BAHRIA UNIVERSITY KARACHI CAMPUS

DEPARTMENT OF COMPUTER SCIENCE

MULTIMEDIA SYSTEMS

BSCS-7A

ANIMAL IMAGE RECOGNITION SYSTEM

Submitted by:

Shahmir Naeem 02-134211-043

Khurram Aimad 02-134211-050

Submitted To:

Dr. Safdar & Ms. Soomal Fatima

1. ABSTRACT:

In this project, we built a machine learning model to classify images of cats, dogs, and pandas. We used a type of neural network called a Convolutional Neural Network (CNN) because it works well with images. To improve our model's accuracy, we used image processing techniques like rescaling, rotating, and flipping images. This helps the model see a variety of examples and learn better. We trained our model using a library called TensorFlow, which makes building and training neural networks easier. We split our image data into two parts: one for training the model and one for testing it to make sure it works well with new images. After training the model, we created an interactive tool that lets users upload their own images to see if the model can correctly identify them as a cat, dog, or panda. If the model is not confident about its prediction, it will simply say "Unknown." To help users see how well the model is performing, we used charts to show the model's accuracy and loss during training. Overall, this project demonstrates how we can use advanced machine learning techniques and multimedia tools to build an image classifier that can recognize different animals.

2. INTRODUCTION:

In this project, we created a machine learning model that can identify whether an image is of a cat, dog, or panda. We used a special type of neural network called a Convolutional Neural Network (CNN) because it is great at analyzing images.

3. Implementation of Multimedia Concepts to Construct a Model:

3.1. Image Processing and Augmentation

- **Concept**: Image processing involves manipulating and transforming images to improve their quality or extract important features. Image augmentation artificially expands the size of a training dataset by creating modified versions of images in the dataset.
- **Implementation**: We use ImageDataGenerator from Keras to rescale images, apply random transformations such as rotations, flips, and zooms. This helps the model generalize better by seeing varied examples during training.

3.2. Convolutional Neural Networks (CNNs)

- **Concept**: CNNs are a class of deep neural networks, most commonly applied to analyzing visual imagery. They use convolutional layers that apply filters to the input image to capture spatial hierarchies.
- **Implementation**: We construct a CNN with multiple convolutional and pooling layers to extract features from images.

4. Utilization of Multimedia Tools to Attain Desired Outcomes:

4.1. Training and Validation Using Multimedia Data

- **Tools**: Use Keras' ImageDataGenerator to load and preprocess the image data for training and validation.
- **Outcome**: Efficient data loading and augmentation ensure that the model sees diverse examples and can generalize well to unseen data.

4.2. Model Training

- **Tools**: TensorFlow and Keras libraries are used for constructing, compiling, and training the CNN model.
- **Outcome**: These libraries provide high-level APIs for model building and training, enabling quick experimentation and efficient training.

4.3. Evaluation and Visualization

- **Tools**: Matplotlib is used to visualize training results, and IPython widgets (ipywidgets) are used to upload and test new images interactively.
- **Outcome**: Visualization tools help monitor the training process and evaluate model performance. Interactive tools allow for real-time testing and validation.

5. Objectives and Goals:

5.1 Build an Accurate Image Classifier:

- Develop a machine learning model that can accurately classify images of cats, dogs, and pandas.
- Use Convolutional Neural Networks (CNNs) for their effectiveness in image recognition tasks.

5.2 Enhance Model Generalization:

- Employ image processing and augmentation techniques to improve the model's ability to generalize from the training data to new, unseen images.
- Include rescaling, rotation, flipping, and zooming transformations during training.

5.3 Scalability and Flexibility:

- Design the model and tools in a way that allows for easy updates and scalability.
- Make it possible to expand the model to include more classes or improve its accuracy with additional data and training.

5.4 Educational Purpose:

- Demonstrate the practical application of machine learning and multimedia tools in solving real-world problems.
- Provide a comprehensive example of how to build, train, and deploy an image classification model using modern machine learning libraries.

By achieving these objectives and goals, the project aims to create a robust, user-friendly, and accurate image classification system while also serving as a valuable educational resource.

6. Project Scope:

This project aims to create a machine learning model that can identify images of cats, dogs, and pandas. It involves building and training a Convolutional Neural Network (CNN) using enhanced and processed images. The project also includes developing an interactive tool that allows users to upload their own images and get predictions in real-time. If the model isn't confident about a prediction, it will label the image as "Unknown." Additionally, the project includes evaluating the model's performance and visualizing its accuracy. The scope covers everything from data preparation to model deployment, ensuring an easy-to-use and accurate image classifier.

7. Project Overview

In this project, we developed a machine learning model to recognize images of cats, dogs, and pandas. We used a Convolutional Neural Network (CNN), a special type of neural network good at handling images. To make the model more accurate, we processed and enhanced the training images. We also created an interactive tool that lets users upload their own images and get instant predictions. If the model isn't sure, it will say "Unknown." The project includes evaluating the model's performance and visualizing its training progress, ensuring a user-friendly and reliable image classification system.

8.Difficulties faced:

8.1. Training Duration: One of the primary challenges encountered was the extensive time required for model training. With 50 epochs initially set, training sessions extended to several hours, consuming substantial computational resources.

8.2. Inaccuracy in Predictions: Despite prolonged training, the model initially produced inaccurate predictions. Images were often misclassified, indicating the need for additional model optimization.

8.3. Computational Resource Intensity: CNNs demand significant computational power, making it challenging to iterate rapidly and explore different architectures and hyperparameters.

9. Mitigation Strategies:

To address these challenges, several strategies were implemented:

- **Early Stopping**: Introduced to halt training when validation accuracy plateaued, preventing overfitting and reducing training time.
- **Fine-tuning Hyperparameters**: Adjusted learning rates, batch sizes, and image augmentation parameters to enhance model performance and efficiency.
- **Model Evaluation and Iteration**: Regularly evaluated model performance using validation metrics and iteratively adjusted architecture and training strategies.

10. Alternative Model Considerations

Several alternative models were evaluated for image classification:

- **Support Vector Machines (SVMs)**: Effective for smaller datasets but require feature extraction and may not capture complex image patterns as effectively as CNNs.
- **K-Nearest Neighbors (KNN)**: Simple and intuitive but computationally expensive for large image datasets.
- **Decision Trees**: Interpretable but may struggle with image data's high dimensionality and complexity.

10.1. Rationale for CNN Selection:

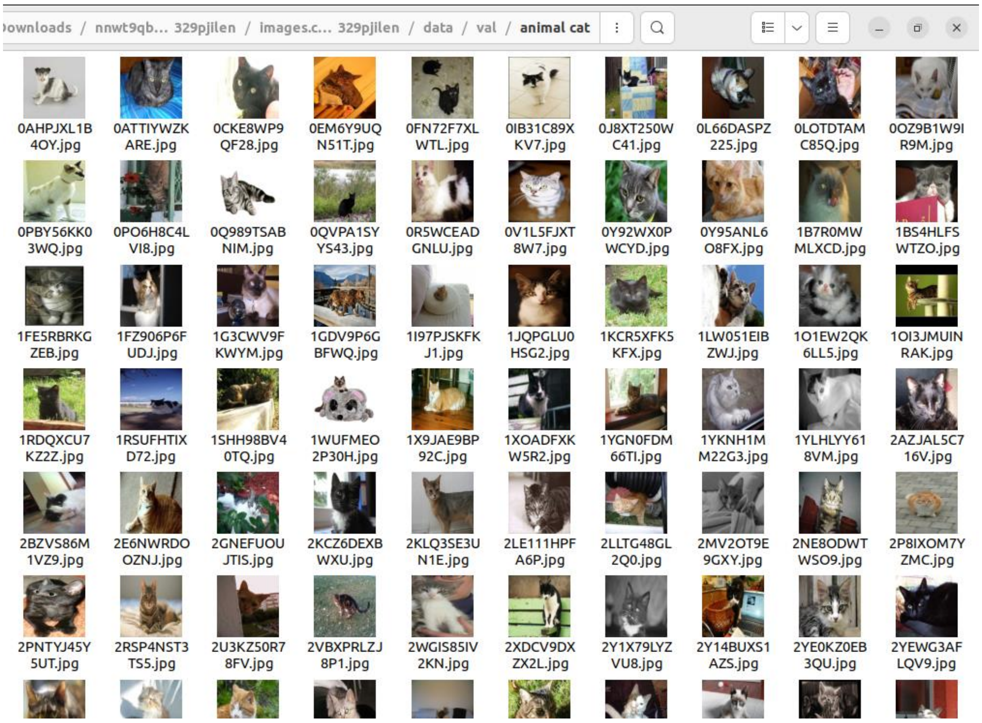
CNNs were chosen over alternative models due to several key advantages:

- **Automatic Feature Extraction**: CNNs can automatically learn relevant features from raw image data, eliminating the need for manual feature extraction.
- **Spatial Hierarchies**: They capture spatial hierarchies in images, crucial for recognizing intricate patterns like textures and shapes.
- **State-of-the-Art Performance**: CNNs have demonstrated superior performance in image classification tasks, leveraging deep learning advancements.

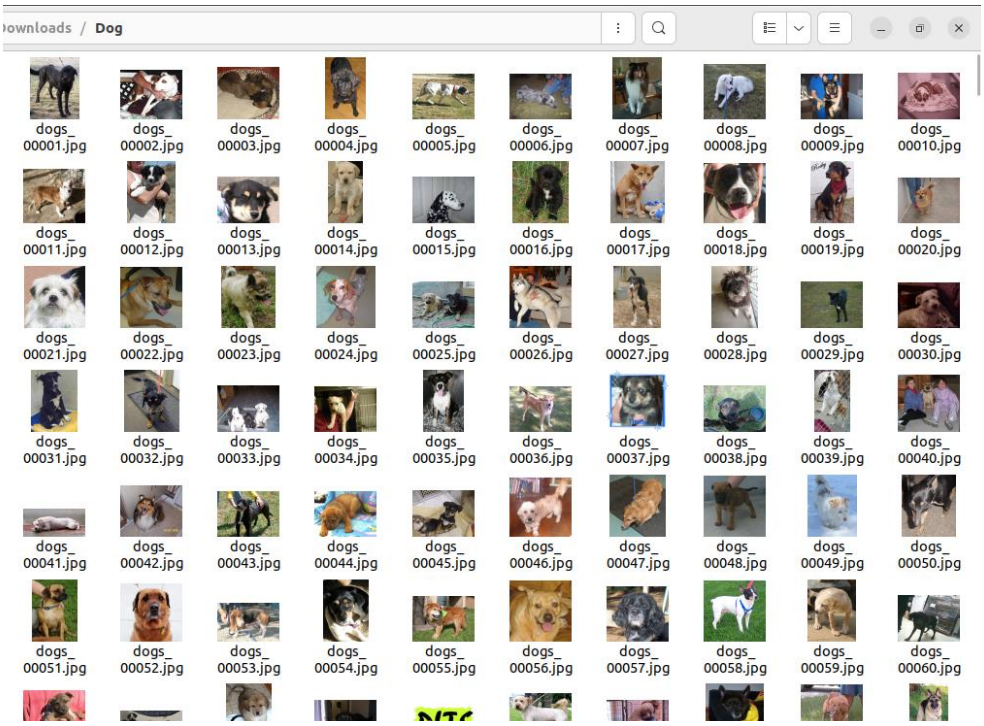
11. OUTPUT:

11.1 Data for Training:

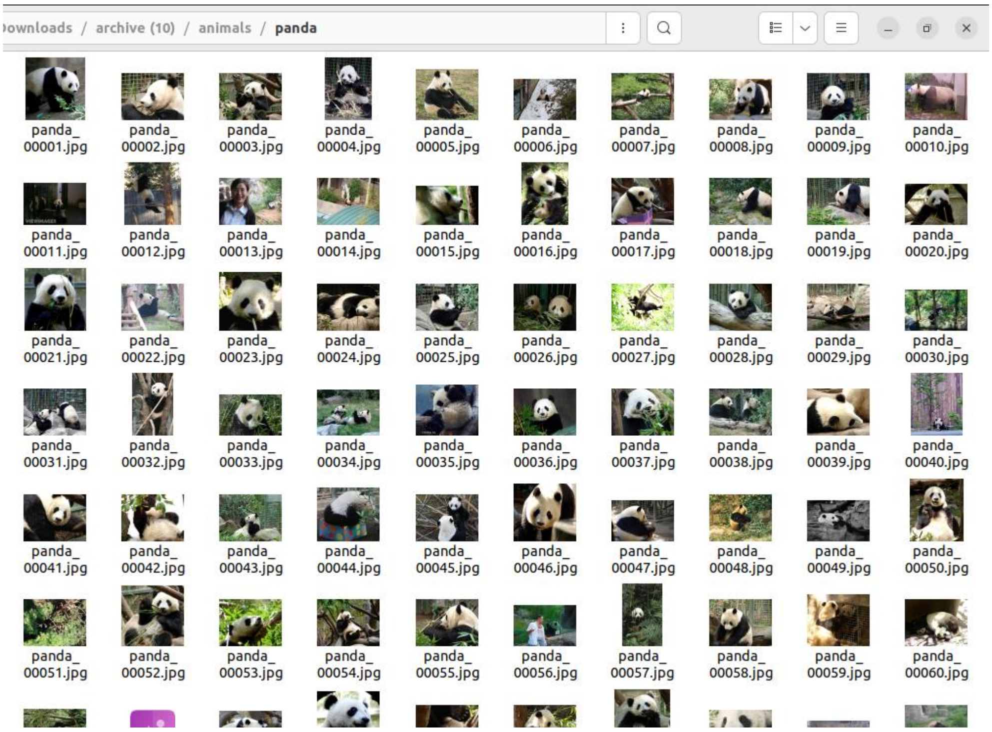
Cat (2000 Pictures):



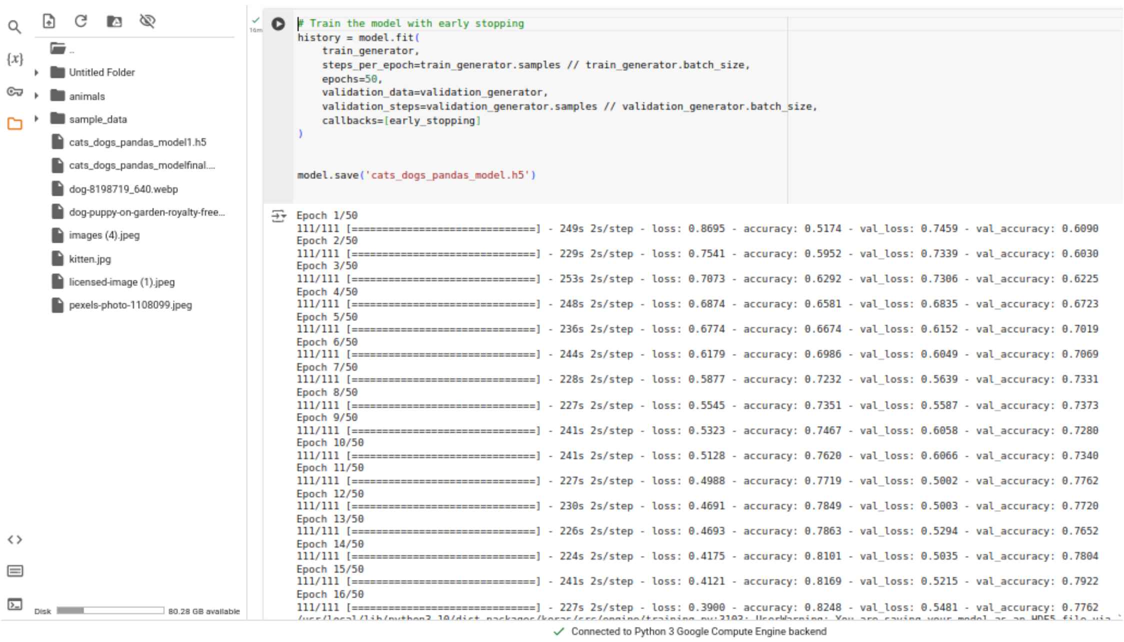
Dog (2000 Pictures):

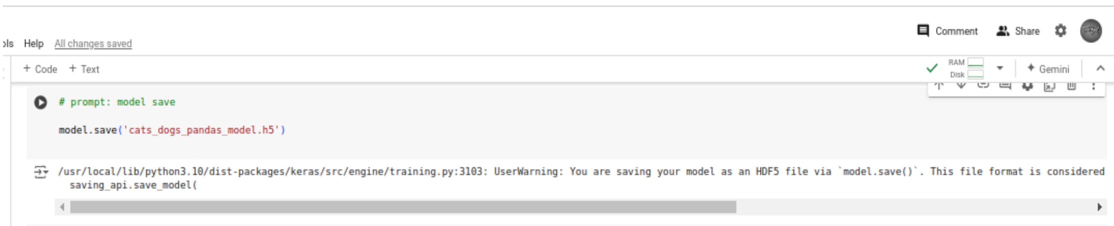


Panda (2000 Pictures):



11.2 Model Training:





11.3 Training Accuracy:

```
acc = history.history['accuracy']
val_acc = history.history['val_accuracy']
loss = history.history['loss']
val_loss = history.history['val_loss']

epochs_range = range(len(acc))

plt.figure(figsize=(8, 8))
plt.subplot(1, 2, 1)
plt.plot(epochs_range, acc, label='Training Accuracy')
plt.plot(epochs_range, val_acc, label='Validation Accuracy')
plt.legend(loc='lower right')
plt.title('Training and Validation Accuracy')

plt.subplot(1, 2, 2)
plt.plot(epochs_range, loss, label='Training Loss')
plt.plot(epochs_range, val_loss, label='Validation Loss')
plt.legend(loc='upper right')
plt.title('Training and Validation Loss')
plt.show()
```

11.4 Model Prediction:

Upload (5)

```
1/1 [==============================] - 0s 38ms/step
```

Predicted: Panda

```
Predicted class: Panda
Probabilities: [2.7134824e-05 4.6238606e-04 9.9951053e-01]
1/1 [==============================] - 0s 44ms/step
```

Predicted: Cat

Predicted: Dog

```
Predicted class: Dog
Probabilities: [0.19816728 0.80083317 0.00099954]
1/1 [==============================] - 0s 47ms/step
```

Predicted: Unknown

```
Predicted class: Unknown
1/1 [==============================] - 0s 50ms/step
```

Predicted: Unknown

```
Predicted class: Unknown
```

12. Conclusion:

In this project, we successfully built a machine learning model to identify images of cats, dogs, and pandas using a Convolutional Neural Network (CNN). We enhanced the training images to improve the model's accuracy and created an interactive tool for users to upload and test their own images. The model reliably labels images and indicates "Unknown" for low-confidence predictions. This project demonstrates the effective use of advanced technology for image classification, providing a practical and user-friendly solution. It also serves as an educational example of how to build and deploy machine learning models for real-world applications.