer and the parameter values for each layer.

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
Block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
Block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
Block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
Block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
Block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
Block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
Block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
Block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
Block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
Block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
Block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
Block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
Block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0

Block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
Block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
Block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
Block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
global_max_pooling2d (GlobalMaxPooling2D)	(None, 512)	0
dense (Dense)	(None, 512)	252656
dropout (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 1)	513
Total params: 14,977,857 Trainable params: 7,342,593 Non-trainable params: 7,635,264		

Table 4- 2: VGG16 model

4.3 Evaluations of ResNet50 model

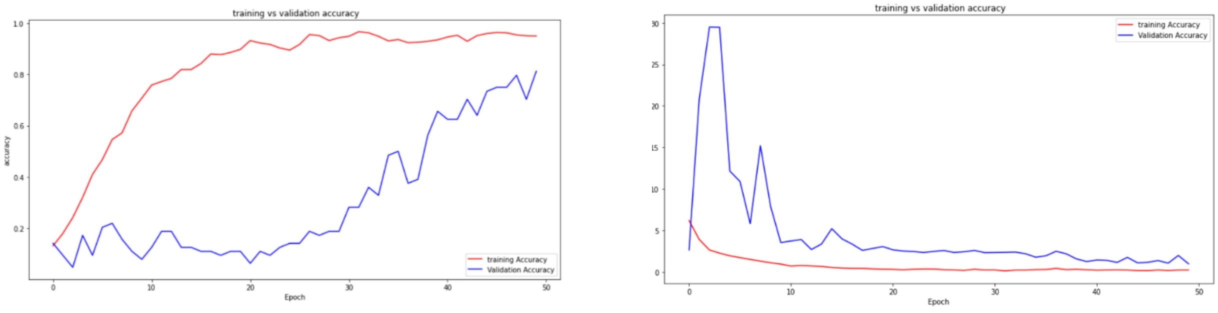

Figure 4- 2: Accuracy and train loss graph for ResNet50 model

The accuracy graph of the ResNet50 model typically displays the training and validation accuracy during the training process. The horizontal axis represents epochs, while the vertical axis represents the accuracy or loss value of the model.

In the initial stage of training, as the model learns to recognize important features and patterns in input data, the accuracy of training often improves rapidly. As shown in Figure 4-2, as the training progresses, the training accuracy has improved and stabilized, while the validation accuracy has increased rapidly from a slow start to 30 epochs. The loss value of training continues to decrease until it stabilizes.

Ideally, the goal is to achieve a high level of accuracy on the training and validation dataset, while minimizing the loss on the training and validation dataset. Among them, the possible factors that may cause large fluctuations in data are overfitting or parameter settings.

This model structure consists of ResNet50 and several fully connected layers, and is a classification model. As shown in Table 4-3, the ResNet50 layer serves as the main body of the model, receiving 224x224x3 input images and outputting a tensor of size (None, 7, 7, 2048). The Flatten layer flattens the output of the ResNet50 layer to a vector of (None, 100352). The Dense layer is a fully connected layer with 2048 neurons, used to convert the output of ResNet50 into higher dimensional feature representations. The Dropout layer randomly disconnects neurons in the Dense layer with a certain probability to prevent overfitting.

Layer (type)	Output Shape	Param #
resnet50 (Functional)	(None, 7, 7, 2048)	23587712
flatten (Flatten)	(None, 100352)	0
dense (Dense)	(None, 2048)	205522944
dropout (Dropout)	(None, 2048)	0
dense_1 (Dense)	(None, 256)	524544
dropout_1 (Dropout)	(None, 256)	0
dense_2 (Dense)	(None, 64)	16448
dropout_2 (Dropout)	(None, 64)	0
dense_3 (Dense)	(None, 10)	650
Total params: 229,652,298 Trainable params: 229,599,178 Non-trainable params: 53,120		

Table 4- 3: ResNet50 model

4.4 Evaluations of Xception model

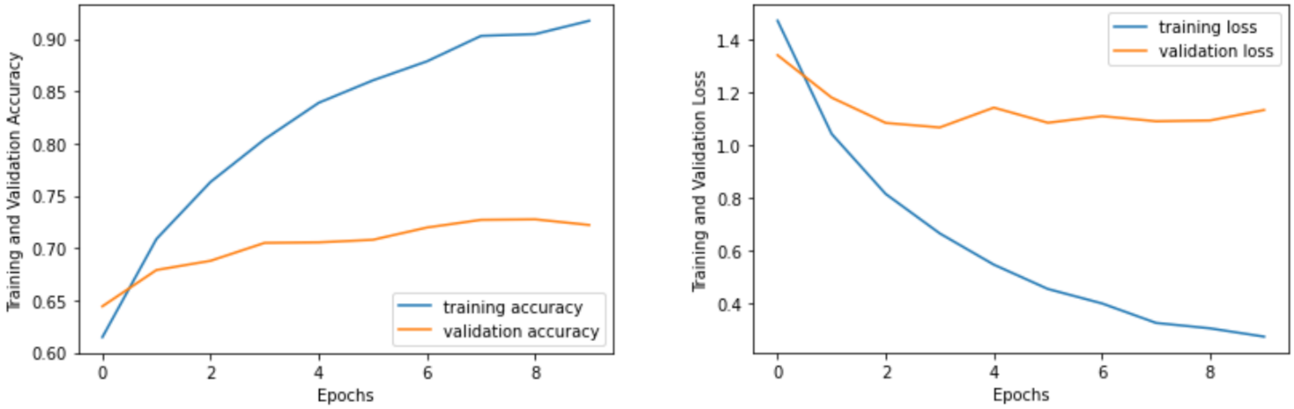

Figure 4- 3: Accuracy and train loss graph for Xception model

In the initial stage of training, the loss value of the Xception model rapidly decreases, and the accuracy value often increases. As the training progresses, the loss value may continue to decrease, but the speed will be slower. The accuracy value can also continue to increase.

The difference between the accuracy and loss values of validation and the accuracy and loss values of training is shown in Figure 4-3. The accuracy of training can reach 90%. However, the accuracy of validation can only reach 70%.

The Xception structure is a linear stack of deeply separable convolutional layers with residual connections. The Xception model has 14 layers of blocks. As shown in Tables 4-4.

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
Block1_conv1 (Conv2D)	(None, 111, 111, 32)	864
block1_conv1_bn (BatchNormalization)	(None, 111, 111, 32)	128
block1_conv1_act (Activation)	(None, 111, 111, 32)	0
block1_conv2 (Conv2D)	(None, 109, 109, 64)	18432

block1_conv1_bn (BatchNormalization)	(None, 109, 109, 64)	256
block1_conv2_act (Activation)	(None, 109, 109, 64)	0
block2_sepconv1 (SeparableConv2D)	(None, 109, 109, 128)	8768
block2_sepconv1_bn (BatchNormalization)	(None, 109, 109, 128)	512
block2_sepconv2_act (Activation	(None, 109, 109, 128)	0
block2_sepconv2(SeparableConv2D)	(None, 109, 109, 128)	8768
block2_sepconv1_bn (BatchNormalization)	(None, 109, 109, 128)	512
block2_sepconv2_act (Activation	(None, 109, 109, 128)	0
block2_sepconv2 (SeparableConv2D)	(None, 109, 109, 128)	17536
block2_sepconv2_bn (BatchNormalization)	(None, 109, 109, 128)	512
conv2d (Conv2D)	(None, 55, 55, 128)	8192
block2_pool (MaxPooling2D)	(None, 55, 55, 128)	0
batch_normalization (BatchNormalization)	(None, 55, 55, 128)	512
add (Add)	(None, 55, 55, 128)	0
block3_sepconv1_act (Activation)	(None, 55, 55, 128)	0
block3_sepconv1 (SeparableConv2D)	(None, 55, 55, 256)	33920
block3_sepconv1_bn (BatchNormalization)	(None, 55, 55, 256)	1024
block3_sepconv2_act (Activation)	(None, 55, 55, 256)	0
block3_sepconv2 (SeparableConv2D)	(None, 55, 55, 256)	67840
block3_sepconv2_bn (BatchNormalization)	(None, 55, 55, 256)	1024
conv2d_1 (Conv2D)	(None, 28, 28, 256)	32768
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
batch_normalization_1 (BatchNormalization)	(None, 28, 28, 256)	1024
add_1 (Add)	(None, 28, 28, 256)	0
block4_sepconv1_act (Activation)	(None, 28, 28, 256)	0
……	……	……
block14_sepconv2_act (Activation)	(None, 7, 7, 2048)	0
Total params: 20,861,480		

```
Trainable params: 20,806,952
Non-trainable params: 54,528
```

Table 4- 4: Xception model

4.5 Comparision of the three models

Generally speaking, it can be concluded that the weight levels of convolution have been well trained and the dog image dataset used for classification can be considered a promising result. To make it more straightforward to see which of the three models' training results is more in line with expectations. Looking back, three different models were considered for comparison: accuracy, loss value, recall, and precision were compared for each model. As shown in Tables 4-5.

Model	Train accuracy	Val accuracy	Train loss	Val loss	precision	recall
VGG16	0.9917	0.9917	0.053	0.049	0.92	0.85
ResNet50	0.9503	0.8125	0.2062	0.9631	0.9660	0.9409
Xception	0.9173	0.7221	0.2724	1.1339	0.84	0.89

Table 4- 5: VGG16, ResNet50, Xception: loss, accuracy, precision, recall

4.6 Comparison Analysis with Other State-of-the-art models

The most advanced breed classification model classifies the Stanford 120 breed dataset. Inception v3 is widely used in image classification tasks. It has achieved impressive results in dog breed classification, with an accuracy of over 90% on the Stanford Canine Dataset. ResNet-50 is another CNN architecture that has been widely used for image classification tasks. It has an accuracy of over 93% on the Stanford Dog dataset. Xception is a deep neural network architecture developed by Google that uses deep separable convolutions to reduce the number of parameters in the model. VGG-16 has achieved impressive results in several image classification tasks. It has an accuracy of over 93% on the Stanford Dog dataset. The accuracy comparison between the model of this project and other methods is shown in Table 4-6.

Methods	Model	Resullts
Uno et al. [23]	VGG-16	Accuracy = 84.8%
Wang et al. [24]	VGG-16	Accuracy = 97%
Kumar et al. [25]	ResNet50	Accuracy = 87.89%
Lee et al. [27]	Inception-ResNet v2 and Xception	Accuracy = 94.49%
This thesis work	VGG16, ResNet50, Xception	Accuracy=99.17%, 95.03%, 91.73%

Table 4- 6 : Results of some state-of -the-art methods

Chapter 5 Professional Issues

5.1 Project Management

5.1.1 Activities

To manage projects, the goal is to organize and complement project development. Here is a detailed description of the steps involved in the project:

a. Learn deep learning and understand the basic knowledge about convolutional neural networks.

b. Select the appropriate data set for the project. The data set image is preprocessed. Can help unify all the images into the same size. The goal is to help achieve better accuracy.

c. Search the literature on historical models and select the appropriate model for this project.

d. Classify data sets into training sets and verification sets. Make sure you don't overfit.

e. Read the image into the program. Read the image file stored in the folder and store it as a tensor with a high width channel shape.

f. Implement the model in code. The model chosen should fit the data set.

g. Improve the data, adjust the parameters and enhance the data to improve the accuracy of the model.

h. Then prepare a progress report documenting the progress made during the project. The report should detail the steps taken, the challenges encountered and the solutions implemented to overcome them.

i. After completing the progress report, complete the code so that the performance of the model can be effectively evaluated. Performance can be evaluated with appropriate parameters.

j. Fine-tuning can be done to specific projects for unsatisfactory results. After the transfer learning process is complete, the model is trained to correct errors in the data set.

5.1.2 Schedule

Task	Start Date	Duration	End Date
Learning the basic of Deep Learning and choose the dataset	2022/10/25	7	2022/11/1
Understand the CNN model and search the literatures	2022/11/2	5	2022/11/7
Prepare Project Proposal and finish it	2022/11/5	6	2022/11/11
Learning Deep Learning Framework and prepare software	2022/11/6	8	2022/11/14
System Design and prepare	2022/11/10	6	2022/11/16
Implementation	2022/11/15	87	2023/2/10
Literature Review	2022/10/18	50	2022/12/7
Prepare Progress Report	2022/11/19	56	2023/1/13
Testing & Evaluation	2023/1/1	78	2023/3/20
Write Final Report	2023/1/25	54	**2023/5/5**
Increasing the words of final report between 8000 - 10000 words	2023/3/10	10	**2023/3/20**
Perfecting the GUI interface	2023/3/14	6	2023/3/20
Create Poster and PPT design for Project	2023/3/21	13	2023/3/27

Table 5- 1 : The schedule of project

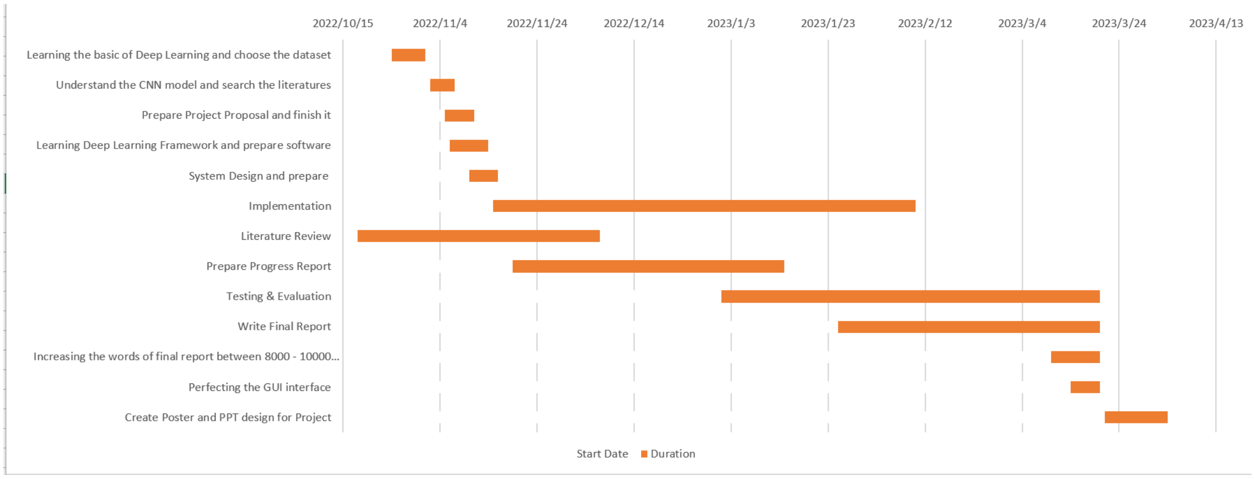

Figure 5- 1: Gantt

5.1.3 Project Data Management

1. It is necessary to maintain version control of modified tasks throughout the project. This includes periodically storing modified tasks in dedicated folders to track progress and ensure that changes are accurately recorded.

2. The project source code should be managed effectively to ensure that it is easily accessible and retrieved. This can be done by recording the source code in a dedicated folder on a version control platform such as giitee or GitHub. This enables multiple team members to collaborate on the project and track changes made to the code over time.

3. During the project, it is critical to ensure that data used for training and validation is properly organized and labeled. This includes checking for any missing or incorrect labels and correcting them promptly to ensure that the model has been trained correctly.

4. Finally, it is critical to document the entire project, including the steps taken, the challenges encountered, and the solutions implemented to overcome them. This document should be comprehensive and easily accessible to ensure that the findings and conclusions of the project can be effectively shared with stakeholders and other interested parties.

5.1.4 Project Deliverables

This will show what the project needs to deliver.

1. Finish the project proposal
2. Finish the progress report
3. Final report.
4. Make source code to the zip file to upload github.
5. Poster and PPT for dog breed classification.

5.2 Risk Analysis

Effective risk management is critical to the success of this project. A comprehensive risk analysis is an important step in identifying and addressing potential threats to a project.

During the risk analysis process, consider a variety of factors that may negatively affect the project. These factors can be technical problems, such as hardware or software failures, or external factors, such as changes in government regulations or unforeseen market conditions.

Regular monitoring of potential risks and their impact on the project is essential to ensure that the project is on schedule and completed smoothly. By actively managing potential risks, project teams can optimize project efficiency and performance and ensure that project objectives are met.

Potential risks	potential	Severity	Likelihood	Risk	mitigation
Bad program	programming is too complicated.	5	1	5	It can be done through study and practice. In addition, use efficient and professional programming techniques to solve,
The final test results is not good	The program is lack of testing.	4	2	8	This can be solved by testing the code more thoroughly.
The model gets terrible accuracy when run it	Data sets cannot be trained well in this model.	2	3	6	Data enhancement is performed on the data set. Resize the dataset and add more data to the dataset.
Date loss	Data is not processed	3	3	9	This can be solved by

	and saved properly.				ensuring that the program backs up data in the event of a sudden loss.
Miss the deadline	Poor GPU system is used in training model	4	3	12	Use a better configuration training model or a different GPU system.

Table 5- 2: Risk analysis

5.3 Professional Issues

This section covers the background to product development for professional, social, ethical, safety and legal issues, as well as content related to finished products.

Legal issues:

When using copyrighted or owned data or images, the project must ensure that it obtains the necessary licenses and permits to use these data or images, or uses publicly available virtual data, in order to avoid any potential legal issues related to intellectual property infringement. When collecting and processing personal information of dog owners or their pets, the use of data must also comply with applicable data privacy laws, such as GDPR or CCPA.

Social issues:

Most importantly, this model may perpetuate bias and discrimination based on dog breeds or races, leading to unfair treatment or stereotypes. To alleviate this situation, teams can consider incorporating diversity and inclusivity principles into their models and data collection processes. In addition, the project team can strive to represent as many dog breeds as possible in their dataset to avoid underrepresentation of certain breeds.

Moral issues:

The team must ensure that the data collection process does not harm the welfare of the dog, nor does it cause excessive stress or harm. In addition, it is crucial to maintain transparency in the limitations of the model and how to collect data to ensure that users fully understand the situation.

Environmental issues:

The project team should consider implementing sustainable practices to reduce the generation of resource waste. For example, projects can use renewable energy or recycle hardware and electronic devices used in the project.

Professional Code of Conduct:

The project team must comply with relevant professional code of conduct, such as ACM ethics and professional norms. These guidelines provide guidance for ethical and responsible behavior when developing artificial intelligence models, including issues such as bias, privacy, transparency, and accountability.

Chapter 6 Conclusion

This paper presents a design and implementation method of dog breed classification based on convolutional neural network. Here is a summary of all the work done and a plan for future work.

The main work is as follows. First, based on convolutional neural network, a dog breed classification project with accuracy of no less than 70% for 120 common dogs is designed and implemented with training set and validation set, which is a relatively mature convolutional neural network application. Compared with other dog breed classification systems using convolutional neural networks, this system reuses three existing models VGG-16, ResNet50, Xception and common features by using transfer learning methods, and compares the efficiency and accuracy of dog breed classification. Reduce the time cost, data resource cost and hardware resource cost of the algorithm. Second, combined with the characteristics of GPU parallel computing, the GPU utilization rate of back-end applications of the system is optimized, so that a single server can realize high-quality services and save the extra equipment cost, maintenance cost and technical difficulty of building server clusters. Finally, deploy the breed classification project, where deployment involves making the trained model available for real-world applications. This could involve developing a mobile app or web service that allows end users to easily classify dog breeds using their own images. It is important to consider such factors as user experience, scalability, and security during the deployment phase to ensure that the application is reliable and efficient. In addition, ongoing maintenance and updates may be required to ensure that the application stays current with the latest advances in deep learning and breed classification.

VGG-16 is a convolutional neural network (CNN) composed of 16 layers. VGG-16 has achieved high precision in dog classification due to its deep structure and many parameters, but there may be overfitting problem. ResNet50 is a CNN architecture that uses skip connections to solve the problem of gradient vanishing, which may occur in very deep neural networks. ResNet50 performs better than vgg-16 in variety classification tasks. Its performance is usually more robust, and it is not easy to overfitting. Xception can reduce the number of parameters and computational complexity of the network. The

Xception model trained in the initial stage performs better than VGG-16 and ResNet50 in variety classification tasks. Its performance is usually more accurate and efficient. However, after fine-tuning the VGG-16 model, the accuracy of VGG-16 can reach 99.17%. Better than ResNet50 and Xception.While my project had successful results in breed classification using the VGG-16, ResNet50, and Xception models, future work has the potential to further improve the accuracy and efficiency of the classification system. One potential avenue for future work is to explore other pre-training models and architectures beyond those used in my project. This may involve experimenting with different neural network configurations, or combining a number of different models. Further fine-tuning of the model used in my project is also possible. Fine-tuning involves adjusting the parameters of the pre-training model to better fit a particular data set or task. By fine-tuning the model used in my project, it is possible to achieve better performance and accuracy in breed classification.

In general, deep learning model for dog breed classification still has a lot of research and development space. By constantly exploring new techniques and methods, it is possible to improve the accuracy and efficiency of classification systems and further advance the field.

References

[1] P. Borwarnginn, K. Thongkanchorn, S. Kanchanapreechakorn, and W. Kusakunniran, “Breakthrough conventional based approach for dog breed classification using CNN with transfer learning,” 2019 11th International Conference on Information Technology and Electrical Engineering (ICITEE), 2019.

[2] FCI International Championship. [Online]. Available: https://www.fci.be/en/FCI-International-Championship-41.html. [Accessed: 08-Nov-2022].

[3] J. Liu, A. Kanazawa, D. Jacobs, and P. Belhumeur, “Dog breed classification using part localization,” Computer Vision – ECCV 2012, pp. 172–185, 2012.

[4] X. Wang, V. Ly, S. Sorensen, and C. Kambhamettu, “Dog breed classification via landmarks,” 2014 IEEE International Conference on Image Processing (ICIP), 2014.

[5] Parkhi O M, Vedaldi A, Zisserman A, et al. Cats and dogs[C]// Computer Vision and Pattern Classification. IEEE, 2012:3498-3505.

[6] Liu J, Kanazawa A, Jacobs D, et al. Dog Breed Classification Using Part Localization[C]// European Conference on Computer Vision. Springer Berlin Heidelberg, 2012:172-185.

[7] Wang X, Ly V, Sorensen S, et al. Dog breed classification via landmarks[C]// IEEE

International Conference on Image Processing. IEEE, 2015:5237-5241.

[8] Z. Raduly, C. Sulyok, Z. Vadaszi, and A. Zolde, “Dog breed identification using Deep Learning,” 2018 IEEE 16th International Symposium on Intelligent Systems and Informatics (SISY), Sep. 2018.

[10] K. Lai, X. Y. Tu, S. Yanushkevich. Dog identification using soft biometrics and neural networks. In Proceedings of International Joint Conference on Neural Networks, IEEE, Budapest, Hungary, pp.1–8, 2019. DOI: 10.1109/ IJCNN.2019.8851971.

[11] X. Y. Tu, K. Lai, S. Yanushkevich. Transfer learning on convolutional neural networks for dog identification. In Proceedings of the 9th IEEE International Conference on

Software Engineering and Service Science, IEEE, Beijing, China, pp.357–360, 2018. DOI: 10.1109/ICSESS.2018. 8663718.

[12] Kumar, A. and Kumar, A., 2020, December. Dog breed classifier for facial classification using convolutional neural networks. In 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS) (pp. 508- 513). IEEE.

[13] Jain, R., Singh, A. and Kumar, P., 2020. Dog Breed Classification Using Transfer Learning. ICCII 2018, p.579.

[14] Borwarnginn, P., Kusakunniran, W., Karnjanapreechakorn, S. and Thongkanchorn, K., 2021. Knowing Your Dog Breed: Identifying a Dog Breed with Deep Learning. International Journal of Automation and Computing, 18(1), pp.45-54.

[15] Liu J, Kanazawa A, Jacobs D, et al. Dog breed classification using part localization[C]//European conference on computer vision. Springer, Berlin, Heidelberg, 2012: 172-185.

[16] Dąbrowski, Marek, Tomasz Michalik. How Effective is Transfer Learning Method for Image Classification. Position Papers of the Federated Conference on Computer Science and Information Systems, vol. 12, 3– 9 https://doi.org/10.15439/2017f526 ISSN 2300-5963 ACSIS

[17] Vrbančič, G., Pečnik, Š. and Podgorelec, V., 2020, August. Identification of COVID-19 X-ray Images using CNN with Optimized Tuning of Transfer Learning. In 2020 International Conference on INnovations in Intelligent SysTems and Applications (INISTA) (pp. 1- 8). IEEE.

[18] Chanvichitkul, M., Kumhom, P. and Chamnongthai, K. (2007) "Face classification based dog breed classification using coarse-to-fine concept and PCA," 2007 Asia-Pacific Conference on Communications [Preprint]. Available at: https://doi.org/10.1109/apcc.2007.4433495.

[19] A. Khosla, N. Jayadevaprakash, B. Yao and L. Fei-Fei. "Novel dataset for Fine-Grained Image Categorization". First Workshop on Fine-Grained Visual Categorization (FGVC), IEEE Conference on Computer Vision and Pattern Classification (CVPR), 2011.

[20] G. Chen, J. Yang, H. Jin, E. Shechtman, J. Brandt, and T. Han, “Selective Pooling Vector for Fine-Grained Classification”, Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on. IEEE, 2015.

[21] C. Kanan, “Fine-Grained Object Classification with Gnostic Fields”, Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on. IEEE, 2014.

[22] “Applying transfer learning on your data pycon 2017 Delhi, India.” [Online]. Available: https://www.researchgate.net/publication/325997539_Applying_Transfer_Learning_on_Your_Data_PyCon_2017_Delhi_India. [Accessed: 28-Apr-2023].

[23] M. Uno, X.-H. Han, and Y.-W. Chen, “Comprehensive study of multiple cnns fusion for fine-grained dog breed categorization,” 2018 IEEE International Symposium on Multimedia (ISM), 2018.

[24] Y. Wang, Q. Fu, N. Lin, H. Lan, H. Zhang, and T. Ergesh, “Identification and classification of defects in PE gas pipelines based on VGG16,” Applied Sciences, vol. 12, no. 22, p. 11697, 2022.

[25] R. Kumar, M. Sharma, K. Dhawale, and G. Singal, “Identification of dog breeds using Deep Learning,” 2019 IEEE 9th International Conference on Advanced Computing (IACC), 2019.

[26] I.-H. Wang, Mahardi, K.-C. Lee, and S.-L. Chang, “Predicting the breed of dogs and cats with fine-tuned keras applications,” Intelligent Automation &amp; Soft Computing, vol. 30, no. 3, pp. 995–1005, 2021.

[27] WTF is a convolutional neural network? (no date) Honeybadger Developer Blog. Available at: https://www.honeybadger.io/blog/convolutional-neural-network-cnn-ruby/ (Accessed: May 5, 2023).