International Journal of Electrical and Computer Engineering (IJECE)
Vol. 16, No. 2, April 2026, pp. 863~872
ISSN: 2088-8708, DOI: 10.11591/ijece.v16i2.pp863-872

Integrated deep learning approach for real-time object detection and color analysis

Srinivas Dibbur Byrappa[1], Kushal Gajendra[1], Rohith Holenarasipura Puttaraju[1], Tumakalahalli Nagaraj Malini[2]
[1]Department of Information Science and Engineering, Nitte Meenakshi Institute of Technology, Bangaluru, India
[2]Department of Management Studies, Nitte Meenakshi Institute of Technology, Bangaluru, India

Article Info

Article history:

Received May 17, 2025
Revised Dec 3, 2025
Accepted Jan 15, 2026

Keywords:

Convolutional neural networks
Feature pyramid network
Gradient class activation
Histogram of oriented gradients
Object detection
Pascal VOC
Region proposal network

ABSTRACT

Object identification is one of the major application areas of deep learning that provides significantly better feature extraction and representation than more conventional methods of recognition. Driven by the growing significance of conjunction of objects detection and color interpretation in contemporary computer vision systems, the current work proposes an integrated, real-time deep learning system that completes the task of object localization and color analysis. It is suggested that the proposed system employs a faster region-based convolutional neural network (Faster R-CNN) with backbone of ResNet-50 and supplemented with a feature pyramid network to perform multi-scale feature aggregation. The model was trained and tested using the Pascal VOC 2012 dataset and it showed good results with the average precision of 0.8114, F1 of 0.6232 and IoU of 0.7096. The large set of experiments on different learning rates and training epochs allowed optimizing the detector to work well in a variety of conditions. To enhance even more, visualization histogram of oriented gradients (HOG) and gradient-weighted class activation mapping (Grad-CAM) was used to gain a more profound understanding of the significance of features and the logic behind a model. This study complements image perception with color by combining object recognition and color in a single architecture, which can result in fruitful applications in areas of autonomous vehicles, industrial automation, and medical imaging.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Srinivas Dibbur Byrappa
Department of Information Science and Engineering, Nitte Meenakshi Institute of Technology
6429, NITTE Meenakshi College Rd, BSF Campus, Yelahanka, Bengaluru, Govindapura, Karnataka 560064, India
Email: srinivas.db@nmit.ac.in

1. INTRODUCTION

Object detection is one of the most important and one of the most popular tasks of computer vision and a fundamental field of artificial intelligence. It provides a basis for a variety of practical applications, such as autonomous navigation, industrial quality control, robotics, and medical diagnostics. Over the last few years, deep learning, particularly convolutional neural networks (CNNs) have had multiple profound implications on object detection, including automated learning of features and providing very robust classification and localization performance. Conventional computer vision methods were intensive on statistical models likely to fail in the presence of variability in the appearance of objects, lighting, and complexity of the background. With the emergence of deep learning, CNNs [1], [2] have so far transformed object detection by automatically acquiring hierarchical feature representations using raw image inputs. The state-of-the-art object detection models like the Faster R-CNN model [3], you only look once (YOLO) [4], [5]. The proposed model in [6] is carried out using less memory and less processing power and processing mass data using Pascal VOC and Microsoft COCO datasets. Quick detection algorithm can discriminate the object in 0.82 and above seconds.

However, there is a critical gap in existing literature that has not been fully used, *i.e.*, the use of intrinsic object attributes, *e.g.* color information in these detection pipelines. The current models are mainly based on features of space and shape without much focus on color features which are essential in various practical situations. To demonstrate, one of the main distinctions in the use of color is in traffic signal detection, sorting of fruits in agriculture, or the scanning of defects on industrial products. The failure to pay attention to color may result in misclassifications in situations where color coded objects or appearance similar objects are separated primarily by color. Moreover, the second significant drawback of the state-of-the-art is the inability to interpret deep learning models. Even though CNN-based detectors are highly performing, they are black boxes, which means that end-users in safety-critical systems (autonomous vehicles or medical devices) cannot rely on model output unless there is an interpretability mechanism. To overcome them, we introduce a new hybrid framework that will involve CNN-based object detection (Fast R-CNN with ResNet50-feature pyramid network (FPN) backbone), color analysis with the k-means clustering algorithm and explainable artificial intelligence (AI) methods through gradient-weighted class activation mapping (Grad-CAM) and histogram of oriented gradients (HOG) [7].

The remainder of this article is structured as: Section 2 provides a brief overview of related work. Section 3 outlines the method. In section 4 results are discussed, and finally section 6 concludes the article.

2. RELATED WORK

Recent advances in object detection technologies have been significantly improved in various fields and architecture. Studies [8] have shown that integrating ResNet-50 with detection with transformers (DETR) (a transformer-based model) achieved approximately 90% better results than traditional CNNs, while transformers combined with CNNs achieved an average precision (AP) of 20.6 for small objects [9]. Faster R-CNN has been effective in niche applications such as manga character and text detection, achieving metrics as high as 0.816 and 0.898, respectively [10]. Satellite imagery analysis [11] using a custom CNN approach achieved an accuracy of 94.65% to detect vehicles, buildings, and trees. Traffic sign detection with a two-stage Faster R-CNN model achieved a detection precision of 88.99% [12], while specialized applications such as unmanned aerial vehicle (UAV) detection showed better results with LSL Net [13]. Detection of household objects using ResNet50 and support vector machine (SVM) is presented in [14]. CNN is used to detect objects, ResNet50 is used to classify the images into objects, and then the SVM is used to train objects and stored in the object database. A two-phase approach combining correlative filter tracking and CNN showed significant improvements in study [15] for various scenarios. A Faster R-CNN model with a regional proposal network [16] showed strong results on benchmarks, such as MS COCO and Pascal VOC [17].

The object identification algorithm based on the characteristic color with the YUV color space is presented in study [18]. A study - expansion algorithm to obtain the spatial distribution of the compatible object color characteristic. The images containing CCTV faces are segmented in study [19], experiments were carried out on a single face image in mask detection, resulting in an accuracy of 97.33%. Similarly, thermal image detection reported a mean AP (mAP) of 26.5% [20]. In other applications, an masked image modeling (MIM)-pretrained vanilla vision transformer (ViT) [21] encoder achieved an AP of 51.5, and vision transformers demonstrated promising open-vocabulary detection, achieving an AP of 31.2% on the LVIS v1.0 dataset [22]. Specialized underwater detection models for low-power systems, such as the Raspberry Pi, demonstrated by using multi-scale feature learning to improve computational efficiency [23]. Images captured from roads or hilly regions often suffer from poor visibility due to haze [24]. Removing this haze can significantly enhance color recognition. The proposed system addresses this by integrating two key techniques: first, it employs the dark channel prior method to eliminate haze; next, it utilizes CNN for feature learning. Once the features are extracted, an effective classification method, such as the SVM, is used to perform the final classification. Python-OpenCV [25] has proven effective in RGB color recognition for color detection in computer vision. Gupta *et al.* [26] highlight how the model utilizes deep learning and natural language processing techniques to generate concise descriptions of input images. Comprehensive review of recent advances, methods, challenges, and future directions in object detection and color identification, focusing on deep learning and practical applications [27].

3. METHOD

The framework as illustrated in Figure 1 introduces a real time object detection and color analysis object of raw images system. It is based on a Faster R-CNN framework with a ResNet-50 backbone and a FPN that is used to detect objects with high levels of robustness and at multiple scales. The initial stage involves pre-processing and augmentation of the images and then through detection network. An expert color recognition unit operates on the identified areas through an analysis of clustering to find the prevailing colors and their percentages. The system can visualize the results with boundaries, color details, and heatmaps of Grad-CAM, which have continuous dynamic updates, to offer better user experience.

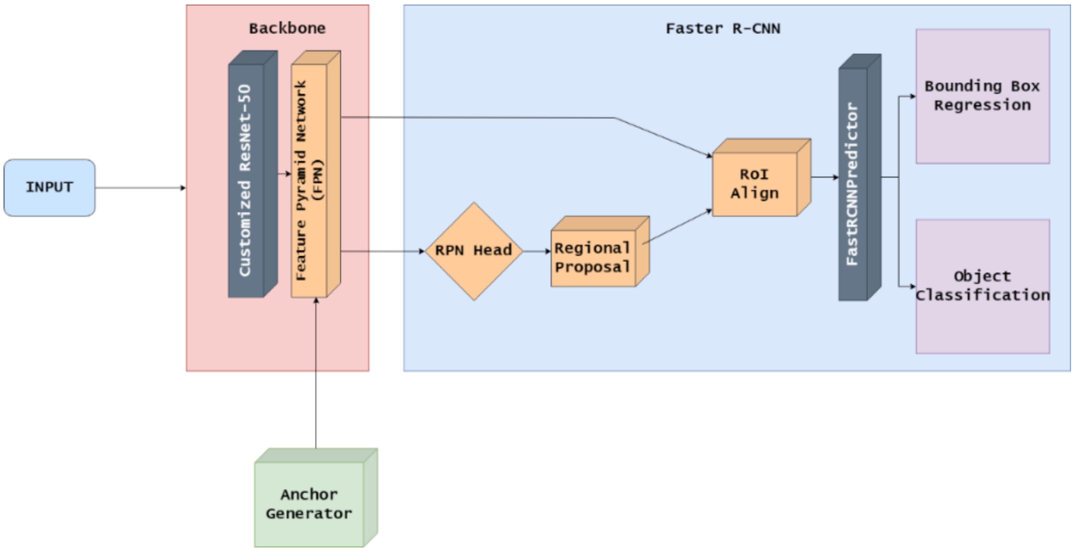

Figure 1. Proposed framework

3.1. Faster R-CNN

Faster R-CNN (Algorithm 1) serves as the primary detection module due to its strong accuracy in region-based object localization. The system uses a ResNet-50 backbone with an FPN to extract rich multi-scale features suitable for detecting objects of different sizes. A region proposal network (RPN) then generates candidate object regions by evaluating anchor boxes of various scales and aspect ratios and predicting object ness scores and refined coordinates. The top proposals are processed using region of interest (ROI) align, which produces spatially consistent, fixed-size feature maps, enabling precise classification and bounding box refinement.

3.2. Color recognition integration

The integration of color recognition into the Faster R-CNN framework involves analyzing the ROIs detected by the object detection model and performing detailed color analysis for each identified object. The process (Algorithm 2) starts by extracting a circular region from the center of each detected object's bounding box, ensuring the analysis focuses on the most representative part of the object while minimizing background interference. This extracted region is then subjected to k-means clustering, an efficient and widely used algorithm for identifying dominant colors within image patches by grouping similar pixel colors into clusters based on their proximity to cluster centers. The number of clusters (k) can be predefined, and the center of each cluster represents a dominant color found in the object region. This method enables robust and automated color identification for each detected object, supporting tasks that require precise color information within object detection pipelines.

3.3. Visualization

The framework integrates explainable AI methods HOG and Grad-CAM to improve transparency and interpretability in object detection and color analysis. HOG provides a traditional, hand-crafted visual explanation by mapping edge orientations and gradients, emphasizing shapes and contours that support object recognition. It allows comparison between conventional feature extraction and neural representations. Grad-CAM complements this by visualizing model attention using gradient-based heatmaps from deep networks like ResNet-50 with FPN. These maps highlight the image regions most influential in the model’s classification and localization decisions, revealing how the network focuses on specific spatial features during detection.

$$d(p,q) = \sqrt{(\Sigma(p_i - q_i))}\ for\ i = 1\ to\ n \quad (1)$$

The pixels within the identified region are used as inputs to the k-means algorithm (Algorithm 3) which clusters them into a specified number of clusters of colors with (1) (Euclidean distance) as the distance measure. The method allows breaking down not only one dominant color but also it gives an allocation of the major colors that form in the object. The most representative colors of an object are then the centroids of each cluster. The centers of these clusters are the most descriptive colors of the object. It has a color-naming component which maps the RGB values of cluster centroid to the closest named colors in a pre-existing color space, *e.g.* CSS3 color names [28], [29]. This assists in converting colors of objects into a human understandable format, which can be used in cases where the application requires the use of natural language colors.

3.4. Implementation

Figure 2 shows the implementation of the proposed framework. The PyTorch deep learning framework is used in the framework. The methods used consisted of random horizontal flipping at 50 percent frequency, random application of the rotations of ±10 degrees and brightness and contrast manipulation. The photos were resized to 800 pixels on the shorter dimension keeping the original aspect ratio to make sure that the input is as homogeneous as possible and that the significant visual information is not lost. The stochastic gradient descent (SGD) algorithm was used in the training stage where a momentum factor of 0.9 and a weight decay rate of 0.0005 were applied. In the analysis of color, k-means clustering algorithm is used to determine the prevailing color patterns in the detected object regions. It was experimentally tested that the five cluster approach (K=5) offered a good balance between computation efficiency and capacity to describe adequate color variability. To make the sampling constant and representative, a circular region is sampled out of any given observed object, and the radius is defined as a quarter of the lesser side of the bounding box of the object. The sampling strategy is uniform, which makes the method consistent in analyzing color information of objects of different shapes and sizes. All experiments were conducted using Python 3.10 with the PyTorch 2.x deep learning framework on a system equipped with an 12th gen intel(r) core(tm) i5-12500, 3000 Mhz 6 cores, 12 threads, 32 GB RAM, and Ubuntu 20.04.

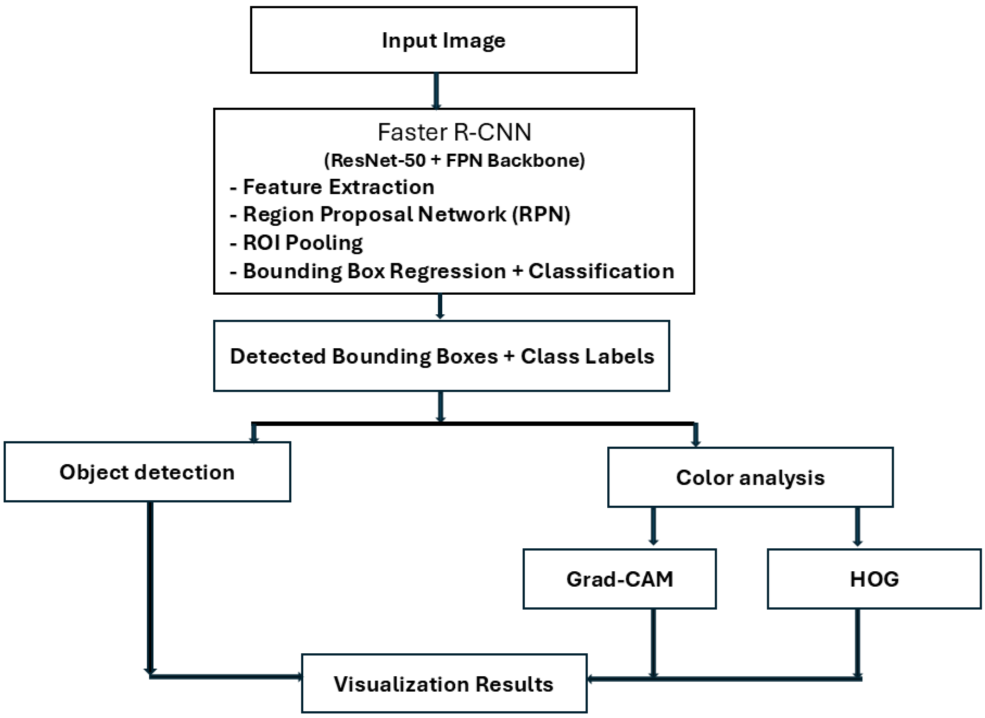

Figure 2. Flow chart

Algorithm 1. Object detection using Faster R-CNN
```
1: Feature Extraction
2: Extract features using ResNet50 backbone
3: Create multi-scale feature maps using FPN
4: Region proposal network (RPN)
5: Generate anchor boxes
6: Predict object proposals
7: Filter and refine proposals
8: Region of Interest (RoI) Processing
9: Align multi-scale feature maps with region proposals to get RoI features
10: Final Prediction
11: for each RoI do
12:     Classify the RoI
13:     Refine bounding box for the RoI
14: end for
15: Return Results
```

Algorithm 2. Color analysis procedure
```
Require: image, bounding box
Ensure: dominant color, color name, cluster centers, counts
1: Extract circular region from the bounding box.
2: Get dominant color:
3: Reshape images to 2D and remove black pixels.
4: if pixels are not empty then
5:     Perform k-means clustering.
6:     Calculate mean color.
7:     Find closest centroid to the mean color.
8:     Store dominant color, cluster centers, and counts.
9: end if
10: Return Results:
11: if dominant color is found then
12: Get color name from RGB value.
13: return dominant color, color name, cluster centers, counts.
14: else
15:     return None, None, None, None.
16: end if
```

Algorithm 3. k-means clustering procedure
```
Require: dataset D, number of clusters k, maximum iterations max iterations
Ensure: centroids, assignments
1: Initialize k centroids randomly
2: for iteration = 1 to max iterations do
3:     Assign each point in D to the nearest centroid
4:     Update centroids as the mean of the assigned points
5:     if centroids do not change then
6:     break
7:     end if
8: end for
9: return Final centroids, Cluster assignments
```

4. RESULTS AND DISCUSSION

The performance of our proposed framework is tabulated in Table 1. Experiment 10 (learning rate = 0.005 and epochs = 50) shows the highest mean precision of 0.81%, mAP 0.81%, and solid F1 score of 0.62%, indicating that more epochs (50) at a moderate learning rate yield the best overall performance. Experiments 1 and 6 also show high precision and mAP (0.77% and 0.74%), suggesting that 30 epochs with 0.005 or 0.001 learning rate also perform well. Similarly, experiment 7 (learning rate = 0.0001, epochs = 20) has the lowest performance in all metrics, especially with an extremely low F1 score (0. 09%) and mAP (0. 15%), indicating that too low a learning rate with fewer epochs leads to underfitting. Experiments 2 (30 epochs) and 8 (50 epochs) show similar metrics, which shows that higher epochs prevent further gain after a point, due to early convergence.

The Pascal VOC 2012 dataset is used for experimental purposes. This data set comprises 1 1,530 images with 27,450 annotated ROIs and 6,929 segmentations in 20 unique object classes. The proposed framework effectively detected and localized objects in various settings. In Figure 3, it accurately identified a horse with high confidence, correctly drawing the bounding boxes. The model also recognized the horse's dominant coat color as dark slate gray (RGB: 54, 54, 52), demonstrating its ability to provide detailed information about detected objects. Similarly, the framework demonstrated strong cow image detection in Figure 4, with a high confidence score of 0.97. The results show that factors such as cow posture, lighting, and the extent to which the animal occupies the frame can significantly influence the model's confidence.

The model could be improved to perform more consistently in different variations of the same object class. Its color analysis system provides accurate color information, as shown by correctly identifying the 'Sienna' colors.

Table 1. Performance metrics across experiments

Exp.	Learning rate	Epochs	Mean precision	Mean recall	Mean F1 score	Mean IoU	mAP
1	0.005	30	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	0.6700	0.6122
4	0.010	20	0.2510	0.1231	0.1542	0.6901	0.2510
5	0.005	20	0.5588	0.3147	0.3737	0.6514	0.5588
6	0.001	30	0.7463	0.5802	0.6242	0.7040	0.7463
7	0.0001	20	0.1500	0.0755	0.0988	0.6693	0.1500
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	50	0.8119	0.5376	0.6232	0.7096	0.8119

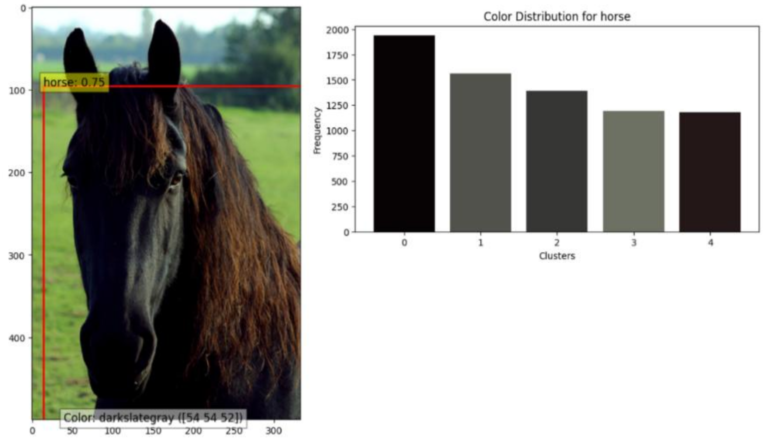

Figure 3. Object detection and color identification of horse

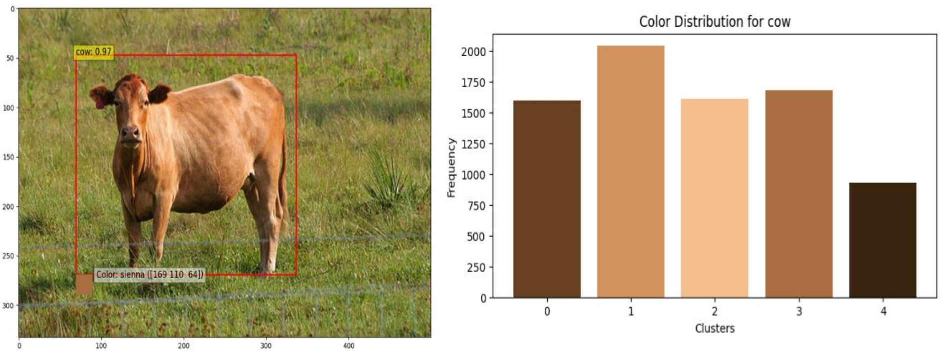

Figure 4. Object detection and color identification of cow

HOG is a widely used feature descriptor in computer vision and image processing for object detection. HOG visualizations provide information about how an image is characterized by its edge orientations and intensities. Figures 5 and 6 show HOG visualization of an aircraft and a bus, respectively. In Figure 5, strong gradient patterns along the wings, fuselage, and tail of the plane highlight its aerodynamic features and show that the framework focuses on sharp edges and contours. Similarly, the HOG visualization in Figure 6 highlights strong horizontal and vertical gradients around key bus features like the windshield and headlights. This indicates that the framework identifies buses by focusing on geometric shapes and edge patterns, effectively using structural and angular characteristics for vehicle recognition.

Grad-CAM visualizations are a powerful tool for interpreting our convolutional neural network, revealing which areas of an image are the most influential in the model predictions. Grad-CAM heatmap in Figures 7 and 8 shows that the model mainly highlights the cow's body and head. The framework focuses on parts of the animal rich in category-defining features and demonstrates its ability to accurately identify and locate cows in the image. This visualization shows that the model relies not only on the overall shape of the cow, but also on variations in surface patterns, colors, and textures when making its predictions.

Figure 5. HOG visualization of plane

Figure 6. HOG visualization of bus

Figure 7. Grad-CAM visualization of cow

Figure 8. Grad-CAM visualization of cow

5. CONCLUSION

Finally, this study also brings a contribution to the integrated examination of object detection and color interpretation because of using large-scale datasets, complex augmentation methods, and the most recent deep learning models, including Faster R-CNN with a ResNet-50 backbone and feature pyramid networks. The proposed system has a high quantitative performance with a mAP of 0.8119 on the Pascal VOC 2012 dataset and shows resilience in a variety of object categories and demanding visual settings due to the integration of k-means clustering to extract dominant colors and explainable AI solutions like HOG and Grad-CAM. The capability of the model to give accurate object localization along with semantically descriptive color features makes it a useful tool in real-world applications in such fields as industrial inspection, autonomous systems and medical imaging. Generally, the work presents a valuable and developed framework that stitches the disparity between theoretical developments and practical aspects of computer vision needs. This framework will be extended in future directions to facilitate real-time processing, adaptive behavior, domain generalization and explainability thus increasing its capability to be further used in practice and real-life situations.

FUNDING INFORMATION

This research did not receive any grants from any funding agencies.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Srinivas Dibbur Byrappa	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	
Kushal Gajendra		✓				✓		✓	✓	✓	✓	✓		
Rohith Holenarasipura Puttaraju	✓		✓	✓		✓			✓		✓	✓	✓	
Tumakalahalli Nagaraj Malini	✓		✓					✓		✓		✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are available on request from the corresponding author.

REFERENCES

[1] R. L. Galvez, A. A. Bandala, E. P. Dadios, R. R. P. Vicerra, and J. M. Z. Maningo, “Object detection using convolutional neural networks,” in *TENCON 2018 - 2018 IEEE Region 10 Conference*, Oct. 2018, pp. 2023–2027, doi: 10.1109/TENCON.2018.8650517.

[2] G. Chandan, A. Jain, H. Jain, and Mohana, “Real time object detection and tracking using deep learning and OpenCV,” in *2018 International Conference on Inventive Research in Computing Applications (ICIRCA)*, Jul. 2018, pp. 1305–1308, doi: 10.1109/ICIRCA.2018.8597266.

[3] A. Körez and N. Barışçı, “Object detection with low capacity GPU systems using improved Faster R-CNN,” *Applied Sciences*, vol. 10, no. 1, p. 83, Dec. 2019, doi: 10.3390/app10010083.

[4] H. Deshpande, A. Singh, and H. Herunde, “Comparative analysis on YOLO object detection with OpenCV,” *International journal of research in industrial engineering*, vol. 9, no. 1, pp. 46–64, 2020.

[5] H. Gong *et al.*, “Swin-transformer-enabled YOLOv5 with attention mechanism for small object detection on satellite images,” *Remote Sensing*, vol. 14, no. 12, p. 2861, Jun. 2022, doi: 10.3390/rs14122861.

[6] M. A. Shehab, A. Al-Gizi, and S. M. Swadi, "Efficient real-time object detection based on convolutional neural network," in *2021 International Conference on Applied and Theoretical Electricity (ICATE)*, May 2021, pp. 1–5, doi: 10.1109/ICATE49685.2021.9465015.

[7] H. Moujahid *et al.*, "Combining CNN and Grad-CAM for COVID-19 disease prediction and visual explanation," *Intelligent Automation & Soft Computing*, vol. 32, no. 2, pp. 723–745, 2022, doi: 10.32604/iasc.2022.022179.

[8] E. Suherman, B. Rahman, D. Hindarto, and H. Santoso, "Implementation of ResNet-50 on end-to-end object detection (DETR) on objects," *SinkrOn*, vol. 7, no. 2, pp. 1085–1096, Apr. 2023, doi: 10.33395/sinkron.v8i2.12378.

[9] C.-L. Lin, Y.-L. Chen, and Y.-C. Lin, "Small objects detection using transformer-CNN," *SSRN Electronic Journal*, 2022, doi: 10.2139/ssrn.4232850.

[10] H. Yanagisawa, T. Yamashita, and H. Watanabe, "A study on object detection method from manga images using CNN," in *2018 International Workshop on Advanced Image Technology (IWAIT)*, Jan. 2018, pp. 1–4, doi: 10.1109/IWAIT.2018.8369633.

[11] S. A. N and S. M. N, "Classification and object detection on satellite images using custom CNN architecture," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 12, no. 07, 2023, doi: 10.15680/ijirset.2023.1207190.

[12] E. Güney and C. Bayilmis, "An implementation of traffic signs and road objects detection using faster R-CNN," *Sakarya University Journal of Computer and Information Sciences*, vol. 5, no. 2, pp. 216–224, 2022.

[13] P. Dsouza, "Detection of small objects using CNN," *International Research Journal of Modernization in Engineering Technology and Science*, vol. 5, no. 5, p. 6818, May 2023, doi: 10.56726/IRJMETS40343.

[14] S. Vatchala, S. Sasidevi, Dhanalakshmi R, and SaiRamesh L, "Smart household object detection using CNN," *Advances in Parallel Computing Algorithms, Tools and Paradigms*, vol. 41, p. 464, 2022, doi: 10.3233/APC220065.

[15] F. M. T. R. Kinasih, C. Machbub, L. Yulianti, and A. S. Rohman, "Two-stage multiple object detection using CNN and correlative filter for accuracy improvement," *Heliyon*, vol. 9, no. 1, p. e12716, Jan. 2023, doi: 10.1016/j.heliyon.2022.e12716.

[16] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.

[17] K. Tong and Y. Wu, "Rethinking PASCAL-VOC and MS-COCO dataset for small object detection," *Journal of Visual Communication and Image Representation*, vol. 93, p. 103830, May 2023, doi: 10.1016/j.jvcir.2023.103830.

[18] J. Sui, L. Yang, Y. Wang, and Z. Hua, "An object color recognition algorithm based on study-expansion method," in *2009 9th International Conference on Electronic Measurement & Instruments*, Aug. 2009, pp. 4–205, doi: 10.1109/ICEMI.2009.5274106.

[19] R. P. Sidik and E. Contessa Djamal, "Face mask detection using convolutional neural network," in *2021 4th International Conference of Computer and Informatics Engineering (IC2IE)*, Sep. 2021, pp. 85–89, doi: 10.1109/IC2IE53219.2021.9649065.

[20] Yuhandri, Musli Yanto, and Eka Naufaldi Novri, "Application of object mask detection using the convolution neural network (CNN)," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 4, pp. 922–929, Aug. 2023, doi: 10.29207/resti.v7i4.5059.

[21] Y. Fang, S. Yang, S. Wang, Y. Ge, Y. Shan, and X. Wang, "Unleashing vanilla vision transformer with masked image modeling for object detection," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2023, pp. 6221–6230, doi: 10.1109/ICCV51070.2023.00574.

[22] M. Minderer *et al.*, "Simple open-vocabulary object detection," in *European conference on computer vision*, 2022, pp. 728–755.

[23] C.-H. Yeh *et al.*, "Lightweight deep neural network for joint learning of underwater object detection and color conversion," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 11, pp. 6129–6143, Nov. 2022, doi: 10.1109/TNNLS.2021.3072414.

[24] K. S. Aarathi and A. Abraham, "Vehicle color recognition using deep learning for hazy images," in *2017 International Conference on Inventive Communication and Computational Technologies (ICICCT)*, Mar. 2017, pp. 335–339, doi: 10.1109/ICICCT.2017.7975215.

[25] P. Raguraman, A. Meghana, Y. Navya, S. Karishma, and S. Iswarya, "Color detection of RGB images using Python and OpenCv," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, pp. 109–112, 2021, doi: 10.32628/CSEIT217119.

[26] S. C. Gupta, N. R. Singh, T. Sharma, A. Tyagi, and R. Majumdar, "Generating image captions using deep learning and natural language processing," in *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, Sep. 2021, pp. 1–4, doi: 10.1109/ICRITO51393.2021.9596486.

[27] K. Gajendra and S. D. Byrappa, "Recent advancements and challenges in object detection and color identification: a survey," *High Technology Letters*, vol. 30, no. 5, 2024. Accessed: May 17, 2025. [Online]. Available: https://drive.google.com/file/d/1N8Oi_xkyMmnnk_eWfw5e0-yYsBsva0R7/view

[28] J. van de Weijer, C. Schmid, J. Verbeek, and D. Larlus, "Learning color names for real-world applications," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1512–1523, Jul. 2009, doi: 10.1109/TIP.2009.2019809.

[29] J. van de Weijer, C. Schmid, and J. Verbeek, "Learning color names from real-world images," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2007, pp. 1–8, doi: 10.1109/CVPR.2007.383218.

BIOGRAPHIES OF AUTHORS

Srinivas Dibbur Byrappa received his Ph.D. degree in computer and information sciences from Visvesvaraya Technological University, India, in 2019. He is currently a professor in the Department of Information Science and Engineering at Nitte Meenakshi Institute of Technology, Bangalore, India. His research interests are in parallel computing, cloud computing, grid/cluster computing, and computer vision. He can be contacted at email: srinivas.db@nmit.ac.in.

Kushal Gajendra pursuing M.Tech. in data science at Nitte Meenakshi Institute of Technology, Bangalore, India and completed his Bachelor of Engineering in electronic and communication from Visvesvaraya Technological University in 2022. His research interest includes real-time object detection and explainable AI models. He can be contacted at email: kushalkowsh@gmail.com.

Rohith Holenarasipura Puttaraju completed my M.Tech. (CSE) from M.S. Ramaiah Institute of Technology, currently working as an assistant professor in the Department of Information Science and Engineering. He is pursuing my Ph.D. in natural language processing from VTU. His areas of interest are artificial intelligence, machine learning and computer networking. He can be contacted at email: rohit.hp@nmit.ac.in.

Tumakalahalli Nagaraj Malini is currently working as a professor at the Department of Management Studies, Nitte Meenakshi Institute of Technology, Bangalore, holding 19 years of teaching and research experience in organizational behavior, human resource management, supply chain and logistics management, image processing and computer vision. She can be contacted at email: malini.tn@nmit.ac.in.