DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

UNIVERSITY COLLEGE OF ENGINEERING (A)
OSMANIA UNIVERSITY, HYDERABAD – 500 007

TELANGANA STATE, INDIA

2023

INTERNSHIP REPORT

On

DOG BREED DETECTION USING RESNET50

At

RESEARCH CENTRE IMARAT LABORATORY
DEFENCE RESEARCH AND DEVELOPMENT ORGANISATION
VigynanaKancha, RCI road, Hyderabad-500069

BY

DHULIPALLA VENU	100520735014
JAPATHI YASHWANTH CHANDRA	100520735018
RITVIK AGGARWAL	100520735029

Under the Guidance of

K.PRAVEEN KUMAR - SCIENTIST F

DECLARATION

We hereby declare that the presented report of Internship entitled "DOG BREED DETECTION USING RESNET50" is uniquely prepared by us under the supervision and guidance of K. PRAVEEN KUMAR, Scientist F, after the completion of one month of internship at RCI DRDO. We also confirm that the report is only prepared for my academic requirement, not for any other purpose.

We do hereby solemnly declare that the work presented in this Internship Report has been carried out by me and has not been previously submitted to any other University, College, and Organization for any academic Qualification, Certificate and Degree.

DHULIPALLA VENU
(100520735014)

JAPATHI YASHWANTH CHANDRA
(100520735018)

RITVIK AGGARWAL
(100520735029)

CERTIFICATES

GOVERNMENT OF INDIA
RESEARCH CENTRE IRAMAT
Dr. APJ Abdul Kalam Missile Complex
DRDO, Ministry of Defence
Vignyanakancha
HYDERABAD – 500 069
Phone : 040 – 24306432
Fax : 040-24306159

CERTIFICATE

This is to certify that **DHULIPALLA VENU,** bearing Roll No **100520735014**, pursuing B.E Third year in Department of **Electronics and Communication Engineering** in **UNIVERSITY COLLEGE OF ENGINEERING, OSMANIA UNIVERSITY**, has successfully completed his Industrial Oriented Summer Internship on "**DOG BREED PREDICTION USING RESNET50 MODEL"** under our supervision and guidance in **Directorate of HILS, RESEARCH CENTRE IMARAT, HYDERABAD-69,** from April 1, 2023 to May 22, 2023.

K PRAVEEN KUMAR
Scientist "F"
R. C. I

K. PRAVEEN KUMAR
Scientist
Research Centre Imarat
DRDO, Govt. of India
Ministry of Defence
Hyderabad-69

ISO 9001:2008 Lab

आर सी आई

ISO 9001 REGISTERED

DNV

MGMT. SYS. RvA C074

DNV Certification B.V., The Netherlands

GOVERNMENT OF INDIA
RESEARCH CENTRE IRAMAT
Dr. APJ Abdul Kalam Missile Complex
DRDO, Ministry of Defence
Vignyanakancha
HYDERABAD – 500 069
Phone : 040 – 24306432
Fax : 040-24306159

CERTIFICATE

This is to certify that **JAPATHI YASHWANTH CHANDRA**, bearing Roll No **100520735018**, pursuing B.E Third year in Department of **Electronics and Communication Engineering** in **UNIVERSITY COLLEGE OF ENGINEERING, OSMANIA UNIVERSITY**, has successfully completed his Industrial Oriented Summer Internship on **"DOG BREED PREDICTION USING RESNET50 MODEL"** under our supervision and guidance in **Directorate of HILS, RESEARCH CENTRE IMARAT, HYDERABAD-69**, from April 1, 2023 to May 22, 2023.

K. Praveen Kumar

K PRAVEEN KUMAR

Scientist "F"

R. C. I

के. प्रवीण कुमार/K. PRAVEEN KUMAR
वैज्ञानिक/Scientist
अनुसंधान केन्द्र इमारत/Research Centre Imarat
डी.आर.डी.ओ, भारत सरकार/DRDO, Govt. of India
रक्षा मंत्रालय /Ministry of Defence
हैदराबाद, Hyderabad-69

GOVERNMENT OF INDIA
RESEARCH CENTRE IMARAT
Dr. APJ Abdul Kalam Missile Complex
DRDO, Ministry of Defence
Vignyanakancha
HYDERABAD – 500 069
Phone : 040 – 24306432
Fax : 040-24306159

CERTIFICATE

This is to certify that **RITVIK AGGARWAL,** bearing Roll No **100520735029**, pursuing B.E Third year in Department of **Electronics and Communication Engineering** in **UNIVERSITY COLLEGE OF ENGINEERING, OSMANIA UNIVERSITY**, has successfully completed his Industrial Oriented Summer Internship on "**DOG BREED PREDICTION USING RESNET50 MODEL"** under our supervision and guidance in **Directorate of HILS, RESEARCH CENTRE IMARAT, HYDERABAD-69,** from April 1, 2023 to May 22, 2023.

K. Praveen Kumar

K PRAVEEN KUMAR
Scientist "F"
R. C. I

के. प्रवीण कुमार/K. PRAVEEN KUMAR
वैज्ञानिक/Scientist
अनुसंधान केन्द्र इमारत/Research Centre Imarat
डी.आर.डी.ओ.भारत सरकार/DRDO, Govt. of India
रक्षा मंत्रालय /Ministry of Defence
हैदराबाद, Hyderabad-69

ACKNOWLEDGEMENT

We record our sincere thanks to the Shri U. RAJA BABU, Director, Research Centre Imarat, Defence Research and Development Organization for providing us an excellent opportunity to undergo training in this esteemed organization.

We would sincerely like to thank Mr. K. PRAVEEN KUMAR, Scientist F, Research Centre Imarat, DRDO for permitting us to work in his division and for the guidance, care and interest shown towards us throughout this internship inspite of his schedules and commitments.

Our sincere thanks to Prof. D. Rama Krishna, Head of the Department, Electronics and Communication Engineering, UCEOU and all the faculty of the Department for the encouragement and guidance provided.

We extend our sincere thanks to Prof. P. CHANDRASEKHAR, Vice Principal, UCEOU, Hyderabad, for co-operation in providing necessary support for internship.

A special note of thanks to Prof. SRIRAM VENKATESH, Principal, UCEOU for providing facilities in doing this internship.

We sincerely thank all of them who have helped us either directly or indirectly during the completion of the internship tenure.

ORGANISATION PROFILE

Research Centre Imarat (RCI) is a premier laboratory of Dr APJ Abdul Kalam Missile Complex, DRDO spearheading R&D in a wide range of Avionics Systems for diversified defence and aerospace applications.

RCI is entrusted with the responsibility of carrying out research and development in the technologies of Control Engineering, Inertial Navigation, Imaging Infrared seekers, RF Seekers & Systems, On-board Computers and Mission Software. RCI has established state-of-the-art test facilities including Hardware-in-Loop Simulation, Environmental Test facilities and Electro-Magnetic Interference/Compatibility/Pulse (EMI/EMC/EMP) facilities and System Integration facilities supported by a strong Reliability and Quality Assurance team. Over the last three decades, RCI has pioneered the development of many advanced technologies and has grown into a major aerospace laboratory.

Vision

To be the leader in the development of guided missile systems for our Armed Forces by developing the frontier technologies, multidisciplinary competence and avant-garde infrastructure leading to self-reliance.

Mission

- Be a premiere institute for developing Frontier Technologies in collaboration with Academic institutions & industry.
- Foster Human Resources for professional Excellence.
- Organize for the induction and production of the guided missile systems on to the Armed forces.

INDEX

LIST OF CONTENTS

LIST OF FIGURES

ABSTRACT

This project investigates an advanced dog breed detection system utilizing deep learning, with a specific emphasis on using the ResNet-50 architecture pretrained on the ImageNet dataset. ResNet-50, a deep convolutional neural network well-recognized for its proficiency in capturing complex hierarchical features, forms the core of the dog breed detection approach. Employing transfer learning, the model is initialized with pretrained weights from the diverse ImageNet dataset, resulting in enhancements in the model's efficiency and convergence rate.

The dataset utilized comprises a diverse compilation of labeled dog images spanning a wide range of breeds. This diverse dataset equips the model with a rich array of features crucial for accurate breed detection. Fine-tuning the pretrained ResNet-50 model on this specialized dog breed dataset is essential, enabling the model to adapt and tailor its features. Preprocessing techniques and data augmentation strategies are employed to strengthen the models robustness.

The model is evaluated on a separate test set, and performance metrics such as accuracy are computed to assess the predictive capability of the trained model. The results demonstrate the effectiveness of utilizing ResNet-50 for accurate and reliable dog breed prediction, showcasing its potential in various applications related to canine-related image recognition.

CHAPTER 1
INTRODUCTION

The project aims to develop an advanced dog breed detection system by harnessing deep learning techniques, specifically leveraging the pre-trained ResNet-50 architecture using the ImageNet dataset. ResNet-50, a powerful convolutional neural network renowned for its ability to capture intricate features within images, is the fundamental model for this dog breed detection initiative. Employing transfer learning, the model is initialized with pre-trained weights from the vast and diverse ImageNet dataset, enhancing its efficiency and convergence.

The dataset utilized for this task is the ImageNet dataset that is a comprehensive collection of labeled dog images representing various breeds. This diverse dataset equips the model with a large number of features that are essential for precise detection. Implementation of various preprocessing techniques is done to enhance the model's robustness and optimize its performance.

This model's potential to accurately classify dog breeds showcases its value in aiding pet owners, veterinarians, and researchers in understanding and managing canine populations effectively.

1.1 PROBLEM STATEMENT AND EXISTING SYSTEM

The conventional methods for dog breed detection often involve manual feature extraction and rule-based classification, which can be inefficient and labor-intensive. Additionally, these approaches may not sufficiently capture the complex visual patterns necessary for accurate breed identification.

The challenge addressed by this project lies in accurately classifying diverse dog breeds using computer vision, especially in scenarios where traditional methods may fall short. While various methodologies have been employed for dog breed classification, the need for a highly efficient and accurate system using deep learning techniques is paramount. Existing approaches might lack in precision or struggle to distinguish between intricate breed features.

1.2 BACKGROUND, MOTIVATION AND PROPOSED SYSTEM

In recent years, the application of deep learning models in computer vision tasks, such as image recognition and classification, has shown promising results. Pretrained deep learning models, like ResNet-50, have demonstrated the ability to learn intricate features from vast amounts of data, enabling them to accurately classify diverse objects.

The motivation for this project is to enhance and automate the process of identifying and classifying various dog breeds accurately. It aims to facilitate various applications such as pet health monitoring, breed specific research and aiding animal shelters in managing and caring for their canine populations effectively.

The proposed system for dog breed detection involves leveraging the ResNet-50 deep learning model, a well-established architecture known for its superior performance in image recognition tasks. Pre-trained weights from ResNet-50 will be utilized to extract features from a diverse dataset of dog images, which will then be used to train a custom classifier. Fine-tuning the model for dog breed recognition ensures it learns specific breed characteristics. To enhance the system, data augmentation techniques will be applied to enrich the training dataset.The system's accuracy and performance will be evaluated rigorously to ensure its effectiveness in breed identification.

Chapter 2

FRAMEWORKS

2.1 INTRODUCTION

Artificial Intelligence (AI) is a broad field of computer science andengineering that aims to create intelligent machines that can perform tasks that typically require human intelligence, such as perception, reasoning, learning, and decision-making.

Machine learning is a subset of artificial intelligence that involves training computer systems to learn and make decisions based on data, without being explicitly programmed. It involves the use of algorithms and statistical models to analyze and learn patterns from data, and thenmake predictions or decisions based on those patterns.

Deep learning is a subfield of machine learning that involves the useof neural networks with multiple layers to learn and make predictions from complex datasets. Neural networks are mathematical models thatare designed to simulate the behavior of the human brain, using interconnected nodes or "neurons" to process and analyze information.

2.2 INTRODUCTION TO RESNET50

ResNet50 is a variant of the ResNet (Residual Network) architecture,which is a type of convolutional neural network (CNN) that was introduced in 2015. ResNet50 is a specific variant of the ResNet architecture that has 50 layers, and it has been trained on large-scale image recognition tasks, such as the ImageNet dataset.

The ResNet architecture is designed to address the problem of vanishing gradients in deep neural networks. In very deep neural networks, the gradients used in backpropagation can become very small, which can make it difficult to train the network.

ResNet introduces skip connections, also known as residual connections, that allow the gradient to be directly propagated through the network, reducing the likelihood of vanishing gradients.

2.3 RESNET50 ARCHITECTURE

ResNet50 has a deep architecture that consists of multiple residual blocks, each of which contains several convolutional layers and skip connections. The architecture of ResNet50 can be divided into several main components:

1. **Input layer:** This layer takes the input image and performs somepreprocessing, such as resizing or normalization.

2. **Convolutional layer and pooling layer:** This is the first set of layers in the network, which perform initial feature extraction from theinput image.

3. **Residual blocks:** These are the building blocks of the ResNet architecture, which contain multiple convolutional layers and skip connections. Each residual block has a shortcut connection that skipsover one or more convolutional layers, allowing the gradient to be directly propagated through the network.

4. **Fully connected layer:** This layer is used to perform the final classification of the input image. It takes the output from the last residual block and produces a probability distribution over the differentclasses.

The residual blocks in ResNet50 can be further divided into several subcomponents, including:

1. **Convolutional layers**: Each residual block contains multiple convolutional layers, which are used to learn increasingly complexfeatures from the input image.
2. **Batch normalization:** This layer is used to normalize the output of the convolutional layers, which helps to speed up training and improve the performance of the network.
3. **Activation function**: This layer applies a non-linear activation function, such as ReLU, to the output of the convolutional layers, which introduces non-linearity into the network.
4. **Skip connection:** This layer adds the output of the previous residualblock to the output of the current residual block, allowing the gradient tobe directly propagated through the network and reducing the likelihood of vanishing gradients

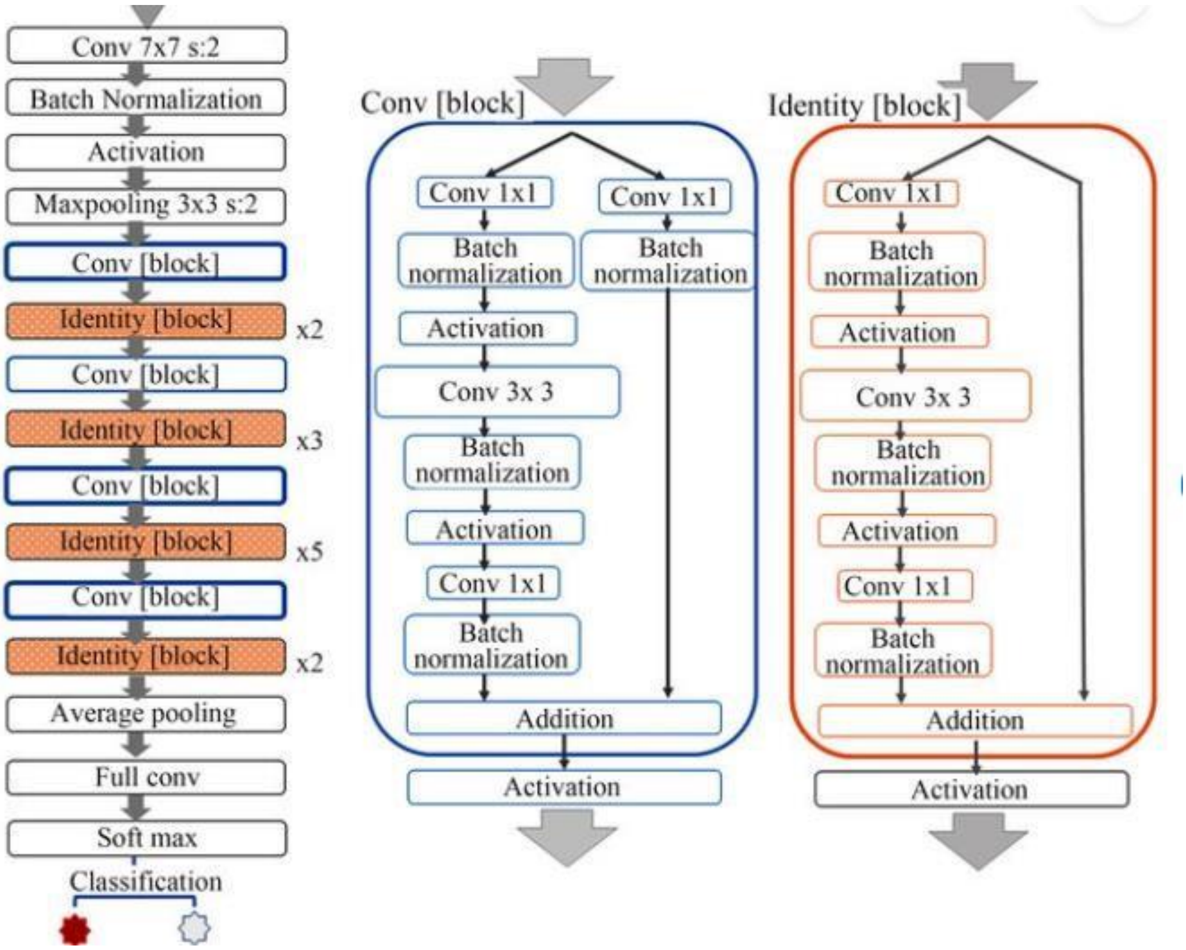

Fig 2.1

RESNET50 ARCHITECTURE

ResNet50 has a total of 50 layers, including 16 residual blocks, whichallows it to learn increasingly complex features from the input imageand achieve state-of-the-art performance on a wide range of image recognition tasks.

Chapter 3

REQUIREMENTS

3.1. SOFTWARE REQUIREMENTS

Python:

Python is a popular high-level, versatile programming language known for its simplicity and readability. It's widely used in various domains, including web development, data analysis, machine learning, artificial intelligence, scientific computing, automation, and more. Python serves as the primary programming language for the project. Python 3.6 or a version higher is preferrable.

Google Colab:

Google Colab is a cloud-based Jupyter Notebook environment provided by Google. It offers a free and accessible platform for running Python code, facilitating collaboration, and harnessing high-performance computing resources, including GPUs and TPUs. Google Colab was employed as the primary development environment for this project.

3.2.LIBRARIES REQUIRED

The various libraries that are necessary to develop the ResNet50 model are

1. **torch (PyTorch):**

PyTorch, an open-source deep learning framework, is a central component of this code. It provides a collection of tools and libraries for building and training neural networks. Here, it's used for defining the architecture of neural networks and implementing various deep learning operations.

2. **torch.nn (PyTorch Neural Networks):**

Within PyTorch, `torch.nn` is a module that contains classes, functions, and utilities for building and training neural network models. It includes predefined layers, loss functions, activation functions, and optimization algorithms essential for constructing neural network architectures.

3. **torchvision.models:**

The `torchvision.models` module from PyTorch Vision provides access to pre-trained models for computer vision tasks. These models, such as ResNet-50 in this case, come with pre-trained weights and architectures that can be used for various image-related tasks without training from scratch.

4. **torchvision.transforms:**

This module from PyTorch Vision provides various image transformations and data augmentation techniques. It allows for preprocessing input images, resizing, normalization, and other transformations necessary for preparing data before feeding it into neural networks.

5. **PIL (Python Imaging Library) – Image:**

The `Image` module from the Python Imaging Library (PIL) provides a way to open, manipulate, and save many different image file formats.

6. **requests:**

The `requests` library in Python enables making HTTP requests to a specified URL. It is employed here to retrieve data (an image) from a given URL.

7. **io (BytesIO):**

The `io` module provides the fundamental classes needed to perform I/O operations. The `BytesIO` class is used to read and write bytes as if it were a file, which is useful for handling image data in memory.

.

Chapter 4

ALGORITHM

1. **Import Necessary Libraries:**

   Import the required Python libraries for deeplearning, image processing, and data handling, including PyTorch, torchvision, and the Python Imaging Library (PIL).

2. **Load the Pretrained ResNet-50 Model:**

   Load the ResNet-50 model pretrained on ImageNet using the torchvision.models module.

3. **Set the Model to Evaluation Mode:**

   Set the model to evaluation mode using `model.eval()` to ensure no gradient calculations are performed during inference.

4. **Load ImageNet Labels:**

   Retrieve the ImageNet labels used for the pretraining of the ResNet-50 model. These labels represent the classes that the model can predict.

5. **Preprocess the Input Image:**

   Define a preprocessing function to resize, crop, normalize, and transform the input image into the format suitable for the ResNet-50 model.

6. **Predict the Dog Breed:**

   Define a function that takes an image URL, fetches the image, preprocesses it, and predicts the dog breed using the pretrained ResNet-50 model. This involves utilizing the model to obtain predictions and converting them into human-readable labels.

7. **Test the Model:**

   Test the dog breed prediction by calling the predict function with an image URL. The predicted dog breed will be printed to the console.

Chapter 5

TESTS AND RESULTS

TEST CASES

URL:

https://th.bing.com/th/id/OIP.teteWZC8EGODe69e0grjLAHaGE?pid=ImgDEt&rs=1

Fig 5.1
TEST CASE 1

RESULT:

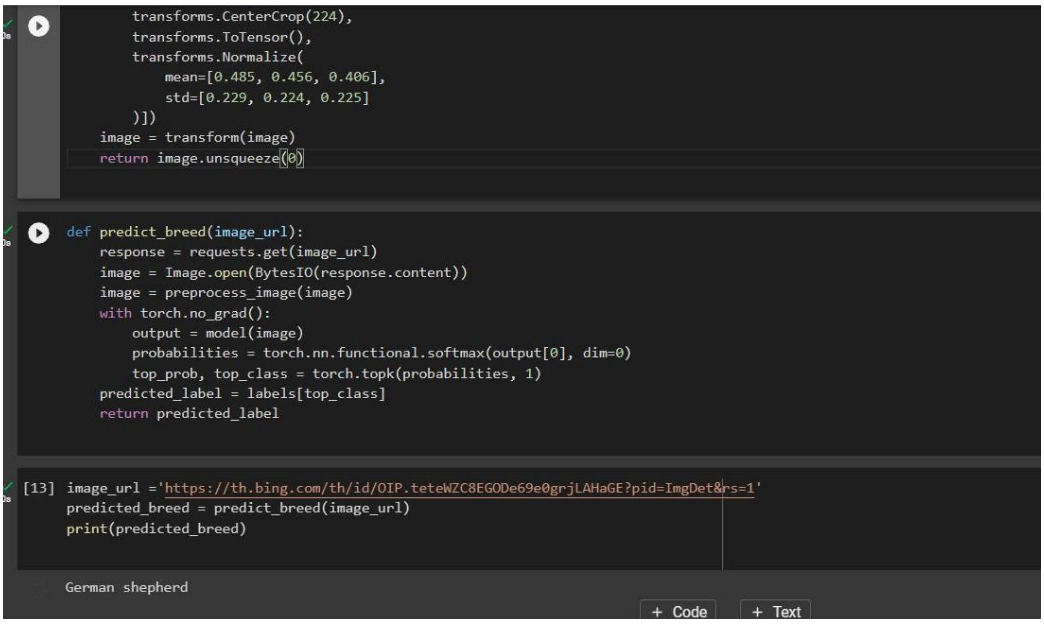

Fig 5.2
PREDICTED OUTPUT 1

URL:

https://images.unsplash.com/photo-1626211596179- d1fe8beaf75c?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxzZWFyY2h8Mnx8cG9tZXJhbmlhbnxlbnw wfHwwfHw%3D&w=1000&q=80

Fig 5.3
TEST CASE 2

RESULT:

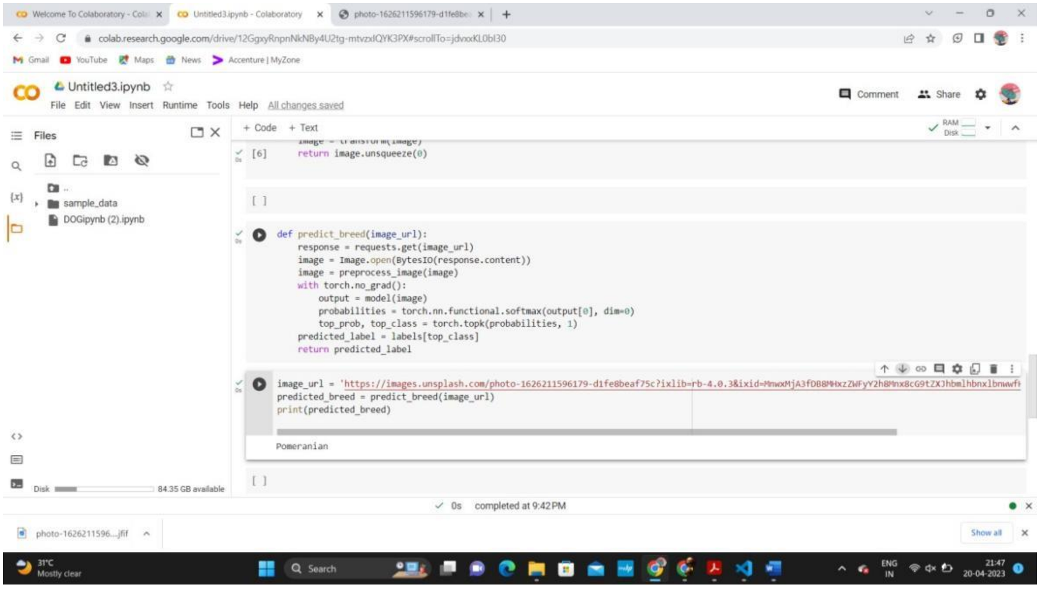

Fig 5.4
PREDICTED OUTPUT 2

CHAPTER 6

CONCLUSION

This project successfully developed a dog breed prediction model using ResNet50.The results demonstrate the potential of deep learning techniques in the field of computer vision, and the model's high accuracy suggests its potential use in various applications, such as animal shelters or veterinary clinics. Further work could explore the use of transfer learning or fine-tuning to improve the model's performance. Overall, this project highlights the importance of developing intelligent systems that can help us better understand and care for the animals that share our world.

FUTURE ENHANCEMENTS

The model used for detection is a pretrained Resnet50 model. For improving the performance further exploration in deeper network variants and integrating attention mechanisms to enhance feature extraction. Fine-tuning approaches, such as domain adaptation and multi-task learning, can be done to tailor the model to specific domains and tasks. Advanced data augmentation and dynamic preprocessing methods are introduced to boost model robustness. Furthermore, regularization techniques and optimization algorithm can be looked into. Techniques like quantization, pruning, ensemble learning, and human recognition using ResNet-50 are highlighted for model optimization and expanded functionality.

REFERENCES

1) A-comprehensive-guide-to-convolutional-neural-networks-

2) Image Database from ImageNet
https://www.image-net.org/

3) Torch-vision ResNet50 Documentation.
https://pytorch.org/vision/main/models/generated/torchvision.models.resnet50.html

Appendix

1) **Import the necessary libraries**:

```
import torch
import torch.nn as nn
import torchvision.models as models
import torchvision.transforms as
transformsfrom PIL import Image
import requests
from io import BytesIO
```

2) **Load the ResNet50 model**:

```
model = models.resnet50(pretrained=True)
```

3) **Set the model to evaluation mode:**

```
model.eval()
```

4) **Load the ImageNet labels**:

```
labels_url='https://raw.githubusercontent.com/pytorch/hub/master/imagenet_classe
s.txt'
response=requests.get(labels_url)
 labels = response.text.split('\n')
```

5) **Define a function to preprocess the input image:**

```
def   preprocess_image(image):
transform                          =
transforms.Compose([
```

```
transforms.Resize(256),
transforms.CenterCrop(224),
transforms.ToTensor(),
transforms.Normalize(
mean=[0.485, 0.456, 0.406],
 std=[0.229, 0.224, 0.225]
 )])
image   =   transform(image)
return image.unsqueeze(0)
```

6) Define a function to predict the dog breed:

```
def predict_breed(image_url):
 response = requests.get(image_url)
image = Image.open(BytesIO(response.content))I
mage = preprocess_image(image)
   with torch.no_grad():
   output = model(image)
probabilities = torch.nn.functional.softmax(output[0], dim=0)
top_prob, top_class = torch.topk(probabilities, 1) predicted_label
= labels[top_class]
return predicted_label
```

7) Test the model by calling the predict breed function with an imageURL:

```
image_url  =  'https://images.dog.ceo/breeds/beagle/n02088364_1513.jpg'
predicted_breed = predict_breed(image_url)
print(predicted_breed)
```