© NOV 2025 | IRE Journals | Volume 9 Issue 5 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I5-1712260

Real-Time Object Detection with OpenCV and Python

HIMANSHU RAI[1], VANSH KHATANA[2], DR. ANUJ CHANDILA[3], PROF. (DR.) SANJAY PACHAURI[4]

[1, 2, 3, 4] *Department of Data Science (DDCS), GNIOT College, Greater Noida, India*

Abstract- This paper presents a real-time object detection system using OpenCV and Python. The proposed model aims to achieve fast, accurate, and reliable identification of objects from a live video stream. The system utilizes deep learning–based pretrained architectures such as MobileNet-SSD and YOLO-tiny, known for their efficiency in real-time detection tasks. The workflow includes frame acquisition, preprocessing, blob formation, and inference using OpenCV's DNN module. Results demonstrate that lightweight models can process video frames in real time on CPU hardware while maintaining good accuracy. This study highlights how computer vision and deep learning can support automation, surveillance, and human–computer interaction systems.

Keywords— Object Detection, OpenCV, Deep Learning, YOLO, MobileNet-SSD, Real-Time Detection.

I. INTRODUCTION

Real-time object detection is a significant application of computer vision that involves locating and identifying objects in images or video frames. With the advancement of deep learning, modern object detection models have shown remarkable improvements in speed and accuracy compared to traditional techniques like Haar cascades and HOG + SVM.

OpenCV, combined with Python, offers a highly flexible platform for implementing these models in real-time applications. Systems developed using these technologies can be applied in security surveillance, autonomous systems, robotics, traffic monitoring, and various intelligent automation tasks.

The objective of this study is to create a fast and efficient real-time object detection system using OpenCV and Python, evaluate its performance with MobileNet-SSD and YOLO-tiny models, and analyze their accuracy and frames-per-second (FPS) output on CPU hardware.

II. METHODOLOGY

The real-time object detection system implements deep learning models using OpenCV's DNN module. The process begins with capturing video frames from a webcam or external camera. Each frame is resized and converted into a blob format suitable for neural network input.

The models used in this study include:
MobileNet-SSD: A lightweight Single Shot Detector architecture designed for high-speed detection on CPUs.
YOLO-tiny: A smaller version of YOLO optimized for real-time detection with improved accuracy over traditional lightweight models.

Frames are passed through the selected network to generate predictions, including class labels and bounding box coordinates. OpenCV functions such as cv2.rectangle and cv2.putText are used to overlay detection results on the frames. FPS is calculated to evaluate system responsiveness.

The dataset used to train these pretrained models includes COCO and VOC datasets, containing common objects such as people, vehicles, animals, and household items. Performance is measured based on detection accuracy and frame processing speed.

III. RESULTS AND DISCUSSION

The system demonstrated strong real-time performance during testing. MobileNet-SSD processed video at 20–25 FPS on ordinary CPU hardware, ensuring smooth detection with low latency. YOLO-tiny achieved 12–18 FPS, offering improved accuracy but slightly reduced speed.

Both models successfully detected multiple objects such as persons, cars, cups, and bottles with satisfactory confidence levels. However, environmental factors such as low lighting, motion blur, and occlusion sometimes affected detection accuracy.

DOI: https://doi.org/10.64388/IREV9I5-1712260

Accuracy and FPS results indicate that MobileNet-SSD is well-suited for real-time applications requiring high-speed processing, whereas YOLO-tiny is preferred where accuracy is more important than speed. Full YOLO models provide very high accuracy but require GPU acceleration for real-time performance.

IV. CONCLUSION

This paper presents a real-time object detection system using OpenCV and Python. The system successfully implements deep learning–based pretrained models to detect objects from live video streams with high speed and accuracy. MobileNet-SSD delivers fast processing suitable for CPU-based systems, while YOLO-tiny provides a balanced trade-off between speed and accuracy.

This research supports automation in robotics, surveillance, and smart monitoring systems. Future work may include integrating advanced models like YOLOv8, optimizing inference using GPU or TensorRT, and deploying the system on embedded platforms such as Raspberry Pi or NVIDIA Jetson.

REFERENCES

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
[2] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed, "SSD: Single Shot MultiBox Detector," European Conference on Computer Vision (ECCV), 2016.
[3] A. Rosebrock, "OpenCV Deep Learning with Python," PyImageSearch, 2017.
[4] A. Bochkovskiy, C.-Y. Wang, and H.-Y.M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," arXiv preprint, arXiv:2004.10934, 2020.
[5] OpenCV Documentation, "Deep Neural Network (DNN) Module," Available at: https://docs.opencv.org