Project 2 - Neural Networks
Dog Breeds Classification

Ms. Sorawee Chirarattanawilai 6322771781 | Mr. Ayush Kwatra 6322774645
Mr. Chayanan Rattanaboon 6322772433

*Department of Information, Computer and Communication Technology (ICT), Sirindhorn Institute of Technology (SIIT), Thammasat University,
Klong Luang, Pathum Thani, Thailand*

1. Teammates

Ms. Sorawee Chirarattanawilai	6322771781
Mr. Ayush Kwatra	6322774645
Mr. Chayanan Rattanaboon	6322772433

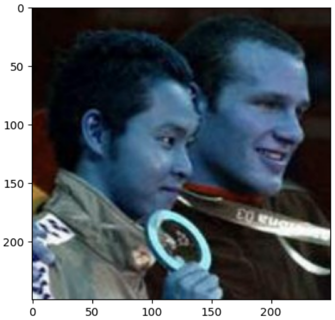
2. Introduction and Data Analysis

The purpose of this project is to be able to detect humans and dogs, and also to be able to classify dog breeds for each dog picture. We have also done dog breed classification for human faces to check which type of dogs they look like.

There are two datasets utilized in this project which are "*dogImages"* and "*lfw"* datasets. The format of the content inside the datasets is an RGB image consisting of the faces of dogs and humans in each dataset. The "*dogImages"* dataset consists of 8,351 dog images from 133 dog categories. We use 6,680 dog images as a training set, 835 dog images as a validation set, and 836 dog images as a test set. In the *"lfw"* dataset, there are a total of 13,233 human images.

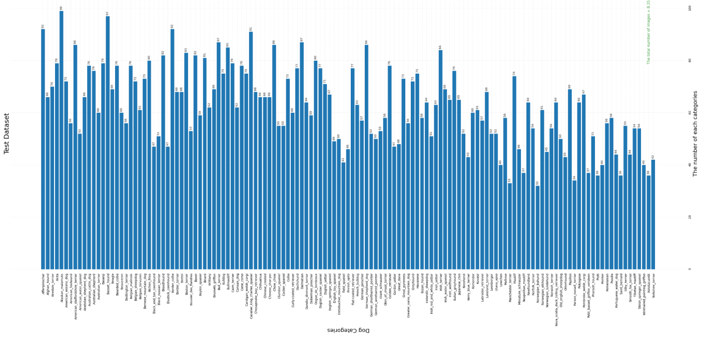
In detecting human faces, we use OpenCV's implementation of Haar feature-based cascade classifiers to detect human faces in images. In detecting dog faces, we use a pre-trained ResNet-50 model to detect dogs in images which includes ResNet-50 model, along with weights that have been trained on ImageNet, a very large dataset used for image classification and other vision tasks. ImageNet contains over 10 million URLs, each linking to an image containing an object from one of 1000 categories. Given an image, this pre-trained ResNet-50 model returns a prediction (derived from the available categories in ImageNet) for the object that is contained in the image.

The quality of human face images in the dataset is very high, about 97% to 99% accurate for detecting a human in an image. One of the problems in the human faces images that leads to an error is that it occurs two faces in the same picture.



The quality of dog face images in the datasets is 100% accurate which means that there are likely no errors in detecting dogs from the images.

We also looked out for biases in this project. We found out that there is a bias in the *"dogImages"* dataset, where the number of images per dog category is not equal to each other. The result is shown below.



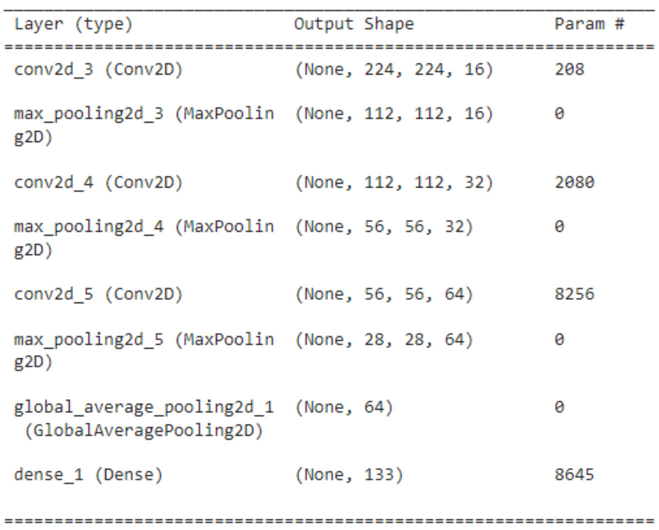
3. Algorithm Design

a. Data preprocessing

We rescale the images' size by dividing every pixel of every image by 255 in order to normalize the range of the image size to be 0 to 1 which is a common range to prepare image data for the machine learning models.

b. CNN Architecture

In our CNN architecture, it has a total of 8 layers consisting of 3 convolutional layers, 3 max-pooling layers, 1 global average pooling layer, and 1 dense layer as an output layer. The model architecture summary is provided below.



```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_3 (Conv2D)           (None, 224, 224, 16)      208

 max_pooling2d_3 (MaxPoolin  (None, 112, 112, 16)      0
 g2D)

 conv2d_4 (Conv2D)           (None, 112, 112, 32)      2080

 max_pooling2d_4 (MaxPoolin  (None, 56, 56, 32)        0
 g2D)

 conv2d_5 (Conv2D)           (None, 56, 56, 64)        8256

 max_pooling2d_5 (MaxPoolin  (None, 28, 28, 64)        0
 g2D)

 global_average_pooling2d_1  (None, 64)                0
  (GlobalAveragePooling2D)

 dense_1 (Dense)             (None, 133)               8645

=================================================================
```

1. Convolutional Layer #1: The first layer is a convolutional layer with 16 filters and a kernel size of 2x2. "Same" padding is used to make sure that the spatial dimensions of the feature maps are still the same. ReLU is used to show non-linearity. This layer is also an input layer where images are of size (224, 224, 3).
2. Max-Pooling Layer #1: This max-pooling layer has a pool size of 2x2. It reduces the spatial dimensions by half and extracts the most important information from the feature maps.
3. Convolutional Layer #2: Everything is the same as the first convolutional layer, except that the second convolutional layer has 32 filters.
4. Max-Pooling Layer #2: Another max-pooling layer follows the second convolutional layer with a pool size of 2x2. It will further reduce the image dimension by half.
5. Convolutional Layer #3: In convolutional layer 3, it differs from the previous one in the filters, which have 64 filters.
6. Max-Pooling Layer #3: A third max-pooling layer follows the third convolutional layer with a pool size of 2x2.
7. Global Average Pooling Layer: A Global Average Pooling (GAP) layer is used to calculate the average value of each feature map, reducing the spatial dimensions to 1x1. This is a way to condense the extracted features into a fixed-size vector.
8. Output Layer: The final layer is a Dense layer with 133 units, representing the number of classes in the classification task. Softmax activation is used to output class probabilities.

4. Evaluation

We computed the confusion matrix on the training set. A confusion matrix is a common tool used in the field of machine learning and statistics to assess the performance of a classification algorithm. It provides a summary of the classification results produced by a classification model, allowing you to evaluate the model's performance in terms of true positive(TP), true negative(TN), false positive(FP), and false negative(FN) classifications. The accuracy is a ratio of the sum of true positive(TP) and true negative(TN) to all predictions(TP + TN + FP + FN)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

In this project, we evaluated the accuracy of how accurately it could classify the dog breed to match with the image of the dog. We have used three pre-trained CNN models in this project: VGG16, ResNet-50, and Xception.

VGG16, a convolutional Neural Network architecture, is widely used for image classification due to its deep and uniform design. Comprising 16 layers, it utilizes 3x3 convolutional filters and 2x2 max-pooling layers uniformly throughout the network. VGG16's depth enables it to capture intricate image features, making it effective for various computer vision tasks. Additionally, pre-trained VGG16 models, trained on large datasets like ImageNet, are available for transfer learning, allowing users to fine-tune the model for specific tasks, saving time and resources. While newer CNN architectures have surpassed VGG16 in terms of performance, it remains a valuable baseline model for its simplicity, ease of use, and widespread availability in deep learning libraries. The drawback of VGG16 is that it’s slow to train the model and the network weights are large due to its depth and number of fully-connected nodes. In this project, the accuracy that we got from using VGG16 is 71.89%.

ResNet-50 is a convolutional neural network architecture primarily used for image classification and object detection. It is better than VGG16 in many ways because of its deeper architecture with residual connections that enable the training of very deep networks more effectively, resulting in improved accuracy. Additionally, ResNet-50 is more computationally efficient, making it a practical choice for real-world applications. Pretrained ResNet-50 models are available for transfer learning, and their scalability allows for the creation of even deeper networks while maintaining performance stability. The accuracy of dog breed classification using ResNet-50 is 82.65%, which is about 11% higher than VGG16.

Xception is a convolutional neural network architecture known for its efficiency and effectiveness in image classification and computer vision tasks. It distinguishes itself from VGG16 and even ResNet50 by employing depthwise separable convolutions, significantly reducing computational demands while still capturing complex image features. This unique architecture allows Xception to outperform VGG16 in terms of both efficiency and accuracy, making it an excellent choice for a wide range of applications. Compared to ResNet50, Xception focuses on reducing computation through depthwise separable convolutions, while ResNet50's strength lies in addressing the vanishing gradient problems with residual connections. The choice between Xception and ResNet50 ultimately depends on the specific task, available resources, and the balance between computational efficiency and performance requirements. The accuracy for classifying dog breeds with dog faces is the highest among the other models with 84.81%.

In our model, without using any pre-trained models, we obtained an accuracy of 1.55% after 5 epochs. The reason we got very low accuracy is that our CNN model is not fully optimized compared to other benchmark models that have solved optimization problems and are ready to be used in real-world applications.

CNN Models	**Accuracy (%)**
Our CNN	1.55
VGG16	71.89
RestNet-50	82.65
Xception	84.81

From this experiment, we recommend the use of transfer learning with Xception for a dog breed classification as it has the highest accuracy among all other models.

5. Conclusion

This study's experiment aims to study the characteristics of Haar feature-based cascade classifiers and evaluate algorithm performances. First, OpenCV's implementation of Haar feature-based cascade classifiers to detect human faces in images, the evaluated performance is more than 97 percent which is a high performance. However, we can see that the most false detection occurs from pictures with an unclear face or has more than one person in a picture. The method to fix this problem is that we are able to use features from the face to help the classifier work better.

In this work, we use 3 pre-trained models, which include VGG16, Resnet-50, and Xception. The accuracy we obtain from the three benchmarks varies. VGG16 has the lowest accuracy with 71.89%. Resnet-50 follows with 82.65% and Xception with the highest accuracy, 84.81%. All of them have much higher accuracy than our CNN model, which has just 1.55% accuracy. To improve this algorithm, we can improve on classification of other animals such as cats. Another improvement could be increasing the training data by including more pictures of the dog breed and developing the preprocessing step.

6. Reference

Sawnhey, P. (2020, May 22). Dog Breed Classification Using Flask.

B. Valarmathi, N. Srinivasa Gupta, G. Prakash, R. Hemadri Reddy, S. Saravanan, P. Shanmugasundaram (2023, July 24) Hybrid Deep Learning Algorithms for Dog Breed Identification—A Comparative Analysis. https://ieeexplore.ieee.org/abstract/document/10192536