RESEARCH PAPER

A Comprehensive Analysis of Approach for Object Detection and Recognition using CNN

Jitendra Sheetlani [1], Isha Dubey[2], Tanvee Nema [3], Harshita Vachhani [4], Dhirendra Kumar Tripathi [5], Abdul Razzak Khan Qureshi[6*]

[1] Professor, CSE Department, Chameli devi group of Institutions, Indore
[2] Asst. Prof., CSE, Medicaps University, Indore
[3] Asst. Prof., Institute of Engineering and Science IPS Academy Indore
[4] Associate Professor, CSE Dept. Pratibha college of commerce and computer studies, Pune
[5] Asst. Prof., Department of Computer Science, MGU, Sehore, Bhopal
[6]Assistant Professor, Department of Computer Science, Medicaps University, Indore, India.*
**Corresponding Author Email: dr.arqureshi786@gmail.com, https://orcid.org/0009-0002-1548-7504*

Abstract:
In computer vision, object detection and recognition processes are used to identify and locate objects in images. Humans can recognize a wide range of things in photos with little effort, even if the images vary somewhat from different angles, sizes, and scales, or when they are moved or rotated. Objects can be identified even when partially blocked from vision. This task remains a challenge for computer vision systems. Several approaches to the task have been implemented over the years. In many computer vision systems, the first duty is object detection, which allows us to learn more about the identified object and the scene. Researchers have recently focused their emphasis on the application of Machine learning techniques. Machine learning uses multiple layers to separate raw information into higher-level features. This paper outlines a recommended research methodology. This method uses a convolutional neural network-based architecture with a limited amount of data to detect objects in an image. The data used is the MAR database, and the process classifies thirty images into four distinct classes with high accuracy. The approaches outlined in this research included data collection and pre-processing, as well as designing and developing a hardware-based machine learning system. This study discusses the many techniques to object identification, their similarities, the limitations of existing work, and, finally, the model given in the last section of the paper.

How to cite this article: Jitendra Sheetlani 1 Isha Dubey2 Tanvee Nema 3 Harshita Vachhani 4 Dhirendra Kumar Tripathi 5 Abdul Razzak Khan Qureshi6*. A Comprehensive Analysis of Approach for Object Detection and Recognition using CNN. Int J Drug Deliv Technol. 2026;16(89s):833-837. DOI: 10.25258/ijddt.16.89s.88.

Introduction

Object detection and recognition are the main key point areas in computer vision fields, which is important for the branch of artificial intelligence. It seeks to identify and locate the exact position of items in given images or videos. Machine learning approaches have resulted in more powerful and robust algorithms for dealing with multi-scale, high-level characteristics, overcoming the constraints of classic object detector pipelines. The popularity of the transformer framework allows for greater capacity datasets by processing self-attention mechanisms, and object detection methods have advanced into a new age.

The process of identifying objects in films and photos is known as object recognition.This computer vision technology allows autonomous vehicles to identify and recognize things in real time.

2. Literature Review:

In this part of the paper, we discuss a review of object detection and recognition systems using machine learning literature. Previous research mainly focused on obtaining results from data mining, where data mining algorithms are used to detect objects.

Burger, W. et al. (2022) presented the most influential early methods as the scale-invariant feature transform (SIFT). SIFT was designed to detect and describe local features in images that were invariant to scale and rotation. By identifying key points in an image and generating descriptors that capture local image gradients around those key points, SIFT became a powerful tool for matching objects across different images. Its robustness to changes in viewpoint and illumination made it a popular choice for early object recognition tasks. However, despite its accuracy, SIFT was computationally intensive, which limited its application in real-time systems.

J. Joby et al. (2021), Medical imaging and healthcare by addressing the critical issue of pneumonia diagnosis. Pneumonia is a common and potentially life-threatening respiratory condition, and early detection is essential for early diagnosis and improved patient outcomes. Ayub et al. focused on using Generative Adversarial Networks (GAN), a class of deep learning models, to detect pneumonia from medical images such as chest X-rays. GANs are known for their ability to generate patterns in real data and are increasingly being used in various fields, including medical image analysis. The methodology used in the study involved the development and training of a GAN-based model specifically designed to diagnose pneumonia.

Afzal et al. (2021) In the realm of road safety, helmet usage stands as a paramount factor in mitigating the severity of head injuries sustained during traffic accidents involving motorcyclists.

Kumar et al. (2020) Computer vision and public safety by addressing the critical issue of helmet compliance among motorcyclists. Wearing a helmet is very important to reduce head injuries in traffic accidents, but the enforcement of helmet rules is a challenge for law enforcement.

3. CNN-based system for object detection and recognition

This work uses a quantitative, experimental approach to create and assess a convolutional neural network (CNN)-based system for object identification and recognition. The process is outlined in the following phases.

1. Problem Definition: The study's goal is to create an efficient model that can detect and distinguish many items in photos using convolutional neural networks (CNNs). The issue is handled using a mix of classification (recognition) and localization (detection).
2. Data Collection: The study uses benchmark datasets for object detection and recognition, such as:
   - COCO (Common Objects In Context)
   - PASCAL VOC
   - ImageNet

These datasets offer labeled images with bounding boxes and class annotations, which are required for supervised training.

3. Data preprocessing:
   - The images are scaled and normalized.
   - Data augmentation techniques like as rotation, scaling, flipping, and color modifications are used to boost dataset diversity.
   - Annotations are transformed to the appropriate format for the chosen CNN architecture (e.g., YOLO format or Pascal VOC XML).
4. Model Selection: research examines different CNN-based architectures for object detection:
   - Faster R-CNN: For accurate region-based detection.
   - YOLOv5: For real-time detection that strikes a decent balance between speed and accuracy.
   - SSD (Single Shot Multibox Detector): Used for lightweight detection in confined situations.
5. Model Training:
   - Deep learning frameworks such as TensorFlow or PyTorch are used for GPU-based training.
   - The loss functions are classification loss, localization loss, and objectness score.
   - Experimentation is used to tune hyperparameters such as learning rate, batch size, and the number of epochs.
6. Model Evaluation: models are assessed using standard metrics:
   - Mean Average Precision (mAP) measures detection accuracy.
   - For bounding box overlap, use intersection over union (IoU).
   - Precision, recall, and F1-score are used to assess recognition performance.
   - A separate test set is used to determine generalizability.
7. Comparative Analysis: Different CNN models are evaluated in terms of accuracy, computational complexity, and inference time. The tradeoffs between detecting speed and accuracy are examined.
8. Deployment and Testing: The best-performing model is implemented in real-time via a camera or video feed. The system is tested for resilience under a variety of illumination situations, occlusions, and object sizes.

4. Proposed method by Utilizes Efficient Multi-Scale Anchor Box

Suggested framework preserves the entire detail of the little items by extracting the image's multi-scale feature, while also ensuring the integrity of the large object's feature. The details are presented in the section below.

4.1 Architecture: In proposed model, there are 23 convolution layers and 5 max-pooling layers. It trains on 608 x 608 image size. We use mostly 3 × 3 and 1 x 1 filters and double the number of filters after every pooling step. Batch normalization is used to stabilize preparation, accelerate convergence and regularize the model.

Layer No.	Layer Type	Filters	Kernel Size	Output
1	Convolutional	32	3 x 3	608 x 608 x 32
2	Max	-	2 x 2	304 x 304 x 32
3	Convolutional	64	3 x 3	304 x 304 x 64
4	Max	-	2 x 2	152 x 152 x 64
5	Convolutional	128	3 x 3	152 x 152 x 128
6	Convolutional	64	1 x 1	152 x 152 x 64
7	Convolutional	128	3 x 3	152 x 152 x 128
8	Max	-	2 x 2	76 x 76 x 128
9	Convolutional	256	3 x 3	76 x 76 x 256
10	Convolutional	128	1 x 1	76 x 76 x 128
11	Convolutional	256	3 x 3	76 x 76 x 256
12	Convolutional	128	1 x 1	76 x 76 x 128
13	Convolutional	256	3 x 3	76 x 76 x 256
14	Max	-	2 x 2	38 x 38 x 256
15	Convolutional	512	3 x 3	38 x 38 x 512
16	Convolutional	256	1 x 1	38 x 38 x 256
17	Convolutional	512	3 x 3	38 x 38 x 512
18	Convolutional	256	1 x 1	38 x 38 x 256
19	Convolutional	512	3 x 3	38 x 38 x 512
20	Convolutional	256	1 x 1	38 x 38 x 256
21	Max	-	2 x 2	19 x 19 x 256
22	Convolutional	512	3 x 3	19 x 19 x 512
23	Convolutional	1024	1 x 1	19 x 19 x 1024
24	Convolutional	512	3 x 3	19 x 19 x 512
25	Convolutional	1024	1 x 1	19 x 19 x 1024
26	Convolutional	512	3 x 3	19 x 19 x 512
27	Convolutional	1024	3 x 3	19 x 19 x 1024
28	Convolutional	675	1 x 1	19 x 19 x 675

Table 1:Architecture Layer

Figure below depicts the architecture of the suggested model in a diagrammatic format. The input of 608X608X3 is considered and then transferred to the feature extraction layers. Layers 1 to 21 are the feature extraction layers. Layers 22–27 are in charge of object detection.

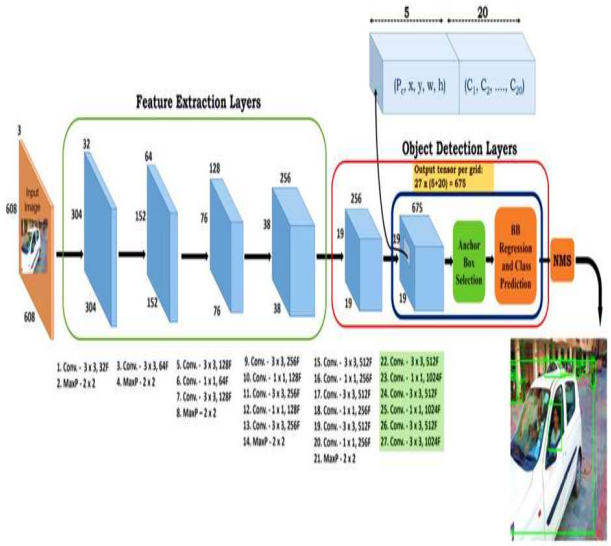


Fig 1 Architecture of Model

A total of 675 output tensors are formed by evaluating 27 feature maps generated each grid cell, together with five coordinates (4 coordinates and 1 confidence value) and 20 classes (based on the PASCAL VOC dataset). The anchor boxes are then selected in the manner specified in the following section. Finally, NMS is used to correctly detect and recognize the objects in the image.

4.2 Layout of Anchor Box and Scale: End-to-end detection methods rely heavily on anchor box quality. Anchor boxes can be generated at a given size and scale. However, the created anchor boxes with set sizes are usually suitable for the most common sizes of items, although the sizes of the objects can vary. After doing the study to detect the YOLOv2 box, it was discovered that several objects of unique size were difficult to detect and distinguish. The object's exceptional size, such as being larger than the vehicle. The automobile was quickly noticed, but the motorcycle was more difficult to detect due to its size. So, to boost detection accuracy, build an anchor box that matches the majority of the ground truth sizes. As a result, in our work, we use a multi-scale anchor box. The grid size used to detect small objects is 19 X 19, as seen in Figure 4.1.

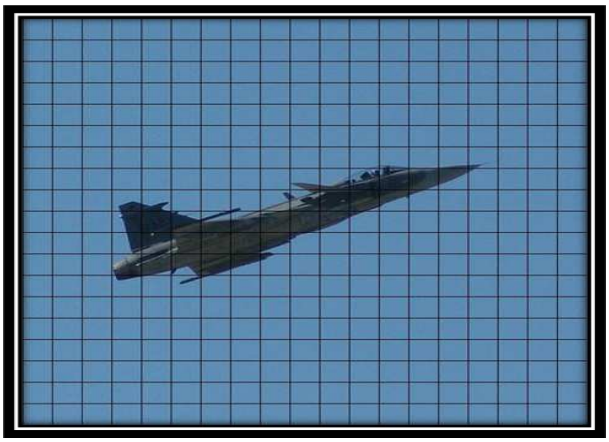


Fig: 2 Grid 19X19

Fig. 2 depicts the application of nine anchor boxes with three distinct scales (1:1, 1:2, 2:1) in each grid cell.

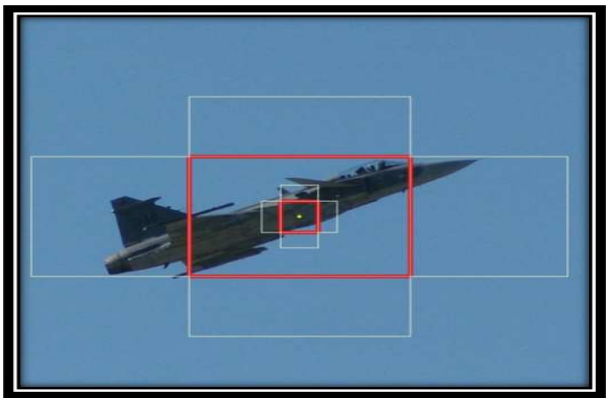


Fig 3: No. of scales of 3 ratios

Above Fig. 3 depicts the generation of 27 feature maps for prediction purposes. The bounding box coordinates of all annotated objects in the training set are clustered using k-means to yield a total of 9 anchor boxes. This anchor box generating method is inherited from YOLOv 2. The anchor box sizes produced after carrying out the preceding technique are: [(40, 47), (58, 108), (132, 90), (115, 308), (235, 180), (308, 392), (538, 248), (508, 444), (585, 556)]. For the entire image, 19 x 19 x 27 = 9747 feature maps will be created for prediction, which will predict category scores for each anchor box, (x, y) coordinates w (width) and h (height) of the anchor box, and objectness or confidence of an object included within that specific cell.

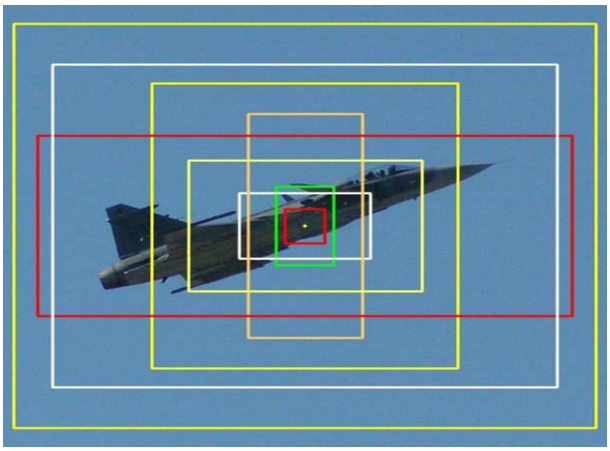


Fig. 4: object with 9 anchor boxes

Whether the object is small or large, our approach can recognize it accurately. Furthermore, regardless of the difference in illumination intensity, our system detects and recognizes small objects, indicating the robustness of our suggested method.

4.3 Innovative Multi-Scale Anchor Box Approach: As noted above, there are 9747 created feature maps. Therefore, there were 9747 predictions carried out. Furthermore, if the majority of the pixels' values are close together, not all anchor boxes receive enough information about the object. The prediction score is calculated for each anchor box, which increases the execution (training and testing) time in object detection. Real-time object detection is time-consuming and decreases FPS rates. To address this limitation, the optimized multi-scale anchor box approach was presented to reduce execution time for both real-time object detection and detecting small objects in a video series.

The algorithm operates in two parts:

(1) An optimized multi-scale anchor box determines whether information is present. This is completed in the following manner:

(a) Applying the canny edge detection algorithm to only the anchor box section. The canny edge detection algorithm needs a minimum and maximum threshold value to decide if the resulting edge is weak or strong. To automate this operation, the otsu binary threshold is utilized. This binary threshold provides a minimum and maximum value threshold.

(b) The above step generates an output image that calculates the frequency of black and white pixels. If the frequency of white pixels is less than 30, it implies that there is no information; otherwise, information is there. The threshold value 30 is a hyperparameter that is determined through trial and error. The value is being examined for the Pascal VOC-2007 dataset.

(2) To perform small item recognition in an optimal manner, the anchor boxes are placed in descending order. If the information is not available in the big scale anchor box, there is no need to switch to the small scale anchor box. This saves execution time for guessing the prediction score of the information included in the specified anchor box.

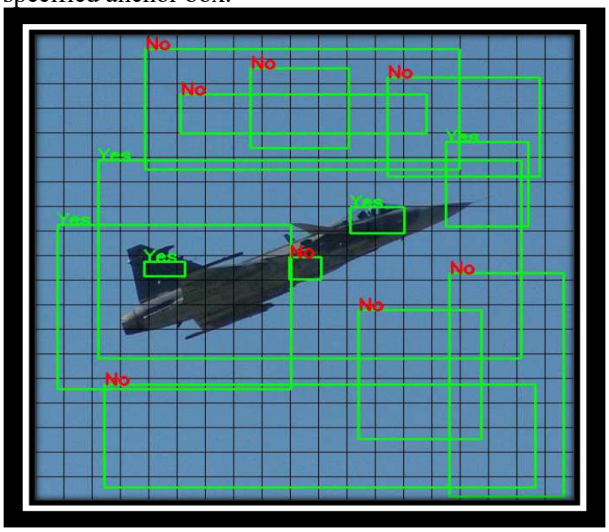


Figure 5: Innovative Multi-scale Anchor Box

Conclusion

This article describes an approach to object detection and recognition that employs a CNN-based architecture supplemented with a multi-scale anchor box mechanism. The suggested method efficiently addresses major issues such as recognizing small objects, managing lighting fluctuations, and maintaining accuracy across a wide range of object sizes. The method or approach decreases computational complexity while increasing real-time detection efficiency by leveraging optimized anchor boxes, k-means clustering, and a unique information-checking strategy. Experimental results show that the model preserves detailed information, improves prediction accuracy, and offers a more reliable detection framework than existing methods. Overall, the proposed method leads to the creation of faster, more accurate, and robust object identification systems suited for practical use.

Reference:

[1]. Afzal, H.U. Draz, M.Z. Khan, M.U.G. Khan. "Automatic helmet violation detection of motorcyclists from surveillance videos using deep learning approaches of computer vision." 2021 International Conference on Artificial Intelligence (ICAI), IEEE, 2021, pp. 252–257.

[2]. Burger, W.; Burge, M.J. Scale-invariant feature transform (SIFT). In Digital Image Processing: An Algorithmic Introduction; Springer: Cham, Switzerland, 2022; pp. 709–763.

[3]. J. Joby, R.J. Raju, R. Job, S.M. Thomas, K.S. Arun. "Pneumogan: a gan based model for pneumonia detection." 2021 2nd International Conference on Advances in Computing, Communication, Embedded and Secure Systems (ACCESS), IEEE, 2021, pp. 102–106.

[4]. S. Kumar, N. Neware, A. Jain, D. Swain, P. Singh. "Automatic helmet detection in real-time and surveillance video." Proceedings of ICM-LIP, 2020, pp. 51–60.

[5]. K.S. Arun, V.K. Govindan, S.M. Kumar. "On integrating re-ranking and rank list fusion techniques for image retrieval." International Journal of Data Science and Analytics, vol. 4, no. 1, 2017, pp. 53–81.

[6]. R. Babu, A. Rathee, K. Kalita, M.S. Deo. "Helmet detection on two-wheeler riders using machine learning." ARSSS International Conference, vol. 41, 2018, pp. 603–623.

[7]. Bochkovskiy, C.Y. Wang, H.Y.M. Liao. "Yolov4: optimal speed and accuracy of object detection." 2020. arXiv preprint arXiv:2004.10934.

[8]. K. Dahiya, D. Singh, C.K. Mohan. "Automatic detection of bike-riders without helmet using surveillance videos in real-time." 2016 International Joint Conference on Neural Networks (IJCNN), IEEE, 2016, pp. 3046–3051.