An Intelligent System for Dog Breed Identification

*Manish Choubey[1], Purushottam Kumar[2], Raunit Anand[3], Abhimanyu Singh[4], Shaktish Prajapati[5], and Moin Hasan[6.]

[1]Lovely Professional University, India
Choubey.m26@gmail.com
[2]Lovely Professional University, India
Raunitanandlpu@gmail.com

* Corresponding Author

Abstract: Dog breed identification is very useful technique in various factors, to get the knowledge about an individual dog's conditions, health condition, Visual interaction behaviour, and natural profession. In this paper we present a clear solution for the identification of all dog breeds by using pictures of their face. The method introduced here uses a deep learning-based perspective in order to distinguish the breeds of dogs. The process starts with a transfer educate by training a pre-trained convolutional neural networks (CNNs) on the available dogs' dataset. Then, the image with various settings is also applied on the training dataset, keeping in mind to increase the classification performance. This method is judged using three different CNNs with different types of augmentation settings and detailed experimental comparisons. The model we have introduced achieves more precision than others with the published dataset with many available dog breeds.

1 Introduction

Everyone says that dogs are men's best friend. They are now becoming important part of one's daily life. Also, they are now considered as the family member in a house. From morning walk to evening walk, shopping to shows, dogs are with human beings like their best friend. But sometimes it is difficult to recognize the breed of the dog as many dogs of different breeds looks somewhat similar to each other. And it is more difficult for us to make computers understand the same. It can be a great help for humans when computer and easily and accurately identify the breed of the dogs. Imagine a situation where a dog has met an accident and its breed needs to be identified and start the treatment as soon as possible to avoid any mis happenings. In this case if an electronic device is able to identify the breed, it can help the doctors to start the treatment as soon as possible, or we can take any other application where we need to identify the breed of dogs. In this project we have come up with a model to identify the dog breeds.

The aim of this project is to classify dog breeds from photographs or simply by the face of dogs. This is a fine-grained and well-defined classification issue: all Canis lupus breeds intimate share similar body features and overall structure, so identification and finding difference between breeds becomes a very huge problem. In addition, there is little inter-breed variation and a lot of intra-breed variation; in other words, there are little variations between breeds and a lot of differences within breeds, with colour, shape, and size differences. This dilemma is not only challenging, but it also has implications for other fine-grained classification issues. The methods used to solve this problem, for example, area fine-grained classification problem can be solved using any number of groups with relatively little variance within them. In the natural world, an identifier like this may be used in biodiversity surveys, saving time and money for scientists doing research on the health and abundance of specific plant populations.

2 Related Work

This section explores various research works carried out on Dog Breed Identification.

- In 2018, Wenting Shi, et al. [] proposed "Image Recognition" technique in which they used the data set from Kaggle, applied data argumentation to pre-processing the data set to increase the training data and multi-method (Dropout, weight Decay) to avoid overfitting. This model used the concept of CNN, ResNet18, VGG16, DenseNet161, and AlexNet classification network. To evaluate the performance, they used SoftMax and Cross-Entropy. The overall accuracy of this project was more than 85.14%. Some challenges of this project are the quantity of the data set is inadequate and less accurate.
- In 2018, Xinyuan Tu, Lai and Yanushkevich created a model of transfer Learning using Columbia Dataset and Stanford Dogs dataset, in which they investigated how this model can help improve identification of dogs. Applied CNN coarse-to-fine approach and focused on transfer learning. And finally, both coarse and fine stage were applied with transfer learning with GoogleNet. However, the model was not giving much accuracy with some dog breeds and could not cross the accuracy more than 83.94%.
- In 2018, Fang Wu, et al. [] developed an iOS mobile application for detection of Dog breeds using deep learning. Which was trained using 120 different breeds of dogs. Normalization of size of the images by clipping it is also supported. In order to catch the features and info from the images through CNN, transfer leaning with fine tuning was applied to the dog breed at lower rate. The final accuracy for 50 classes of images was 85% and 63% with 120 classes of images. However, there were some issues with devices with different screen sizes.

- In 2019, Aydin Ayanzadeh and Vahidnia created a project for Dog Breed Identification using Deep Neural Networks using Stanford Dog Breed dataset. In this they have leveraged and modified the state of art models on ImageNet using a pretrained model to extract the edges from the features in Dog Breed dataset. They used DenseNet-121, DenseNet-169, ResNet-50, and GoogleNet to compare the performance of the ImageNet and found the accuracy of 89.66% on ResNet50, 85.37% on DenseNet-121, 84.01% on DenseNet-169 and 82.08 on GoogLeNet showed the better performance to the method proposed in the previous works but accuracy remained below 90%.

- In 2019, Punyanuch Borwarnginn, at al. [] used Columbia Dataset and created a model through CNN using two image processing approaches. For first approach they used conventional approach of Local Binary Pattern and Oriented Gradient. And for second process they used Deep learning by using tutored CNN. It carried accuracy of 96.75% as compared to that of HOG having 79.25% rightness. However, accuracy of the CNN model can be improved by training more dogs of the same breed to the model.
- In 2019, Hsu, David did a project to classify dog breeds using CNN by using tow network architecture i.e. LeNet and GoogLeNet. In the GoogLeNet architecture “inception layers” were included. Used open software package Caffe for learning framework. They found the highest accuracy for different layers of LeNet and GoogLeNet as: 8.9% and 9.5%. Lacking accuracy in recognizing face, eye and ear which could be increased by part localization using different filters.
- In 2020, Sneha I. Kadari et al. [] created a project with web interface for Dog Breed Prediction using CNN and web development tools. In the whole process they chose an image for input , performed object recognition to connect Tensorflow containing 1000 images. Then input image was given as input parameter to ReadImage()to convert the image into pixels. Then to convert the pixels into tensor StepInput() was executed and finally tensor the image as input parameter is passed as input which predicted the output based on the probability. The model was not much accurate because of lesser number of training images.
- In 2020, Zalan Raduly, et al. [] proposed "Image Recognition" technique using Machine learning, CNN concepts, they took the data set of Stanford. They used fine tuning of images and they have also used the mobile application called "Sniff" which determines the breed of the dog using its image eve`n when there is no internet connection. Had rightness of more than 90%, but is quite less when it comes to determining some breeds of dogs.
- In 2020, Punyanuch Borwarnginn, et al. [] applied a deep learning to identify breed of dogs. They started the process with a transfer learning by retraining a pre-trained dataset. Then, they did image augmentation on the training dataset, for improving classification performance. The suggested model's accuracy was 89.92 percent. But the model remained with a challenge for them to improve the classification.

- In 2020, Ding-Nan Zou, et al. [] developed an image dataset for classification of dog breeds named, 'The Tsinghua Dogs Dataset'. They used CUB200-2011 fine-grained technique and had trained the dataset by using PMG, TBMSL-Net & WS-DAN. The breed classification model tested on Tsinghua Dogs dataset achieved an accuracy of 82.65% which was greater than that of

  accurcy of Stanford Dogs dataset which achieved 58.14%. However, this research is mainly focused on breeds found in china and might not perform in another region.
- In 2012, Jiongxin Liu et al. [] found an idea about a solution for identifying a dog with their face images. The method has been used in this paper is deep learning. Created a labelled dataset of 8,351 real-world photographs of 133 dog breeds recognised by the American Kennel Club (AKC). we also create mirror image of the dataset, npm so its quantity doubled.Accuracy they got is 67% which not that much good for machine learning terms. Their accuracy was decreased due to low accuracy, object stability, lower number of images, and their data only detect American dogs.

- In a paper, Voith, Ingram, Mitsouras, and Irizarry compared visual breed identifications of 20 mixed-breed dogs to DNA analysis. The Mars Veterinary Wisdom Panel MXTM was developed by Mars used to test blood samples sent to the MARS VETERINARYTM laboratory. The A statistical model that infers breed from a pattern of 300 genetic markers was used to investigate the accuracy of breed. Although they don't have enough data to improve accuracy.

- Olson, Levy, et al. [] visual breed identification's ubiquity among experts and validity as compared to DNA analysis were investigated in a report. Blood samples from the dogs were sent to MARS Veterinary, Wisdom Panel Canine Genetic Study forDNA identification. The A DNA test found that 21% of dogs had a mutation "pit-bull type" heritageIn Overall, even with a limited goal for identification, the overall validity did not achieve a satisfactory standard. Although they have to work on different bread also, they work on limited data set.
- Kenneth Lai, et al. [] conducted an Image separation method follow a standard pipe where a collection of features removed from the image and given a split. They built a database of 8,351 new wide-angle images. Designed a complete concept app It was made available as a free iPhone app. This approach recognizes the dog's face automatically, aligns the face and removes that greyscale SIFT features and histogram color and predicts the type of the breed. Although their data-set is limited which effect their accuracy.
- In 2015, K.R. Olson, et al. [] developed a model for identification of pit bull type dog. There were 16 breed assessors and 120 dogs in this prospective cross-sectional sample. For breed recognition, the total amount of blood sent a commercial DNA testing facility. Though DNA breed signatures identified only 25 dogs as pit bull type (21%), shelter workers identified 62 dogs as pit

bull type (52%). Although they have to increase data of pit bull from various region and they have to decrease the time consumption.

- Kenneth Lai, et al. [] conducted an investigation of the division animal in the image. They introduced a new pet database with new annotations which includes 37 species of cats and dogs. The model includes the make-up, which is taken up by the appearance, taken by word bag model describing animal fur. Second, they compare two distinctions. They might got good result if they have more data from different places.
- In 2020, Rakesh Kumar, et al. [] have used a fine-grained image cataloguing. To begin, the image is divided into parts several lattices, and the size of the training batch is determined accordingly, after which An algorithm is employed in order to separate and merge the descriptors, as well as image's knowledge about the channel was obtained as the convolutional neural network's data, and finally. Although the Data train on prediction which may decrease accuracy.
- Kenneth Lai, et al. [] conducted investigation on two different possible structures. The upper part of the frame is designed for type separation and performance. Depending on the portion of the selected network, the output of the program is by it can be a predicted dog ownership or a limited breed dog. In this paper, they suggest using coarse-to-fine how to improve the accuracy of predicting identity dog based on the use of "soft" dog-like biometrics give birth.

3 Praposed Framework

3.1 Methods

This fig 3.1 depicts the proposed framework for dog breed classification. It is divided into three phases: data preparation, training, and testing. The data preparation step is needed because we are working with dog face images.

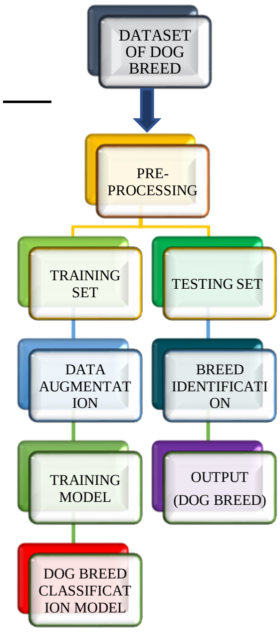

Fig. 1. Overview Flowchart Of Proposed Framework

3.1.1 ResNet18

The term "residual neural network" refers to a network that contains residual information the use of a residual learning framework can make it easier to train networks that are much deeper than those previously used. The output of one layer and the input of the previous layer are combined as the input of the next layer in a residual block.

ResNet18 was used in this case, and it was trained using ImageNet.

$$y = F(x, +Wi) + x \tag{1}$$

$F(x, +Wi)$ is represented as the residual mapping function that we need to train.

3.1.2 AlexNet

AlexNet consists of 5 convolutional layers, 3 full connected layer, few of them are max pooling layers and lastly with a final 1000-way SoftMax. It has 60 million parameter and 650000 neurons. To minimize overfitting in the fully connected layer Dropout is used.
We used AlexNet, which was trained on ImageNet, in this case

$$c^j_{y,z} = b^j_{y,z}/(l + b\sum_{k=\max\left(0,j-\frac{m}{2}\right)}^{\min\left(M-1,J+\frac{M}{2}\right)}((b^j_{y,z})^2))^{\beta} \quad (2)$$

$b^j_{y,z}$ is the output of ReLU of the I'th channel's at (y,z), l, m, b, β are constants.

3.1.3 Data augmentation

Data augmentation is common method for minimizing overfitting on data which is been trained by taking help of different transformations earlier than the feedforward pass while training.
Here: - g(y) represents the function and y represents image.

$$g(y) = P\left(\frac{y}{255.0} - 0.5\right) * 2.0 \quad (3)$$

3.1.4 Stochastic Gradient Descent (SGD)

Stochastic gradient descent is one of the well-known optimization algorithms. The SGD algorithm is a significant reduction in complexity.
Because the gradient of $En(fw)$ is not computed exactly. It estimates this gradient in each iteration using a single randomly chosen example a_u

$$x_{u+1} = x_u - z_u \nabla_x R_{(a_u,x_u)} \quad (4)$$

During each iteration, a random example determines the stochastic process. When we compare it to traditional gradient descent SGD can converge with high probability under technical assumptions hence saving time on computation.
We used SGD with minimum batch size, learning rate schedule, momentum, and weight decay in our task. We then repeat these steps until we get the best results.

3.2 Face Parts Localization

We use the consensus of model's method of to localize portions of the dog's face. The procedure accurately locates the eyes and nose; we work with more complex dog sections during the breed recognition process. To model component locations, low-level detectors are paired with labelled photographs. . For each dog section, we start by training a sliding window SVM detector. J is a query image, pJ is the position of the image's components, and D is the Our objective is to compute detector responses for the sections in J

$$\hat{p}^J = arg \max_{pI} P(p^I \mid D) \tag{5}$$

In exemplars that have been manually labelled dictate possible sites for these components.

These examples assist in the establishment of different types of conditional independence components. pl represent Eq. 5 and the positions of the sections in the l'th exemplar picture becomes

$$\hat{p}^J = arg \max_{p^I} \sum_{l=1}^{n} \int_{u \in U} \prod_{j=1}^{o} P(\Delta p_{l,u}^{(j)} P(p^{(j)J} | D^{(j)}) dt \tag{6}$$

$$\hat{p}^J = arg \max_{p^I} \sum_{l=1}^{n} \int_{u \in U} \prod_{j=1}^{o} P(\Delta p_{l,u}^{(j)} P(p^{(j)J} | D^{(j)}) dt \tag{7}$$

The summation of our case is of all m exemplars, or all labelled examples of dog face components in our context. The integral is over the exemplars' transformations of resemblance t. In the query picture, there is a difference in the location of part I and the exemplar transformed by t in the kth model, is denoted by p(j) l,u, and the part J in the question image and the transformed exemplar are not in the same place is denoted by p(j) l,u. This requires incorporating a part generative model positions, which transforms and places a randomly chosen example in the image, Noise is also there. Following part freedom in variance We exclude the model from the equation.

The optimization is then solved using a RANSAC-style method, in which a large number of exemplars are randomly chosen and u equals the detector output modes. The model part locations are compared with the detector performance for each hypothesis in which a model is converted into an image, and the best fitting fits pool information to form a consensus about the part locations. On the first attempt,

(8)

$$\hat{p}^{(j)J} = arg \max_{p^{(i)I}} \sum_{l,u \in N} P(\Delta p_{l,u}^{(j)} P(p^{(j)J} | D^{(j)})$$

Where ∆p_(l,u)^((j)) is modelled, We sun over the best N between the exemplars and detectors provided by RANSAC as a Gaussian distribution.

3.3 Breed Identıfıcatıon

The face of the dog is the sole focus of our classification algorithm. This is partially because the face is mostly a rigid entity, making it easy to compare photographs of different dogs. We are, however, motivated by the idea that dog breeds should be classified primarily by their looks. The body outline of a dog is not only difficult to recognise and barely visible in pictures, but it also provides little additional information. detail, unless the in the most serious circumstances (e.g., dachshunds).

3.3.1 Classification of Dog Breeds Using Part Localization

Our goal is to compute the dog's breed, denoted by C.

$$\ddot{C} = arg \max_{C} P(C|J) \tag{9}$$

Allow the positions of the parts in the query image to be determined by Allow pJ to specify the component positions in the query image J.
Then

$$\ddot{C} = arg \max_{C} \int P(C|J, p^J) P(p^J|J) dp^J \tag{10}$$

Here, we combine all possible positions of the pJ sections in the picture. If the true positions of the components can be accurately localised, P(pJ |J) is a delta function about them.
Then if we write

$$\hat{p}^J = arg \max_{p^J} P(p^J |J) \tag{11}$$

we have: -

(12)

$$\hat{C} \quad = arg \max_{C} \; P(\,C|\hat{p}^{J})P(\hat{p}^{J}|J)$$

Note that $P(\mathbf{p}J\,|J)$ is not dependent upon C, so that

$$C = arg \max_{C} P\;(C|J,\hat{p}^{J}) \tag{13}$$

As a result, we can divide our issue into 2 segments.
First, as described in the previous section, we must compute $agr\;max_{p^J}\,P(p^J|J)$
After that, we must compute arg $max_C\;P(C|J,\hat{p}^J)$.

Note that: -

$$P(C|J\hat{p}^{J}) = \frac{P(J|C,\hat{p}^{J})P(C|\hat{p}^{J})}{P(J|\hat{p}^{J})} \tag{14}$$

Here the denominator $P(J|\hat{p}^J)$ is a constant that has no bearing on which breed has the best chance of succeeding.
So,

$$\hat{C} = arg \max_{C} P\;(J|C,\hat{p}^{J})P(C|\hat{p}^{J}) \tag{15}$$

Our understanding of what makes up a breed, on the other hand, is based entirely on our collection of labelled exemplar images. The information in these images is divided into two sections. To begin, we'll use the abbreviation pC to denote the known positions of all parts in all breed C exemplars. Then we use DC to denote descriptors that describe the appearance of the breed C exemplars. These descriptors are extracted at the pC-specified component locations.
So Eq: -

$$\hat{C} = arg \max_{C} P(J|D^{C},p^{C},\hat{p}^{J})P(D^{C},p^{C}|\hat{p}^{J}) \tag{16}$$

To approximate, We presume that the breed appearance descriptors DC have a standardised distribution across breeds and are unrelated to their roles.
Hence Eq: -

$$\hat{C} = arg \max_{C} P(J|D^{C},p^{C},\hat{p}^{J})P(p^{C}|\hat{p}^{J}) \tag{17}$$

Hence, we compute

$$P(J|D^C, p^C, \hat{p}^J) \quad (20)$$

by comparing the appearance of the picture at and around the p^Jagreed component locations to the appearance of our exemplars in their corresponding position p^C.

4 Experımental Result & Evaluatıon

The Stanford Dogs dataset is a picture dataset of dog breeds that is available to the public. There are 120 dog groups in total, with a total of 20580 in which 8580 photographs were used for assessment, while 12000 were used for training. The boxes of bounding which contain the dogs in the images are symbolized by annotations over the dataset of images. Within one unified class, the bounding boxes and original pictures are all different sizes, and the scenes are all different types, with occlusion, different poses, different backdrop subjects, and different fur colours.

The first step in the learning process is to convert the raw image files into a functional collection of images for training and testing. The first thing we did was crop all the pictures using the bounding boxes that were annotated. For training and testing purposes, resize all the resulting images to 256x256. We decided against just resizing it by hand because we doubted that filters could measure a squished or strained image in the same way. As a result, we chose to discard all cropped images of one of the two dimensions less than 256 pixels, resize all dimensions until the smaller dimension was 256, and then remove rows and columns [1:256] from the file. We realise that taking repetitive random samples of rows and columns from the image will generate an augmented dataset that can help with training, but we didn't do so for this project. There were 5678 training photos and 4007 research images at the end after this pre-processing, which we converted to LMDB format for use with Caffe.

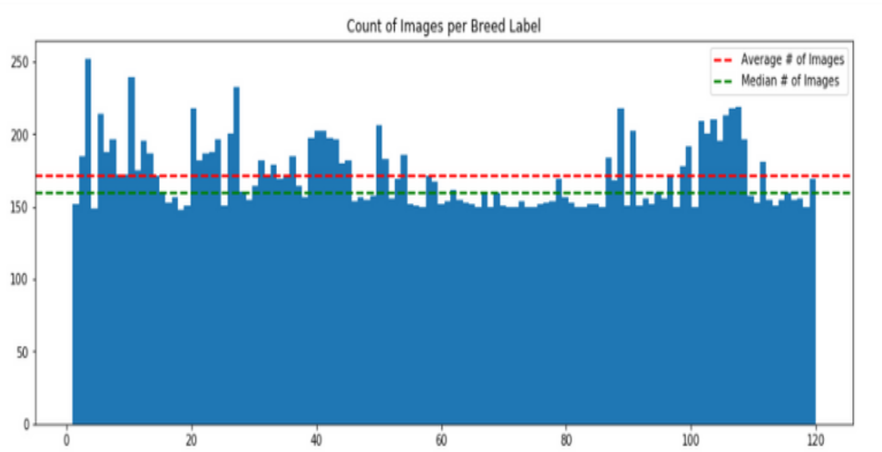

Fig. 2. Image Per Breed

4.1 Loss Analysıs And Accuracy

In our project, we compare 4 different pretrained models: ResNet18, AlexNet, DenseNet161, and VGG16.

We use SoftMax and cross entropy to evaluate their output in terms of training and testingfailure.The softmax function is a generalisation of the logistic function that "squashes" a L dimensional vector Y of arbitrary real values into a L dimensional vector (Y) of real values, each entry in the range (1,0) and all entries adding up to 1. The role is depicted in Fig: -

$$\sigma : S^L \rightarrow \left\{\sigma \,\epsilon\, S^L \middle| \sigma_j > 0,\;\; \textstyle\sum_{j=1}^{L} \sigma_J = 1\right\} \tag{21}$$

$$\sigma(Y)_k = \frac{f^{yk}}{\sum_{l=1}^{L} f^{yk}} \tag{22}$$

The cross-entropy loss is a metric for evaluating the effectiveness of a classification model that produces a probability value between 0 and 1. As the predicted probability varies from the actual label, cross-entropy loss increases. Cross-entropy can be calculated in multiclass classification as follows:

(23)

$$-\sum_d^N z_o, d^{\log(q_o,d)}\text{b}$$

C= accuracy %
C= (CP/TP) *100
CP= No. of Images (correctly predicted)
TP= Total predictions

$$ACCURACY = \frac{NO.OF\ CORRECT\ PREDICITIONS}{TOTAL\ NO.OF\ PREDICITIONS}$$

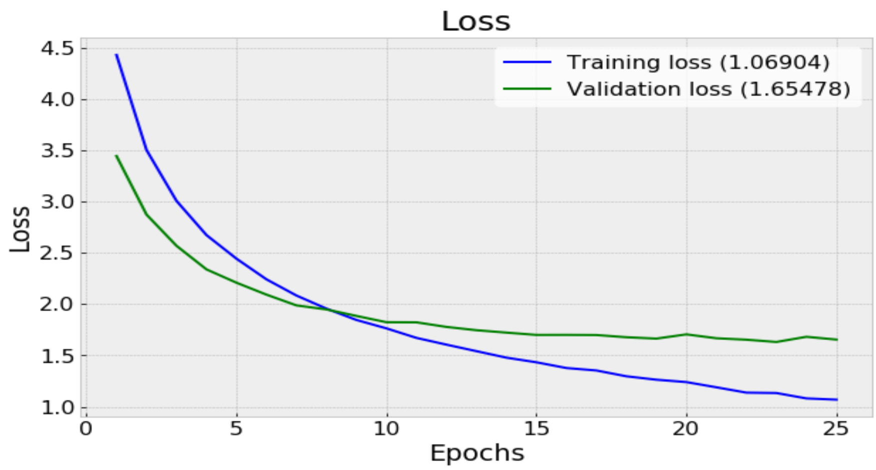

Fig. 3. Graph(Loss)

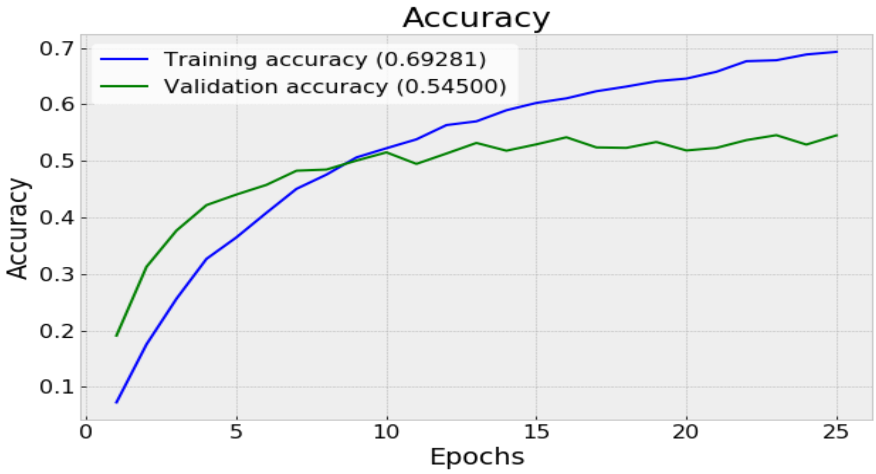

Fig. 4. Graph(Accuracy)

Uploaded Image...
Woof! The model predicted the breed as...papillon!
...with a confidence of 99.50%.

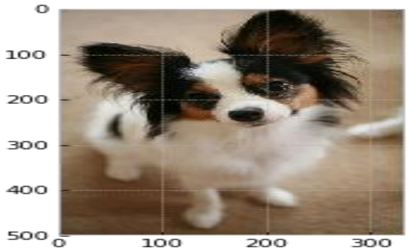

Fig. 5. Forecasting of Papillon

Uploaded Image...
Woof! The model predicted the breed as...West_Highland_white_terrier!
...with a confidence of 54.76%.

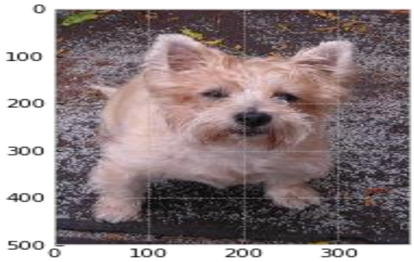

Fig. 6. Forecasting of West Highland White Terrier

Uploaded Image...
Woof! The model predicted the breed as...French_bulldog!
...with a confidence of 97.15%.

Fig. 7. Forecasting of French BullDog

5. Conclusions and Future Scope

In this project we used only 5678 training images and 4007 testing images. As we can see from above used images as examples, the image of the French bulldog is quite clear and easy to identify, because of which the system gives accuracy of 97.15%. Whereas in case of the image of Saluki, the system gives accuracy of only 18.06% because the image of the dog is not clear. From this we can say that more image data can give better result. So rather working on optimization one can use different image resources and add more images for training and testing purposes by labelling them or one can also use NASNet, Xcaption, MobileNet, DenseNet, etc. models for higher accuracy.