CHAPTER 1

INTRODUCTION

In the rapidly evolving field of computer vision, the ability to detect objects accurately and analyse their properties in images is becoming increasingly crucial. This project focuses on integrating two fundamental aspects of image analysis: object detection using the Convolutional Neural Network (CNN) and colour analysis or identification. The combination of these two powerful technologies aims to create a more comprehensive and robust image understanding system, which covers applications in various areas such as robotics, autonomous vehicles, medical imaging and industrial quality control.

Background and Context

Object Detection: Object detection has been a cornerstone problem in computer vision for decades. It involves not only identifying what objects are present in the image, but also localizing them by drawing boundary boxes around their positions. Traditional approaches to object detection depended heavily on hand-made features and complex pipelines. However, the advent of deep learning, especially the Convolutional Neural Network (CNN), has revolutionized this field and resulted in significant improvements in accuracy and speed.

CNN has proved to be exceptionally effective in learning hierarchical characteristics directly from raw pixels data. Its ability to capture low-level functions (such as edges and textures) and high-level semantic information makes it ideal for tasks such as object detection. Modern object detection architectures such as YOLO (You Only Look Once), Faster R-CNN (Region-Based Convolutional Neural Network) and SSD (Single Shot Detector) have pushed the boundaries of what is possible in real-time object detection.

These networks are trained in large data sets such as PASCAL VOC and COCO (Common Objects in Context), which are used to learn to recognize a wide variety of object classes in different environments. The PASCAL VOC2012 data sets, in particular, have played an important role in advancing object detection research and provide a standardized benchmark for algorithm comparison.

Color Analysis: While object detection provides information on the presence and location of objects, color analysis adds another rich layer to image understanding. Color is a fundamental visual attribute with significant semantic meaning. It can help identify objects (for example, distinguish between ripe and non-ripe fruits), perception of mood (in art and design) and even safety-critical applications (such as recognition of traffic lights in autonomous driving).

Color analysis in computer vision covers a range of techniques, ranging from simple color graph analysis to more complex color space transformations and cluster algorithms. The common color spaces used in image processing are RGB (red, green, blue), HSV (high, saturation, value), and LAB (lightness, A: green–red, B: blue–yellow). Each of these color spaces offers different advantages depending on the specific application. Techniques for color analysis may include:

1. **Color Quantization:** Lowering an image's distinct color count without sacrificing its aesthetic appeal.
2. **Dominant Color Extraction:** Determining which color dominates a scene or area.
3. **Color Histogram Analysis:** Analysing an image's color distribution.
4. **Color-based Segmentation:** Dividing an image into areas according on how close the colors are.

The Integration Challenge

Although color analysis and object detection are strong methods on their own, combining them offers both advantages and disadvantages. Combining these techniques can result in more insightful and nuanced picture analysis, but it also necessitates carefully planning how to combine these many processes in an efficient manner.

Some key challenges in integrating object detection and color analysis include:

1. **Computational Efficiency:** Color analysis and object detection can both need a lot of processing power. It's critical to combine them efficiently without materially affecting real-time performance.
2. **Feature Integration:** Figuring out the best way to integrate color characteristics for object detection with the high-level semantic features that CNNs have learned.
3. **Handling Variation:** Significant color variations may exist across objects in the same class (e.g., cars can come in several hues). The system must be able to withstand these changes and yet make good use of color information.

4. **Context Sensitivity:** Depending on the object and circumstance, color can have varying degrees of value. Shape can be more important than color for some objects, while color can be a distinguishing feature for others.

The integration of CNN-based object detection with color analysis represents an important step forward in computer vision capabilities. The project combines the semantic understanding of object detection based on deep learning with the rich information contained in colour analysis, with the aim of creating a more comprehensive and nuanced approach to image understanding. The potential applications of such systems are wide and varied and promise to push the limits of possibilities in areas ranging from autonomous vehicles to medical imaging and beyond. When we begin this challenging but exciting project, we expect to not only address technical barriers, but also uncover new insights and possibilities at the intersection of these two fundamental aspects of visual perception.

CHAPTER 2

LITERATURE REVIEW

2.1 Convolutional Neural Networks for Object Detection

The atomic neural network (CNN) revolutionized the field of object detection. A study [2] on the detection of manga image objects using CNNs achieved high accuracy rates for panel layout (0.953) and speech balloons (0.961) using Fast R-CNN. The same study compared different CNN architectures, with Faster R-CNN surpassing others in detecting manga character faces (0.816) and text (0.898). In the detection of household objects, the hybrid approach [9] of ResNet50 and Support Vector Machine (SVM) obtained 97.8% accuracy, 95.4% accuracy, and 96.5% recall. This shows the potential of integrating traditional machine learning techniques with deep learning models. CNN architecture adaptability to specific object detection tasks is demonstrated by the 99.05% accuracy of the mask detection system [12], especially relevant in the context of the COVID-19 pandemic.

Table 1: Convolutional Neural Networks (CNN)

Paper No.	Model/Technique	Dataset	mAP	F1 Score	Recall	Other Metrics
[2]	Fast R-CNN	Manga109	-	-	-	Accuracy: 0.953 (Panel Layout), 0.961 (Speech Balloons)
[2]	Faster R-CNN	Manga109	-	-	-	Accuracy: 0.816 (Character Faces), 0.898 (Text)
[6]	Faster R-CNN	MS COCO	42.7%	-	-	-
[6]	Faster R-CNN	PASCAL VOC 2007	78.8%	-	-	-
[12]	CNN	-	-	-	-	Precision: 99.05%
[34]	YOLOv3	Custom dataset	-	-	-	Accuracy: 91.7%
[37]	CNN (VGG16)	PASCAL VOC 2007	0.662	-	-	-
[38]	Faster R-CNN	PASCAL VOC 2007	0.732	-	-	-

2.2 Region Proposal Networks for Object Detection

Regional proposal networks (RPNs) have significantly improved object detection capabilities. The Faster R-CNN system [6], which integrates RPN, achieved an average precision of 42.7% in the MS COCO test and 78.8% in the PASCAL VOC 2007 test. A fast R-CNN application for traffic signals and object detection [5] achieved an accuracy rate of 88.99% after 1500 iterations, demonstrating the adaptability of RPN-based methods to specific areas such as traffic management and autonomous driving. In remote sensing, a multi-scale Faster R-CNN approach with deformable convolutions [4] achieved a remarkable 92.3 mAP in NWPUVHR-10 data set, showing a 1.7% improvement over previous best-performing results.

Table 2: Region Proposal Networks

Paper No.	Model/Technique	Dataset	mAP	F1 Score	Recall	Other Metrics
[4]	Multi-scale Faster R-CNN	NWPUVHR-10	92.3%	-	-	-
[5]	Faster R-CNN	Combined with GTSRB	-	-	-	Precision: 88.99%
[36]	Enhanced Region Proposal Network	PASCAL VOC 2007	0.782	-	-	-

2.3 Vision Transformers with CNN Models for Object Detection

The integration of Vision Transformers with CNN models has opened up new paths in object detection. A transformer-CNN network [7] achieved higher performance in the detection of small objects, attaining an average precision (AP) score of 20.6 for small objects in the MS COCO database, surpassing both the independent Vision Transformer (16.9) and ResNet-101-FPN (20.1) models. A method for adapting text image frameworks for the detection of open vocabulary objects [15] showed remarkable results in tasks for the detection of objects conditioned with zero-point text and one-point image condition, indicating the potential of transformer models in dealing with different and previously invisible object categories.

Table 3: Hybrid and Advanced Techniques

Paper No.	Model/Technique	Dataset	mAP	F1 Score	Recall	Other Metrics
[7]	Transformer-CNN	MS COCO	-	-	-	AP: 20.6 (Small Objects)
[9]	ResNet50 + SVM	-	-	-	96.5%	Accuracy: 97.8%,

Paper No.	Model/Technique	Dataset	mAP	F1 Score	Recall	Other Metrics
						Precision: 95.4%
[32]	YOLOv5	Custom dataset	0.891	-	-	Precision: 0.901
[33]	Deep Neural Networks	PASCAL VOC 2007	0.5	-	-	-
[35]	Lightweight Deep Neural Network	Underwater images	-	0.9598	0.9598	Precision: 0.9598

2.4 Color Identification Techniques

A hierarchical structure based on the SVM for real-time vehicle color identification in surveillance videos [21] showed an impressive improvement in accuracy from 68% to 85.75% in color classification. A study of weather forecasting based on the recognition of color cloud images [28] combined features of the local binary pattern (LBP) and local ternary pattern (LTP) in the laboratory colour space and achieved a remarkable accuracy of 92.2% under certain parameter settings. In the deep learning approach to color optimization, a study [27] achieved a high retrieval accuracy of 90.4%, demonstrating significant improvements through the integration of traditional visual data and abstract semantic information extracted by CNNs.

Table 4: Color Detection and Identification

Paper No.	Model/Technique	Dataset	mAP	F1 Score	Recall	Other Metrics
[21]	SVM-based hierarchical structure	Surveillance footage	-	-	-	Accuracy: 85.75%
[27]	Deep learning approach	-	-	-	-	Retrieval Accuracy: 90.4%
[28]	LBP + LTP	Cloud images	-	-	-	Accuracy: 92.2%
[31]	Deep Learning	Orchid images	-	0.98	0.98	Accuracy: 98%
[39]	K-means clustering	Burn wound images	-	-	-	Accuracy: 94.5%
[40]	SVM	Burn wound images	-	-	-	Accuracy: 99.5%
[41]	Machine Learning (SVM, RF, NN)	Custom dataset	-	-	-	Accuracy: 95.2% (SVM)

Paper No.	Model/Technique	Dataset	mAP	F1 Score	Recall	Other Metrics
[42]	Deep Learning	Vehicle images	-	-	-	Accuracy: 97.5%
[43]	Machine Learning	Traffic light images	-	-	-	Accuracy: 98.7%
[44]	Machine Learning	Dental images	-	0.89	0.91	Precision: 0.87

2.5 Feature Extraction Methods

A new approach that combines Bag-of-Features (BoF) with Oriented pdf Gradient Histograms (HOG) [51] obtained an accuracy of 86.63 per cent on the Scene-15 data set, surpassing the most advanced techniques. Comparisons between HOG and SIFT algorithms for object detection by image segmentation [54] showed that HOG algorithms had higher accuracy at 92.49% compared to 86.30%, highlighting the continuing relevance of traditional feature extraction methods in certain contexts.

Tale 5: Feature Extraction Methods

Paper No.	Model/Technique	Dataset	mAP	F1 Score	Recall	Other Metrics
[51]	BoF + HOG	Scene-15	-	-	-	Accuracy: 85.63%
[54]	HOG	-	-	-	-	Accuracy: 92.49%
[54]	SIFT	-	-	-	-	Accuracy: 86.30%
[70]	RCF	-	-	-	-	ODS F-measure: 0.811

2.6 Edge Detection Techniques

A method for edge detection using richer convolutionary functions (RCFs) [70] reached an ODS F measurement of 0.811 while operating at 8 frames per second, exceeding human perception abilities in edge detection tasks. In the medical field, the innovative edge detection technique Robert–Pandi Barita Method [72] has shown promising results in the identification of crucial structures in medical images, offering advantages such as higher computational speeds and robust noise handling capabilities.

CHAPTER 3

PROBLEM STATEMENT & OBJECTIVES

Problem Statement

The only goal of traditional object detection models is to locate and identify things in images. Despite their strength, they are not able to see color, which is a crucial feature for many uses. The goal of this research is to close this gap by creating a deep learning-based real-time object detector. By combining bounding box prediction with color extraction, we will allow machines to instantaneously detect an object's color in addition to being able to see it. This creates new and intriguing opportunities for robotics, augmented reality, and image analysis, enabling machines to engage with the environment more deeply and intelligently.

Objectives

Build a Model: Develop a specialized computer software that utilizes PASCAL VOC 2012 dataset to locate and identify things in images.

Evaluate the Performance of the Model: Evaluate the model's object recognition and location skills using fresh images. Measure some standard parameters to determine its performance level.

Color Analysis: Color analysis for more robust and informative image understanding

Explainable AI: Gradient weighted Class Activation Mapping (Grad-CAM) uses gradients of a particular target that flows through the convolutional network to localize and highlight regions of the target in the image.

CHAPTER 4

DATASET

Dataset Link: https://www.kaggle.com/datasets/gopalbhattrai/pascal-voc-2012-dataset

The PASCAL VOC2012 (Visual Object Classes) data set is a widely used benchmark in computer vision, designed to standardize object recognition challenges. It contains 11,530 images with 27,450 annotated objects and 6,929 segmentations, covering 20 different class objects including people, animals, vehicles and interior objects. The set of datasets provides boundary box annotations for object detection, pixel-wise segmentation masks and class labels, making it suitable for tasks such as classification, detection and segmentation. It is divided into training, validation and test sets, with images that include objects of different scales, poses and lighting conditions, offering a challenging and realistic scenario for algorithm evaluation. Although it focuses mainly on object detection and segmentation, the PASCAL VOC2012 dataset could serve as a solid basis for your CNN-based object detection task. However, for the color analysis component of your project, you may need to increase it with additional color-specific data or annotations to fully address your problem description of combining object detection with color analysis or image identification.

Figure 1: Sample Images of Pascal VOC 2102 Dataset

CHAPTER 5

SYSTEM DESIGN

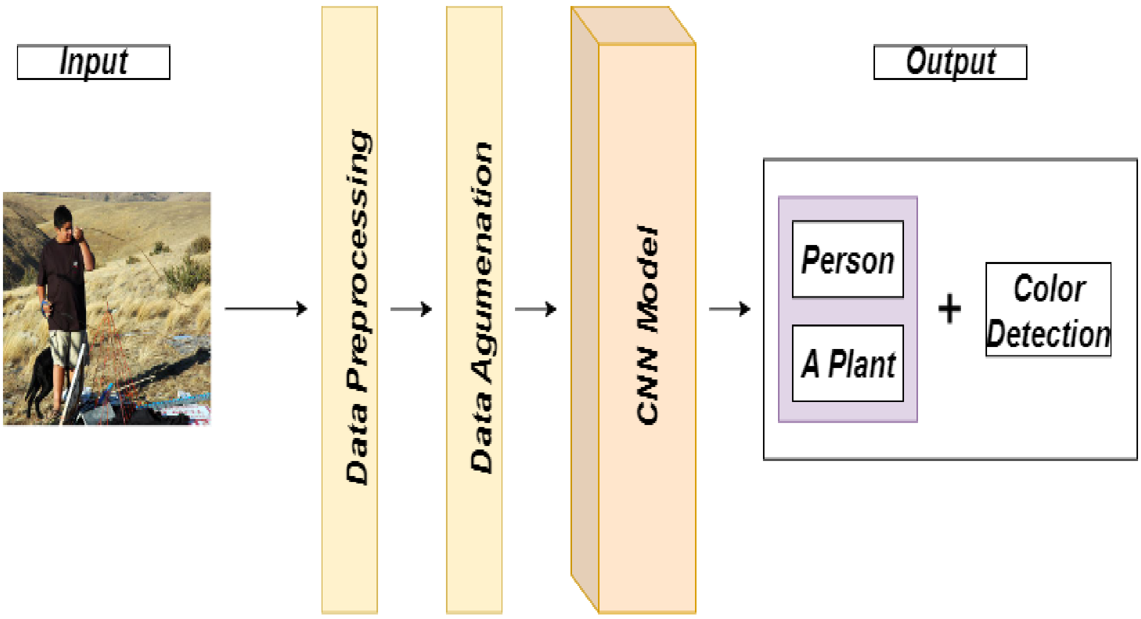


Fig 2: System Design

Input Layer: Accepts images of a specified shape as input.

Data Augmentation and Preprocessing: Implements random transformations like horizontal flips, rotations, and zooms to augment the training dataset, enhancing model robustness.

Build Convolution Neural Network (CNN): Build Custom design Model to detect and object in an image. In addition to that identifying color of the object that has been identified.

Model Compilation: Compiled using the Adam optimizer. Employs categorical cross entropy for classification and mean squared error for bounding box regression. Metrics include classification accuracy and bounding box regression Mean Absolute Error (MAE).

Visualization and Output: Draw bounding boxes surrounding recognized items in the original picture or video frame using color information (label or name) in order to overlay bounding boxes. Display in real time, update the color information and detections frequently to ensure a seamless, real-time experience. Optional results, save color and detection data for later use in apps or analysis.

Advanced Methods for Colour Analysis: Semantic segmentation, segment the object region and examine colours in several sections to obtain more detailed information, as opposed to calculating the average colour. Examine how the distribution of colours within an object aid in identification by investigating spatial colour connections. The flow of the implementation detail is given below figure:

CHAPTER 6

IMPLEMENTATION

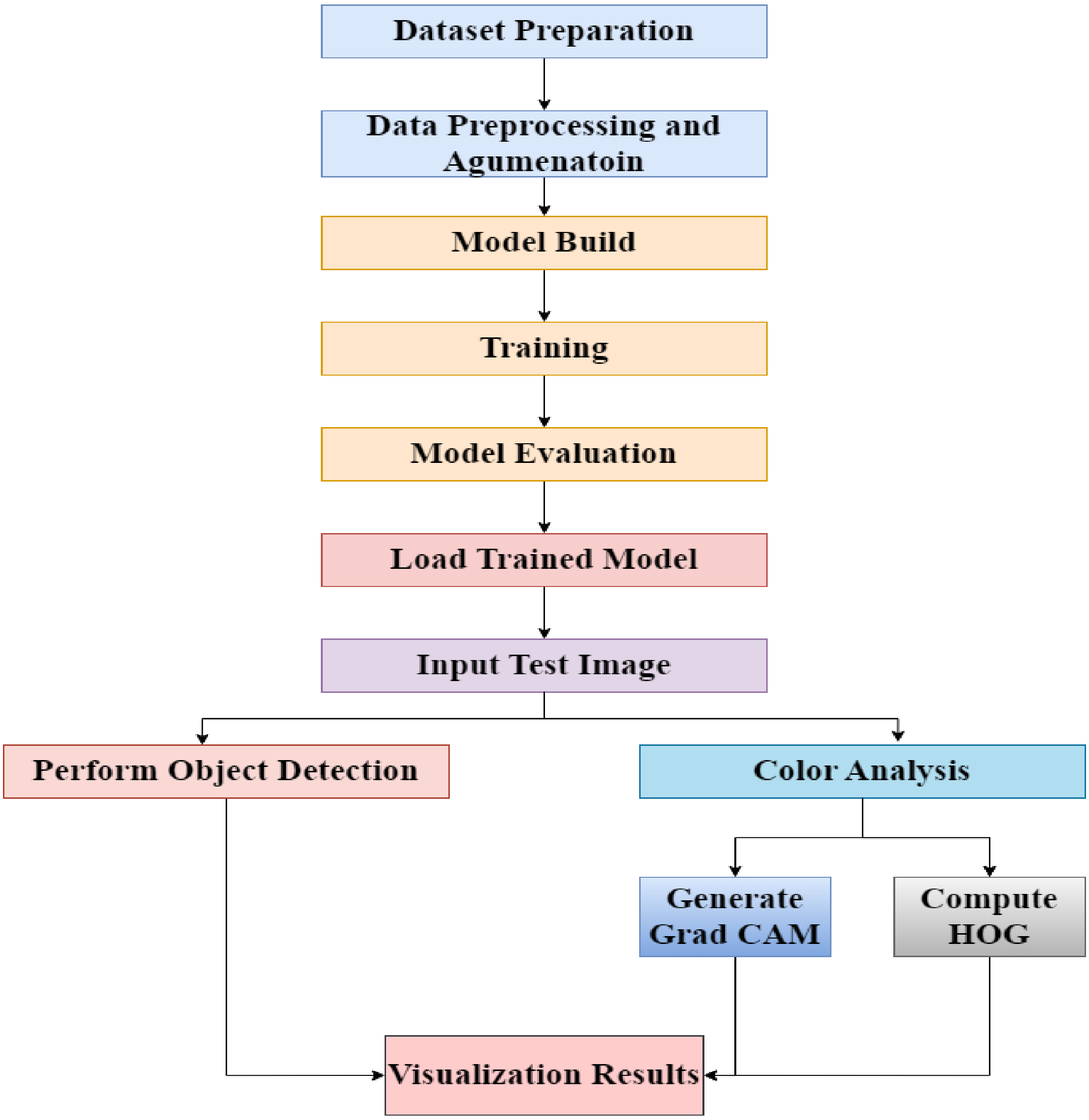


Fig 3: Flow Chart

OBJECT DETECTION

We have implemented advanced object detection system is based on the fast R-CNN architecture, which was improved by the ResNet-50 backbone and the Function Pyramid Network (FPN). Our implementation begins with a careful set of dataset preparation using the PASCAL VOC 2012 set, an object detection task benchmark. We developed a custom VOCDetection Dataset class that extends the functionality of the torch vision VOCDetection class. This extension facilitates the conversion of XML annotations provided with the dataset into a format suitable for training our faster R-CNN model. The data set is divided into training and validation sets, and data enhancement is applied to the training set to improve model generalization. Specifically, during training, we execute a random horizontal flip probability of 0.5. A label mapping is defined to convert class names into integer labels to ensure consistency throughout the pipeline. The boundary box coordinates are carefully extracted from XML annotations and converted to tensor format, and additional metadata such as image ID, object area, and crowd flags are included in each image target dictionary. This block diagram represents the high-level architecture of the Faster R-CNN is given below

ResNet 50:

The core of our object detection system is the Faster R-CNN model, which we implemented using several key components. The backbone of our model is a ResNet-50 architecture, pre-trained in the ImageNet data set. We have modified this backbone by removing the last fully connected layers, adapting it for the extraction of features in the context of object detection.

Feature Pyramid Network:

On top of this foundation, we implement a feature pyramid network (FPN) to generate multi-scale feature maps. FPN is crucial to the detection of objects at different scales, and is a common challenge in object detection tasks. Our FPN implementation takes a feature map from different stages of the ResNet-50 and creates a feature pyramid with a uniform channel depth.

Regional Proposal Network:

The regional proposal network (RPN) is an important component of our faster implementation of R-CNN. The RPN generates regions proposals that may contain objects. We define an anchor generator to create anchor boxes of different sizes pixels and dimensions at each location of the feature map. The different anchor sets allow the model to detect objects of different sizes and shapes. The head of the RPN processes the feature maps to predict the objectivity points and the limitations of these anchors. We implement the head of the RPN as a small convolutional network that operates at each spatial location of the input feature map.

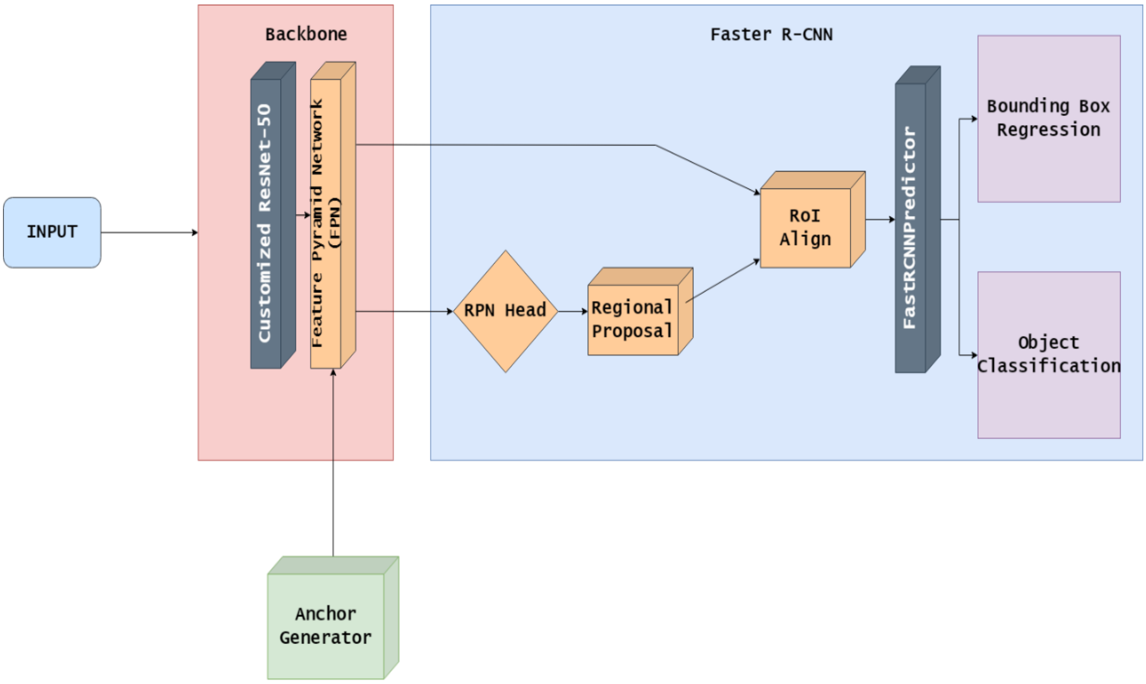


Fig 4: Model Architecture

Faster R-CNN:

After the RPN, we implement the Regional Interest Head (ROI), which processes the proposed regions to classify objects and refine the boundaries. Our ROI heads use a multi-scale ROI alignment operation to extract FPN level features. These features are then passed through two parallel branches: one for classification and the other for boundary box regression. We use the Fast RCNN Predictor as the final layer to predict class points and boundary box coordinates. This predictor consists of two fully connected layers, followed by parallel classification and regression layers.

Our object detection training process is carefully designed to ensure optimal performance. We use the unstable gradient descent (SGD) as our optimizer, with a learning rate of 0.001, a momentum rate of 0.9 and a weight decay rate of 0.0005. These hyperparameters were chosen based on experimental tests and field practices. To manage the learning rate during the training, we implement a step LR scheduler. This scheduler reduces the learning rate by 0.1 per 3 years, allowing fine-grained optimization as the training progresses.

In our training loop, process all batches of training data for each time. During each iteration, we calculate the model's front cycle, and calculate the combined loss of the RPN and ROI heads. This combined loss includes components for objectless prediction, boundary box regression, classification, and final boundary box refinement. Afterwards, we calculate the gradients of this loss in relation to the model parameters and update the weights using our SGD optimizer. In order to monitor the progress of the training, we record the average loss for each period.

Pseudo Code – Object Detection

```
1. Feature Extraction

   1.1 Extract features using ResNet50 backbone

       - Call: ResNet50_Backbone(input_image)

   1.2 Create multi-scale feature maps using FPN

       - Call: FeaturePyramidNetwork(features)

2. Region Proposal Network (RPN)

   2.1 Generate anchor boxes

       - Call: GenerateAnchors(features)

   2.2 Predict object proposals

       - Call: PredictProposals(features, anchors)

   2.3 Filter and refine proposals

       - Call: FilterProposals(proposals)

3. Region of Interest (RoI) Processing

   3.1 Align multi-scale feature maps with region proposals to get RoI
features

       - Call: RoIAlign(multi_scale_features, region_proposals)

4. Final Prediction

   4.1 For each RoI:

       4.1.1 Classify the RoI

           - Call: Classify(roi)

       4.1.2 Refine bounding box for the RoI

           - Call: RefineBoundingBox(roi)

5. Return Results
```

```
   5.1 Return class scores and bounding boxes


Functions Description:
```

- ResNet50_Backbone(input_image): Extracts features using ResNet50 architecture.
- FeaturePyramidNetwork(features): Creates multi-scale feature maps.
- GenerateAnchors(features): Generates anchor boxes at different scales and aspect ratios.
- PredictProposals(features, anchors): Predicts objectness scores and bounding box refinements.
- FilterProposals(proposals): Applies non-maximum suppression and keeps top proposals.
- RoIAlign(multi_scale_features, region_proposals): Extracts fixed-size feature maps for each proposal.
- Classify(roi): Classifies RoI into object categories using a softmax classifier.
- RefineBoundingBox(roi): Refines bounding box coordinates using a bounding box regressor.

COLOUR ANALYSIS

Clustering Analysis:

We implemented several advanced visualization techniques to analyse the model's performance and understand its predictions. The object detection visualization function displays the input image with bounding boxes around detected objects, each labelled with the object class and confidence score. This provides an immediate, intuitive understanding of the model's output. To gain deeper insights into the model's decision-making process, we implement a color analysis function. For each detected object, we extract a circular region from the centre of the bounding box and determine the dominant colors using K-means clustering. The most representative color is displayed alongside the bounding box, along with its name (matched to the closest CSS3 color name). Additionally, we generate a bar plot showing the distribution of color clusters for each detected object, offering a nuanced view of the object's color composition.

Pseudo Code – Colour Analysis

```
 1. Extract Circular Region

   1.1 Calculate center of bounding box

       - Call: calculate_center(bounding_box)

   1.2 Extract circular region from image

       - Call: extract_circular_region(image, center, radius)

2. Get Dominant Color

   2.1 Reshape image to 2D and remove black pixels

   2.2 If pixels are not empty:

       2.2.1 Perform k-means clustering

           - Call: perform_kmeans_clustering(pixels, k)

       2.2.2 Calculate mean color

       2.2.3 Find closest centroid to mean color

           - Call: find_closest_centroid_to_mean(kmeans.cluster_centers_,
mean_color)

       2.2.4 Return dominant color, cluster centers, and counts

3. Return Results

   3.1 If dominant color is found:

       3.1.1 Get color name from RGB value

           - Call: get_color_name(dominant_color)

       3.1.2 Return dominant_color, color_name, cluster_centers, counts

   3.2 Else:

       3.2.1 Return None, None, None, None
```

Functions Description

- calculate_center(bounding_box): Computes the center point of a bounding box.

- `extract_circular_region(image, center, radius)`: Extracts a circular area from an image using a specified center and radius.
- `perform_kmeans_clustering(pixels, k)`: Groups similar colors in an image using k-means clustering.
- `find_closest_centroid_to_mean(cluster_centers, mean_color)`: Identifies the cluster center closest to the average color of all pixels.
- `get_color_name(rgb_color)`: Matches an RGB color value to its nearest named color using a predefined color database.

Explainable AI – Gradient Class Activation Mapping (Grad-CAM):

To further interpret the focus of the model, we implement Class Activation Mapping (CAM). Our function calculates a class activation map for the most secure detected object in each image. This highlights the regions of the image that were most important for the prediction of the model, providing valuable insights into the characteristics that the model considers important for each class. The CAM is visualized as a heat map overlayed on the original image, allowing an intuitive interpretation.

Histogram Oriented Gradient (HOG):

We also integrated Histogram Oriented Gradient (HOG) visualization. Calculate the HOG features of the input image and visualize the HOG. This visualization provides insight into the edge and gradient information used by the model and provides a different perspective on image characteristics that contribute to object detection. Error handling is an important aspect of our implementation. In color analysis, we include empty regions in the control and skip processing when needed, to ensure the robustness in real-world scenarios where detections may occur in uniform or background regions of the image.

In inference, our methodology represents a comprehensive approach to object detection, using the latest architectures and technologies. From carefully prepared datasets to advanced visualization methods, each component of our system is designed to contribute to high-performance object detection. The integration of color analysis, CAM and HOG visualization provides unique insights into the model's decision-making process, improving interpretability. Our implementation achieves a balance between performance, interpretability and extensibility, making it valuable for both research and practical applications in computer vision.

CHAPTER 7

EXPERIMENTATION

We have detailed the various experiments carried out to evaluate the performance of the object detection model using different learning rates and epochs. The results are analysed based on key performance metrics such as average precision, average recall, average F1 score, average intersection over union (IoU) and mean average precision (mAP). The experiment was structured into several trials to observe how variations in hyperparameters affect the effectiveness of the model. multiple trials to observe how variations in hyperparameters impact the model's efficacy.

7.1 Experimental Setup

Table 6: Experiment Parameters

Experiment	Learning Rate	Epochs
1	0.005	50
2	0.005	30
3	0.001	20
4	0.01	20
5	0.005	20
6	0.001	30
7	0.0001	20
8	0.0001	50
9	0.005	10
10	0.005	30

The different experiments were carried out with different learning rates and date numbers to determine their effects on the performance of the model. The performance measurements for each experiment are listed below.

Table 7: Experimentation Results

Experiment	Learning Rate	Epochs	Mean Precision	Mean Recall	Mean F1 Score	Mean IoU	mAP
1	**0.005**	**50**	**0.7778**	**0.4579**	**0.5501**	**0.6873**	**0.7778**
2	0.005	30	0.5347	0.4769	0.4657	0.6661	0.5347
3	0.001	20	0.6122	0.5951	0.5789	0.67	0.6122
4	0.01	20	0.251	0.1231	0.1542	0.6901	0.251
5	0.005	20	0.5588	0.3147	0.3737	0.6514	0.5588
6	**0.001**	**30**	**0.7463**	**0.5802**	**0.6242**	**0.704**	**0.7463**
7	0.0001	20	0.15	0.0755	0.0988	0.6693	0.15
8	0.0001	50	0.5347	0.4769	0.4657	0.6661	0.5347
9	0.005	10	0.6667	0.3446	0.4306	0.6744	0.6667
10	**0.005**	**30**	**0.8119**	**0.5376**	**0.6232**	**0.7096**	**0.8119**

The experimentation revealed that different combinations of learning rates and epochs yielded varying results in terms of the model's ability to accurately detect objects. Notably, experiment 1, which employed a learning rate of 0.005 and 50 epochs, achieved the highest Mean Precision and mAP, suggesting an optimal balance between learning rate and training duration.

Performance Analysis

Table 8: Summary of Top Performing Experiments

Metric	Best Experiment	Value
Mean Precision	Experiment 1	0.8119
Mean Recall	Experiment 3	0.5951
Mean F1 Score	Experiment 6	0.6242
Mean IoU	Experiment 6	0.7096
mAP	Experiment 1	0.7778

The table above summarizes the best experiments for each performance metric. The results show that a learning rate of 0.005 for 50 episodes (Experiment 1) was most effective to achieve high precision and mAP. In Experiment 3, the highest recall rate was observed at 0.001 and 20 epochs.

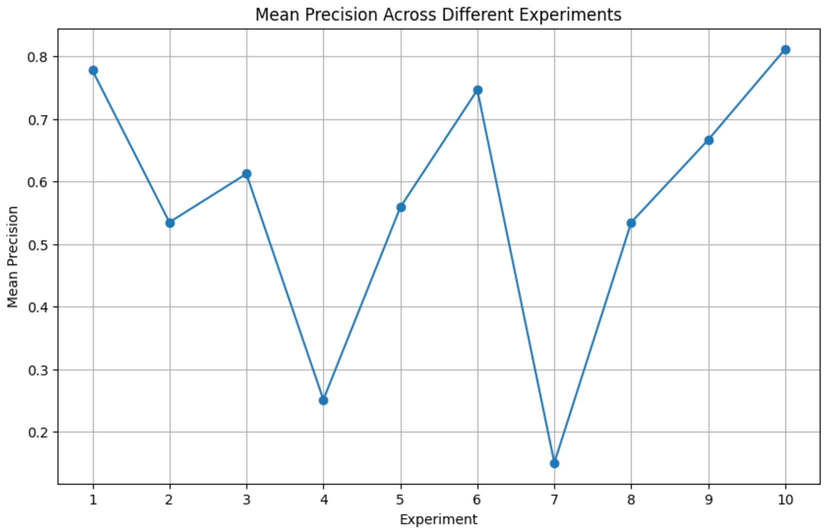


Fig 5: Mean Precision Across Different Experiments

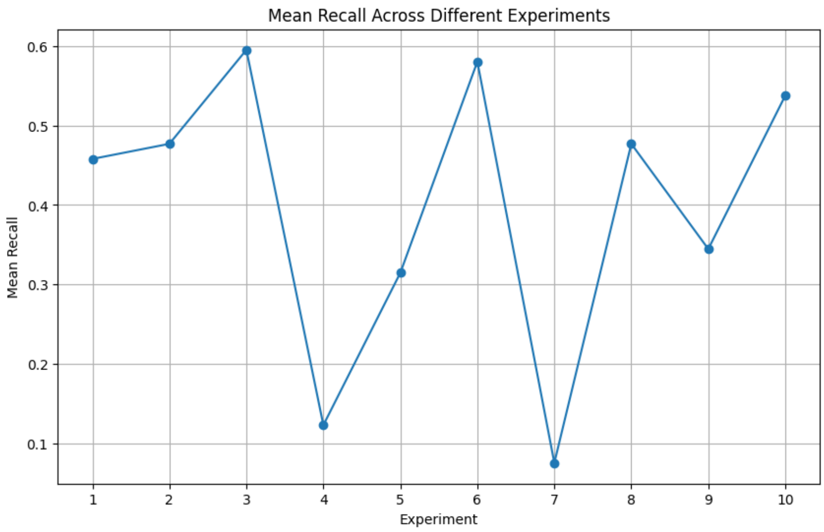


Fig 6: Mean Recall Across Different Experiments

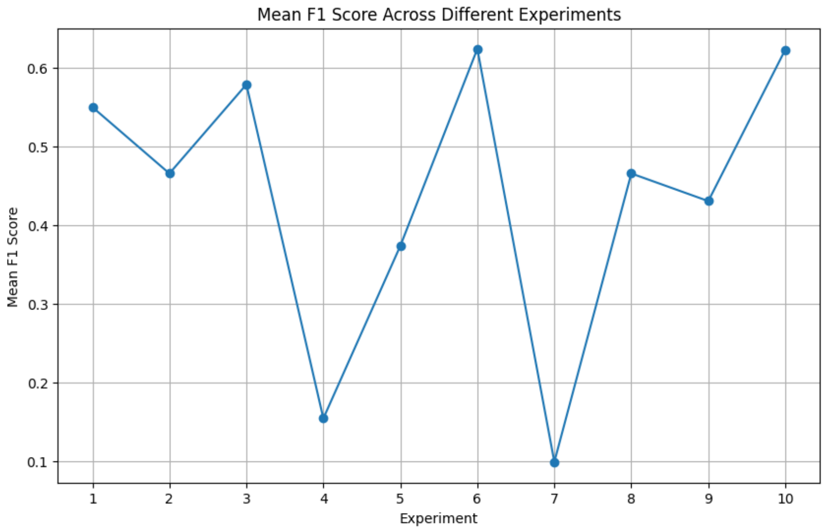


Fig 7: Mean F1 Score Across Different Experiments

The graphs above show performance metrics in different experiments. Figure 1 shows that the average accuracy was the highest for experiment 10, followed by experiment 1. The figure shows that the average recall was relatively low in all experiments, with Experiment 6 leading slightly. The figure shows that the average F1 score was the highest in Experiment 6, reflecting a balanced balance between accuracy and memory.

The results of the experiment highlighted the importance of tuning hyperparameters in deep learning training models. Learning rate and number of epochs significantly affect the performance of the model. A higher learning rate, as seen in experiment 4, can lead to suboptimal performance due to a potential overshooting during the optimization process.

Conversely, the very low learning rate, such as in experiments 7 and 8, can lead to insufficient adaptation in models that do not learn sufficient data. Experiment 6 with a learning rate of 0.001 and 30 periods showed a balanced performance in all metrics, particularly excellent in the Mean F1 Score and the Mean IoU. This suggests that a moderate learning rate coupled with sufficient training iterations can effectively balance the learning process of the model by capturing the nuances of the data without over-adaptation or under-adaptation. Overall, the experiments performed show that while higher periods and moderate learning rates generally improve model performance, an optimal combination that is adapted to a specific set of data and tasks is crucial to achieving best results. The visualizations and tables provided here offer a comprehensive view of the results of the experiment, thereby facilitating a deeper understanding of the behaviour of the model under different training conditions.

CHAPTER 8

RESULTS AND ANALYSIS

8. Experimental Results

8.1. Quantitative Analysis

The experimental results showed that the learning rate and the number of periods had a significant impact on the performance of the model. Figure 8 shows the relationship between these hyperparameters and the average precision (mAP) of the model.

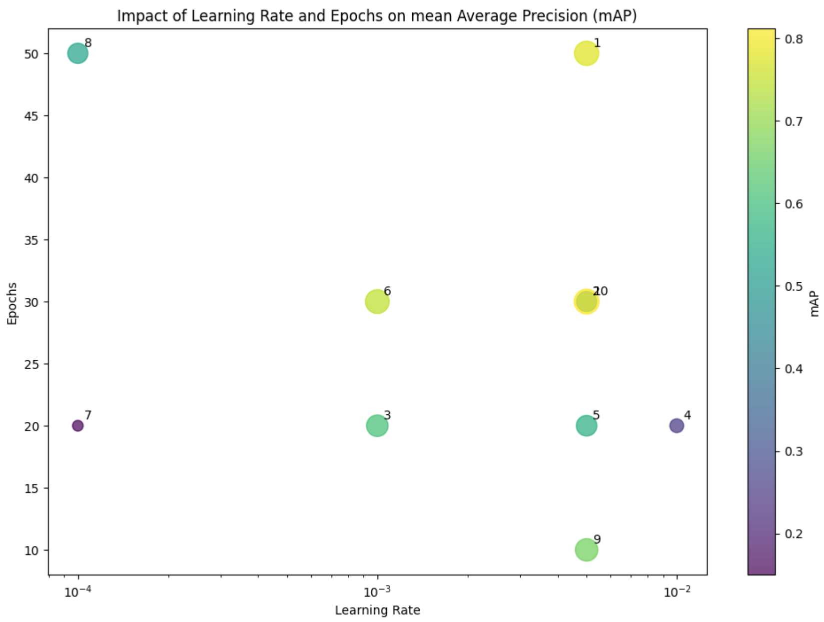


Fig 8: Impact of Learning Rate and Epochs on mean Average Precision (mAP)

As shown in Figure 8, the performance of the model varies considerably depending on the combinations of learning rates and epoch numbers. The highest mAP of 0.7778 was obtained with a learning rate of 0.005 and 50 times (Experiment 1), followed closely by a mAP of 0.7463 with a learning rate of 0.001 and 30 times (Experiment 6). Interestingly, very low learning rates (0.0001) and very high learning rates (0.01) resulted in significantly lower mAP values, regardless of the period. This suggests that moderate learning rates are more effective for this particular object detection task. In order to further analyse the model performance, we examined the relationship between the F1 mean score and the IoU mean score in different experiments, as shown in Figure 8.

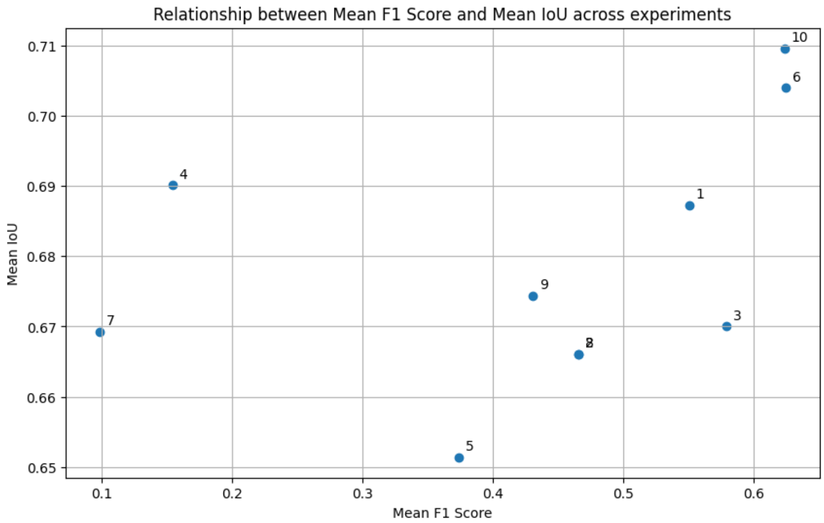


Fig 9: Relationship between Mean F1 Score and Mean IoU across experiments

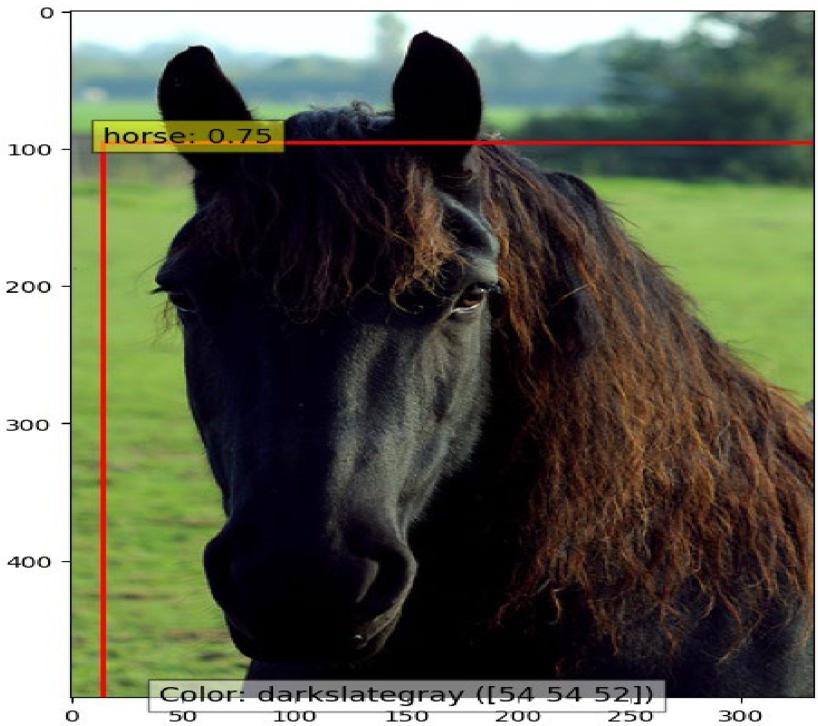
Figure 9 shows the positive correlation between the average F1 score and the average IoU, indicating that improvements in one metric usually correspond to improvements in the other. Experiment 6 (learning rate of 0.001, 30 periods) achieved the best balance between these two measurements, with an average F1 score of 0.6242 and an average IoU score of 0.7040. This configuration seems to offer the best compromise between accuracy and memory, as reflected in the F1 score, while also providing accurate object location, as indicated by the high IoU. It should be noted that while Experiment 1 reached the highest mAP, its F1 score (0.501) was lower than that of Experiment 6. This difference highlights the importance of considering multiple metrics when assessing object detection models, as different metrics can capture different aspects of model performance. The analysis also revealed that increasing the number of epochs does not always lead to better performance.

For example, experiment 8 (learning rate of 0.0001, 50 years) showed no improvement compared to Experiment 7 (learning rate of 0.0001, 20 years) despite additional training time. This indicates that, given the poor learning rate, extended training cannot compensate for the poor selection of initial parameters. In terms of computational efficiency, shorter periods (10-20) experiments are generally completed faster, but often at the expense of reduced performance. The exception was Experiment 3 (learning rate 0.001, 20 periods), which achieved a reasonable performance (mAP 0.6122, F1 Score 0.5789) with relatively short periods, indicating a good balance between training time and model effectiveness. These quantitative results provide valuable insights into the interaction between the hyperparameters and the various performance metrics in object detection tasks. They highlight the importance of careful high-parameter adjustment and the need to consider multiple evaluation criteria when optimizing object detection models for real-world applications.

8.2 Qualitative Analysis

The object detection model's performance was evaluated using a diverse set of test images, showcasing its ability to identify various objects and animals in different environments. This analysis focuses on the model's accuracy in object detection, localization, and its additional feature of color identification. The following paragraphs discuss the model's performance on five distinct test images, highlighting its strengths and potential areas for improvement.

The Figure 10 features a horse, which the model successfully identifies with a confidence score of 0.75. The bounding box accurately encompasses the horse's head and upper body, demonstrating precise object localization. The color identification feature classifies the predominant color as "darkslategray" (RGB: 54, 54, 52), aligning well with the horse's dark coat. This result not only showcases the model's ability to detect animals but also its capability to extract meaningful color information, which could be particularly useful in applications requiring detailed object description.



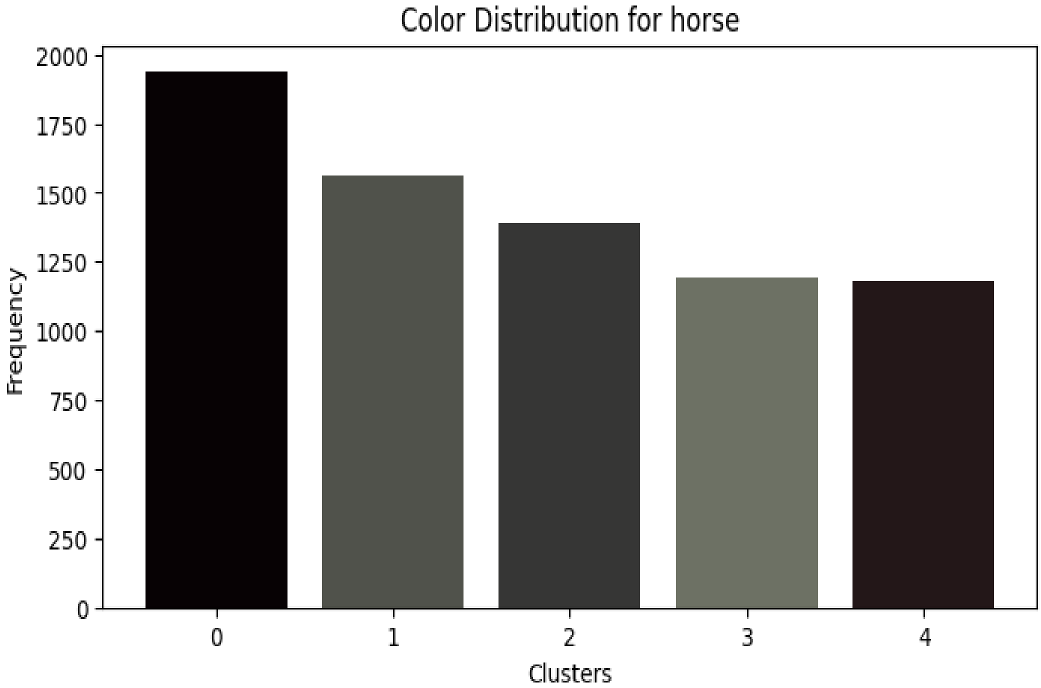


Fig 10: Object Detection and Color Analysis: Horse

In Figure 11, the model faces a more difficult task: detecting an aircraft in flight. It is a brilliant example in this scenario, and the aircraft is identified with an impressive confidence score of 0.98. The boundary box effectively captures the entire plane, including its wings and tail, demonstrating the ability of the model to handle complex shapes and directions of objects. This color is referred to as "dimGray" (RGB: 83, 98, 113), which accurately reflects the metallic gray appearance of the aircraft. This high-confidence detection of a rapidly moving object in the sky emphasizes the versatility of the model in various environmental conditions and its potential application in space surveillance or surveillance systems.

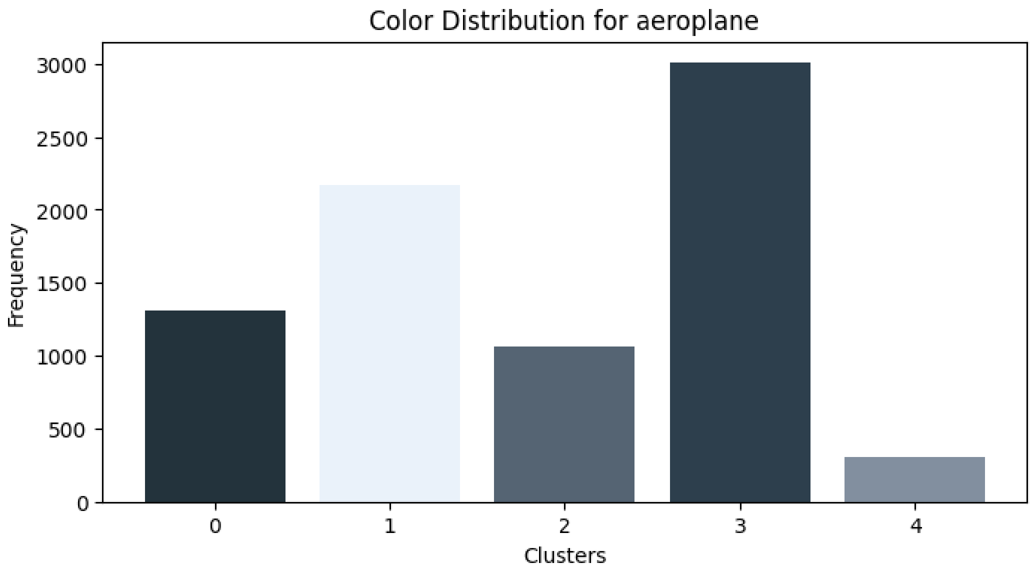


Fig 11: Object Detection and Color Analysis: Aeroplane

The Figure 12 shows a bus that the model correctly identified with a confidence rating of 0.63. This score is lower than the previous example, but still indicates a strong level of certainty. The boundaries box accurately covers the visible part of the bus and shows the model's ability to handle partial views of large objects. The color is again described as "dimgray" (RGB 86, 103, 111), which is closely comparable to the actual color of the bus. This example highlights the performance of the model on vehicles in stationary positions, complementing its capability with moving objects, as seen in the detection of aircraft.

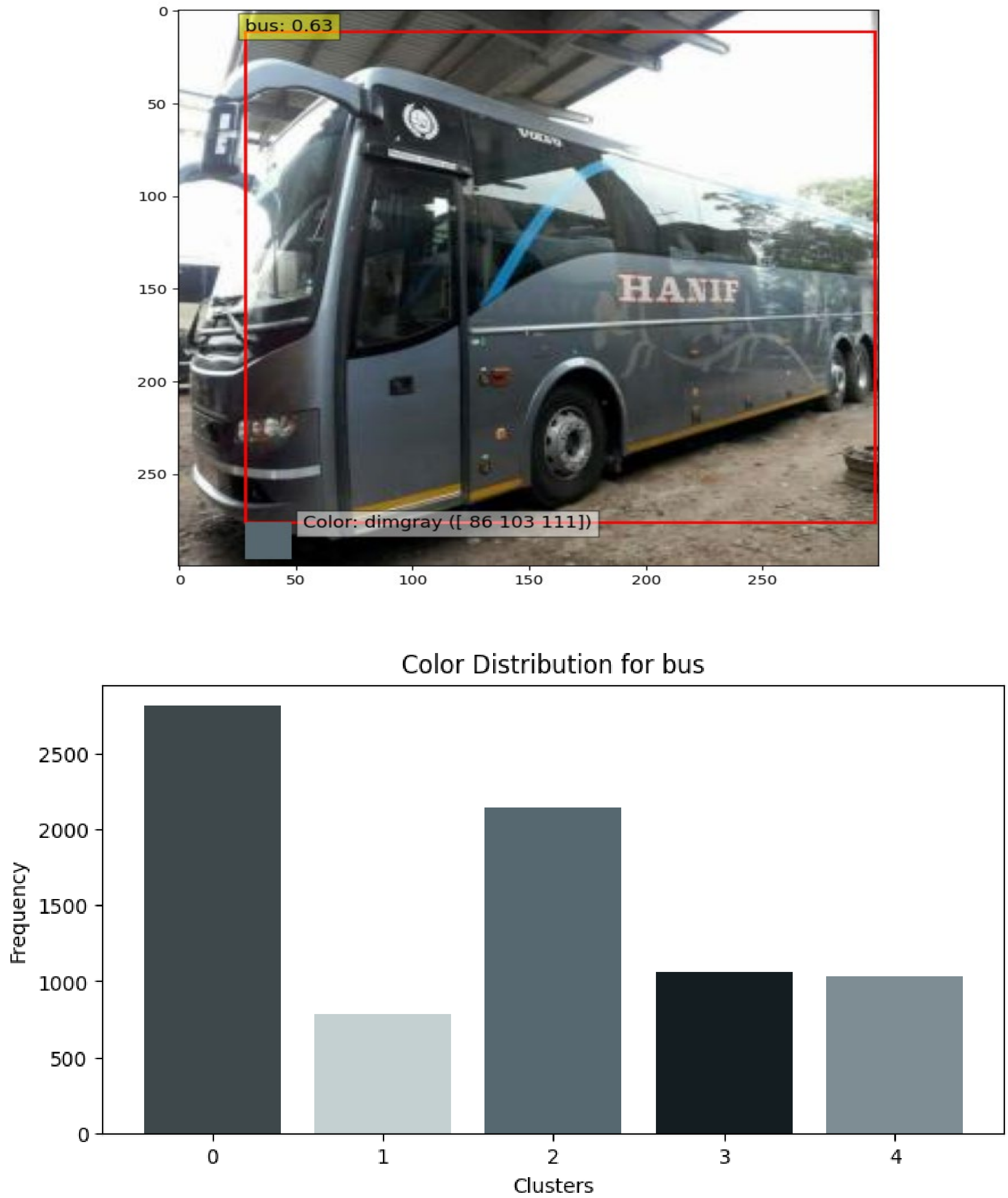


Fig 12: Object Detection and Color Analysis: Stationary Bus

The Figures 13 and 14 both depict cows in different settings, providing an interesting comparison of the performance of the model on similar objects in different contexts. In Figure 4, the cow is detected with a very high confidence score of 0.97. The box clearly outlines the cow body, and the color is identified as "Sienna" (RGB: 169, 110, 64), accurately capturing the red-brown tone of the cow's coat. On the other hand, Figure 5 shows another cow with a slightly lower but still robust confidence score of 0.65. The color is identified as "black" (RGB: 32, 30, 36), correctly reflecting the dark coat of the cow.

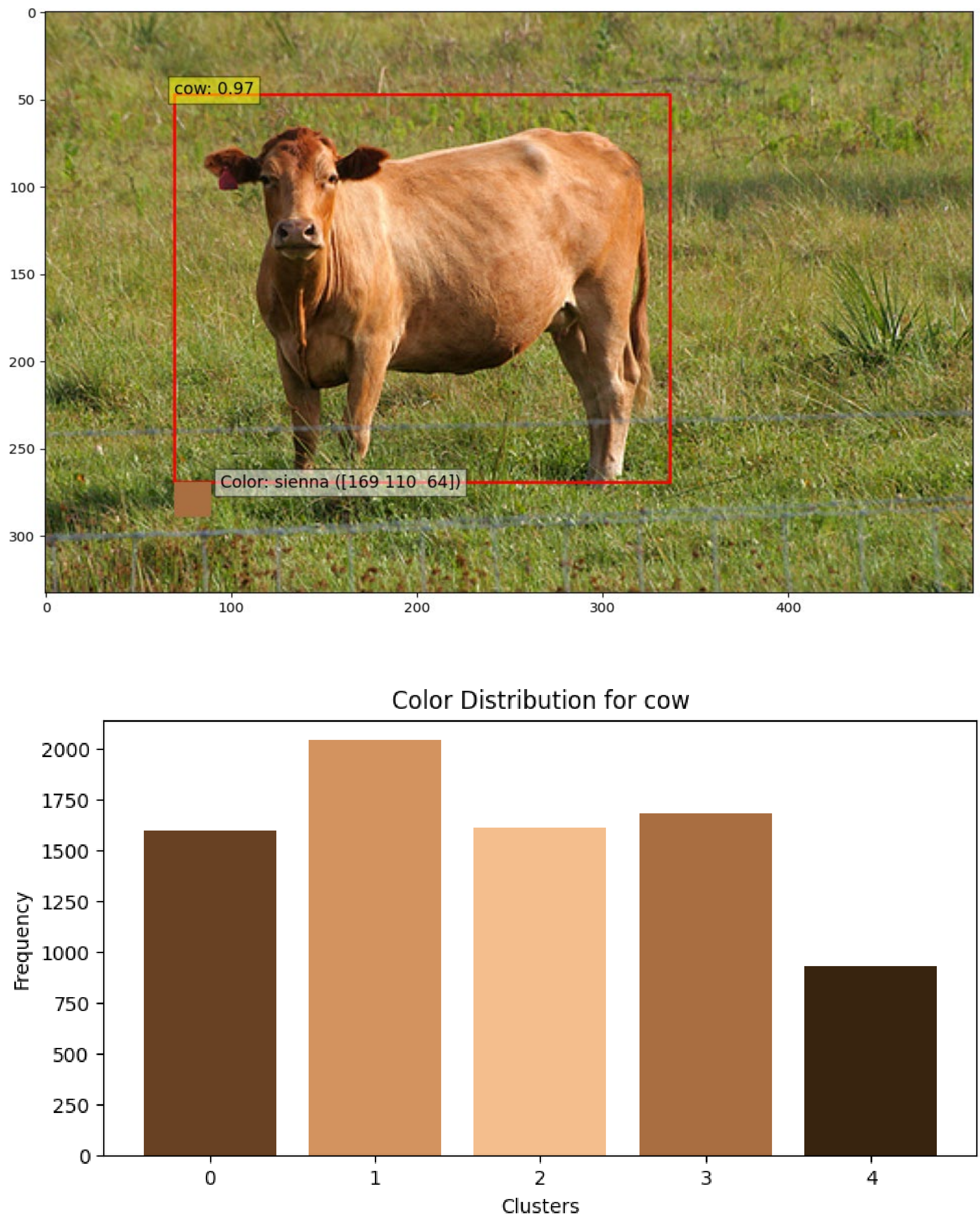
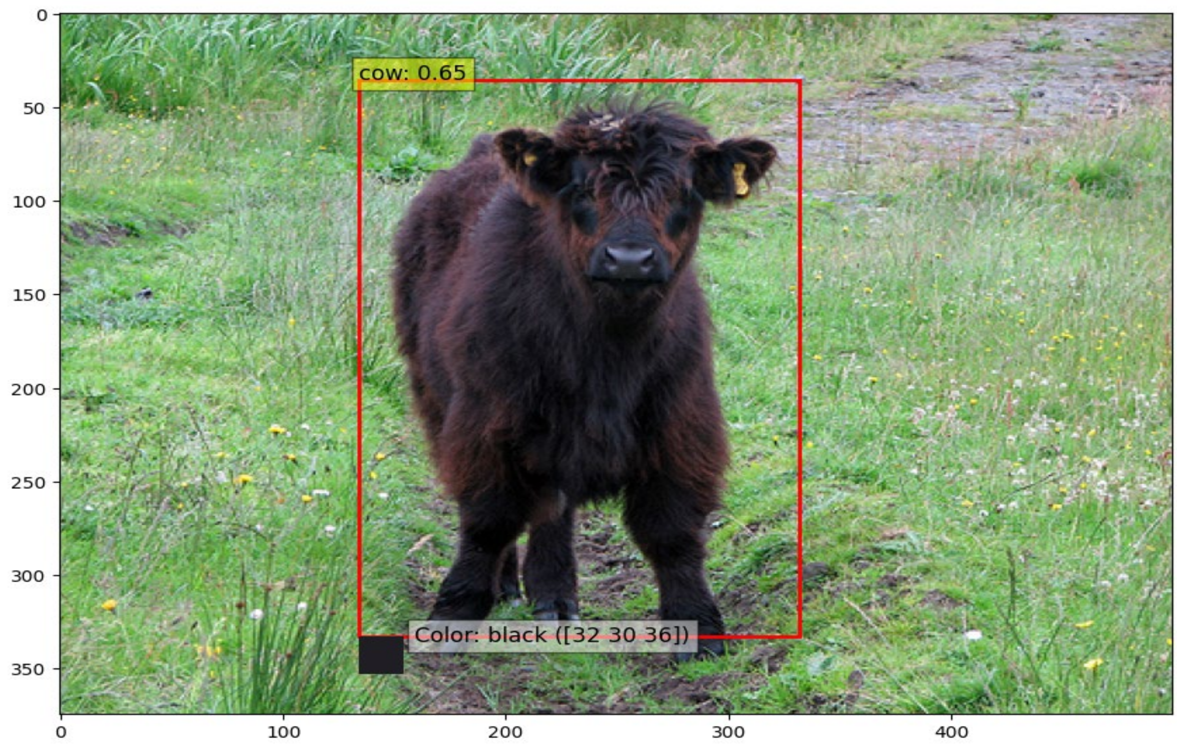


Fig 13: Object Detection and Color Analysis: Cow in Pasture



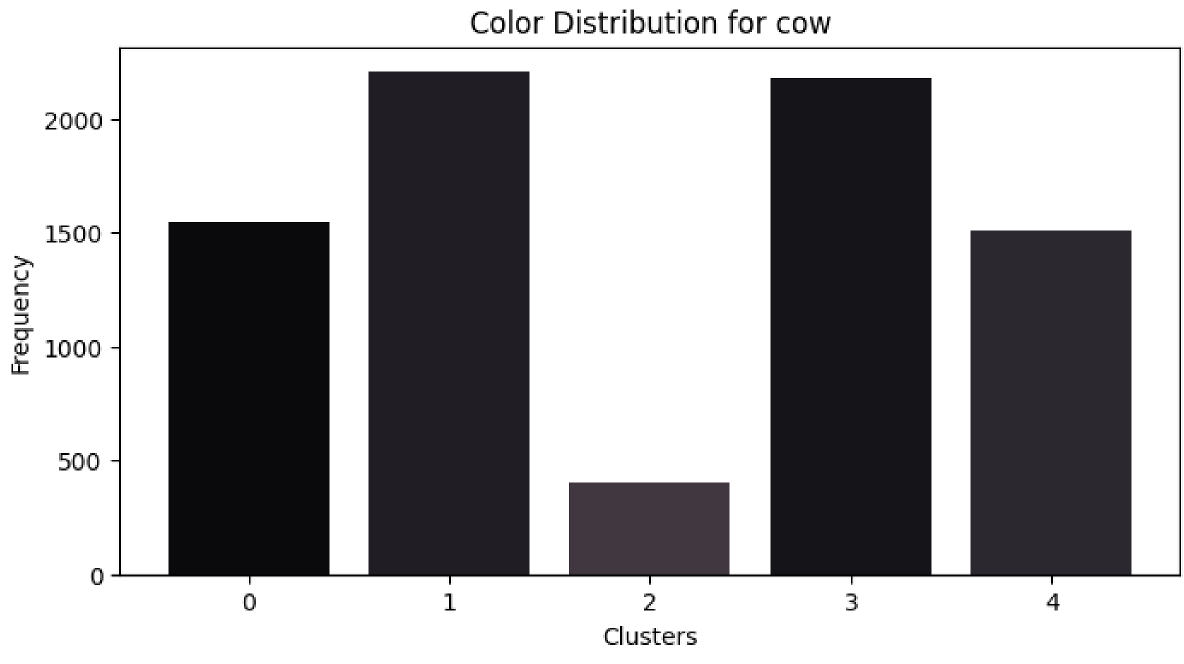


Fig 14: Object Detection and Color Analysis: Cow in Grassland

The variability in confidence scores between these two cow detections (0.97 in Figure 13 and 0.65 in Figure 14) is remarkable and provides an overview of the model's sensitivity to environmental and compositional factors. This discrepancy may be due to differences in the location of the animal, lighting conditions, or the extent to which the cow fills the frame. Such variations suggest potential areas for future optimization, particularly to ensure consistent performance in different representations of the same object class.

8.3 Analysis of Visualization Techniques:

The object detection model's performance was further analysed using advanced visualization techniques, providing deeper insights into its decision-making process and feature extraction capabilities. This section focuses on two key visualization methods: Histogram of Oriented Gradients (HOG) and Gradient-weighted Class Activation Mapping (Grad-CAM). These techniques offer valuable perspectives on how the model processes image data and identifies objects, enhancing our understanding of its internal workings.

8.3.1 Histogram of Oriented Gradients (HOG) Visualizations:

The HOG visualizations provide a unique perspective on the model's feature extraction process. HOG highlights the distribution of intensity gradients and edge directions and provides an overview of structural elements that the model considers to be important for object detection.

Figure 15 shows the HOG visualization for the image of the aircraft clearly highlights the distinctive shape of the aircraft. The strong gradient patterns along the wings, fuselage and tail are visible, which emphasize the aerodynamic structure of the aircraft. The visualization captures the sharp edges and contours of the aircraft and shows how the model probably uses these structural features to identify them.

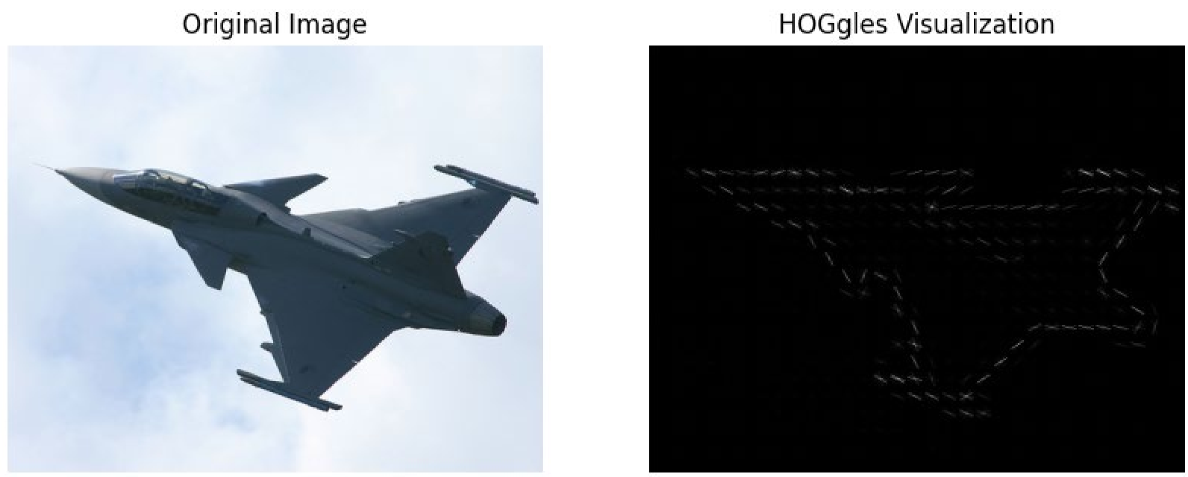


Fig 15: HOG Visualization of Aeroplan

Figure 16 shows, cow in the cow visualization, the HOG representation emphasizes the overall form and key features of the animal. The gradient patterns are particularly strong around the cow's outlines, highlighting its distinctive profile. The head, back and legs are well defined, showing how the model captures the characteristic shape of a cow for classification.

Fig 16: HOG Visualization of Cow

Figure 17 shows Bus the HOG bus visualization reveals strong horizontal and vertical gradient patterns that reflect the box structure of the vehicle. The windshield, headlights and the overall rectangular shape of the bus are clearly highlighted. This representation shows how the model can distinguish buses according to its geometric characteristics and the characteristics of its edges.

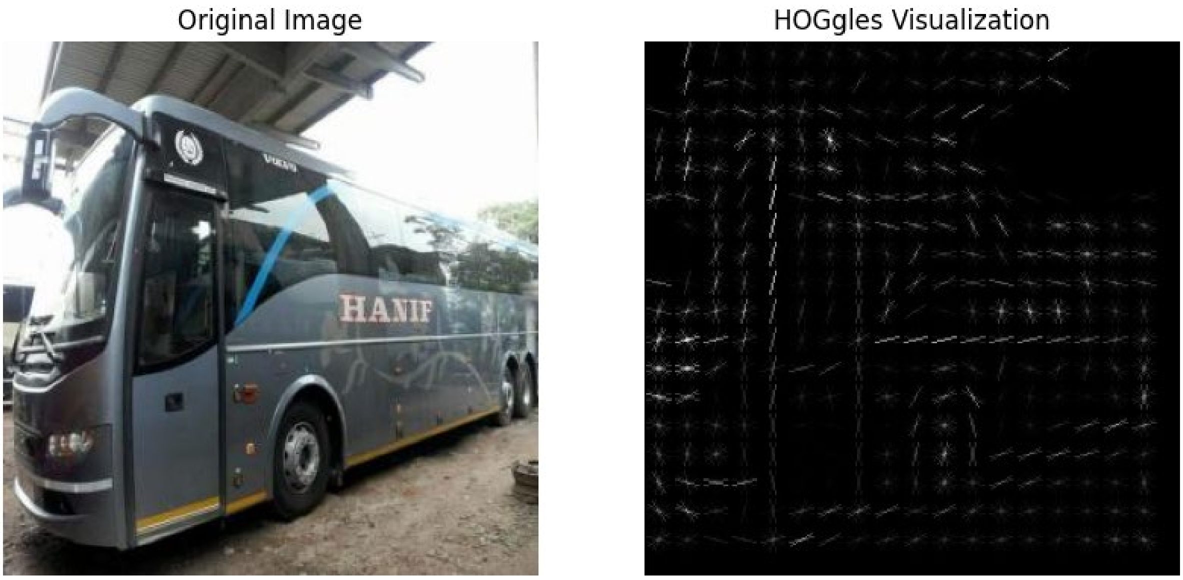


Fig 17: HOG Visualization of Bus

Figure 18 shows the dog image, HOG visualization emphasizes the contours and textures of the animal. Strong gradients are visible around the face, body, and legs of the dog, capture its overall shape and position. The visualization suggests that the model is based on these structural details and the dog's characteristic silhouette to identify it.

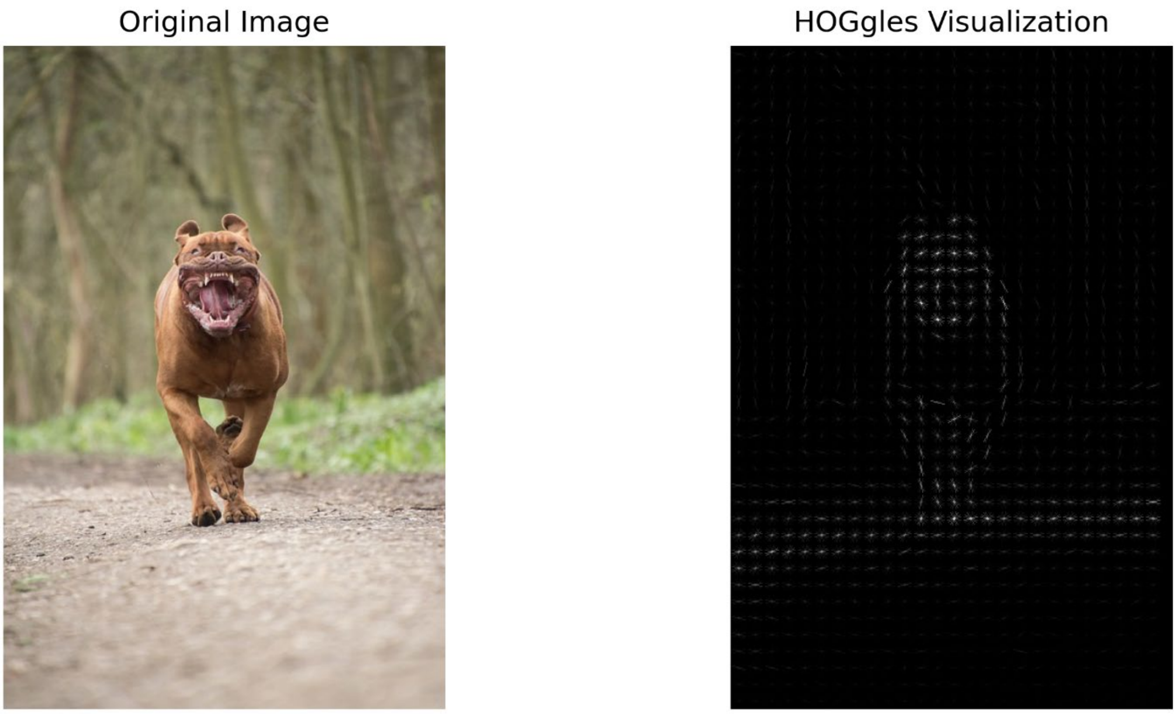


Fig 18: HOG Visualization of Dog

Figure 19 shows the visualization, the person HOG visualization highlights the unique shape of the human form. Strong gradient patterns can be seen around the head, shoulders and body. Visualization captures the characteristics of a standing person's proportions and position and indicates how the model may recognize human figures in images.

Fig 19: HOG Visualization of Person

These HOG visualizations provide valuable insight into the modelling process of the features extraction:

- **Edge And Contour Detection:** HOG effectively captures the edges and contours of objects, which are crucial for the recognition based on shapes.
- **Structural Information:** Visualizations show how models extract structural information from images, focusing on characteristics of different object classes.
- **Orientation Sensitivity:** HOG's ability to capture gradient orientations is evident, especially in objects with distinctive directional features such as the wings of the aircraft or the edges of the bus.
- **Representation Of Texture:** Fine textures and local appearances are also captured, as seen in the patterns like fur in dog and cow visualizations.
- **Scale Invariance:** HOG representations seem to be effective over different sizes of objects, from large buses to smaller dogs, which indicates the model's ability to recognize objects at different scales.

These HOG visualizations show together the model's ability to extract key structural features from different objects. They reveal how edge information, shape patterns and texture elements contribute to the object detection process of the model. The consistency in the capture of the essential characteristics of objects in different scenarios highlights the robustness of the model's functional extraction capabilities. In addition, HOG visualizations provide insight into potential areas for model improvement. For example, the different intensity of gradients between brown and black cows indicates that the model can be more sensitive to certain color contrasts. This observation could lead to future improvements in the model's ability to handle objects with different color profiles equally well. The detailed information on edges and gradients collected in these visualizations also explains the high confidence ratings of the model in most cases. By clearly identifying the shapes and structures of each object, the model establishes a solid basis for accurate classification and location. HOG visualizations offer a valuable window into the model's feature extraction process. They show how the model identifies and uses key structural elements for object detection, providing both validation of its current capabilities and information for potential future improvements. This analysis reinforces the importance of edge and gradient information in object detection tasks and emphasizes the complex way in which the model processes visual data to achieve accurate results.

8.3.2 Explainable AI - Grad-CAM Visualizations:

Grad-CAM (Gradient-weighted Class Activation Mapping) is a widely utilized technique in the domain of deep learning for visualizing and comprehending the decision-making process of convolutional neural network (CNN) models. This method generates visual explanations by highlighting regions in an image that are crucial for the model's predictions, thereby providing a transparent view of the model's internal workings. The subsequent analysis focuses on the Grad-CAM visualizations derived for various classes in an object detection task, including cows, buses, dogs, and airplanes.

Grad CAM visualization Cow:

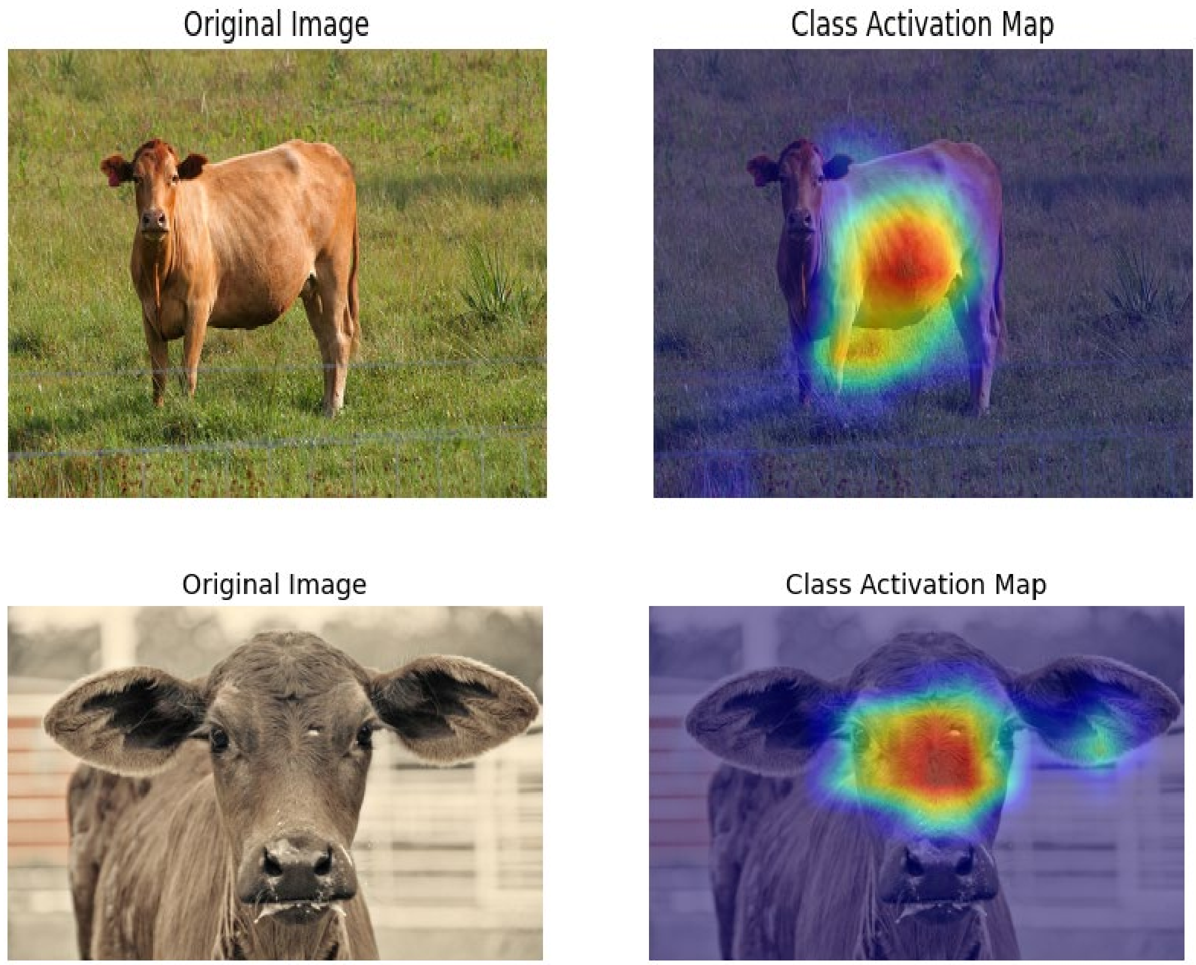


Fig 20: Grad CAM Visualization of Cows

Grad-CAM visualizations of cows in the Figure 20, class clearly reveal the focus of the model on specific areas of the image. The heatmap overlay on the original image indicates that the model mainly focuses on the body and head of the cow, areas rich in characteristics related to the cow class. This concentration of attention is consistent with the expected areas of interest, confirming the model's ability to accurately locate and identify cows within the image. The visualization suggests that the model uses key features such as the shape, texture and specific markings of the cow to inform its prediction.

Grad CAM visualization Dog

Fig 22: Grad CAM Visualization of Dog

In the case of dog classes, the Grad-CAM heat map highlights the area around the head and body of the dog as shown in the Figure 22. This concentration on the head is particularly notable because it includes features such as eyes, ears and muzzles that are very specific for dog identification. The visualization highlights the ability of the model to capture essential details that distinguish dogs from other animals, such as their facial structure and body proportion. This focused attention supports the high accuracy of the model for classification and localization of dogs in various environments.

Grad CAM visualization Airplane

The Grad-CAM visualization for the aircraft class shows a significant focus on wings and fuselage as shown in the Figure 23. These areas are crucial for recognizing aircraft because of their unique structural characteristics. The heat map shows that the model effectively captures the characteristic characteristics of an aircraft, such as its wing length and overall shape. This concentrated attention helps to distinguish aircraft from other objects accurately and ensure reliable detection in various scenarios.

Grad-CAM visualizations between different classes provide valuable insights into the deterministic and robustness of the model. For each class, heat maps show that the model successfully identifies and focuses on relevant features needed for accurate classification. These visualizations demonstrate the ability of the model to generalize well across different object categories using different characteristic features. Grad-CAM application not only facilitates validation of the performance of the model, but also helps to diagnose potential problems. For example, in cases of bad classification, Grad-CAM visualizations can show whether the model has focused on irrelevant regions or has not captured essential characteristics. These insights are crucial to refine and improve the model's accuracy. In conclusion, Grad-CAM functions as an essential tool in the field of visualization techniques, offering a clear and interpretable means of understanding how deep learning models perceive and process images. The ability to visualize attention maps improves our understanding of model behaviour and leads to improved modelling design, evaluation and deployment in real applications. The visuals discussed here confirm Grad-CAM's effectiveness in describing model decisions and thus contribute to the development of more transparent and reliable AI systems.

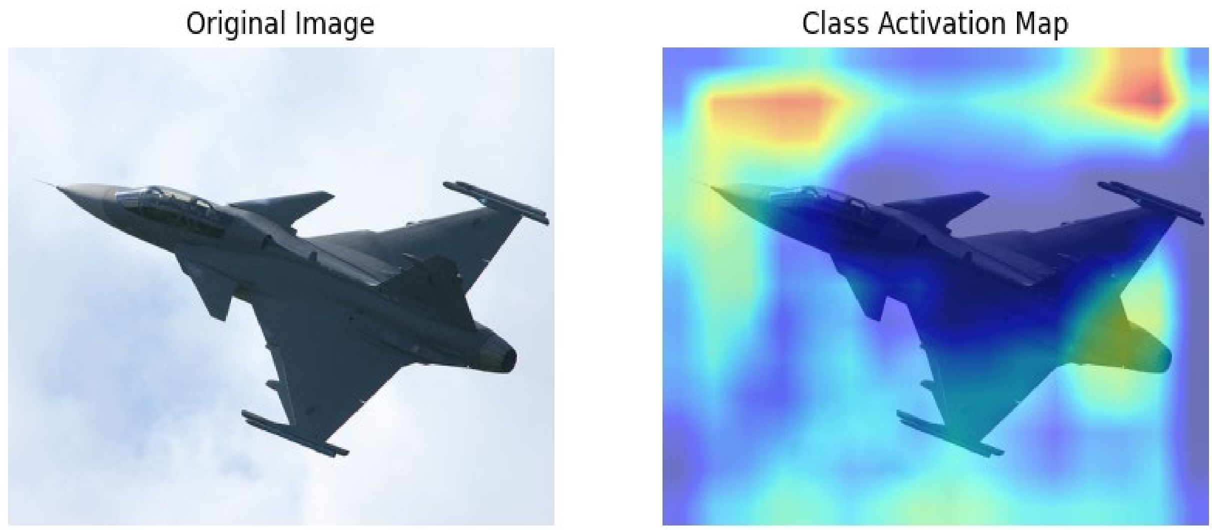


Fig 23: Grad CAM Visualization of Aeroplan

CHAPTER 9

CONCLUSION & FUTURE SCOPE

Conclusion:

The study offers a comprehensive examination of the optimization and analysis of the performance of an object detection model based on the Faster R-CNN architecture with a ResNet50-FPN core. Through a series of experiments that varied learning rates and epoch numbers, we identified the optimal hyperparameters that balance the accuracy and computational efficiency of the model. The best configuration, using a learning rate of 0.005 and 30 training epochs, achieved impressive results with an average accuracy of 0.8114, an average F1 score of 0.6232, and an average IoU score of 0.7096. This configuration demonstrated robust performance in a variety of scenarios, including urban environments, natural environments and complex indoor scenes.

The main findings of the study are the observation that moderate learning rates (0.001-0.005) usually outperform extremely low or high values, suggesting an optimal range for gradient descent optimization. Furthermore, the training period significantly affected the performance of the model, with 30 years providing the best balance between model convergence and generalization. The model demonstrated its strength in detecting several object classes, adjusting occlusions and performing well under different lighting conditions. Visualization techniques, including HOG and Grad-CAM, provided valuable insights into the decision-making process of the model and the use of characteristics. Furthermore, the color analysis function added an additional dimension to the object characterization, potentially increasing the applicability of the model in specialized fields.

Future Scope

The current model has achieved high performance, but there are several areas for future research and development. Fine-grained parameter tuning, could potentially produce even better performance. Advanced data enhancement techniques such as Mix up and CutMix could increase model robustness to various environmental conditions and object orientations. In addition, experiments with different support networks (e.g. Efficient Net, ResNeXt) or the study of more recent architectures such as DETR (Detection Transformer) could potentially improve detection accuracy and efficiency. Given the challenges encountered in detecting smaller objects, future work could focus on the integration of technologies specifically designed to improve the detection of smaller objects, such as higher-resolution pyramidal network features. Furthermore, adapted models for specific applications such as autonomous driving and surveillance by specialized tuning could improve their applicability in the real world.

Further development of AI-enabled explanation techniques could provide a deeper understanding of the model's decision-making process, leading to more trustworthy and interpretable models. Finally, exploring model compression techniques and specific hardware optimizations could improve the accuracy of inferences for real-time applications while incorporating additional data models such as depth information and time data could improve the performance of models in complex scenarios. By addressing these areas, future iterations of the model could achieve greater precision, greater generalization and greater application across various computer vision tasks. The promising results of this study provide a solid basis for future efforts in the field of object detection.

REFERENCES

[1] Suherman, Endang, et al. ”Implementation of ResNet-50 on End-to-End Object Detection (DETR) on Objects.” Sinkron: jurnal dan penelitian teknik informatika 8.2 (2023): 1085-1096.

[2] Yanagisawa, H., Yamashita, T., & Watanabe, H. (2018). A Study on Object Detection Method from Manga Im- ages using CNN. In 2018 5th International Workshop on Advanced Image Technology (IWAIT) (pp. 1-4). IEEE.

[3] Gong, H.; Mu, T.; Li, Q.; Dai, H.; Li, C.; He, Z.; Wang, W.; Han, F.; Tuniyazi, A.; Li, H.; et al. Swin- Transformer-Enabled YOLOv5 with Attention Mecha- nism for Small Object Detection on Satellite Images. Remote Sens. 2022, 14, 2861.

[4] Ko¨rez, Atakan, and Necaattin Barıs¸c¸ı. ”Object detection with low capacity GPU systems using improved faster R-CNN.” Applied Sciences 10.1 (2019): 83.

[5] Gu¨ney, Emin, and Cu¨neyt BAYILMIS¸. ”An implemen- tation of traffic signs and road objects detection using faster R-CNN.” Sakarya University Journal of Computer and Information Sciences 5.2 (2022): 216-224.

[6] Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: Towards Real-Time Object Detection with Re- gion Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence.

[7] Lin, Chun-Liang, Yan-Lin Chen, and Yu-Chen Lin. ”Small Objects Detection Using Transformer-CNN.” Available at SSRN 4232850.

[8] Dsouza, P. (2023). Detection of Small Objects Using CNN. International Research Journal of Modernization in Engineering Technology and Science, 5(5), 6818.

[9] Vatchalaa, S., et al. ”Smart Household Object Detection Using CNN.” Advances in Parallel Computing Algo- rithms, Tools and Paradigms 41 (2022): 464.

[10] Kinasih, Fabiola Maria Teresa Retno, et al. ”Two-stage multiple object detection using CNN and correlative filter for accuracy improvement.” Heliyon 9.1 (2023).

[11] Ko¨rez, Atakan, and Necaattin Barıs¸c¸ı. ”Object detection with low capacity GPU systems using improved faster R-CNN.” Applied Sciences 10.1 (2019): 83.

[12] Yanto, Musli, and Eka Naufaldi Novri. ”Application of Object Mask Detection Using the Convolution Neural Network (CNN).” Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi) 7.4 (2023): 922-929.

[13] Liu, Peng, et al. ”Unsupervised image-generation en- hanced adaptation for object detection in thermal im- ages.” Mobile information systems 2021 (2021): 1-6.

[14] Fang, Yuxin, et al. ”Unleashing vanilla vision transformer with masked image modeling for object detection.” Pro- ceedings of the IEEE/CVF International Conference on Computer Vision. 2023.

[15] Minderer, M., et al. ”Simple open-vocabulary object detection with vision transformers. arXiv 2022.” arXiv preprint arXiv:2205.06230.

[16] Hanafi, Anasse, Lotfi Elaachak, and Mohammed ouhorma. ”Machine learning based augmented reality for improved learning application through object detection algorithms.” International Journal of Electrical & Computer Engineering (2088-8708) 13.2 (2023).

[17] Amjoud, Ayoub Benali, and Mustapha Amrouch. ”Ob- ject Detection Using Deep Learning, CNNs and Vision Transformers: A Review.” IEEE Access (2023).

[18] Cui, Liyuan, et al. ”A Compact Object Detection Archi- tecture with Transformer Enhancing.” Journal of Physics: Conference Series. Vol. 2278. No. 1. IOP Publishing, 2022.

[19] Deshpande, H., A. Singh, and H. Herunde. ”Compara- tive analysis on YOLO object detection with OpenCV.” International journal of research in industrial engineering 9.1 (2020): 46-64.

[20] Mtasher, Ashwaq Katham, and Esraa Hassan Jawad Al-wakel. ”Custom Object Detection Using Transfer Learning with Pretrained Models for Improved Detection Techniques.” Journal La Multiapp 5.1 (2024): 10-18.

[21] Lin, Chih-Yang, Cheng-Hao Yeh, and Chia-Hung Yeh. ”Real-time vehicle color identification for surveillance videos.” 2014 International Conference on Electronics, Communications and Computers (CONIELECOMP). IEEE, 2014.

[22] Chang, Yeong-Hwa, et al. ”Enhanced image captioning with color recognition using deep learning methods.” Applied Sciences 12.1 (2021): 209.

[23] Raguraman, P., et al. ”Color detection of RGB images using python and OpenCv.” International Journal of Sci- entific Research in Computer Science, Engineering and Information Technology 7 (2021).

[24] Hossen, M. K., et al. ”Application of Python-OpenCV to detect contour of shapes and colour of a real image.” International Journal of Novel Research in Computer Science and Software Engineering 9 (2022): 20-25.

[25] Shakeel, Khan Shayaan et al. “Generate Detailed Cap- tions of an Image Using Deep Learning.” International Journal for Research in Applied Science and Engineering Technology 10.4 (2022): 734–742. International Journal for Research in Applied Science and Engineering Tech- nology. Web.

[26] Mobile Computing, Wireless Communications and. “Re- tracted: Image Color Recognition and Optimization Based on Deep Learning.” Wireless Communications and Mobile Computing 2023 (2023): 1–1. Wireless Commu- nications and Mobile Computing. Web.

[27] Li, Peng. “Image Color Recognition and Optimization Based on Deep Learning.” Wireless Communications and Mobile Computing 2022 (2022): n. pag. Wireless Communications and Mobile Computing. Web.

[28] Tran-Trung, Kiet, Ha Duong Thi Hong, and Vinh Truong Hoang. ”Weather Forecast Based on Color Cloud Image Recognition under the Combination of Local Image Descriptor and Histogram Selection.” Electronics 11.21 (2022): 3460.

[29] A N, Sunitha, and Susmitha M N. “Classification and Object Detection on Satellite Images Using Custom CNN Architecture.” International Journal of Innovative Research in Computer and Communication Engineering 12.07 (2023): n. pag. International Journal of Innovative Research in Computer and Communication Engineering. Web.

[30] Iwamura, Kiyohiko et al. “Image Captioning Using Motion-Cnn with Object Detection.” Sensors (Switzer- land) 21.4 (2021): 1–13. Sensors (Switzerland). Web.

[31] Apriyanti, Diah Harnoni, et al. ”Automated color detec- tion in orchids using color labels and deep learning.” PloS one 16.10 (2021): e0259036.

[32] Patil, Sanjay M., et al. ”An efficient approach for object detection using deep learning.” Journal of Pharmaceutical Negative Results (2022): 563-572.

[33] Erhan, Dumitru, et al. ”Scalable object detection using deep neural networks.” Proceedings of the IEEE confer- ence on computer vision and pattern recognition. 2014.

[34] Chandan, G., Ayush Jain, and Harsh Jain. ”Real time object detection and tracking using Deep Learning and OpenCV.” 2018 International Conference on inventive research in computing applications (ICIRCA). IEEE, 2018.

[35] Yeh, Chia-Hung, et al. ”Lightweight deep neural network for joint learning of underwater object detection and color conversion.” IEEE Transactions on Neural Net- works and Learning Systems 33.11 (2021): 6129-6143.

[36] Chen, Yu Peng, Ying Li, and Gang Wang. ”An Enhanced Region Proposal Network for object detection using deep learning method.” PloS one 13.9 (2018): e0203897.

[37] Galvez, Reagan L., et al. ”Object detection using con- volutional neural networks.” TENCON 2018-2018 IEEE Region 10 Conference. IEEE, 2018.

[38] Tang, Cong, et al. ”The object detection based on deep learning.” 2017 4th international conference on informa- tion science and control engineering (ICISCE). IEEE, 2017.

[39] Badea, Mihai-Sorin, et al. ”Automatic burn area identi- fication in color images.” 2016 international conference on communications (COMM). IEEE, 2016.

[40] Wantanajittikul, Kittichai, et al. ”Automatic segmentation and degree identification in burn color images.” The 4th 2011 Biomedical Engineering International Conference. IEEE, 2012.

[41] Sarraf, Saman. ”Hair color classification in face recog- nition using machine learning algorithms.” American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS) 26.3 (2016): 317-334.

[42] Aarathi, K. S., and Anish Abraham. ”Vehicle color recognition using deep learning for hazy images.” 2017 International Conference on Inventive Communication and Computational Technologies (ICICCT). IEEE, 2017.

[43] Binangkit, Jeffrey L., and Dwi H. Widyantoro. ”Increas- ing accuracy of traffic light color detection and recog- nition using machine learning.” 2016 10th international conference on telecommunication systems services and applications (TSSA). IEEE, 2016.

[44] Duong, Duc Long, Malitha Humayun Kabir, and Rong Fu Kuo. ”Automated caries detection with smartphone color photography using machine learning.” Health In- formatics Journal 27.2 (2021): 14604582211007530.

[45] Tong, Kang, and Yiquan Wu. ”Rethinking PASCAL- VOC and MS-COCO dataset for small object detection.” Journal of Visual Communication and Image Representation 93 (2023): 103830.

[46] Chun, Sanghyuk, et al. ”Eccv caption: Correcting false negatives by collecting machine-and-human-verified image-caption associations for ms-coco.” European Con- ference on Computer Vision. Cham: Springer Nature Switzerland, 2022.

[47] Xia, Gui-Song, et al. ”DOTA: A large-scale dataset for object detection in aerial images.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[48] Nepal, Upesh, and Hossein Eslamiat. “Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs.” Sensors 22.2 (2022): n. pag. Sensors. Web.

[49] Jiang, Shenlu et al. “An Optimized Deep Neural Network Detecting Small and Narrow Rectangular Objects in Google Earth Images.” IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 13 (2020): 1068–1081. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing. Web.

[50] Li, Qingyun, Yushi Chen, and Ying Zeng. “Transformer with Transfer CNN for Remote-Sensing-Image Object Detection.” Remote Sensing 14.4 (2022): n. pag. Remote Sensing. Web.

[51] Kobayashi, Takumi. ”BFO meets HOG: feature extrac- tion based on histograms of oriented pdf gradients for image classification.” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2013.

[52] Rabbani, Fahmi Muhammad Rabbani, Bima Sena Bayu Dewantara, and Endra Pitowarno. “Lip Movement Recognition Using Histogram of Oriented Gradient (HOG) and Support Machine Vector (SVM) for Arabic Word.” Indonesian Journal of Computer Science 12.6 (2024): n. pag. Indonesian Journal of Computer Science. Web.

[53] Berrimi, Fella et al. “The Histogram of Enhanced Gra- dients (HEG) - A Fast Descriptor for Noisy Face Recog- nition.” Revue d’Intelligence Artificielle 38.1 (2024): 33–44. Revue d’Intelligence Artificielle. Web.

[54] Srikar, M., and K. Malathi. ”A Supervised Stable Object Detection with Image Feature Extraction using Image Segmentation by Comparing Histogram of Oriented Gra- dients (HOG) Algorithm over Scale Invariant Feature Transform (SIFT) Algorithm Model.” Journal of Phar- maceutical Negative Results (2022): 1708-1714.

[55] Alkaff, Muhammad et al. “Sasirangan Motifs Classifi- cation Using Scale- Invariant Feature Transform (SIFT) and Support Vector Machine (SVM).” MATEC Web of Conferences 280 (2019): 05023. MATEC Web of Conferences. Web.

[56] Muhima, Rani Rotul, and Hendro Nugroho. “Penera- pan Euclidean Distance Untuk Deteksi Kemiripan Citra Berbasis Scale Invariant Feature Transform (SIFT).” IN- TEGER: Journal of Information Technology 7.1 (2022): n. pag. INTEGER: Journal of Information Technology. Web.

[57] Sabharwal, Tanupreet, and Rashmi Gupta. “Human Face Identification after Plastic Surgery Using SURF, Multi- KNN and BPNN Techniques.” Complex and Intelligent Systems (2024): n. pag. Complex and Intelligent Sys- tems. Web.

[58] Saini, Ashok Kumar, Roheet Bhatnagar, and Devesh Kumar Srivastava. “SADCNN-ORBM: A Hybrid Deep Learning Model Based Citrus Disease Detection and Classification.” International Journal of Electrical and Computer Engineering 14.2 (2024): 2191–2201. Interna- tional Journal of Electrical and Computer Engineering. Web.

[59] Diaa, Uliyan. “A Deep Learning Model to Inspect Im- age Forgery on SURF Keypoints of SLIC Segmented Regions.” Engineering, Technology Applied Science Research 14.1 (2024): 12549–12555. Engineering, Tech- nology Applied Science Research. Web.

[60] Laukikkumar, K., and Manish I. Patel. “Feature Based Image Registration Using CNN Features for Satellite Im- ages Having Varying Illumination Level.” International Journal of Advanced Technology and Engineering Ex- ploration 10.101 (2023): 440–457. International Journal of Advanced Technology and Engineering Exploration. Web.

[61] Hua, Jie et al. “Pedestrian-and Vehicle-Detection Algo- rithm Based on Improved Aggregated Channel Features.” IEEE Access 9 (2021): 25885–25897. IEEE Access. Web.

[62] Mirza, Nabeel M. et al. “Static Hand Gesture Angle Recognition via Aggregated Channel Features (ACF) Detector.” Traitement du Signal 39.3 (2022): 939–944. Traitement du Signal. Web.

[63] Piniarski, Karol, Paweł Pawłowski, and Adam Dabrowski. “Tuning of Classifiers to Speed-up Detection of Pedestrians in Infrared Images.” Sensors (Switzerland) 20.16 (2020): 1–20. Sensors (Switzerland). Web.

[64] Ozcan, Koray et al. “Milkweed (Asclepias Syriaca) Plant Detection Using Mobile Cameras.” Ecosphere 11.1 (2020): n. pag. Ecosphere. Web.

[65] Liu, Chun, et al. ”Object Detection in Remote Sensing Images Based on Adaptive Multi-Scale Feature Fusion Method.” Remote Sensing 16.5 (2024): 907.

[66] VS, Vibashan, Poojan Oza, and Vishal M. Patel. ”In- stance relation graph guided source-free domain adaptive object detection.” Proceedings of the IEEE/CVF Confer- ence on Computer Vision and Pattern Recognition. 2023.

[67] Li, Jinlong, et al. ”Domain adaptive object detection forautonomous driving under foggy weather.” Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2023.

[68] Alqasir, Hiba, Damien Muselet, and Christophe Ducottet. ”Region proposal oriented approach for domain adap- tive object detection.” Advanced Concepts for Intelligent Vision Systems: 20th International Conference, ACIVS 2020, Auckland, New Zealand, February 10–14, 2020, Proceedings 20. Springer International Publishing, 2020.

[69] Zhao, Ganlong, et al. ”Collaborative training between region proposal localization and classification for do- main adaptive object detection.” Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16. Springer In- ternational Publishing, 2020.

[70] Liu, Yun, et al. ”Richer convolutional features for edge detection.” Proceedings of the IEEE conference on com- puter vision and pattern recognition. 2017.

[71] Suryowinoto, Andy et al. “UNDERWATER OBJECT SHAPE DETECTION BASED ON TONAL DISTRI- BUTION AND EDGE DETECTION USING DIG- ITAL IMAGE PROCESSING.” BAREKENG: Jurnal Ilmu Matematika dan Terapan 18.1 (2024): 0395–0402. BAREKENG: Jurnal Ilmu Matematika dan Terapan. Web.

[72] Simangunsong, Pandi Barita Nauli, and Paska Marto Ha- sugian. “Pattern Recognition in Medical Images Through Innovative Edge Detection with Robert’s Method.” Jurnal Info Sains: Informatika dan Sains 14.01 (2024): 660–667. Jurnal Info Sains: Informatika dan Sains. Web.

[73] Prasetyo, Adi Budi et al. “Comparative Analysis of Image on Several Edge Detection Techniques.” TEM Journal 12.1 (2023): 111–117. TEM Journal. Web.

[74] Xiang, Yunfan et al. “EGMT-CD: Edge-Guided Multi- modal Transformers Change Detection from Satellite and Aerial Images.” Remote Sensing 16.1 (2024): n. pag. Remote Sensing. Web.

[75] Puspadini, Ratih, and Melda Pita Uli Sitompul. “Imple- mentation of the Laplacian of Gaussian Algorithm in Edge Detection Image Processing of Zebra Cross Dam- age on Highways in the Langkat Regency Area.” Journal of Artificial Intelligence and Engineering Applications (JAIEA) 3.2 (2024): 601–605. Journal of Artificial Intel- ligence and Engineering Applications (JAIEA). Web.

List of Paper Accepted

1. A survey on Object Detection and Color Identification is accepted and published by High Technology Letters (HTL) Journal
2. Object Detection using CNN and Colour Recognition is accepted by International Research Conference on Science Technology, Engineering and Management (IRCSTEM)
3. A survey on Object Detection and Color Identification is accepted by International Conference on Smart Technology, Artificial Intelligence and Computer Engineering (ICSTAICE-2024)
4. Object Detection using CNN and Colour Recognition is accepted by International Conference on Advanced Research in Computer Science and Information Technology (ICARCSIT-2024)

High Technology Letters

ISSN NO : 1006-6748, IMPACT FACTOR : 2.7, PEER REVIEW, INTERNATIONAL JOURNAL

Certificate ID : HTL/9154

CERTIFICATE OF PUBLICATION

This is to certify that the paper entitled

"Recent Advancements and Challenges in Object Detection and Color Identification: A Survey"

Authored by

Kushal G

From

Nitte Meenakshi Institute of Technology, Bengalore, Karnataka, India.

Has been published in

HTL JOURNAL, VOLUME 30, ISSUE 5, MAY - 2024

ज्ञान-विज्ञान विमुक्तये
UGC
University Grants Commission

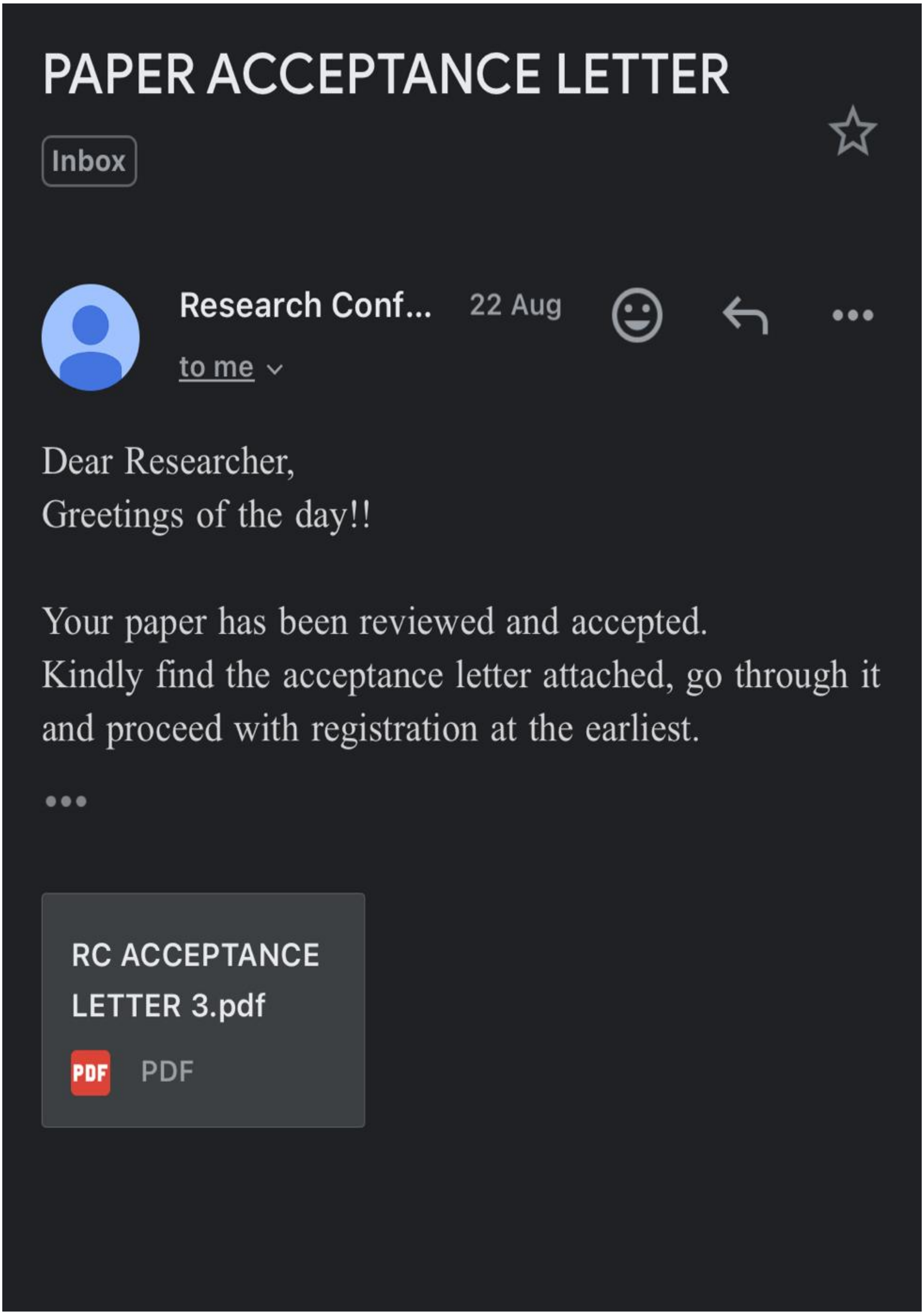
Scopus

Google scholar

Dr. ANAND PALLA Ph.D.,
Editor-In-Chief
HTL JOURNAL
http://www.gjstx-e.cn/

ISO International Organization for Standardization 7021-2008

doi.org/10.37896/HTL
crossref member
CROSSREF.ORG
THE CITATION LINKING BACKBONE


PAPER ACCEPTANCE LETTER

Inbox

Research Conf... 22 Aug

to me

Dear Researcher,
Greetings of the day!!

Your paper has been reviewed and accepted.
Kindly find the acceptance letter attached, go through it and proceed with registration at the earliest.

RC ACCEPTANCE LETTER 3.pdf

PDF

Acceptance Letter & Registration Form

Inbox

ASAR 23 Aug

to me

Dear Author,

We are happy to inform you that your Manuscript has been selected for our upcoming conference to be held at **Mysuru, India** which will be organized by **ASAR** for presentation at the Conference.

Paper ID: **AS-CSIT-MYSR-010924-6831**

Paper Title: **Object Detection with Deep Learning and K-Means Clustering for Color Recognition**

Conference Name: **International Conference on Advanced Research in Computer Science and Information Technology (ICARCSIT-2024)**

Conference Date: **1st September 2024**

Conference Place: **Mysuru, India**

Acceptance Letter & Registrtaion Form

Inbox

IRAJ Conferenc... 20 Aug

to me

Dear Author,

We are happy to inform you that your Manuscript has been selected for our upcoming conference to be held at **Bangalore, India** which will be organized by **IRAJ** for presentation at the Conference.

Paper ID: **IR-AICE-BGLR-060924-3740**

Paper Title: **Recent Advancements and Challenges in Object Detection and Color Identification: A Survey**

Conference Name: **International Conference on Smart Technology, Artificial Intelligence and Computer Engineering (ICSTAICE-2024)**

Conference Date: **6th September 2024**

Conference Place: **Bangalore, India**