Dog breed Identification

Hari Kiran, Sasidhar, Vamsi

Problem Statement and Importance

Our main aim with this project is to identify the breed of the dog given the picture of the dog.

Importance:

Improved accuracy: Automatic identification using deep learning techniques can improve the accuracy of identifying dog breeds compared to human identification, as humans may have biases or lack knowledge about certain dog breeds.

Improved dog breeding: Automatic identification can aid in dog breeding by ensuring the correct breed of dogs are bred together, leading to healthier and more desirable offspring.

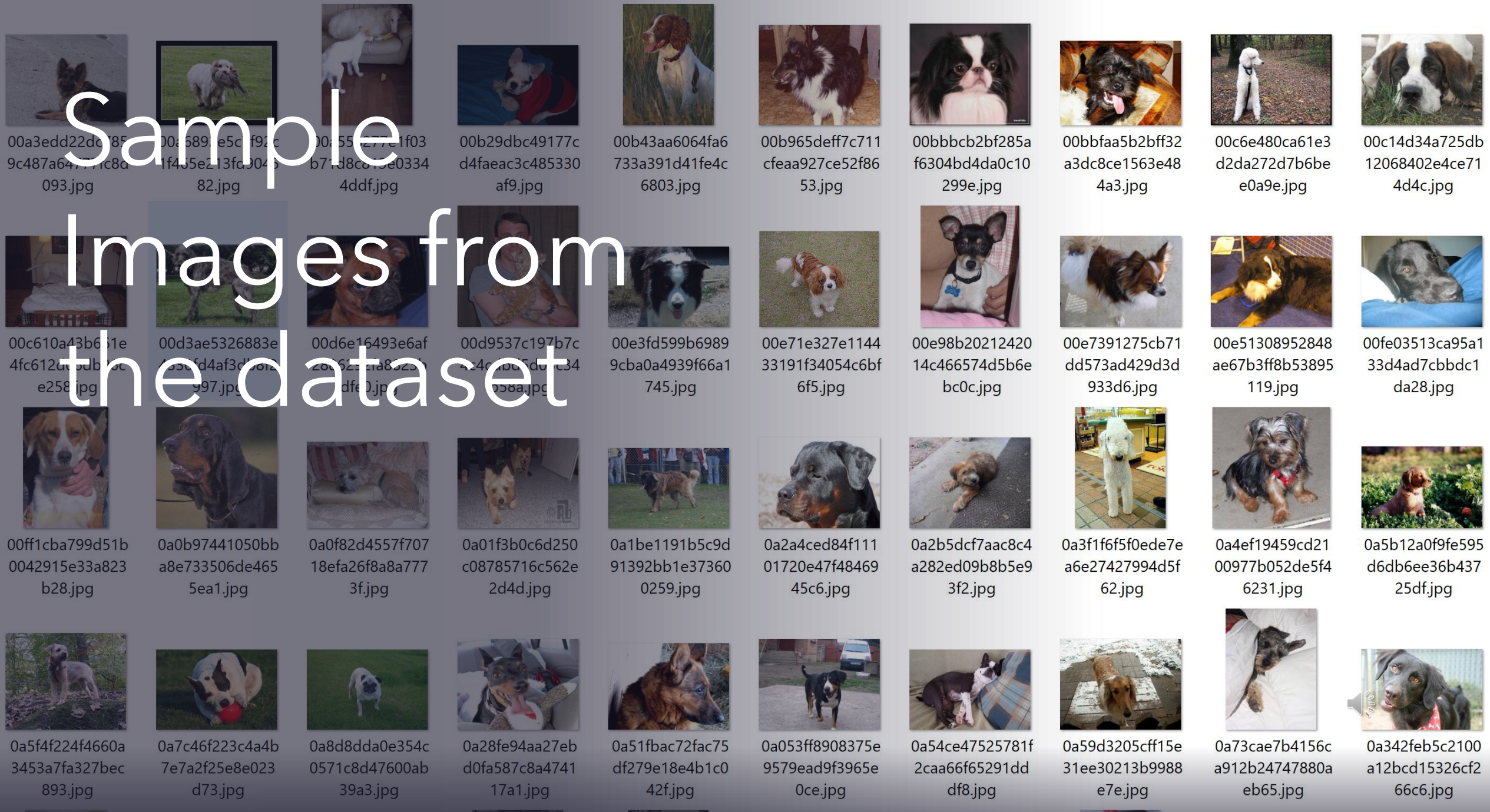
Better dog care: Knowing the breed of a dog can help with understanding its specific needs, such as dietary requirements, exercise needs, and potential health issues, leading to better care for the dog.

Time-saving: With automatic identification, it can save time for veterinarians or dog breeders to identify the breed of a dog quickly, especially in cases where the dog's breed is unknown.

Dataset

- **The dataset used for this project is taken from Kaggle which contains images of dogs belonging to 120 different breeds. The dataset contains a total of 10222 dog images, which are divided into three sets: training set (6,517 images), validation set (1,533 images), and test set (2,1712 images).**
- **Each image is given a unique name and the image name and the dog breed are provided in a separate CSV file. This can help us in determining what the dog breed is of an image.**
- **The dataset presents a challenging computer vision problem due to the high level similarity between some of the dog breeds.**

Sample Images from the dataset



	A	B	C	D
1	id	breed		
2	000bec180eb18c7604dcecc8fe0dba07	boston_bull		
3	001513dfcb2ffafc82cccf4d8bbaba97	dingo		
4	001cdf01b096e06d78e9e5112d419397	pekinese		
5	00214f311d5d2247d5dfe4fe24b2303d	bluetick		
6	0021f9ceb3235effd7fcde7f7538ed62	golden_retriever		
7	002211c81b498ef88e1b40b9abf84e1d	bedlington_terrier		
8	00290d3e1fdd27226ba27a8ce248ce85	bedlington_terrier		
9	002a283a315af96eaea0e28e7163b21b	borzoi		
10	003df8b8a8b05244b1d920bb6cf451f9	basenji		
11	0042188c895a2f14ef64a918ed9c7b64	scottish_deerhound		
12	004396df1acd0f1247b740ca2b14616e	shetland_sheepdog		
13	0067dc3eab0b3c3ef0439477624d85d6	walker_hound		
14	00693b8bc2470375cc744a6391d397ec	maltese_dog		
15	006cc3ddb9dc1bd827479569fcdc52dc	bluetick		
16	0075dc49dab4024d12fafe67074d8a81	norfolk_terrier		
17	00792e341f3c6eb33663e415d0715370	african_hunting_dog		
18	007b5a16db9d9ff9d7ad39982703e429	wire-haired_fox_terrier		
19	007b8a07882822475a4ce6581e70b1f8	redbone		
20	007ff9a78eba2aebb558afea3a51c469	lakeland_terrier		
21	008887054b18ba3c7601792b6a453cc3	boxer		
22	008b1271ed1addaccf93783b39deab45	doberman		
23	008ba178d6dfc1a583617470d19c1673	otterhound		
24	009509be3ca7cce0ff9e37c8b09b1125	otterhound		
25	0097c6242c6f3071762d9f85c3ef1b2f	bedlington_terrier		
26	00a338a92e4e7bf543340dc849230e75	dingo		
27	00a366d4b4a9bbb6c8a63126697b7656	golden_retriever		
28	00a862390341c5be090dd72bd2bc19ef	standard_schnauzer		
29	00b7d114bc5166a629a3cc03d9329120	irish_water_spaniel		
30	00ba244566e36e0af3d979320fd3017f	black-and-tan_coonhound		
31	00bee065dcec471f26394855c5c2f3de	cairn		
32	00ca18751837cd6a22813f8e221f7819	affenpinscher		
33	00cc68a50b2d016a6b29af628ea4e04b	labrador_retriever		

Sample CSV file

Proposed Models

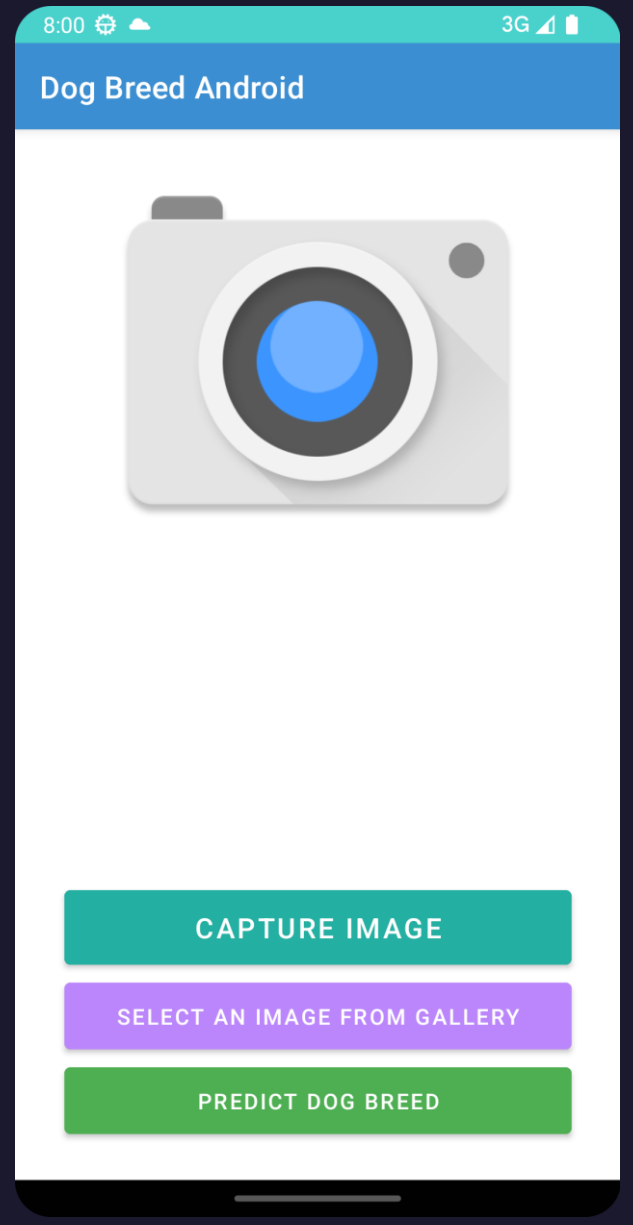
- **We are using multiple CNN architectures which will be commonly used for image classification tasks in computer vision.**
- **These are the architectures we have used:**

1. **Resnet50**
2. **InceptionV3**
3. **EfficientNet**
4. **VGG16**
5. **MyCNN (our own CNN built using Pytorch**

Proposed Models

- **We have fine-tuned the pre-trained models on our dog breed dataset by training the last layer of the model while freezing the rest. The final output layer will have 120 features, each representing a dog breed. We have used the softmax activation function to obtain the probability of each breed and the breed with the highest probability will be predicted as output.**
- **Out of all the models, we have seen that InceptionV3 and EfficientNet produced highest accuracy. So, we saved those models and deployed them to Android Application.**

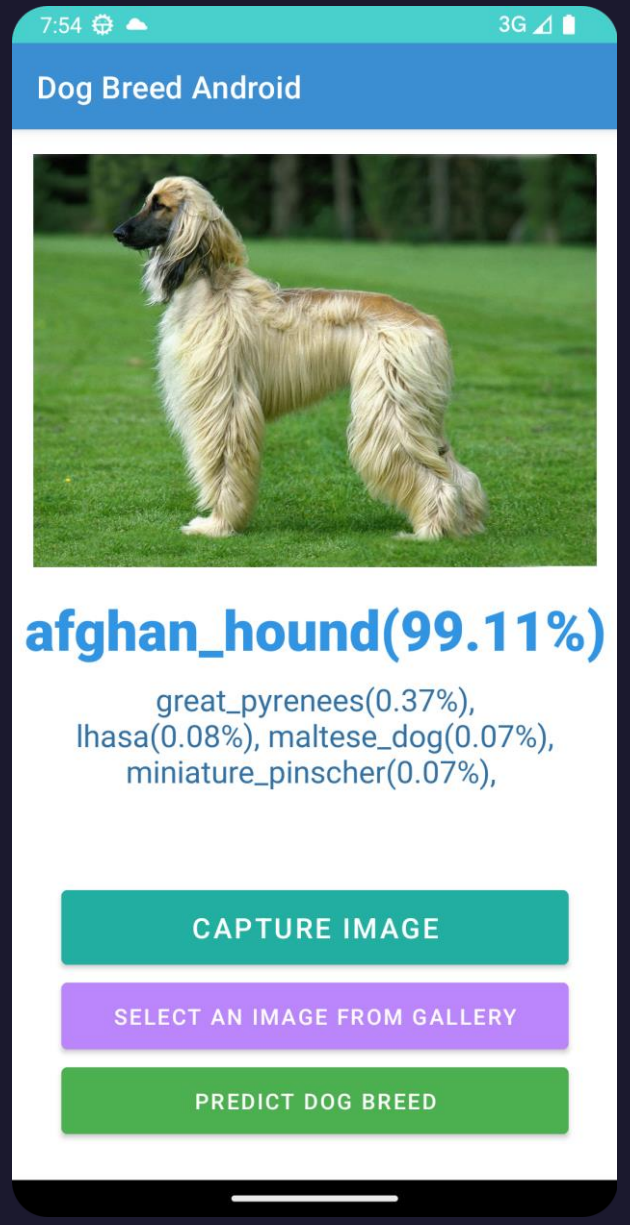
Android App Preview - Home Screen



Android App Preview – Capture Image

Android App Preview – Select Images from Gallery

Android App Preview – Predict dog breed



Android App Preview

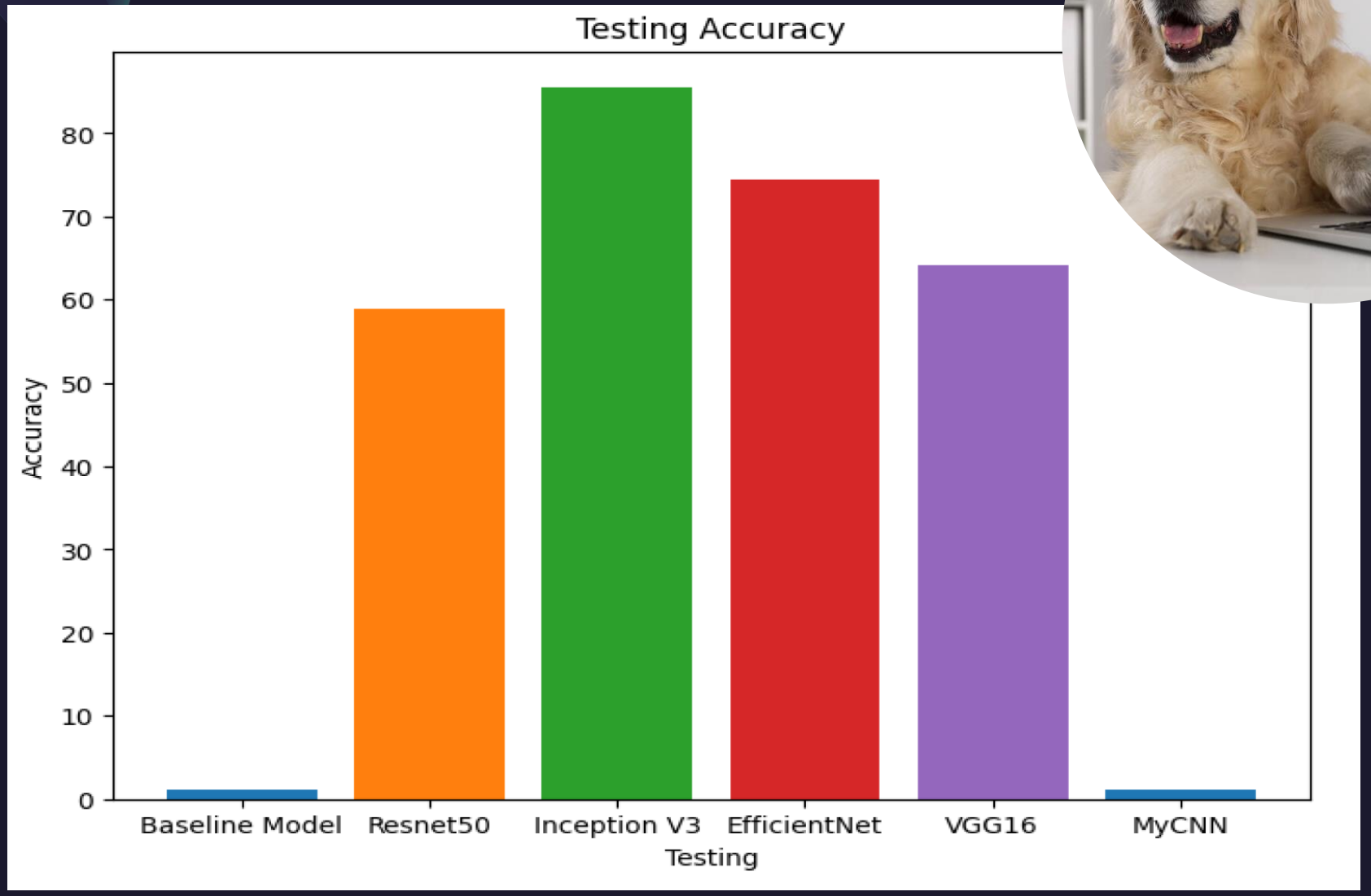
- **If you have an Android Device try our app by downloading it from here!**
  - **https://drive.google.com/file/d/1IR_H0E9-MomQvZa7heD4QEbatMtw2DOn/view?usp=sharing**

Results

Table 1. Models Performance on Dog Breed Dataset

Model	Validation Accuracy	Testing Accuracy
Baseline Model	1.09%	1.09%
Resnet50	57.14%	58.84%
Inception V3	**82.19%**	**85.59%**
EfficientNet	**65.68%**	**74.54%**
VGG16	60.59%	64.21%
MyCNN	1.11%	didn't check

Results


Testing Accuracy
Accuracy
0
10
20
30
40
50
60
70
80
Baseline Model
Resnet50
Inception V3
EfficientNet
VGG16
MyCNN
Testing

Discussion

- The InceptionV3 model achieved a classification accuracy of 85.59% on the test dataset, which is a pretty good result considering the complexity of the task. But one of the main drawbacks of this project is that the dataset size is very small. We could have scraped the web for more dogs and searched online for more dog datasets and combine them.

- However, we can see the potential of pretrained models and transfer learning in the field of Computer Vision.

- Our final conclusion is that **pretrained models work**! So, next time you think of solving a computer vision problem, the first thing that you have to think is of using pretrained models and modifying them to fit your own image classification tasks. There is no need to spend huge amounts of time and computing to train a model from scratch because as we can see that these models are robust and work quite well to most of the Computer Vision Problems.

Special Thanks to Professor Yan Huang
and to the Teaching Assistants
Riyad Bin Rafiq and Zhaomin Xiao.

This project helped us gain valuable experience in doing real world projects and using CNNs and Transfer Learning

Any Questions?