Domestic Animal Breed Identification

CSA301 Deep Learning
Bachelor of Computer Science (AI Development and Data Science
(Year III, Semester I)

RESEARCHER(S)

Sonam Wangchuk (12220069)
Sonam Yangzom (12220070)
Tashi Norbu Dema (12220079)
Tenzin Wangmo (12220083)

GUIDED BY

Ms. Pema Wangmo

Gyalpozhing College of Information Technology
Kabjisa : Thimphu

List of Figures

Contents

1 Abstract

Accurate and reliable breed identification of domestic animals from images is one the most promising but challenging tasks in identifying the breed. Traditional methods for domestic animal breed are very costly and time consuming. Therefore, there is a need for faster and cheaper technique for animal breed identification, which can be used by anyone without much technical knowledge. Deep learning based animal breed identification from images can used to solve this problem. The algorithms, Convolutional Neural Network (CNN) has improved the accuracy of image recognition systems, but choosing the optimal model for the required task is very important for best performance. In this study, the performance of four different deep CNN-based models have been analyzed to find the optimal model which can precisely determine the breed identity of individual domestic animal from its images.

Keywords: Domestic animal classification, Breed identification, Deep learning models, Convolutional neural networks (CNNs)

2 Introduction

2.1 Project Background

The ability to accurately identify the breed of domestic animals is important in veterinary science, animal husbandry and so on as it facilitates the provision of appropriate care, breeding and management practices. Nevertheless, it is still difficult to identify breeds of all domestic animals because there are several reasons such as the high level of genetic diversity within any breed, similar phenotypic characteristics of different but related breeds,which often results in inaccurate classification and poor management. Before, classification of animal breeds has been basically

dependent on the physical appearance of the given breed. This has always been subjective and has many possibilities of inaccuracies in the case of animal breeds that are rare. Understanding the importance of improved methods, this project aims at building a deep learning model for breed identification using image recognition. The system will finally be implemented as an easy to use web application which will enhance efficiency and accuracy on livestock management. This project is in support of animal welfare and also boosts breeding and conservation strategies. The project activities comprise data collection, model development, web integra-

tion, and performance testing. There are issues such as data quality and breed complexity that are expected, but the system gives an up to date approach to the age-old problem of assessing breeds and categorizing them. The Era of Molding

very intelligent Machines – With the advent of machines that can be taught and the wide application of deep learning, particularly convolutional neural networks (CNNs) used for image processing, many industries such as those associated with facial recognition, medical imaging, and even the classification of species have also undergone great transformation. The use of these technologies for the purpose of identification of animal breeds is preferable to the conventional methods for obvious reasons. Nevertheless, it should be noted that this is only possible when data quality is good, the breeds in question are not complex, and there is acceptance from the users.

2.2 Problem Statement

The accurate identification of domestic animal breeds is crucial aspects of veterinary science, agriculture, and animal conservation. However, the identification of domestic animal breeds is challenging due to the high level of genetic diversity, similar physical traits among the breeds and the prevalence of crossbreeding leading to misidentification that can adversely affect animal care, breeding practices, and conservation efforts. Traditionally, breed identification has depended on the expertise of individuals who visually inspect animals which can be subjective and prone to error, particularly with mixed or less common breeds. Given the importance of precise breed identification and the limitations of traditional methods, there is a pressing need for more efficient, reliable, and accessible techniques. Therefore, we are developing a breed identification system that leverages deep learning and image recognition technology to provide accurate, efficient, and user-friendly identification of domestic animal breeds, ultimately improving animal welfare and supporting effective breeding and conservation strategies.

2.3 Aim

The project aims to develop a deep-learning model for breed identification of domestic animals through a user-friendly web application to enhance livestock management practice.

2.4 Objectives

- Collect datasets and perform processing of data
- Build adeep learning model to recognize the domestic breed animals.
- Integrate the model with training datasets.
- Integrate the trained models into a web application.
- Testthe performance of the trained models using unseen datasets.

2.5 Scope

2.5.1 System Scope

- Data Integration: The system will receive input in the form of an image (Domestic animal images)
- Image Analysis: The system should analyze the animal images to identify the breed of that particular domestic animal.
- Reporting: The system will assist in generating reports regarding the findings of the animal breed.

2.5.2 User Scope

- The user scope includes anyone interested in identifying the breed of domestic animals.

2.6 Limitations

- Data quality and availability: Lack of high quality images and less of datasets.
- Breed Complexity: Small differences in appearance can make it hard to classify and identify them correctly.
- Technology access: The application may face limited use if users lack skills and are reluctant to use and adapt to new methods and technology.

2.7 Requirements

2.7.1 Functional Requirements

- Image Upload: Users should be able to upload images of any domistic animal.
- Preprocessing: The system should include preprocessing steps such as image resizing, .jpg format conversion and contrast enhancement to improve the quality of the images for analysis.
- Image Analysis: The system should use image analysis algorithms to classify the breed of the domestic animal accurately.
- Reporting: The system should be able to generate reports from the findings of the animal breed.

2.7.2 Non-Functional Requirements

- Reliability: The system should work in stable and reliable manner with minimum chances of failures
- Performance: The system needs to perform exceptionally well, be highly accurate and productive while minimizing interruptions or delays.
- Portability : without any change in its behavior and performance, the web application should work properly in different systems.

2.8 Data source

We compiled a dataset from three distinct sources on Kaggle for three different domestic animal namely; cat, dog and cattle.

Source 1: https://www.kaggle.com/datasets/yapwh1208/dogs-breed-dataset

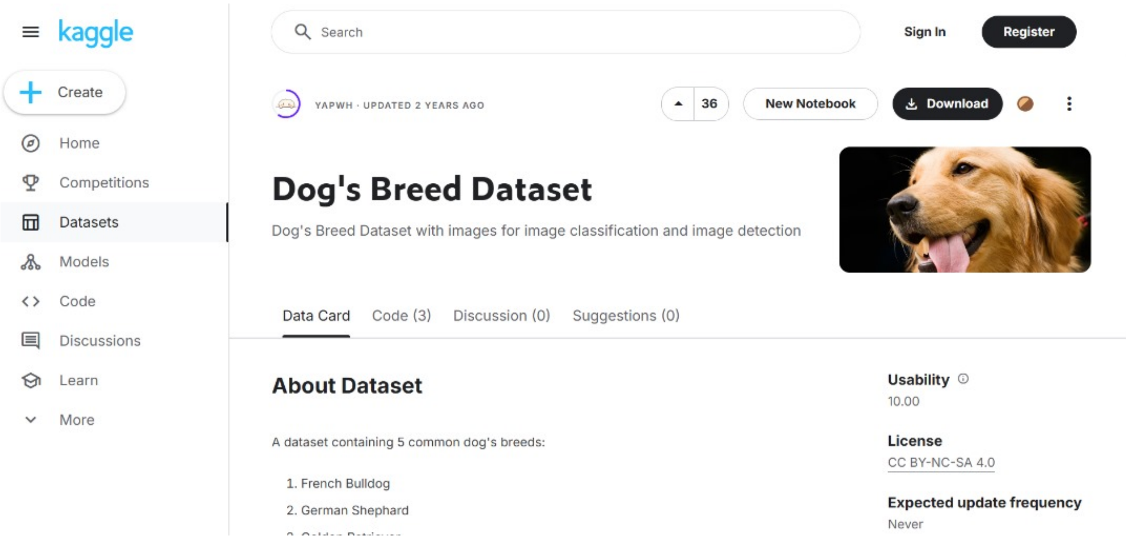

Figure 1: Dogs dataset form kaggle

Source 2: https://www.kaggle.com/datasets/yapwh1208/cats-breed-dataset

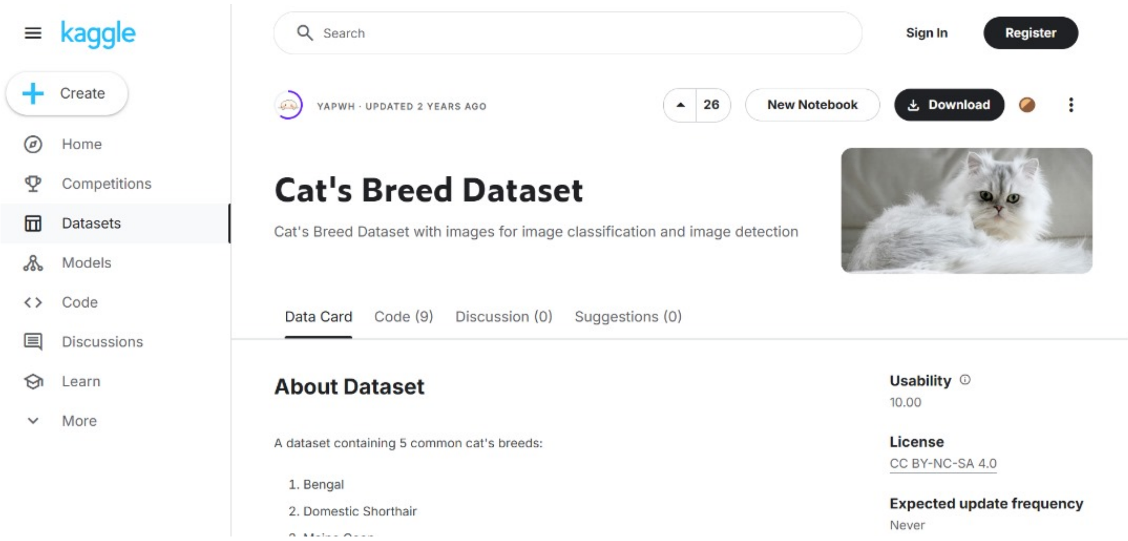

Figure 2: Cats dataset form kaggle

Source 3: https://www.kaggle.com/datasets/anandkumarsahu09/cattle-breeds-dataset

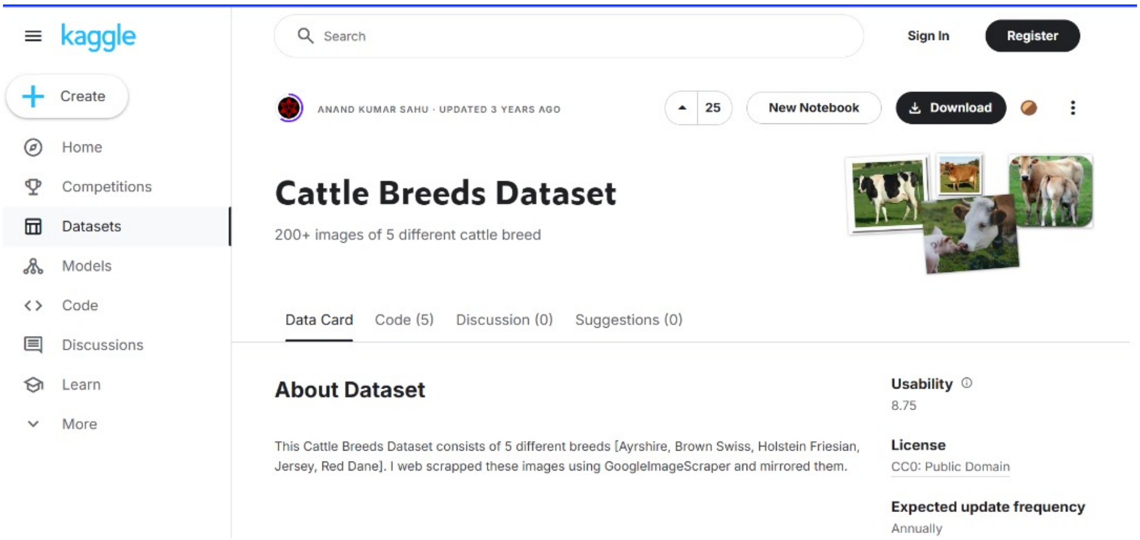

Figure 3: Cattles dataset form kaggle

3 Related Work

Animal breed identification has emerged as a critical area of research, particularly with the advent of deep learning techniques, which offer promising solutions for accurately classifying and distinguishing various breeds. For the domestic breed identification, deep learning model can be trained on the datasets containing images of different breeds, enabling the models to learn the distinct features of each breed. In this research [1], Chao Ma et al. explored the development of a sheep

recognition algorithm based on the deep learning, specifically utilizing the Faster R-CNN (Region-Convolutional Neural Network) model enhanced with the Soft-NMS (Non-Maximum Suppression) to improve the accuracy and efficiency of sheep breed identification . Faster R-CNN is a powerful tool for object detection that integrates a region proposal network (RPN) with a convolutional neural network (CNN) for feature extraction, classification, and bounding box regression. The research demonstrated that the Faster R-CNN model with Soft-NMS achieved a detection accuracy of 95.32 percent. This approach allows for real-time detection and localization of sheep, providing a solid foundation for further research. In this article [2], it high-

lights the use of Deep Neural Networks (DNNs) for animal image identification and classification, particularly focusing on predators. It is employed to deal with noisy labels within data sets. Some of the works include the following: iterative label refinement, feature extraction from pretrained networks and network optimization. If the problem of noisy labels is solved, then the models will be able to produce more accurate classifications, which will benefit from the monitoring of various populations of animals. In the context of image classification, features such as deeper

networks like ResNet use a variety of Convolution Neural Network (CNN). This particular type of models is built special to identify various animal species and are the basis of datasets such as the Snapshot Serengeti and the Panama-Netherlands.

These datasets are in form images which are captured through camera traps; they offer large datasets, through which the CNNs can in the process learn features that can enable them to distinguish between different species. CNNs proves pivotal in enhancing high levels of accuracies on the species identification hence being a strong pillar in wildlife research. Camera trapping is important in this study as

a non-intrusive approach to assessing populations of these animals in their ecosystem. Unlike other intrusive methods, for example, GPS that require fixing devices on animals' bodies or the use of sensor networks that would interfere with animal's environments, camera traps provide non-intrusive ways of observing animal behaviour. The images obtained after that are analysed by the deep learning models for the detection and classification of animals from the visual data obtained from the traps. This has the effect of improving the quality of the data gathered as well as extending the overall subject area of monitoring to more species and habitats. Sometimes it is also difficult to differentiate between two closely related

animal species or breeds and hence Fine-Grained Classification (FGC) techniques are also used in the study. FGC, on the other hand, entails the process of extracting more detailed features from the images which are used to classify the animals deeper. The incorporation of FGC techniques in the deep learning models makes the study more comprehensive and practical since the techniques improve the model performance in identifying and classifying the animals. The study shows how in-

tegrating deep learning and image processing in animal detection and classification systems is efficient. By looking into critical areas and considering the techniques to minimize the effect of noisy labels, the study presents how the efficacy of these systems enhances in the long run. Employing such technologies as camera traps also supports the applicability of these methods in conservation and studying the wild fauna and flora. Deep learning has the ability to automate animal monitoring

and thereby improving efficiency and accuracy. The paper [3], presents a detailed approach to wild animal detection and identification by using YOLOv3 algorithm. The studies have investigated the applications of deep learning in the classification of wildlife images and object detection in natural settings. Nonetheless, challenges such as variations in the appearance of animals, poor quality camera trap images as well as the need for large data sets still exist. The reviewed paper addresses these challenges with support from YOLOv3, and state-of-the-art object detection model. The method used in this paper involves collection of animal videos data set,

pre-processing the data, and then training YOLOv3 model. The findings demonstrate a great accuracy when it comes to recognizing various species of animals, demonstrating the potential of deep learning for this task. Even though the article provides insightful information; however additional research is required to deal with limitations and challenges experienced while monitoring wildlife in real life scenarios. Deep learning offers a promising avenue for automating wild animal detection and identification. The paper by Pukale et al., shows how effective YOLOv3 can be on such projects. In the study "Dog Breed Prediction Using Deep Learning" [4], the

researchers developed a CNN model trained using a large collection of dog images, covering a wide range of breeds. Before feeding the images into the model, they were pre-processed to improve their quality and ensure they were all the same size and format. The CNN model itself included several layers of convolution and pooling, followed by fully connected layers that ultimately made the breed predictions. To evaluate the model's performance, the researchers used metrics like accuracy, precision, and recall, which demonstrated that the model was quite effective at identifying breeds. The results were impressive, with the model correctly predicting

dog breeds 90 percent of the time. The researchers also emphasized the importance of having a diverse dataset and how data augmentation techniques helped make the model more robust. These findings suggest that deep learning can significantly reduce the time and effort required for breed identification compared to traditional methods. In this paper [5], it shows that snake biologists conducted studies using

computer vision algorithms to identify various snake species. For instance, a CNN was trained on its own to identify up to an amazing 45 types of snakes better than humans could really do it. Another study focused on categorizing into poisonous or non-poisonous snakes and achieved accuracy levels of 91.3 percent with knowledge acquired from the evaluation of 1766 images dataset drawn from various parts of Africa. Transfer learning has also become a prevalent approach in animal species

identification with transfer learning using such pre-trained models, researchers can adapt existing networks to new datasets, significantly improving classification accuracy with limited data. Vision transformers (ViT) have come up strongly as a

great replacement for CNNs. The self-attention which the ViTs employ allows them to process images without convolutional layers. This alternative has done well in different image recognition tasks with some benchmarks recording state of the art score . The Data-Efficient Image Transform (DeiT) increases this capability further making it possible to effectively train using small data sets, achieving top-1 accuracy on ImageNet without external data. Incorporation of deep learning especially

CNNs as well as Vision Transformers has greatly enhanced the identification and even classifications of animals in tasks such as species and breed identification with high accuracy. Transfer learning and Ensemble methods, which combine multiple models to improve prediction accuracy, have also been explored in animal identification. By stacking predictions from diverse classifiers, the researchers have been able to improve their overall performance.

4 Methodology

4.1 Methodology of the Study

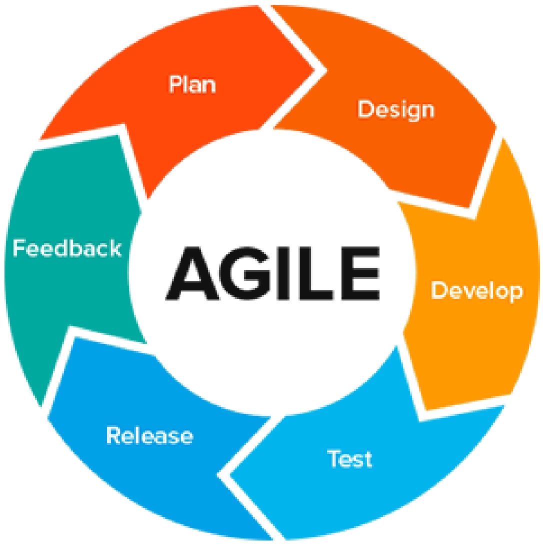

Figure 4: Overview of Agile Methodology

We used Agile methodology in our deep learning project for domestic animal breed identification because it allowed us to remain adaptable and make refinements along the way. This approach enabled us to modify our models and resolve problems efficiently. During the debugging process, Agile methodology addressed issues by allowing the project to be implemented and delivered in phases. This ensured that testing and correction of defects were done upfront, making the whole process more efficient and faster.

4.2 Machine Learning Workflow

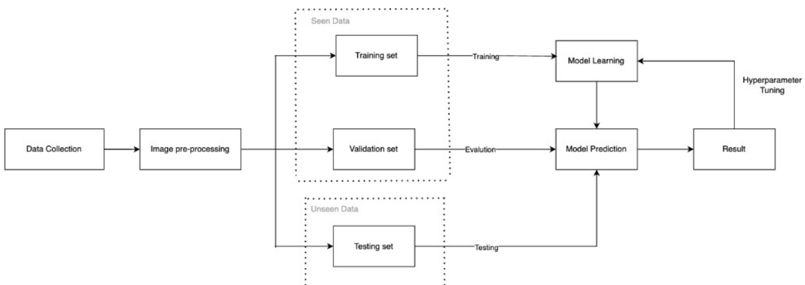

Figure 5: Machine Learning Workflow

1. **Data Collection:** Gather a diverse set of domestic animal breed data (e.g., cattle, cat, dog) to create a comprehensive dataset for breed identification.
2. **Image Pre-processing:** Prepare and clean the images to enhance model performance by addressing issues such as noise, normalization, image format, and resizing.

3. **Data Splitting (Training, Validation, Testing):** Divide the dataset into three subsets:

   - **Training set:** Used to teach the model to learn patterns from the data.
   - **Validation set:** Used to fine-tune hyperparameters and monitor model performance to avoid overfitting.
   - **Test set:** Used for final evaluation to assess the model's generalization ability.

4. **Model Learning on Training Set:** Train the CNN model on the training set to recognize patterns indicative of specific animal breeds.

5. **Validation on Validation Set:** Validate the model's performance on the validation set and adjust parameters to enhance accuracy and prevent overfitting.

6. **Testing on Test Set:** Assess the trained model on the test set to measure its ability to accurately identify domestic animal breeds in new, unseen data, ensuring the reliability of the system.

4.3 System Design

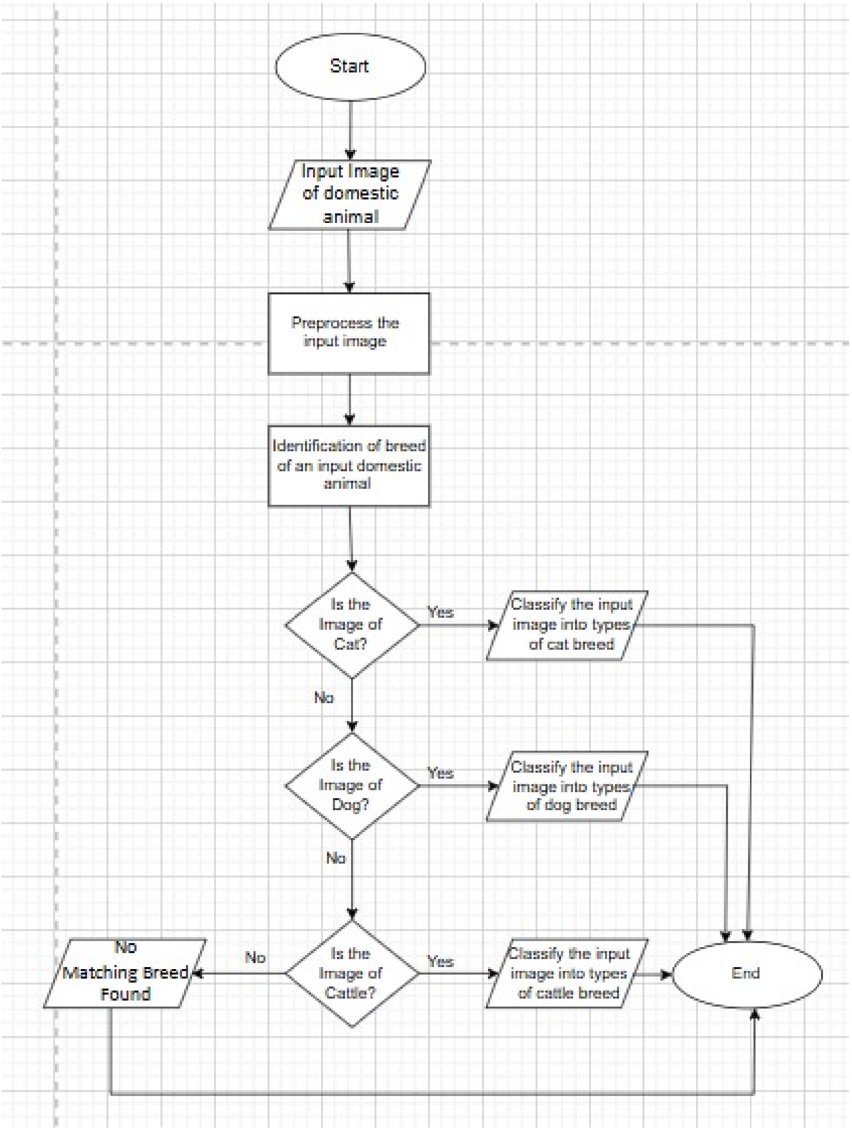

Figure 6: System Design

1. **Start:** The process begins when the user initiates it by providing an image of a domestic animal.

2. **Input Image of Domestic Animal:** The user provides an image of a domestic animal for analysis.

3. **Preprocess the Input Image:** The images will be preprocessed for consistency by adjusting the image shape, brightness, and other factors to make them suitable for analysis by the system.

4. **Identification of Breed of an Input Domestic Animal:** The system will identify the breed of the animal in the image.

5. **Decision Process:** The system goes through series of checks to determine the type of animal in the image. It checks one by one:

   - Is the image of a cat? - If yes, it moves to classify the specific breed of the cat.

   - Is the image of a dog? - If yes, it proceeds to identify the specific breed of the dog.

   -Is the image of cattle? - If yes, it goes to classify the specific breed of cattle.

6. **No Matching Breed Found:** If the system determines that the images does not match any specified domestic animals (cat, dog, or cattle), it concludes that there is no matching breed found.

7. **End:** After successfully identifying the breed or determining no match , the process ends.

4.4 System Architecture

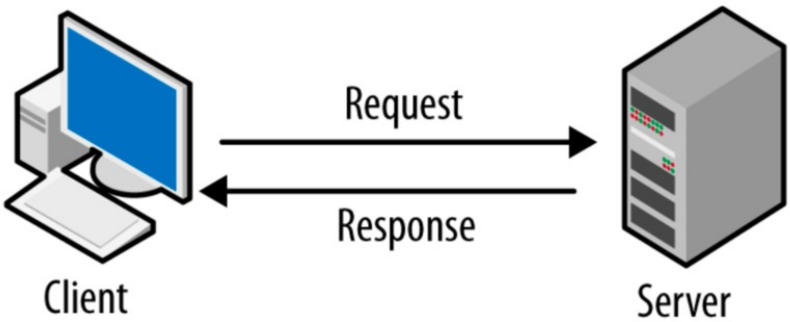

Figure 7: Two-Tier System Architecture

Two-tier architecture comprises a client/user interface layer for user interaction and a server/application layer for processing. Communication is direct between the client and server.

Client/User Interface Layer:

- In the domestic animal breed identification project, the client layer represents the user interface or application through which users interact with the system.
- Users can input relevant data into the system, potentially providing domestic animal images.
- This layer is responsible for capturing user inputs, managing the user interface, and presenting the results of the domestic animal breed.

Server/Application Layer:

- The server layer includes the core application logic and the domestic animal breed identification model.
- The detection model, typically implemented using machine learning techniques, processes input data, analyzes input images, and makes predictions or identify the breed of the given input image..
- This layer is responsible for receiving user inputs from the client layer, running the detection algorithm, and sending the results back to the user interface for display.

4.5 System Overview

4.5.1 Dataset

The dataset was sourced from Kaggle and consists of images of three animal species: cats, dogs, and cattle. Each species has five different breeds, resulting in a total of 15 categories. The dataset contains thousands of labeled images, organized into folders by species and breed. Initially we had the total of 3190 datasets combining all the classes of animal breed.

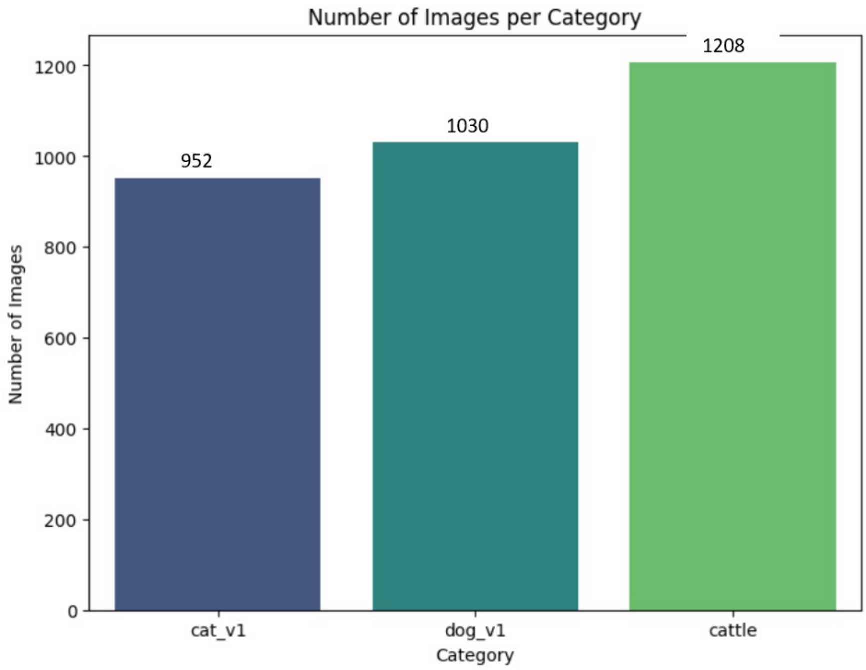

Figure 8: Total dataset before augmentation

The number of datasets for each animal breed before augmentation is shown below:

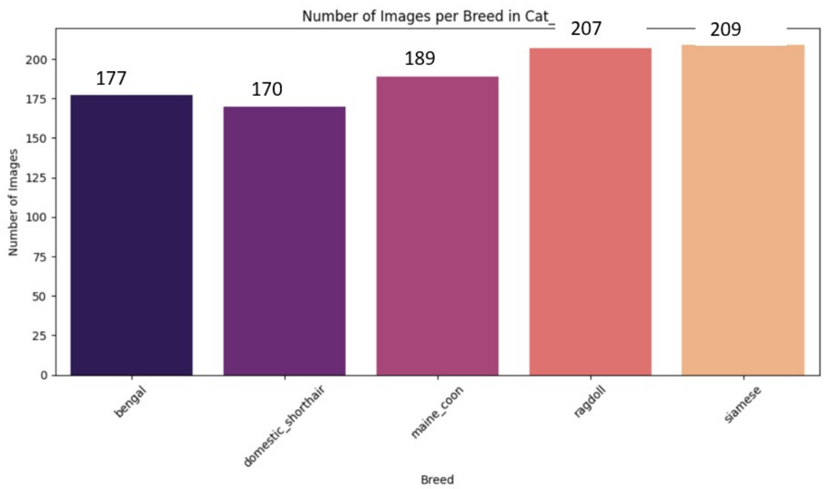

Figure 9: Number of dataset for different breeds of Cat

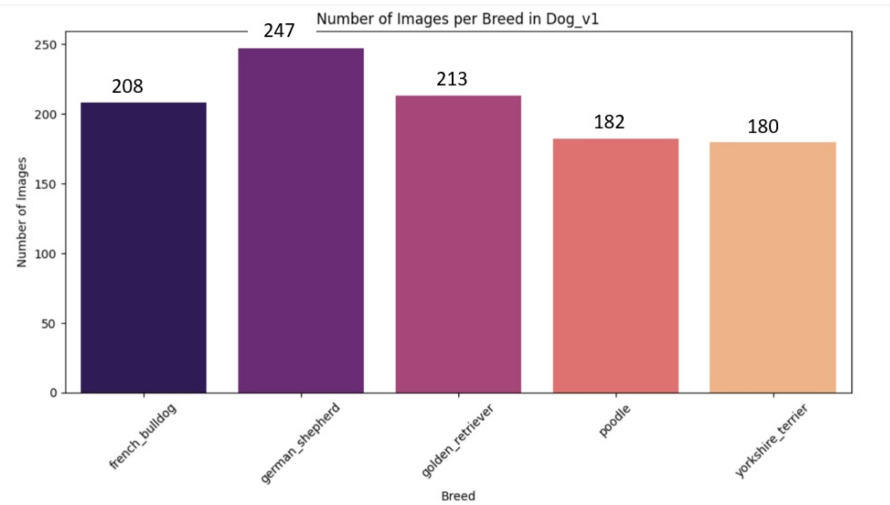

Figure 10: Number of dataset for different breeds of Dog

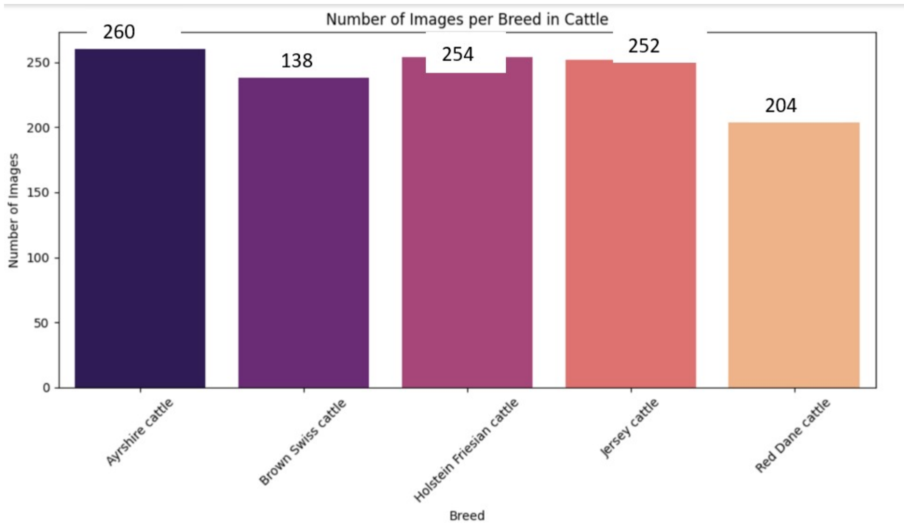

Figure 11: Number of dataset for different breeds of Cattle

Since the data sets were not balanced, to make them balanced we did data augmentation by applying transformations such as rotation, width shift, height shift, shear, zoom, and horizontal flip, we aimed to create a more diverse and representative dataset.This approach helped us improve the robustness and generalization of our model. After augmentation we made the balanced datasets with 500 numbers in each breeds.

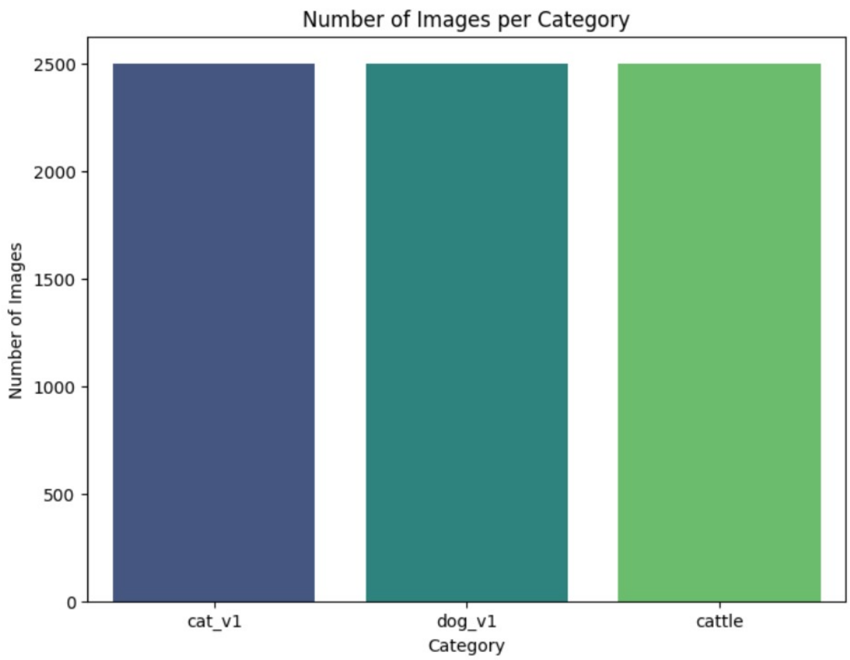

Figure 12: Number of dataset per cataegory after augmentation

The number of datasets for each animal breed after augmentation is shown below:

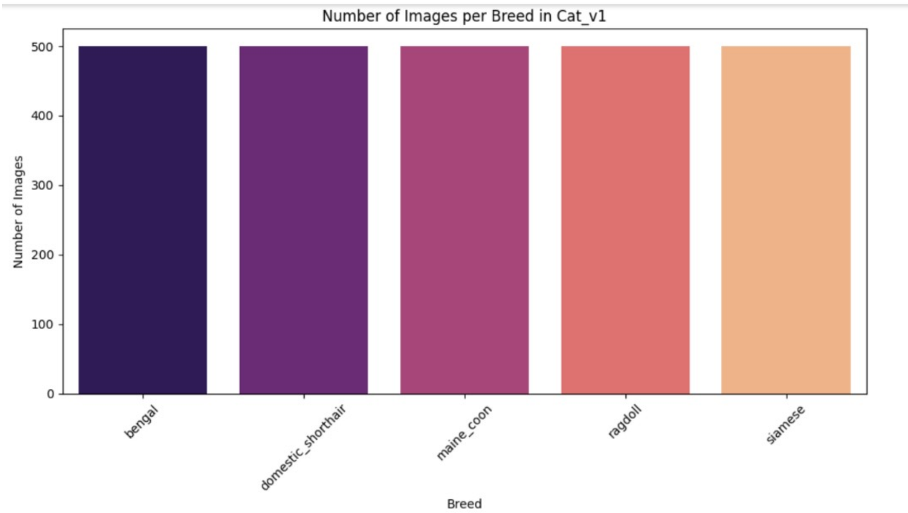

Figure 13: Number of dataset per breed for cat after augmentation

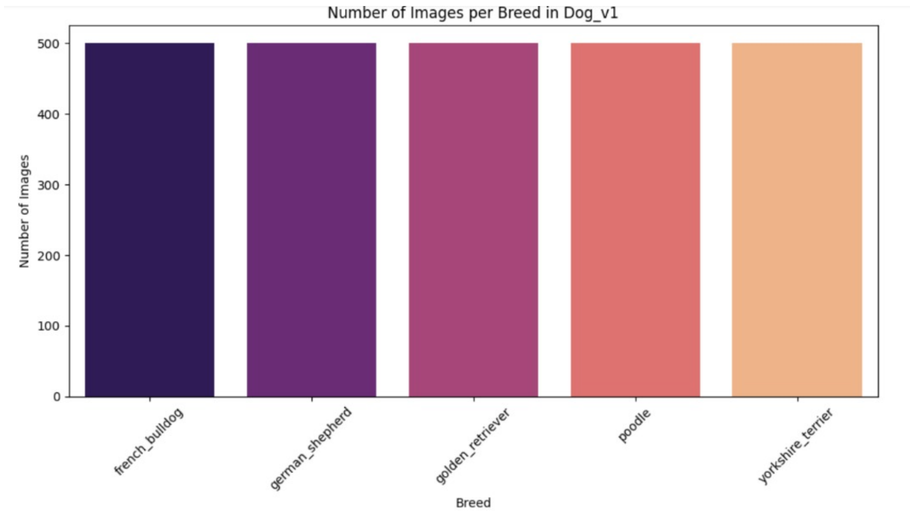

Figure 14: Number of dataset per breed for dog after augmentation

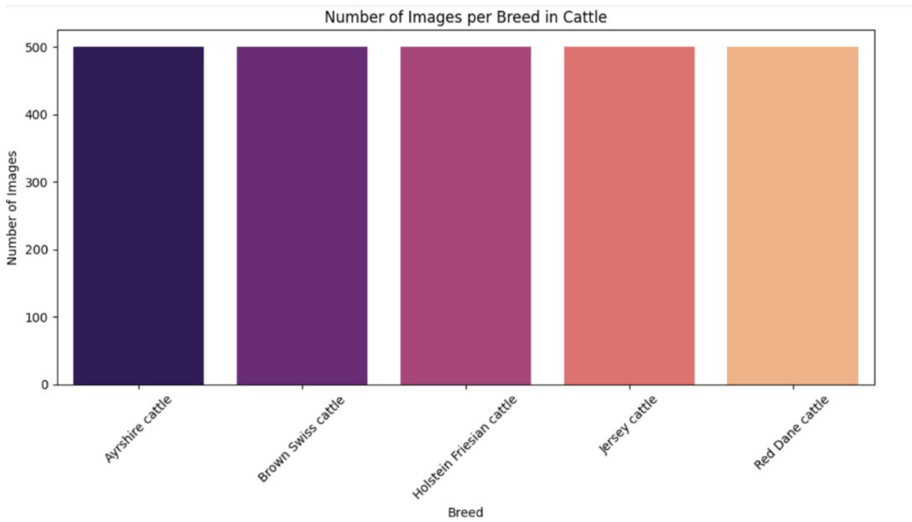

Figure 15: Number of dataset per breed for cattle after augmentation

Beside augmentation we also preprocessed our data using following techniques:

- **Resizing**: All images were resized to 224×224 pixels to fit the input size required by the CNN architectures.
- **Normalization**: Each pixel value was divided by 255 in order to bound the value of the pixels between 0 and 1, allowing for faster convergence during training.
- **JPG Format**: The images were converted to JPG format to ensure consistency and compatibility across the dataset.

After the preprocessing steps the dataset was split with 15% for testing and 75% for training and from the 75% allocated for the training data, we further split 15% for validation. This approach was adopted to ensures that our model is trained, validated, and tested on separate datasets, allowing us to accurately evaluate its performance.

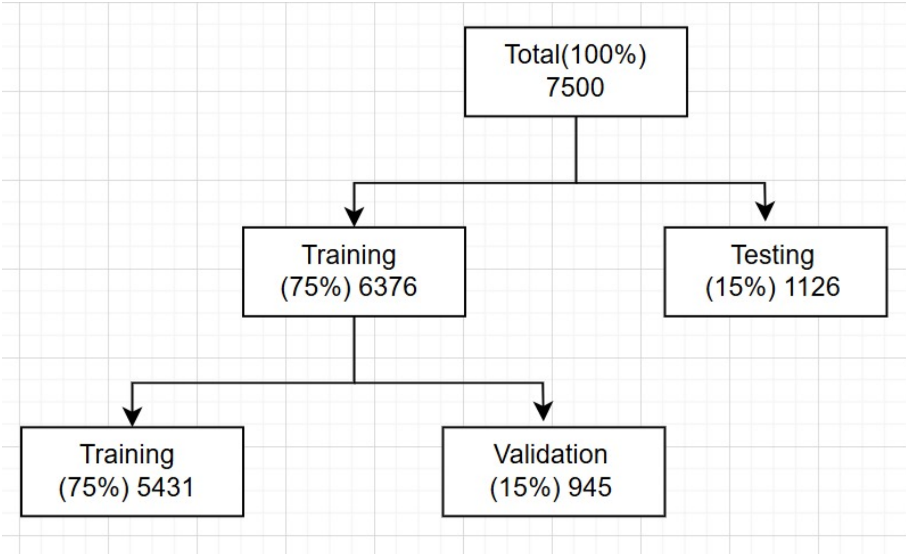

Figure 16: Spliting of Data

Figure 17: Sample images of animals: Cat, Dog, and Cattle

4.5.2 Algorithm

For the Domestic Animal Breed Identification project, we worked on Convolutional Neural Network (CNN) algorithm for model training. Since the project involves image datasets, CNNs are particularly well-suited for the image classification tasks due to their ability to capture spatial hierarchies or relationships and patterns within the images.They can automatically relevant features such as texture, shape, and patterns, which are essential for distinguishing different domestic animal breed.

Key Parameters

- **Input Layer:** The input to the model will be the images of domestic animals (Dogs, Cats, Cattles), resized to a standard resolution (e.g., 224x224 pixels) to maintain uniformity.

Convolutional Layers

- **Number of Filters:** Starting with 32 filters in the first layer and progressively increasing in deeper layers.

- **Filter Size:** 3x3 filters, commonly used for feature extraction, will be used to capture detailed patterns without losing too much spatial resolution.

Activation Function:

- **ReLU (Rectified Linear Unit):** ReLU is used to introduce non-linearity while being computationally efficient. It helps the model learn complex patterns by avoiding vanishing gradients.

Pooling Layers:

- **Max Pooling:** 2x2 max pooling layers will be used after certain convolutional layers to reduce the spatial dimensions of the feature maps, reducing computational load and preventing overfitting.

- **Fully Connected (Dense) Layers:** After flattening the feature maps, one or more dense layers will be used to make the final classification.

- **Output Layer:** The output layer will use a softmax activation function to classify the image into one of the predefined breed categories

- **Loss Function:** Categorical Cross-Entropy will be used as the loss function since this is a multi-class classification problem.

- **Optimizer:** Adam Optimizer will be employed due to its adaptive learning rate and robustness in handling noisy gradients, leading to faster convergence.

4.5.3 Model Architecture

Four different CNN architectures were utilized for breed classification from which VGG16 was selected for final model training:

- **VGG16**: Known for its simplicity and depth, VGG16 employs a series of convolutional layers followed by max-pooling, which captures spatial features effectively.
- **ResNet**: A deeper CNN with residual connections, ResNet allows for better gradient flow, enabling the network to be more effective in training deep layers without vanishing gradients.
- **AlexNet**: One of the earlier CNN architectures, AlexNet is shallower than ResNet but still capable of learning important spatial features, making it a good candidate for comparison.

- **LeNet-5**: One of the earliest CNN architectures, LeNet-5 was initially designed for digit recognition tasks. It consists of alternating convolutional and pooling layers, followed by fully connected layers. Though relatively shallow compared to modern architectures like VGG16 and ResNet, LeNet-5 is efficient in extracting basic spatial features, making it a solid candidate for simpler image classification tasks

4.5.4 Training and Evaluation

The models were trained on the dataset using a supervised learning approach. The Adam optimizer with learning rates of 0.0001 and 0.001 was used, along with categorical cross-entropy as the loss function. The models were trained for 50 epochs with a batch size of 32.

The performance of each architecture was evaluated using the following metrics:

- **Accuracy**: The proportion of correct breed predictions.
- **Precision**: The ratio of correctly predicted positive observations to the total predicted positives.
- **Recall**: The ratio of correctly predicted positive observations to all actual positives.
- **F1-Score**: The harmonic mean of precision and recall, balancing the trade-off between false positives and false negatives.
- **Confusion Matrix**: The confusion matrix was used to gain deeper insights into the classification performance, providing a clear visualization of how well the model predicted each breed category. It displays the number of correct predictions (true positives and true negatives) as well as the number of incorrect predictions (false positives and false negatives) for each class. This helped in understanding potential misclassifications between similar breeds, highlighting where the model struggled the most.

4.5.5 Tools and Technologies

1. **Programming Languages:** Python and React.js
2. **UI UX Designs:** Figma
3. **Version Control System:** GitLab
4. **Code editors:** Visual Studio Code, Jupyter Notebook, Google colab
5. **Machine Learning Frameworks:** tensorflow, Keras, OpenCV

5 Results and Discussions

5.1 Model Performance Comparison Table

Table 1: Comparison of Model Performance Metrics

Model	Accuracy	Precision	Recall	F1-Score
LeNet-5	71%	72.8%	71.04%	70.99%
VGG16	88%	89%	88%	88%
ResNet	46%	55%	46%	44%
AlexNet	77.35%	79.1%	77.35%	77.11%

5.2 Confusion Matrices

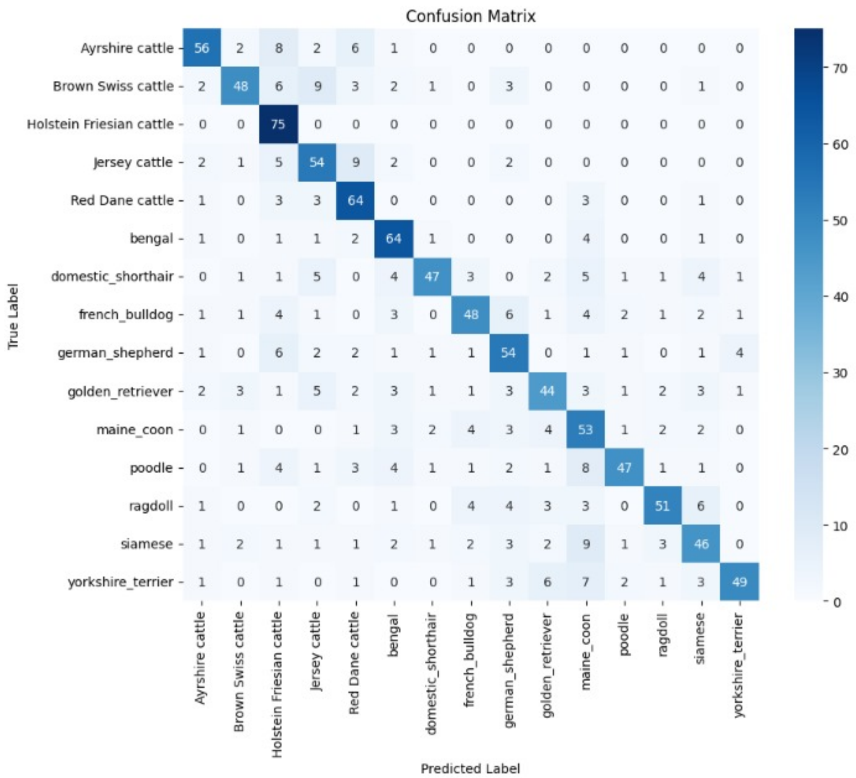

Figure 18: Confusion matrix of LeNet-5

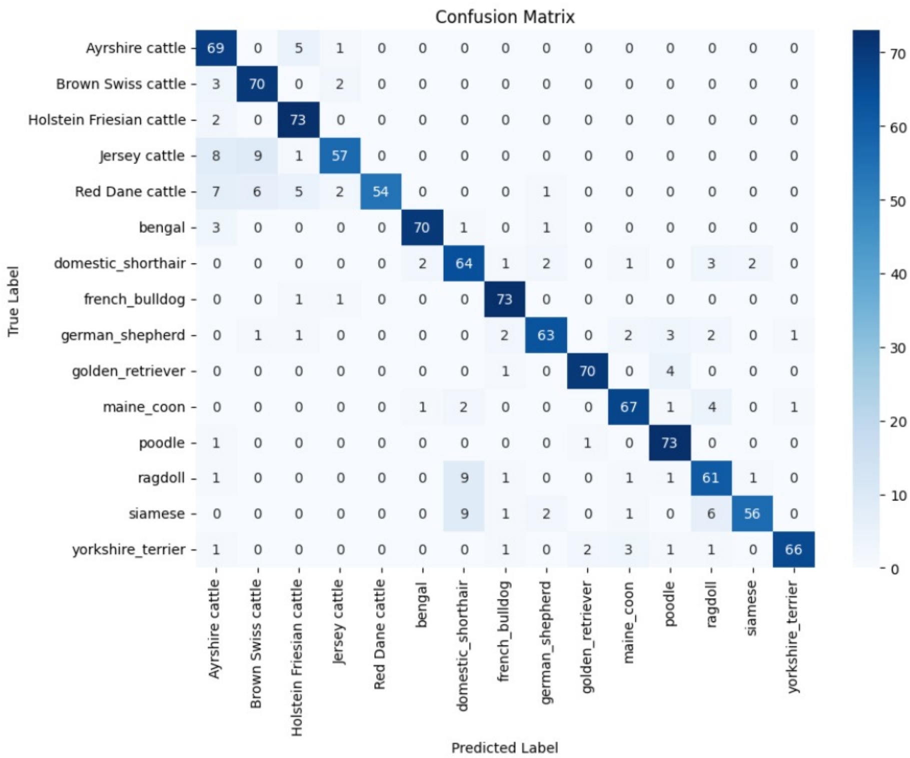

Figure 19: Confusion matrix of VGG16

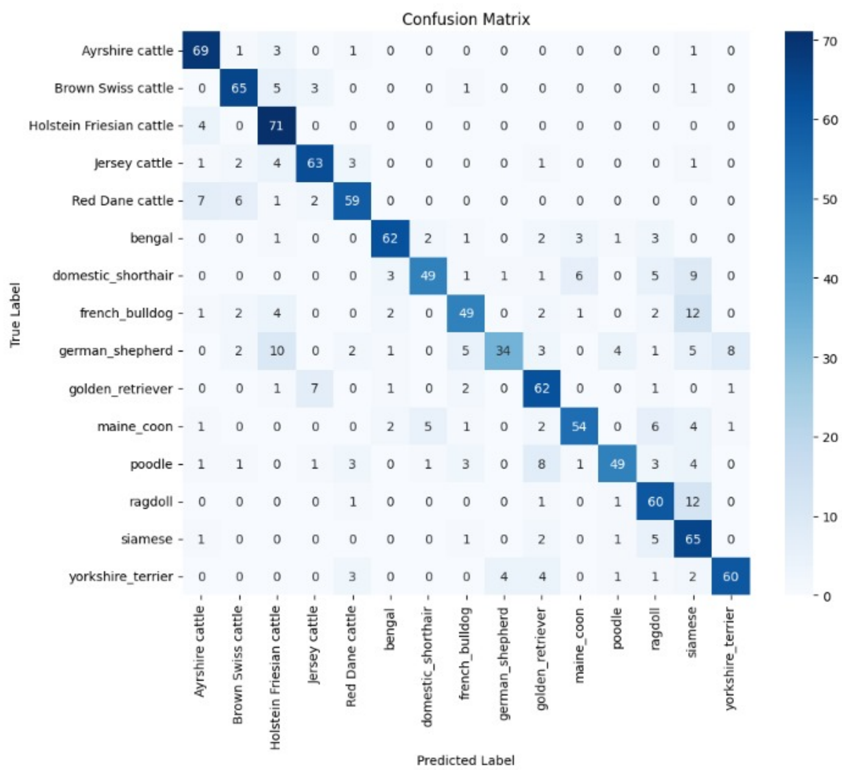

Figure 20: Confusion matrix of AlexNet

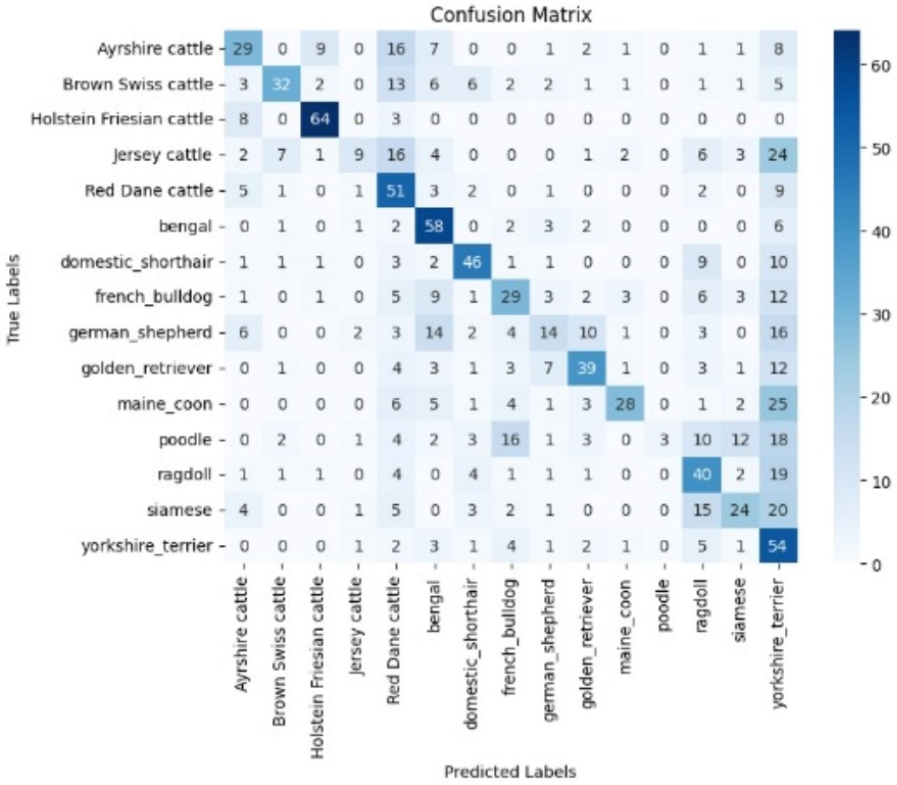

Figure 21: Confusion matrix of ResNet

1. VGG16

   Among most metrics, the performance of VGG16 was the best: 88% accuracy, 89% precision, 88% recall, and an F1-score of 88%. The good result emphasizes that VGG16 is very good at feature extraction and deep enough to capture even minute details of spatial hierarchies in images. Because of its high precision and recall, it has a good job in picking out true positive classes while minimizing false positives; hence, it is the most reliable for that task.

2. AlexNet

   AlexNet, while being a much older architecture, gave reasonably good results: an accuracy of 77.35%, precision of 79.1%, recall of 77.35%, and F1-score of 77.11%. While nowhere near as deep as VGG16, the architecture of AlexNet nonetheless captures vital features in its design for classification. Its somewhat moderate performance points to the simplicity compared to more recent architectures but still quite serviceable for image classification tasks.

3. LeNet-5

   The results for LeNet-5, one of the earliest CNN architectures, gave an accuracy of 71% with precision at 72.8%, recall at 71.04%, and an F1-score at 70.99%. Although it outperforms ResNet in this context, the shallower architecture of LeNet-5 reduces its capacity to catch critical features relevant for classifying between similar animal breeds. This reflects on the overall lower metrics of this network, although it still holds up quite decently on this dataset.

4. ResNet

   ResNet was the poorest performer, with an accuracy of 46%, a precision of 55%, a recall of 46%, and an F1-score of 44%. Generally, ResNet does very well in deeper learning because residual connections help to avoid the vanishing gradient problem in very deep networks. This may, however, be due to overfitting or low fine-tuning on the dataset since its performance in this case is lower. The relatively low recall and F1-score suggest that ResNet does poorly on false negatives, failing to identify many instances correctly.

Since VGG16 has proved to be best model while comparing with the other models, we use above mentione evaluation matrix to further evaluate the VGG16 model:

Classification Report:

	precision	recall	f1-score	support
Ayrshire cattle	0.73	0.92	0.81	75
Brown Swiss cattle	0.81	0.93	0.87	75
Holstein Friesian cattle	0.85	0.97	0.91	75
Jersey cattle	0.90	0.76	0.83	75
Red Dane cattle	1.00	0.72	0.84	75
bengal	0.96	0.93	0.95	75
domestic_shorthair	0.75	0.85	0.80	75
french_bulldog	0.91	0.97	0.94	75
german_shepherd	0.91	0.84	0.88	75
golden_retriever	0.96	0.93	0.95	75
maine_coon	0.89	0.88	0.89	76
poodle	0.88	0.97	0.92	75
ragdoll	0.79	0.81	0.80	75
siamese	0.95	0.75	0.84	75
yorkshire_terrier	0.97	0.88	0.92	75
accuracy			0.88	1126
macro avg	0.88	0.88	0.88	1126
weighted avg	0.89	0.88	0.88	1126

Figure 22: Evaluating the Precision, Recall, and F1-Score for Various Animal Breeds

```
Overall Precision: 0.89
Overall Recall: 0.88
Overall F1-Score: 0.88
Support: 1126.0
Accuracy: 0.88
```

Figure 23: Performance metrics of a VGG16

Overall, these results indicate that complex networks such as VGG16 will be better suitable for tasks of breed classification since the subtle differences in pictures have to be grasped. However, simpler architectures like AlexNet and LeNet-5 still yield competitive results and therefore may be preferred in limited computational resource scenarios.

6 Interface Design

After constructing an initial idea of how the website would function, we designed its interface. The prototype was developed using Figma.

1. **Landing Page:** The user will be directed to the landing page of the Animal Breed Identification website.

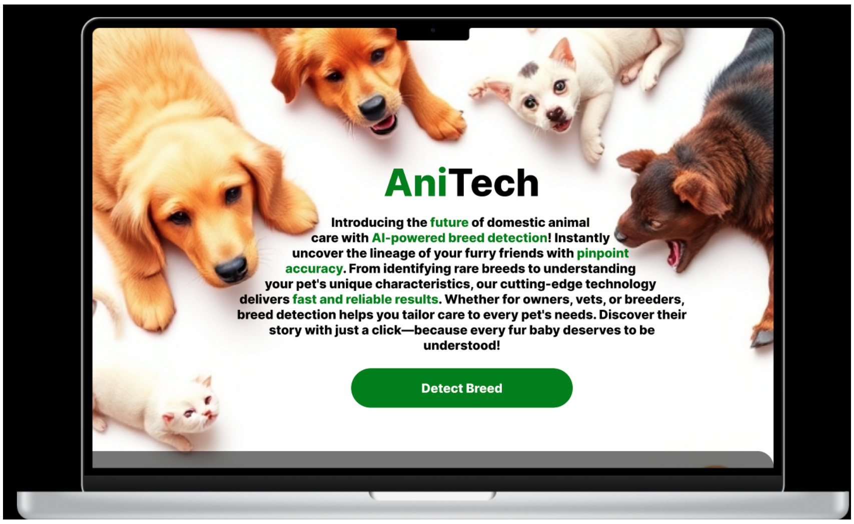

Figure 24: Landing Page

2. **Upload Image:** To detect the breed, the user needs to click on the "Detect Breed" button on the landing page. The system will then redirect the user to the upload-image page. The user can click on the "Upload Image" button to select an image from their system.

Figure 25: Upload Image Page

3. **Check Breed:** After uploading an image, the user clicks on the "Check Breed" button to detect the breed. The system will identify whether the uploaded image corresponds to a cat breed, dog breed, cattle breed, or if the breed is not recognized.

Figure 26: Check Breed Page

4. **Result:** Once the system processes the image, the user will be presented with the result, indicating the type of breed identified or showing that the breed could not be found.

Figure 27: Result Page

7 Conclusion

This study has proven that deep learning models are suitable for the classification of domestic animal breeds and in particular, cats, dogs, and cows. Of the models considered in this study, VGG16 achieved the highest accuracy rate of 88% proving to be effective in outlining the intricate details that are critical when identifying the specific dog breeds. AlexNet and LeNet-5 also yielded decent results, hence can be applied in scenarios where there are limited processing capabilities. ResNet on the other hand exhibited the least classification performance and this is due to model's overfitting or extreme lack of training. The results presented in this study emphasize the importance of deep learning methods to improve the preciseness and efficiency of breed classification which is very critical in veterinary medicine, animal farming and conservation.

More exploration of ResNet could improve the model's performance, addressing its current weaknesses. Further research could also consider other improved networks like EfficientNet or DenseNet that could yield high accuracy with less computational investment. The training dataset could also be enhanced by other species and breeds to increase proper generalization in various animals. Finally, the practical implementation of this particular system is in the advantage of animal welfare, breeding and conservation, and addresses some of the challenges faced in identification of animal breeds with the support of technology.

8 Milestones

8.1 Gantt Chart

The following Gantt chart provides a detailed timeline of the project activities and milestones.

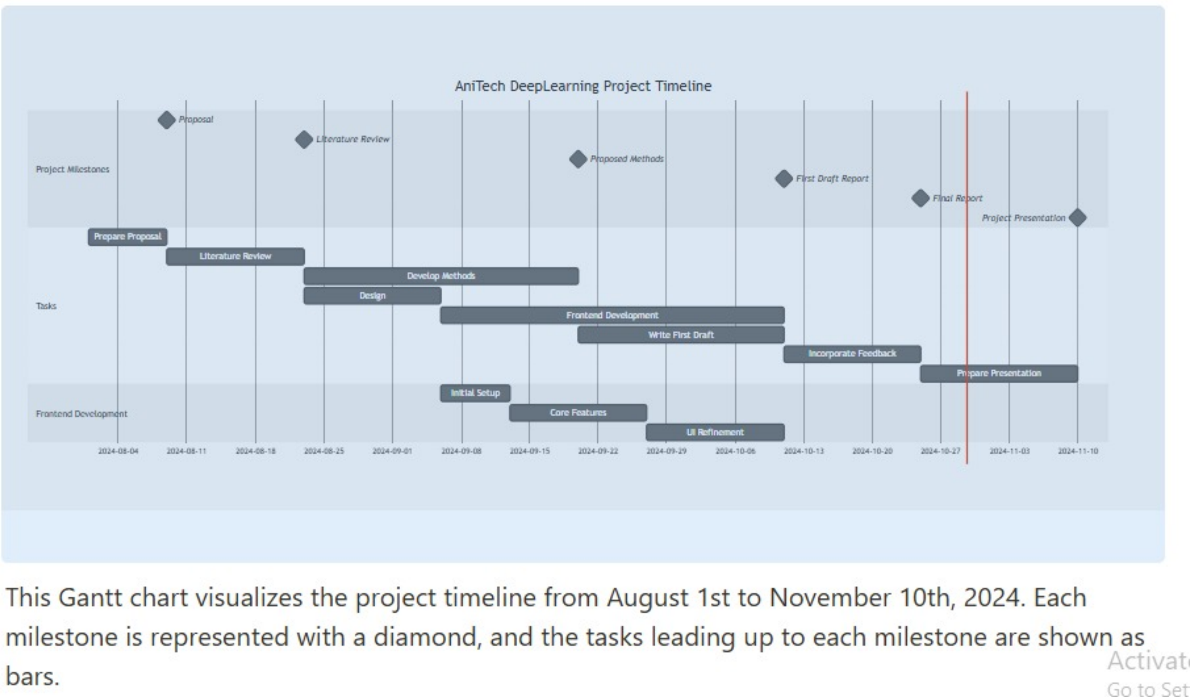

Figure 28: Project Gantt Chart for Domestic Animal Breed Identification

GitLab Project Repository:
https://gitlab.com/mine7662321/group1_domesticanimalbreedidentification

References

[1] C. Ma, X. Sun, C. Yao, M. Tian, and L. Li, "Research on sheep recognition algorithm based on deep learning in animal husbandry," in *Journal of Physics: Conference Series*, vol. 1651, p. 012129, IOP Publishing, 2020.

[2] T. Battu and D. S. R. Lakshmi, "Animal image identification and classification using deep neural networks techniques," *Measurement: Sensors*, vol. 25, p. 100611, 2023.

[3] D. D. Pukale, K. Dudhane, S. Dawghat, A. Deshmukh, and S. Deshmukh, "Wild animal detection and identification using deep learning," *International Journal of Creative Research Thoughts (IJCRT)*, vol. 11, no. 7, pp. a477–a482, 2023.

[4] K. Jagadesh, P. Kavya, A. Lavanya, A. L. Priya, B. Harshini, and D. G. Jerith, "Dog's breed prediction using deep learning," *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*, vol. 12, no. VI, pp. 1161–1166, 2024.

[5] D. Dandeniya, B. Wickramasinghe, and C. Dasanayaka, "A web-based application for snake species identification using vision transformer and cnn-based ensemble meta classifier," in *2022 IEEE Pune Section International Conference (PuneCon)*, pp. 1–5, IEEE, 2022.