ISSN No:-2456-2165

Object Detection Classification and Tracking of Everyday Common Objects

Shiwansh Bhargav, Somil Singh
Dept. of CSE,
RVCE Karnataka, Bangalore

Abstract:- This project presents an advanced computer vision system for object detection, classification, and tracking utilizing the cutting-edge YOLOv4 algorithm. Recent advances in deep learning have led to significant improvements in the accuracy and speed of object detection models. The project focuses on training the YOLOv4 model on large-scale datasets with diverse object categories. By employing transfer learning techniques, the model will be fine-tuned to adapt to specific target objects of interest, achieving a high level of accuracy and generalization. The Object detection, classification and tracking model achieves high accuracy in detecting and tracking objects. The performance analysis of the system showcases promising results. The model achieves the accuracy of over 95% for most of the objects, dropping till 75% for few objects and rarely till 50%. The fluctuation results due to the model not being very robust to occlusions. Overall, the model significantly improves the accuracy of existing model by detecting the targets that are very close to the edges of the frame to by focusing on them before they exit the frame. The model counts the objects and get their position information when tacking. However, ongoing improvement efforts are necessary to address potential challenges, such as real time multi object tracking, object association and occlusion handling.

Keywords:- *Yolov4, detection, classification, tracking, OpenCV.*

I. INTRODUCTION

Recent advances in deep learning have led to significant improvements in the accuracy and speed of object detection models. These models are now able to detect a wider range of objects, including everyday common objects such as people, cars, animals, and food. In addition, object tracking algorithms have been developed that can track objects over time more accurately, even when they are partially occluded or moving quickly. One of the most significant recent developments in object detection is the YOLOv4 model. YOLOv4 has been shown to achieve state-of-the-art accuracy on a variety of object detection benchmarks, while also being significantly faster than previous versions of YOLO. It also supports over 80 object categories, making it a powerful tool for detecting everyday common objects.

Another important development in object detection is the DeepSORT algorithm. DeepSORT is an object tracking algorithm that can track objects over time more accurately than previous methods. DeepSORT is able to track objects even when they are partially occluded or moving quickly, making it a valuable tool for applications such as self-driving cars and video surveillance.

As research in this area continues, we can expect to see even more improvements in the accuracy, speed, and capabilities of object detection and tracking models. These models will have a wide range of applications, such as self-driving cars, video surveillance, and robotics.

II. LITERATURE SURVEY

This paper[1] proposes a method for object tracking and counting in a zone using YOLOv4, Deep SORT, and TensorFlow. The results of the experiments are promising and suggest that the proposed method is a viable option for object tracking and counting in a zone. The key points of the paper are that YOLOv4 is a fast and accurate object detection algorithm, Deep SORT is a tracking algorithm that can track objects over time, TensorFlow is a machine learning framework that can be used to train and deploy YOLOv4 and Deep SORT, and the proposed method was able to achieve high accuracy and speed in object tracking and counting.

In this paper[2] a new method proposed for multiple object tracking in surveillance videos uses a Spatio Temporal Markov Random Field (ST-MRF) model to track moving vectors (MVs) and blocks coding modes (BCMs) from a compressed bitstream. The results show that the proposed method outperforms other methods on the MOTChallenge benchmark. The paper also investigates the use of visual features in the tracking phase of a tracking system using Deep SORT. The results show that the use of visual features can improve the performance of the tracking system.

This paper[3] proposes a new object detection framework called YOLOv4-5D, which is based on the YOLOv4 architecture. The framework introduces several new techniques to improve the accuracy and efficiency of object detection for autonomous driving. These techniques include using a new backbone network, replacing the last output layer with deformable convolution, designing a new feature fusion module, and using a new network pruning algorithm. The results of the experiments show that YOLOv4-5D outperforms the YOLOv4 baseline on the BDD and KITTI datasets. The framework is also able to run in real time, making it suitable for use in autonomous driving applications.

This paper[4] proposes a real-time vehicle detection and tracking system based on the YOLOv4-tiny object detection model. The system uses a pre-trained YOLOv4-tiny model to detect vehicles in real time and the Deep SORT algorithm to track the detected vehicles. The system was evaluated on the

ISSN No:-2456-2165

KITTI dataset, where it achieved an accuracy of 94.17% and can run in real time at 25 frames per second.

The paper[5] proposes a vehicle detection and tracking system based on the YOLO object detection model and the Deep SORT tracking algorithm. The system first uses YOLO to detect vehicles in a video frame. The detected vehicles are then tracked using Deep SORT, which uses a Kalman filter and a Hungarian algorithm to associate detections across frames. The results of the experiments show that the proposed system can achieve real-time performance while maintaining high accuracy.

III. PROPOSED WORK

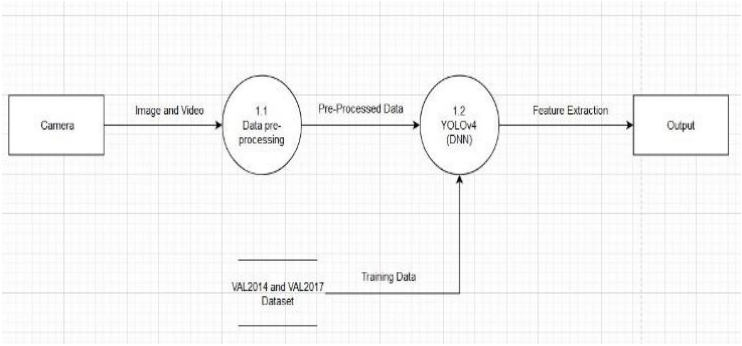

Fig 1: Architecture Diagram

Fig 1 shows the architecture diagram of the application. The details of the application are expanded in this section of the report. The user interacts with the project through the command line terminal. The input to the project is given via web cam or by-passing file location of video through terminal. The input video then preprocessed and send to the Yolov4 model (DNN). The model is trained and evaluated via val2014 and val2017 COCO dataset. Finally, the video after feature extraction is played on a prompt and the tracking details and the detected objects are displayed on the terminal.

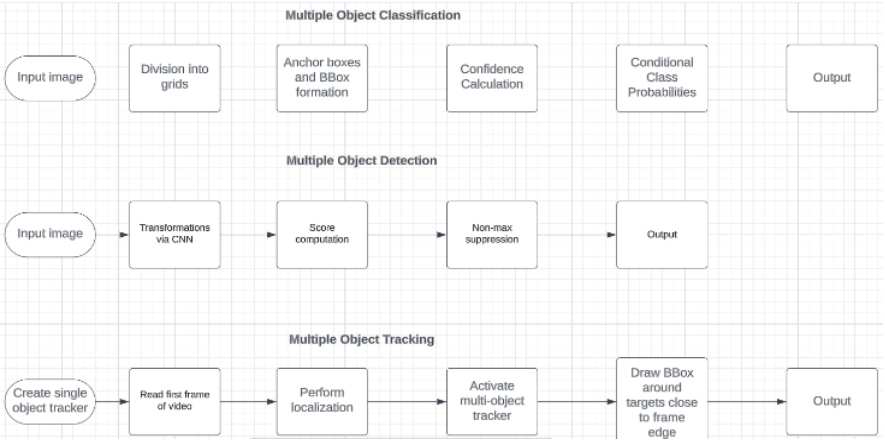

Fig 2: Structure Chart

A. Functional Description of the Modules

There are three major modules in this project which includes multiple object detection followed by multiple object classification and finally multiple object tracking. Fig 1: shows the structure chart for all 3 modules.

- Multiple Object Detection

An image is given as the input to algorithm and transformation is done using CNN. These transformations are done so that, input image is compatible to specifications of algorithm. Following this, flattening operation is performed. Flattening is converting data into a 1-dimensional array for inputting it to next layer. Most of approaches employ a sliding window over feature map and assigns foreground/background scores depending on features computed in that window. The neighborhood windows have similar scores to some extent and are considered as candidate regions. This leads to hundreds of proposals. This leads to a technique, which filters proposals based on some criteria called Non-Maximum Suppression calculation is actually used to measure the overlap between two proposals.

- Multiple Object Classification

Multi-object classification is a computer vision task aimed at simultaneously identifying and categorizing multiple objects within an image. anchor boxes are placed across the image during training and act as reference frames for the model to predict object locations. The model calculates the offsets between the anchor boxes and the actual boundary boxes enclosing the objects, which helps in precisely localizing each object. The boundary boxes represent the predicted bounding boxes for each object detected in the image. These boxes are generated by adjusting the anchor boxes based on the predicted offsets. For each object detected,

ISSN No:-2456-2165

the model computes a confidence score representing the probability that the predicted boundary box contains an object. Conditional class probabilities are probabilities associated with each detected object's class label. For every predicted boundary box with high confidence, the model calculates the conditional probabilities for various classes.

- Multiple Object Tracking

Here train the multi object tracker using yolov4 and deep learning methods and optimize the detector success rate. Consists of Detection phase, where an object detector is used to identify and localize objects in each frame of the video. It then has Data Association phase where the detected objects in consecutive frames are linked based on their appearance, motion, and temporal coherence to establish object tracks. State Estimation phase is where the motion and state of each object are estimated using filtering techniques like Kalman filters or particle filters. This helps to predict the object's position and update its trajectory over time. and final Track Management phase which handles track initialization, termination, and handling of occlusions or object appearance/disappearance.

IV. RESULTS AND DISCUSSION

The evaluation begins with assessing the model's performance on the MS COCO datasets, val2014 and val2017 each containing around 5000 images across 80 categories.

The results generated from the model were evaluated on the following metrics. Computing GIoU Loss: GIoU (Generalized Intersection over Union) is used to measure the similarity between predicted and ground truth bounding boxes. The function utils.bbox_giou() computes GIoU. Bounding Box Loss Scaling: The loss for the bounding box coordinates is scaled based on the size of the ground truth bounding boxes. Confidence Loss (Focal Loss): The confidence loss is computed using Focal Loss, which helps to focus on hard examples and mitigate the effect of easy negatives. Probability Loss: The probability loss is calculated using the standard sigmoid cross-entropy loss for multi-class classification.

Fig. 3: Tracking of detected objects

Fig 3 shows the tracking of the objects detected by our model. The model detects the objects, classify them across the 80 categories and shows the probability of match. The Fig 4 shows the tracking position of the detected object in different frames of the videos.

The performance analysis of the system showcases promising results. The Object detection, classification and tracking model achieves high accuracy in detecting and tracking objects. The model achieves the accuracy of over 95% for most of the objects, dropping till 75% for few objects and rarely till 50%. The fluctuation results due to the model not being very robust to occlusions.

```
Object found: person, Confidence: 0.94, BBox Coords (xmin, ymin, xmax, ymax): 110.0, 99.0, 227.0, 360.0
Object found: backpack, Confidence: 0.87, BBox Coords (xmin, ymin, xmax, ymax): 605.0, 145.0, 639.0, 196.0, Close To Bou
ndary
Object found: bicycle, Confidence: 0.73, BBox Coords (xmin, ymin, xmax, ymax): 374.0, 173.0, 407.0, 207.0
Object found: handbag, Confidence: 0.70, BBox Coords (xmin, ymin, xmax, ymax): 129.0, 177.0, 189.0, 332.0
Object found: chair, Confidence: 0.54, BBox Coords (xmin, ymin, xmax, ymax): 261.0, 205.0, 299.0, 284.0
FPS: 3.18
Object found: person, Confidence: 0.99, BBox Coords (xmin, ymin, xmax, ymax): 520.0, 106.0, 588.0, 287.0
Object found: person, Confidence: 0.98, BBox Coords (xmin, ymin, xmax, ymax): 114.0, 100.0, 238.0, 360.0
Object found: car, Confidence: 0.97, BBox Coords (xmin, ymin, xmax, ymax): 407.0, 135.0, 490.0, 214.0
Object found: person, Confidence: 0.96, BBox Coords (xmin, ymin, xmax, ymax): 496.0, 143.0, 521.0, 220.0
Object found: person, Confidence: 0.94, BBox Coords (xmin, ymin, xmax, ymax): 584.0, 117.0, 640.0, 286.0, Close To Bound
ary
Object found: backpack, Confidence: 0.83, BBox Coords (xmin, ymin, xmax, ymax): 604.0, 145.0, 639.0, 197.0, Close To Bou
ndary
Object found: bicycle, Confidence: 0.69, BBox Coords (xmin, ymin, xmax, ymax): 374.0, 173.0, 406.0, 207.0
FPS: 3.18
```

Fig. 4: Position of object being tracked

V. CONCLUSION AND FUTURE WORKS

Despite the promising features and capabilities of the project, there are some limitations that need to be acknowledged. These limitations are as follows.

- Accuracy: YOLOv4 is a fast and efficient object detection model, but it might not be as accurate as some slower, more complex models. The trade-off between speed and accuracy could impact the detection and tracking performance, especially in challenging scenarios or with small objects.
- Training Data: The quality and diversity of the training data can significantly impact the model's performance. If the training data is limited or biased, the model may struggle to generalize to unseen situations or objects.
- Computational Resources: YOLOv4 can be resource-intensive, particularly during training and inference, requiring powerful GPUs or specialized hardware. This might limit its usage on devices with limited computational capabilities.
- Complex Scenes: The model's performance might degrade in complex scenes with occlusions, clutter, or overlapping objects. These situations can challenge the tracker's ability to maintain accurate object associations over time.
- Variability in Object Appearance: Objects with large variations in appearance, such as different scales, rotations, or lighting conditions, might be challenging for YOLOv4 to detect and track consistently.

To address the limitations and further improve the project, several enhancements could be considered. The future works may include, fine-tuning the YOLOv4 model on domain-specific or more diverse datasets could improve the accuracy and robustness of object detection and tracking in specific scenarios. Applying various data augmentation techniques during training can help the model generalize better to different object appearances and environmental conditions.

Combining the outputs of multiple object detection models or tracking algorithms using ensemble methods could potentially improve overall performance and reliability. Implementing object re-identification techniques can enhance the tracker's ability to handle occlusions and re-establish associations when objects briefly leave the camera's view.

ACKNOWLEDGMENT

We would like to express our gratitude to our advisors and mentors for their guidance and support throughout the course of this research. We acknowledge the contributions of R.V College of Engineering for providing us with the necessary resources and facilities. We also extend our thanks to our guide Dr. Hemavathy R. who generously gave her time and effort to make this study possible. We appreciate the feedback and insights provided by our peers and colleagues, which greatly improved the quality of our work. Lastly, we thank the peer-reviewers for their invaluable suggestions and constructive criticism, which have helped to refine and strengthen our findings.

REFERENCES

[1.] "Multiple Object Tracking using STMRF and YOLOv4 Deep SORT in Surveillance Video", International Journal of Science & Engineering Development Research (www.ijrti.org), ISSN:2455-2631, Vol.7, Issue 6, page no.43 - 51, June-**2022**.

[2.] Diwan T, Anirudh G, Tembhurne JV. Object detection using YOLO: challenges, architectural successors, datasets and applications. Multimed Tools Appl. 2023;82(6):9243-9275. doi: 10.1007/s11042-022-13644-y. Epub **2022** Aug 8. PMID: 35968414; PMCID: PMC9358372.

[3.] Jiang, Yue & Li, Wenjing & Zhang, Jun & Li, Fang & Wu, Zhongcheng. (**2022**). YOLOv4-dense: A smaller and faster YOLOv4 for real-time edge-device based object detection in traffic scene. IET Image Processing. 17. n/a-n/a. 10.1049/ipr2.12656.

[4.] Li, Fudong & Gao, Dongyang & Yang, Yuequan & Zhu, Junwu. (**2022**). Small target deep convolution recognition algorithm based on improved YOLOv4. International Journal of Machine Learning and Cybernetics. 14. 1-8. 10.1007/s13042-021-01496-1.

[5.] Amrouche, Y. Bentrcia, A. Abed and N. Hezil, "Vehicle Detection and Tracking in Real-time using YOLOv4-tiny," **2022** 7th International Conference on Image and Signal Processing and their Applications (ISPA), Mostaganem, Algeria, 2022, pp. 1-5, doi: 10.1109/ISPA54004.2022.9786330.

[6.] Zhang, F. Kang, and Y. Wang, "An Improved Apple Object Detection Method Based on Lightweight YOLOv4 in Complex Backgrounds," *Remote Sensing*, vol. 14, no. 17, p. 4150, Aug. **2022**, doi: 10.3390/rs14174150.

[7.] L. Hou, C. Chen, S. Wang, Y. Wu, and X. Chen, "Multi-Object Detection Method in Construction Machinery Swarm Operations Based on the Improved YOLOv4 Model," *Sensors*, vol. 22, no. 19, p. 7294, Sep. **2022**, doi: 10.3390/s22197294.

[8.] U. P. Naik, V. Rajesh, R. K. R and Mohana, "Implementation of YOLOv4 Algorithm for Multiple Object Detection in Image and Video Dataset using Deep Learning and Artificial Intelligence for Urban Traffic Video Surveillance Application," **2021** Fourth International Conference on Electrical, Computer and Communication Technologies (ICECCT), Erode, India, 2021, pp. 1-6, doi: 10.1109/ICECCT52121.2021.9616625.

[9.] Dewi, R. -C. Chen, Y. -T. Liu, X. Jiang and K. D. Hartomo, "Yolo V4 for Advanced Traffic Sign Recognition with Synthetic Training Data Generated by Various GAN," in IEEE Access, vol. 9, pp. 97228-97242, **2021**, doi: 10.1109/ACCESS.2021.3094201.

[10.] Y. Cai et al., "YOLOv4-5D: An Effective and Efficient Object Detector for Autonomous Driving," in IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1-13, **2021**, Art no. 4503613, doi: 10.1109/TIM.2021.3065438.

[11.] M. A. Bin Zuraimi and F. H. Kamaru Zaman, "Vehicle Detection and Tracking using YOLO and Deep SORT," **2021** IEEE 11th IEEE Symposium on Computer Applications & Industrial Electronics

(ISCAIE), Penang, Malaysia, 2021, pp. 23-29, doi: 10.1109/ISCAIE51753.2021.9431784.

[12.] K. Shetty, I. Saha, R. M. Sanghvi, S. A. Save and Y. J. Patel, "A Review: Object Detection Models," **2021** 6th International Conference for Convergence in Technology (I2CT), Maharashtra, India, 2021, pp. 1-8, doi: 10.1109/I2CT51068.2021.9417895.

[13.] Zhu, G. Xu, J. Zhou, E. Di and M. Li, "Object Detection in Complex Road Scenarios: Improved YOLOv4-Tiny Algorithm," **2021** 2nd Information Communication Technologies Conference (ICTC), Nanjing, China, 2021, pp. 75-80, doi: 10.1109/ICTC51749.2021.9441643.

[14.] Guo, Dongwei & Cheng, Lufei & Zhang, Meng & Sun, Yingying. (**2021**). Garbage detection and classification based on improved YOLOV4. Journal of Physics: Conference Series. 2024. 012023.10.1088/1742-6596/2024/1/012023.

[15.] S. Kumar, Vishal, P. Sharma and N. Pal, "Object tracking and counting in a zone using YOLOv4, DeepSORT and TensorFlow," **2021** International Conference on Artificial Intelligence and Smart Systems (ICAIS), Coimbatore, India, 2021, pp. 1017-1022, doi: 10.1109/ICAIS50930.2021.9395971.

[16.] R. L. A, A. K. S, K. B. E, A.N. D and K. K. V, "A Survey on Object Detection Methods in Deep Learning," **2021** Second International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2021, pp. 1619-1626, doi: 10.1109/ICESC51422.2021.95 32809.

[17.] Bochkovskiy, Alexey & Wang, Chien-Yao & Liao, Hong-yuan. (**2020**). YOLOv4: Optimal Speed and Accuracy of Object Detection.

[18.] C. Kumar B., R. Punitha and Mohana, "YOLOv3 and YOLOv4: Multiple Object Detection for Surveillance Applications," **2020** Third International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2020, pp. 1316-1321, doi: 10.1109/ICSSIT48917.2020.92 14094.

[19.] Y. Li et al., "A Deep Learning-Based Hybrid Framework for Object Detection and Recognition in Autonomous Driving," in IEEE Access, vol. 8, pp. 194228-194239, **2020**, doi: 10.1109/ACCESS.2020.3033289.

[20.] Jiang, Zicong & Zhao, Liquan & Li, Shuaiyang & Jia, Yanfei. (**2020**). Real-time object detection method based on improved YOLOv4-tiny.