YOLOv8 – Based Cat and Dog Detection Using Deep Learning

Nishalini M[1], Shobhik R, Shiva Prakash M, Prathikshana E K, Dr. Amsaveni M
[1]Student, Department of Computer Science with Artificial Intelligence and Data Science, KPR College of Arts, Science and Research, Coimbatore, India.
E-mail: nishalini01m@gmail.com; Shobhikraja222009@gmail.com; Shivaprakash9585@gmail.com; eswaranprathi@gmail.com; amsaveni.m@kprcas.ac.in

Abstract:
Object detection is an important application of computer vision that enables automated identification and localization of objects in images and videos. This study presents a deep learning-based Cat and Dog Detection System using YOLOv8 (You Only Look Once version 8). The main objective of the proposed system is to accurately detect and distinguish cats and dogs from input images using bounding boxes and class labels. A dataset containing images of cats and dogs is used for training and validation. The images are preprocessed and annotated before being provided to the YOLOv8 model for learning. The model is trained using Google Colab with GPU acceleration, allowing efficient deep learning model training. YOLOv8 is selected because it provides a suitable balance between detection accuracy and processing speed, making it useful for real-time object detection applications. The performance of the trained model is evaluated using important metrics such as Precision, Recall, mAP@50, and mAP@50–95. The experimental results show that the proposed model achieves an overall Precision of 92.3%, Recall of 84.0%, mAP@50 of 89.9%, and mAP@50–95 of 73.0%. The model demonstrates effective detection performance for both cat and dog classes under different image conditions. The proposed system can be applied in pet monitoring, animal recognition, smart surveillance, automated image analysis, and educational computer vision applications. The results demonstrate that YOLOv8 provides an efficient and reliable approach for automatic cat and dog detection, while future improvements can focus on increasing robustness under occlusion, poor lighting, and complex backgrounds.

Keywords— YOLOv8, Cat Detection, Dog Detection, Object Detection, Deep Learning

1. INTRODUCTION

Object detection is an important area of computer vision that enables machines to identify and locate objects within images and videos. With the rapid development of deep learning, object detection systems have become more accurate, faster, and suitable for real-world applications. Among the different object detection algorithms, YOLO (You Only Look Once) is widely used because of its ability to detect multiple objects quickly in a single image.

Cats and dogs are two of the most commonly found domestic animals, and automatically distinguishing between them is a useful application of image-based object detection. Traditional image classification methods generally identify the main object in an image but may not provide its exact location. Object detection methods overcome this limitation by identifying the object class and drawing a bounding box around the detected object.

In this project, YOLOv8 is used to develop an automated system for detecting cats and dogs. YOLOv8 is a modern deep learning model designed to provide high detection accuracy while maintaining fast inference speed. The model is trained using a dataset containing images of cats and dogs and learns important visual features such as shape, texture, body structure, and appearance.

The proposed system is implemented using Python and Google Colab, with GPU acceleration used for model training. The trained model is evaluated using performance measures such as Precision, Recall, mAP@50, and mAP@50–95. The experimental results demonstrate that the system can effectively detect and distinguish cats and dogs in different images.

The developed system can be useful in applications such as pet monitoring, animal recognition, smart surveillance, wildlife and domestic-animal image analysis, and automated image processing.

2. RELATED WORKS

Table 1. Comparative Analysis of Cat and Dog Object Detection Methods

Author	Method	Accuracy / Performance	Limitations
Ren et al. [1]	Faster R-CNN	High detection accuracy	Computationally expensive
Liu et al. [2]	SSD	Good detection performance	Lower accuracy for small objects
Redmon et al. [3]	YOLO	Real-time detection	Difficulty with small objects
Bochkovskiy et al. [4]	YOLO v4	Improved speed and accuracy	High computational requirements
Jocher et al. [5]	YOLO v5	Fast and efficient	Depends on dataset quality
Wang et al. [6]	YOLO v7	High speed detection	Requires GPU resources
Ultralytics [7]	YOLO v8	High accuracy and speed	Requires annotated training data
Proposed work	YOLO v8 Cat-Dog Detection	mAP@50 = 89.9%	Sensitive to occlusion and poor-quality images

3. PROPOSED WORK

This study proposes a YOLOv8-based cat and dog detection system to automatically identify and locate cats and dogs in images. The dataset is prepared with annotated images of both classes and divided into training and validation sets. The YOLOv8 model is trained using Google Colab with GPU acceleration and evaluated using Precision, Recall, mAP@50, and mAP@50–95. The proposed model achieved 92.3% Precision, 84.0% Recall, 89.9% mAP@50, and 73.0% mAP@50–95. The system generates bounding boxes, class labels, and confidence scores for detected cats and dogs, providing an efficient solution for automated animal detection.

3.1 Data Collection

The dataset consists of images containing two target classes: Cat and Dog. Each image is prepared for object detection by identifying the location of the animal and assigning the appropriate class label. The dataset contains variations in animal appearance, pose, orientation, scale, and background. These variations are important because a robust object detector should not depend on a single visual pattern. For model validation, the trained system was evaluated on 349 images containing 420 object instances. Among these validation instances, 202 correspond to cats and 218 correspond to dogs.

Table 2. Validation Dataset Distribution

Class	Validation Images	Object Instances
Cat	184	202
Dog	177	218
Total	349	420

3.2 Data Pre-Processing

3.2.1 Image Resizing

All input images are resized to 640 × 640 pixels, which is the image size used during YOLOv8 model training. Resizing provides a consistent input dimension and helps the model efficiently extract visual features.

Equation (1): $I' = \text{Resize}(I, 640, 640)$, where I represents the i-th image in the dataset.

3.2.2 Pixel Normalization

The pixel intensity values of the images are normalized from the standard 0–255 range to 0–1 during the model input process. This provides consistent input values for the deep learning model.

Equation (2): $I_n(x,y) = I(x,y) / 255$. Normalization helps improve the stability of model training and supports faster convergence during the learning process.

3.2.3 Data Augmentation

Data augmentation is applied during YOLOv8 training to increase the variation in the training images and improve the model's generalization ability. The augmentation process includes horizontal flipping, scaling, translation, cropping, and changes in image properties. These

transformations help the model recognize cats and dogs under different positions, sizes, orientations, backgrounds, and lighting conditions. Data augmentation also reduces the possibility of overfitting and improves detection performance on unseen images.

3.2.4 Class Preparation

The dataset contains two object classes: Cat and Dog. Each class is assigned a unique numerical identifier for YOLOv8 training.

Equation (3): C = {0: Cat, 1: Dog}. Here, class 0 represents Cat and class 1 represents Dog. This class mapping enables the YOLOv8 model to distinguish between cats and dogs during training and prediction.

3.3 Data Annotation

The images are annotated with bounding boxes around each cat and dog. The annotations provide the ground-truth information required for training the YOLOv8 object detection model. Each object is represented using its class ID, bounding-box center coordinates, width, and height. The YOLO annotation format stores the bounding-box coordinates in normalized form between 0 and 1. The annotations allow the model to learn both the location and category of each object.

3.4 Module Selection and Training

For the proposed Cat and Dog Detection system, YOLOv8 was selected as the object detection model because it provides fast and accurate detection while being suitable for real-time applications. The model is capable of identifying both the class and location of objects in an image.

The YOLOv8 model was trained using the prepared Cat and Dog dataset. The dataset was divided into training and validation data, with appropriate class labels and bounding-box annotations. During training, the model learned the visual features of cats and dogs from different poses, sizes, orientations, and backgrounds.

The training was performed using 120 epochs with an image size of 640 × 640 pixels. A GPU-enabled Google Colab environment was used to speed up the training process. The best-performing model weights were automatically saved for further validation and testing.

After training, the model was validated using 349 images containing 420 object instances. The validation results showed that the trained model achieved good detection performance for both cat and dog classes. The trained YOLOv8 model was then used for detecting cats and dogs in new input images and generating bounding boxes with the predicted class labels and confidence scores.

3.5 Module Evaluation

Precision

Precision quantifies the proportion of predicted positive detections that are actually correct. It indicates how reliable the model's predicted Cat and Dog detections are. Precision = TP / (TP + FP), where TP represents true positive detections and FP represents false positive detections. A high precision value indicates that the model produces fewer incorrect Cat or Dog detections.

Recall

Recall measures the ability of the model to correctly identify the actual Cat and Dog objects present in the validation images. Recall = TP / (TP + FN), where FN represents false negative detections, which are actual objects that the model failed to detect. A high recall value indicates that the model successfully detects a larger proportion of the animals present in the images.

Average Precision (AP)

Average Precision represents the area under the Precision-Recall curve for an individual object class. AP is calculated separately for each class, allowing the detection performance of Cat and Dog to be analyzed independently.

Mean Average Precision (mAP)

Mean Average Precision represents the average AP across all object classes. $mAP = (1/N) \Sigma AP_i$, where N = 2, representing the two classes, Cat and Dog. The mAP@0.5 metric evaluates detection performance using an Intersection over Union (IoU) threshold of 0.50. The mAP@0.5:0.95 metric provides a stricter evaluation by averaging the AP values across IoU thresholds ranging from 0.50 to 0.95.

4. RESULT AND PRECISION

4.1 System Configuration

The proposed YOLOv8-based cat and dog detection system was implemented in Google

Colab using GPU acceleration. The hardware and software configurations used for model training and evaluation are given below.

Hardware configuration: Processor: Google Colab virtual CPU; GPU: NVIDIA Tesla T4; GPU Memory: Approximately 14.9 GB; Storage: Google Drive; Computing Platform: Google Colab.

Software Stack: Programming Language: Python 3.12.3; Deep Learning Framework: PyTorch; Object Detection Framework: Ultralytics YOLOv8; Development Environment: Google Colab; Dataset: Cat and Dog Object Detection Dataset; Classes: Cat and Dog; Model: YOLOv8; Visualization: Matplotlib; Training Epochs: 120.

4.2 Performance Analysis

The trained YOLOv8 model was evaluated on 349 validation images containing 420 object instances. The performance was measured using Precision, Recall, mAP@50, and mAP@50–95. These metrics provide a detailed assessment of the model's ability to correctly detect and classify cats and dogs.

Table 3. Class-wise Performance Analysis

Class	Images	Instances	Precision	Recall	mAP@50	mAP@50–95
Cat	184	202	0.942	0.883	0.939	0.769
Dog	177	218	0.904	0.798	0.858	0.691
Overall	349	420	0.923	0.840	0.899	0.730

The overall Precision of 92.3% indicates that the majority of the objects predicted by the model were correctly identified. The Recall of 84.0% shows that the model was able to detect most of the cat and dog instances present in the validation dataset. The mAP@50 of 89.9% demonstrates strong detection performance at an IoU threshold of 0.50. The mAP@50–95 of 73.0% provides a stricter evaluation across different IoU thresholds.

The cat class achieved a higher mAP@50 of 93.9% compared with the dog class, which achieved 85.8%. This indicates that the model was slightly more effective at detecting cats in the validation images. Differences in pose, object size, background, occlusion, and visual appearance may affect the detection of dogs.

Figure 1. Precision-Confidence Curve for Cat and Dog Detection.

The Precision-Confidence Curve shows the relationship between the confidence threshold and the precision of the YOLOv8 model. As the confidence threshold changes, the model filters predictions according to how certain it is about each detection. A high precision value indicates that the model produces fewer incorrect detections.

Figure 2. Recall-Confidence Curve for Cat and Dog Detection.

The Recall-Confidence Curve represents the relationship between the confidence threshold and the recall of the model. It indicates how many actual cat and dog objects can be detected at different confidence levels. A lower confidence threshold generally allows more detections, while a higher threshold can remove uncertain predictions.

Figure 3. Precision-Recall Curve for Cat and Dog Detection.

The Precision-Recall (PR) Curve illustrates the trade-off between precision and recall for the cat and dog classes. A curve closer to the upper-right region indicates better detection performance. The area represented by the curve contributes to the Average Precision (AP) of each class.

Figure 4. YOLOv8 Prediction Results for Cat and Dog Detection.

The trained YOLOv8 model was tested on sample images to verify its ability to detect cats and dogs. The prediction results display bounding boxes, class names, and confidence scores around the detected objects. The model successfully identifies the target animals in different image conditions.

5. CONCLUSION

The proposed YOLOv8-based Cat and Dog Detection System successfully demonstrates the effectiveness of deep learning and computer vision techniques for automatically detecting and classifying cats and dogs in images. The system was developed through a complete workflow involving dataset collection, image preprocessing, annotation, training, validation, evaluation, and prediction. The YOLOv8 model was trained using a dataset containing two target classes, namely cat and dog, and the training process was performed using Google Colab with GPU acceleration.

The performance of the trained model was evaluated using important object detection metrics, including Precision, Recall, mAP@50, and mAP@50–95. The experimental results achieved an overall Precision of 92.3%, Recall of 84.0%, mAP@50 of 89.9%, and mAP@50–95 of 73.0%, demonstrating that the proposed model can effectively identify and localize the target animals. The cat class achieved a higher detection performance with 94.2% Precision and 93.9% mAP@50, while the dog class achieved 90.4% Precision and 85.8% mAP@50.

The prediction results further confirm that the model can generate accurate bounding boxes, class labels, and confidence scores for newly provided images. The precision-confidence, recall-confidence, and precision-recall curves provide additional insight into the model's performance at different confidence levels and demonstrate the balance between accurate predictions and object detection coverage.

Although the model performs well, detection accuracy can still be affected by factors such as complex backgrounds, poor lighting, object occlusion, different animal poses, and variations in image quality. Future work can focus on increasing the size and diversity of the dataset, applying advanced augmentation techniques, improving model optimization, and deploying the trained model in real-time applications. Overall, the obtained results demonstrate that YOLOv8 provides an efficient, accurate, and practical solution for automated cat and dog object detection and can serve as a foundation for further animal recognition and computer vision applications.

REFERENCES

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788, 2016.

[2] J. Redmon and A. Farhadi, “YOLO9000: Better, Faster, Stronger,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7263–7271, 2017.

[3] J. Redmon and A. Farhadi, “YOLOv3: An Incremental Improvement,” arXiv preprint arXiv:1804.02767, 2018.

[4] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection,” arXiv preprint arXiv:2004.10934, 2020.

[5] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” Advances in Neural Information Processing Systems, vol. 28, 2015.

[6] W. Liu et al., “SSD: Single Shot MultiBox Detector,” European Conference on Computer Vision (ECCV), pp. 21–37, 2016.

[7] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 2961–2969, 2017.

[8] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, “YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors,” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7464–7475, 2023.