AUTHORS Joseph Babalola-Davies, Grace Y. Lin, Rahul Reddipalli, Youssef Magdi (Group 12)
AFFILIATIONS University of Surrey, Deep Learning and Advanced AI (COM3025)

Hierarchical Dog & Cat Breed Classification Using Object Detection and Fine-tuned Visual Categorisation

Image source: [i1]

1. Introduction and Objectives

INTRODUCTION

Furry companions are a big part of many people's lives. In fact, almost 60% of UK households own at least one cat or dog (almost 25 million households). Contrastingly, almost 2 million dogs and cats enter shelters every year, and for dogs, 50% are stray dogs who tend to be mixed breeds. The UK has seen a 6% increase in stray dogs and cats entering shelters [1]. Shelters classify stray dogs and cats based on their physical appearance as information about their heritage is usually not available [2]. The high cost of conducting DNA tests increases the difficulty of identifying the breed of an animal. This could result in dangerous and harmful treatment of misidentified animals, who could be given unsuitable treatment. Additionally, identifying the breed of a stray can make shelters aware of breed-specific diseases or health conditions that the dog or cat may be more susceptible to.

Image source: [i2]

PROJECT AIM

The **aim** of this project is to research and design a machine learning classifier that is able to identify cats and dogs in a photo and return the breeds of the dogs and cats present. Additionally, it will provide likely predictions for mixed-breed dogs and cats.

GOALS & OBJECTIVES

We introduce the following criteria that must be met in order for this project to be considered a success:

- Findings discovered from the research stage of the project are displayed in a clear and understandable manner
- At least two different model architectures for image classification are experimented with
- An object detection model is implemented that can identify the dog and cats from an image (such as in the picture on the left)
- An image classification model is implemented that can identify the breeds of dogs and cats found from a larger image
- Evaluation and comparison of the final methods in this poster must be explained thoroughly

Image source: [i3]

2. Problem Analysis

THE CHALLENGE OF BREED CLASSIFICATION

Breed classification is an example of a fine-grained visual categorisation problem. This class of problems refers to the challenge of identifying intricate details in an image in order to distinguish between similar objects. In the scope of our project, this refers to pets that may look similar, such as differentiating between a labrador retriever and a golden retriever. Research in this area is abundant, with many new technologies and advances being made. The following research papers were reviewed to discover up-to-date techniques and the most robust and best-performing machine learning algorithms for breed classification.

1) Dog Breed Classification using Part Localization: [3]

Jiu et. al. tackle the identification of dog breeds by focusing on the key points of a dog's face, such as its ears, eyes, and nose shape. This is done by using a sliding window detector to find the dog's face and then tracing the location of its eyes and nose to determine the location of the other facial features.

They trained their model on a dataset consisting of 8,351 images collected across a multitude of sources including Flickr, Image-Net, and Google across 133 dog breeds in total. They use a Support Vector Machine (SVM) to detect a central location of the face with fixed positions of features relative to the "centre point" which are chosen to approximately align with the geometry of a dog's face."

Overall, their model reaches 67% accuracy on the first prediction, increasing up to 93% after 10 passes. Their findings highlight the potential of classifying dogs based on certain features, rather than their body as a whole, as a dog's body shape is difficult to identify as it is not present in most images, and is also not a distinguishing feature between breeds besides for extreme cases such as chihuahuas.

2) Classification of Animal Breeds using Multi-Part Convolutional Neural Networks (MP-CNN) [4]

Divya Meena et al. published their framework for classifying animals and a subset of breeds that does not only focus on an animal's face. First, the image is analysed to detect whether an animal is present. Then the images of the identified animals are split into patches, which highlight key distinguishable features of an animal's body as a whole. Examples include their extremities or fur patterns.

Since the object identification is separate from the part selection, two CNN models are used, where the first CNN (a pre-trained model called FilterNet) is used for object detection, while a different pre-trained model named DomainNet is used for the fine-grained classification. FilterNet also checks which patches are part of the animal and which are part of the background. A dataset of 35,992 images across 27 classes was used for training. A variety of experiments using said dataset, such as class imbalance or unbiased datasets, were also performed.

The MP-CNN proposed reached an average of 96-99.5% across their experiments. While their model was extremely accurate, it should be noted that their proposed framework is not lightweight and requires a significant amount of computational power and time for model training.

3. Relevant Datasets

As mentioned previously, certain breeds of dogs may look similar to other breeds, but the breed prediction method that is developed should be able to distinguish them. To achieve this functionality, and in any machine learning model development, it is imperative to find a robust and comprehensive dataset for training, validation, and testing.

SUITABLE DATASETS

To train the classification model, a large dataset of dog and cat images is required, along with labels for the breed of each image. The following two datasets were decided upon for training the breed classification model.

Dogs: Stanford Dogs Dataset [5] - contains 20,580 images - split into 17,493 for training and 3,087 for testing across 120 dog breed classes.

Cats: Cat Breed Dataset [6] - contains 126,605 images - split into 107,617 for training, 18,990 for testing across 67 cat breed classes.

The images of the Stanford Dogs Dataset also contain bounding box data, however, they were not used for this project. It should also be noted that the Cat Breed Dataset is not balanced across all classes.

Image source: [i4]

4. Object Detection

EXPLORING OBJECT DETECTION APPROACHES

The first component of our project is a method to detect and locate a dog or a cat in an image.

To first find a dog or a cat within a larger image, an object detection technique must be applied. When designing an object detection method, there are a few considerations to make. There are multiple pre-trained models and frameworks for object detection, but we must also consider creating a custom solution and training it.

To create a custom solution, a large collection of images of objects must be collected, and the bounding boxes around every dog and cat in those images must be manually drawn. Then a model can be trained to draw bounding boxes and identify cats and dogs in any given image.

Using a pre-trained model, however, offers the benefits of being able to build upon a well-trained base without the need to collect large amounts of training data. Pre-trained models are often trained on incredibly large datasets that require many hours of modelling.

Object detection methods typically belong to one of two classes: one-stage detection models, and two-stage detection models. The key features of both methods are provided in the table below.

ONE-STAGE DETECTION	TWO-STAGE DETECTION
- Able to identify all objects in a single pass of the image	- Detects regions in which a classifiable object may exist (regional proposals)
- Breaks the image into a grid, each grid cell has its own detector that returns a vector of the properties of its contents (e.g., the probability of an object existing in the cell)	- Uses a new algorithm to classify the subset image within the proposals
- Objects of interest can be bounded and detected simply by analyzing the vector data	- Multiple objects of interest can can result in a singular regional proposal over all the objects, rather than a single region per object
- Generally a faster and more popular object detection method than two-stage	- Generally slower as it requires data to be passed through 2 separate models

FINAL IMPLEMENTATION CHOICE

We initially explored a few state-of-the-art model options [7], including Faster R-CNN (region-based CNN features) [8], which is a two-stage detection algorithm that performs accurately but slowly. It is important to note that no singular algorithm performs the best across all tasks, as performance is tied to the use case and application of the model. For this project, we chose to implement YOLOv5 [9] which is a popular one-stage detection model. The architecture of YOLOv5 consists of a backbone of a pre-trained deep CNN (New CSP-Darknet53). The general architecture of YOLOv5's predecessor, YOLOv4 [10], is shown below.

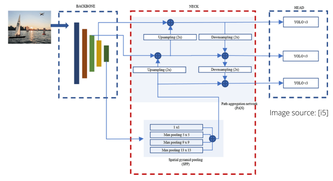


Image source: [i5]

We chose to use YOLOv5 pre-trained on the COCO dataset as training a robust custom object detection model required too much computational power, and manually creating bounding boxes for image training was unfeasible for the timeframe of this project.

During testing, YOLOv4 is able to accurately identify a dog or a cat from an image and return its location on an image as a bounding box. We then take the dog or cat image to perform the next step: **breed classification.**

5. Data Pre-processing

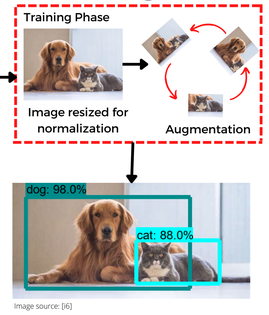


Image source: [i6]

Before the images from the datasets (left) are passed to the model for training, the photos are pre-processed and augmented. This is to reduce the possibility of the model overfitting data and improve the model's ability to generalise and abstract.

First, the images are normalized and resized. Then we apply random augmentations, specifically cropping and rotation.

The datasets are then split into training and validation sets.

6. Image Classification

For this component of the pipeline, we researched, designed, and implemented multiple image classification techniques to compare their performance for our use case. As mentioned previously, different methods will perform differently depending on the application.

We explored the performance of a simple convolutional neural network model with custom layers, a CNN model using the InceptionV3 network as a backbone, and also a model implementing vision transformers.

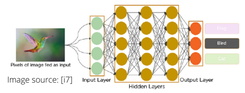


Image source: [i7]

1. CNN MODEL

This custom model is a lightweight CNN model based on ideas gathered from lectures and research

Architecture:

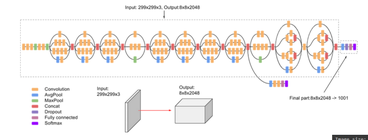
- Rescaling Layer (Input shape 300 x 300 x 3)
- 3 Conv2D Layers (3 x 3 Kernel, with 16, 32, 64 neurons respectively)
- Max Pooling Layer (2 x 2 Kernel) every Convolutional Layer
- Dropout Layer rate: 0.2 at layer 5, 8 and 10 respectively
- Flattening Layer
- Fully Connected Layer (128 neurons)
- Output Layer depending on cats or dogs (67 or 120 classes respectively)

2. INCEPTIONV3 BACKBONE:

Architecture:

- Inception V3 Layers (42 in total, mixture of Conv, Max Pooling Layers, etc.)
- Global Average 2D Pooling Layer
- Dropout Layer rate: 0.2
- Fully Connected Layer (120 neurons)

We use the Inception V3 model, as introduced by Szegedy et al in [11], pre-trained on the ImageNet database which is a state-of-the-art CNN backbone for image analysis tasks and object detection. We then add a light head to the model for our application.



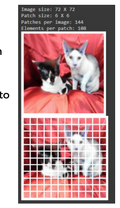
3. VISION TRANSFORMERS (VIT):

Architecture:

- Input layer (Input shape 128 x 128 x 3))
- Patches (4)
- Patch Encoder (4, 64)
- Transformer Layer
- Normalization Layer
- Dropout Layer (0.4)
- MLP Layer
- Dense Output Layer Layer (67 or 120)

A ViT model as introduced in [12] is a newer model architecture that utilises internal transformers that link a relationship between token pairs. The model adapts transformer technology for image analysis by generating image "patches" which represent a token, linked to a respective layer. Patches are used to split the image into a grid in order to capture the animal's features more accurately.

ViT requires a significant amount of computational power especially for a large dataset, so we train the model on reduced images and for fewer epochs. This will provide faster training for the model and help reduce the load on the RAM and GPU during training.



7. Challenges Faced

During the development of the solutions, several challenges and issues occurred that impacted the results of our implementations. Key challenges faced include:

- **Unbalanced cat dataset:** The Cat Breed dataset, while consisting of a large number of images, is largely unbalanced across its classes, with some having as many as 5000 and others as little as 30. This can cause a bias in the models as they will have been trained to recognise certain breeds more accurately than others.
- **Image corruption:** In the process of retrieving the images for use in the notebooks, some images would be extracted incorrectly and become unusable for training. As we trained the models using data loaders the corrupted images would interrupt model training and throw an error. To accommodate for this, a function was created to iterate through the images and locate the corrupted files so that would be removed.
- **Overfitting:** As can be seen in Section 8, some of the models overfit whilst training on the data, where models fit against training data too closely and reduce their ability to generalise. The final implementation added early stopping, learning rate reduction, and model saving measures as hyperparameters during training. This was implemented to mitigate the effect of overfitting.
- **Transformer Inaccuracy:** During testing, ViT models developed are extremely inaccurate. This is because generally, Vision Transformers require a large amount of data for optimal model results. Attempts were made to increase the dataset count via augmentation, however, this increased processing times to an inordinate amount, making training infeasible due to time constraints.

Image source: [i8]

8. Experiment Results

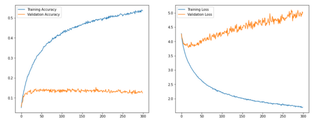
The following results were found for the models when using the Stanford Dogs Dataset:

MODEL NAME	LEARNING RATE	ACCURACY
CNN	0.001	54.02%
Inception V3	0.00001	84.95%
Vision Transformers	0.001	8.80%

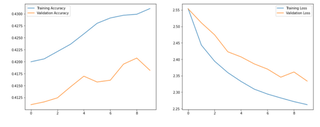
The following results were found for the models when using the Cat Breed Dataset:

MODEL NAME	LEARNING RATE	ACCURACY
CNN	0.001	43.11%
Inception V3	0.001	45.60%
Vision Transformers	0.001	42.54%

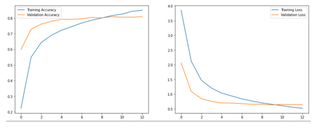
CNN training metrics (Stanford Dogs):



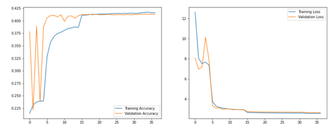
CNN training metrics (Cat dataset):



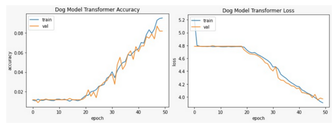
Inception V3 training metrics (Stanford dogs)



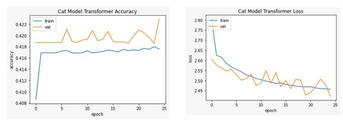
Inception V3's accuracy (left) and loss (right) for cats:



ViT' accuracy (left) and loss (right) for dogs:



ViT's accuracy (left) and loss (right) for cats:



9. Conclusions

ANALYSIS

From our experimentation, the CNN model with a pre-trained InceptionV3 backbone outperformed the other models on the given task. This model had the highest validation accuracy for both datasets.
Vision Transformers seemed to perform the worst out of the three models, despite research showing that they should perform to a similar or higher standard than DCNNs.
Another method to have improved the accuracy of the models would have been to apply methods to prevent overfitting, such as early stopping, to the other models outside of the Inception V3 ones.

EVALUATION

- Research completed has been discussed and design decisions are explained in-depth
- Three model architectures for breed classification were experimented with
- YOLOv5 is implemented as an object detection model is implemented that can identify the dog and cats from an image successfully
- A CNN model based on InceptionV3 architecture is proposed for the breed classification component
- Evaluation and comparison of the final methods in this poster must be explained thoroughly

CONCLUSION

Overall, this project has largely achieved the aims and objectives as set out in the Introduction. We successfully implemented an object detection model that can identify a dog or a cat from an image, and experimented with different approaches to classify these animals into their respective breed categories. We present these results in a concise manner for the reader's convenience and display key graphs and metrics that are easy to interpret. However, it was found that the models struggled to identify cats very accurately compared to dogs. This could be improved by further training or exploring different dataset options. Continuing from this project, it would be beneficial to explore ways to implement transformer technology for images more successfully. Future work could also involve implementing the pipeline suggested into a practical solution such as a mobile application.

10. References

[1] Cosgrove, N., 2022. 12 UK Animal Shelter Statistics & Facts to Know in 2022: Benefits, Facts & More | Pet Keen. [online] Pet Keen. Available at: <https://petkeen.com/animal-shelter-statistics-uk/#3_The_number_of_stray_cats_and_dogs_arriving_at_shelters_in_the_United_Kingdom_has_increased_by_6_over_the_last_ten_years> [Accessed 25 May 2022]
[2] Gunter, L., Barber, R. and Wynne, C., 2018. A canine identity crisis: Genetic breed heritage testing of shelter dogs. PLOS ONE, 13(8), p.e0202633.
[3] J. Liu, et al. Dog breed classification using part localization. Computer Vision: ECCV 2012, pages 172–185, 2012.
[4] Divya Meena, S. and Agilandeeswari, L., 2019. An Efficient Framework for Animal Breeds Classification Using Semi-Supervised Learning and Multi-Part Convolutional Neural Network (MP-CNN). IEEE Access, 7, pp.151783-151802.
[5] Vision.stanford.edu. 2022. Stanford Dogs Dataset. [online] Available at: http://vision.stanford.edu/aditya86/ImageNetDogs/main.html [Accessed 25 May 2022].
[6] Kaggle.com. 2022. Cat Breeds Dataset. [online] Available at: <https://www.kaggle.com/datasets/ma7555/cat-breeds-dataset> [Accessed 25 May 2022].
[7] P. Rajeshwari, P. Abhishek, P. Srikanth, T. Vinod "Object Detection: An Overview" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 24566470, Volume-3 | Issue-3, April 2019, pp.1663-1665, URL: https://www.ijtsrd.com/papers/ijtsrd23422.pdf
[8] S. Ren, K. He, R. Girshick, and J. Sun, 'Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks', arXiv, arXiv:1506.01497, Jan. 2016. doi: 10.48550/arXiv.1506.01497.
[9] 'YOLOv5 (6.0/6.1) brief summary · Issue #6998 · ultralytics/yolov5', GitHub. https://github.com/ultralytics/yolov5/issues/6998 (accessed May 26, 2022).
[10] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, 'YOLOv4: Optimal Speed and Accuracy of Object Detection', arXiv, arXiv:2004.10934, Apr. 2020. doi: 10.48550/arXiv.2004.10934.
[11] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, 'Rethinking the Inception Architecture for Computer Vision', arXiv, arXiv:1512.00567, Dec. 2015. doi: 10.48550/arXiv.1512.00567.
[12] A. Dosovitskiy et al., 'An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale', arXiv, arXiv:2010.11929, Jun. 2021. doi: 10.48550/arXiv.2010.11929.

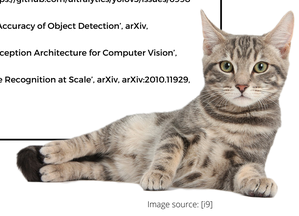


Image source: [i9]

IMAGE SOURCES

[i1] https://www.schoolfinder.com/Discover/Article/I/5082/How-to-De-stress-with-Dogs-and-Cats
[i2] https://blog.invisiblefence.com/tips-for-living-with-both-cats-and-dogs
[i3] https://www.pngwing.com/en/free-png-tjdbd
[i4] https://www.freepnglogos.com/pics/dog
[i5] https://www.mathworks.com/help/vision/ug/yolov4architecture.png
[i6] https://www.mygreatlearning.com/blog/yolo-object-detection-using-opencv/2.
[i7] https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/
[i8] https://www.pngkey.com/maxpic/u2a9o0u2w7r5r5i1/
[i9] https://www.freepnglogos.com/pics/cat