KLE Technological University

Creating Value
Leveraging Knowledge

School

of

Electronics and Communication Engineering

Mini Project Report

on

Semi automated annotation tool towards object detection and tracking in videos

By:

1. **Shyam Desai** 01FE21BEC110
2. **Shridhar Naragund** 01FE21BEC116
3. **Vinayak Nayak** 01FE21BEC305

Semester: V, 2023-2024

Under the Guidance of

Uma Mudengudi
Ramesh Ashok Tabib

K.L.E SOCIETY'S
KLE Technological University,
HUBBALLI-580031
2023-2024

SCHOOL OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that project entitled **"Semi automated annotation tool towards object detection and tracking in videos"** is a bonafide work carried out by the student team of **"Shyam Desai (01FE21BEC110), Shridhar Naragund (01FE21BEC116), Vinayak Nayak (01FE21BEC305)"**. The project report has been approved as it satisfies the requirements with respect to the mini project work prescribed by the university curriculum for BE (V Semester) in School of Electronics and Communication Engineering of KLE Technological University for the academic year 2023-2024

Uma Mudengudi		
Ramesh Ashok Tabib	**Suneetha Budihal**	**B.S. Anami**
Guide	**Head of School**	**Registrar**

External Viva:

Name of Examiners with date **Signature**

1.

2.

ACKNOWLEDGMENT

We would like to express our sincere gratitude to all the people who have assisted us in the completion of this project. All their contributions are deeply appreciated and acknowledged.We would like to place on record our deep sense of gratitude to Suneetha Budihal, Professor and Head of the Department of School of Electronics and Communication for having the opportunity to extend our skills in the direction of this project. We express our heartfelt gratitude to our guide Uma Mudenagudi,Ramesh Ashok Tabib and our Seniors whose valuable insights proved to be vital in contributing to the success of this project.

By:

Project Team

ABSTRACT

We propose Semi automated annotation tool towards object detection and tracking in videos. In Computer vision and Image processing accurate object detection and tracking in images and videos is important. Our problem statement includes three parts that are object detection,object tracking and annotation tool.In object detection there are many methods like R-CNN, Fast R-CNN,Yolo out of that we used YOLO V8 .For object tracking we used deep appearance descriptor(Deep sort).It will track the object throught the frame.Since the demand for annotated images is exponentially increasing.If model doesn't annotate some objects at that time human intervention is necessary.For that we build one GUI to annotate the objects correctly.With the help of tool we can detect more objects.Users can easily correct, verify, and enhance annotations, contributing to the overall precision of the system.The combination of YOLO v8 and Deep Sort enhances the efficiency of both detection and tracking tasks.

Contents

List of Figures

Chapter 1

Introduction towards Semi automated annotation tool

Object detection involves identifying and locating objects within an image.They generate bounding boxes around the objects and may also classify them into specific categories.Tracking algorithms help maintain the identity of objects as they move, providing information about their trajectories and motion patterns.Human intervention is particularly valuable in complex scenes, where automated methods may struggle due to occlusions, lighting changes, or ambiguous situations.

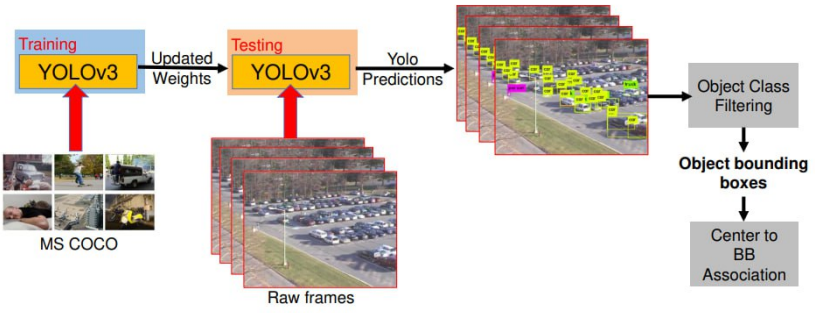

Figure 1.1: Object Detection

1.1 Motivation

- **Challenging Real-World Scenarios:** In many real-world situations, object detection andtracking can be exceptionally challenging due to factors such as occlusions, complex backgrounds, lighting variations, and diverse object types
- **Enhanced Accuracy and Reliability:** The primary motivation is to improve the accuracy and reliability of object detection and track-

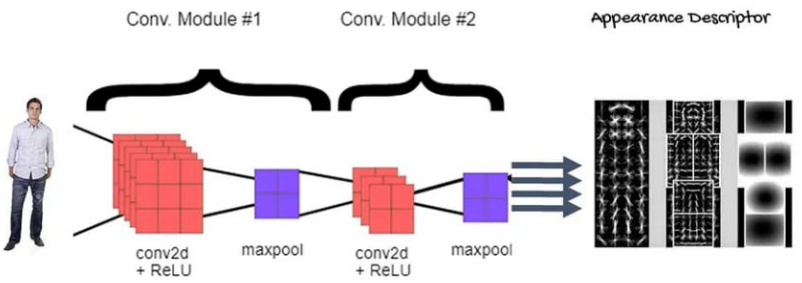

Figure 1.2: Deep appearance descriptor(Object Tracking)

ing.

1.2 Objectives

- Employ detection and tracking algorithms to detect and track objects (bounding boxes will act like annotations) within video frames.
- Develop annotation tool towards object detection and tracking
- Integrate the detection and tracking algorithm to annotation tool.
- Develop logic to hand-correct annotations in case of errors in annotations.

1.3 Literature survey

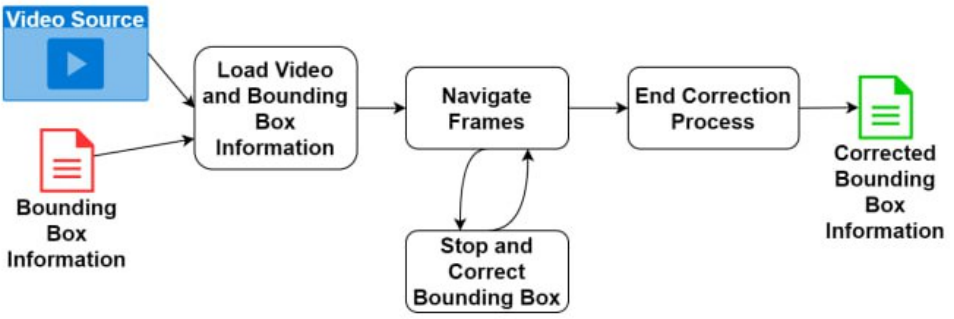

Figure 1.3: Annotation Tool,

- Existing literature underscores OpenCV's role as a foundational framework for understanding object detection, providing a basis for subsequent developments.
- Literature highlights the revolutionary impact of YOLO in real-time object detection,especially its grid-based approach for simultaneous predictions, which significantly improves speed and efficiency.
- Various studies delve into the advancements in object tracking, with focus on methods like Optical Flow and Mean Shift for improving detection continuity, and more sophisticated techniques like SORT and DeepSORT for identity-aware tracking
- Literature emphasizes the synergy between traditional computer vision methods (OpenCV, Optical Flow, Mean Shift) and deep learning techniques (YOLO, DeepSORT) to achieve a balance between real-time efficiency and tracking accuracy.
- The surveyed literature underscores the need for a comprehensive approach that addresses diverse scenarios, with applications in different environments and varying object types.

1.4 Problem statement

Semi automated annotation tool towards object detection and tracking in videos

1.5 Application in Societal Context

- **Traffic Management**:Improving traffic flow, monitoring road safety.
- **Environmental Monitoring**:Monitoring and tracking animals, changes in vegetation, and environmental conditions. itemize

Chapter 2

System design

In this chapter, we will be looking towards the functional block diagram, the design alternatives and also about the final design which is being implemented.

2.1 Design alternatives

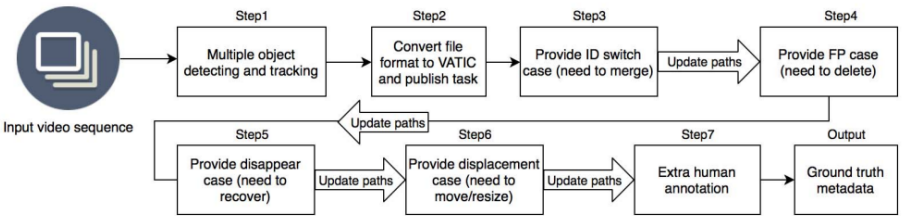

Figure 2.1: Alternate Design

2.2 Final design

Whatever the objects that we detected we have to generate bounding boxes for them.Deep appearance descriptor is CNN which is trained to detect a similar objects in different images.As an input the deep descriptor receives a cropped image of the object detected and as an output we tried to receive a vector that encodes the information that is present in that cropped image.These encoded vectors would allow us to compare different objects.The score by deep descriptor is given using cosine disctance matric.If the model doesn't detect the object

correctly with the help of annotation tool human can annotate the objects correctly .

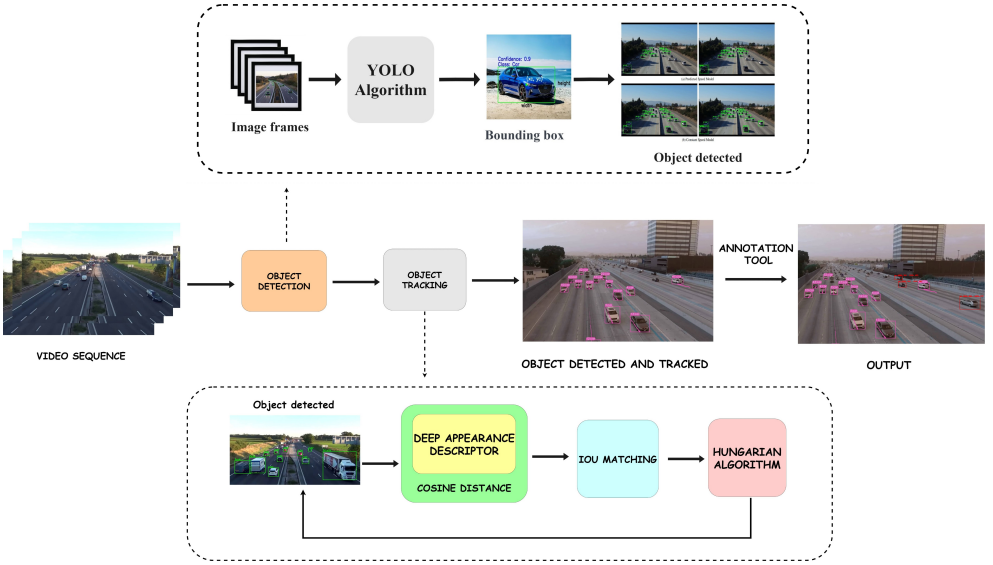

Figure 2.2: Block Diagram of Final Design

Chapter 3

Implementation details

3.1 Specifications and final system architecture

3.1.1 Object Detection

- The YOLO algorithm takes an image as input and then uses a simple deep convolution neural network to detect object in that image.
- For multiple object detection in single frame YOLO algorithm devides an input image into SxS grid cell,if the centre of the image falls into a grid cell, that grid cell is responsible for detecting that object in image.
- Each grid cell predicts B bounding boxes and confidence scores for those boxes. YOLO predicts multiple bounding boxes per grid cell ,we only want one bounding box so YOLO assigns one predictor to be responsible for predicting an object based on which prediction has the highest current IOU with the ground truth.

3.1.2 Deep appearance descriptor

Deep appearance descriptor is CNN which is trained to detect a similar objects in different images.As an input the deep descriptor receives a cropped image of the object detected and as an output we tried to receive a vector that encodes the information that is present in that cropped image.These encoded vectors would allow us to compare different objects.The score by deep descriptor is given using cosine distance matrix

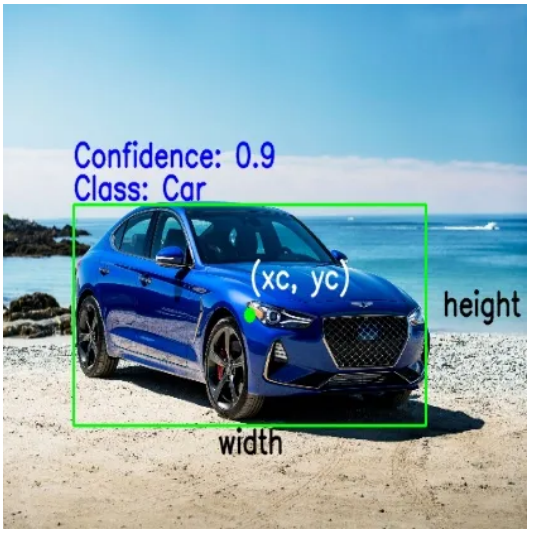

Figure 3.1: Bounding box and Object class of Image

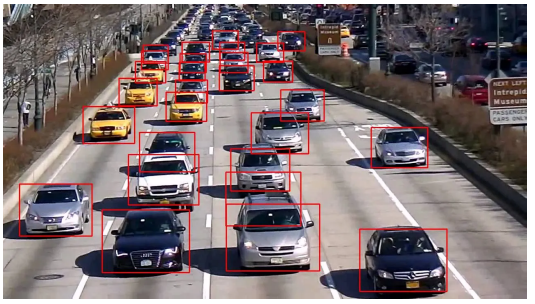

Figure 3.2: Detection of Multiple Objects

3.1.3 Cosine distance matrix

If we are given two entity A and B, from the origin we draw a vector joining these two vectors.The cosine value of the angle is going to give us cosine distance matrix.
If we have two vectors taht are overlaping each other the angle between them is 0.The cosine value of 0 is 1.These two vectors are similar to each other.
If two vectors are perpendicular to each other the cosine value of 90 is 0. These two vectors are dissimilar to each other

3.1.4 IOU matching

It gives quantitative score to determine how much two bounding boxes are similar to each other based on their location as well as size.ID's are useful to determine which object we are tracking.

3.1.5 Hungarian algorithm

Solves this linear assignment problem.Once we have assigned these N predictions to N ID's. We solved tracking problem for the particular frame.Then we can run this again in a loop and again keep track of the objects that are in the next frame of video.

3.1.6 Annotation tool

If model doesn't detect objects correctly, With the help of annotation tool we can annotate the objects

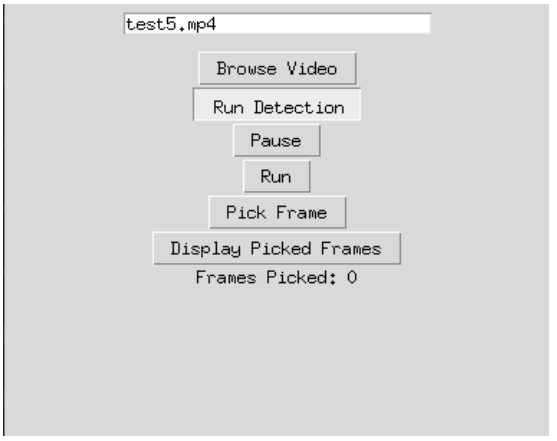

Figure 3.3: Annotation Tool

3.2 Algorithm to semi-automated annotation tool

Object Annotation Tool
Require : Video Frames
Ensure : Annotated Video Frames
Procedure : Object Annotation
1: Load Video File ← load_video()
2: Preprocess Frames ← preprocess_frames(Video Frames)
3: Select Objects ← select_objects(Preprocessed Frames)
4: Annotation Mode ← interactive
5: **for** each frame in Video Frames **do**:
6: **if** first frame **then**:
7: Annotate Objects ← annotate_objects(Select Objects, Annotation Mode)
8: **else**:
9: Annotate Objects ← annotate_objects(frame, Annotation Mode)
10: **end if**
11: **end for**
12: Save Annotations ← save_annotations(Annotated Frames)
13: Initialize Annotation Editor ← initialize_editor(Video Frames,

Annotations)
14: Display Annotation Editor ← display_editor(Annotation Editor)
End Procedure.

3.3 Flowchart

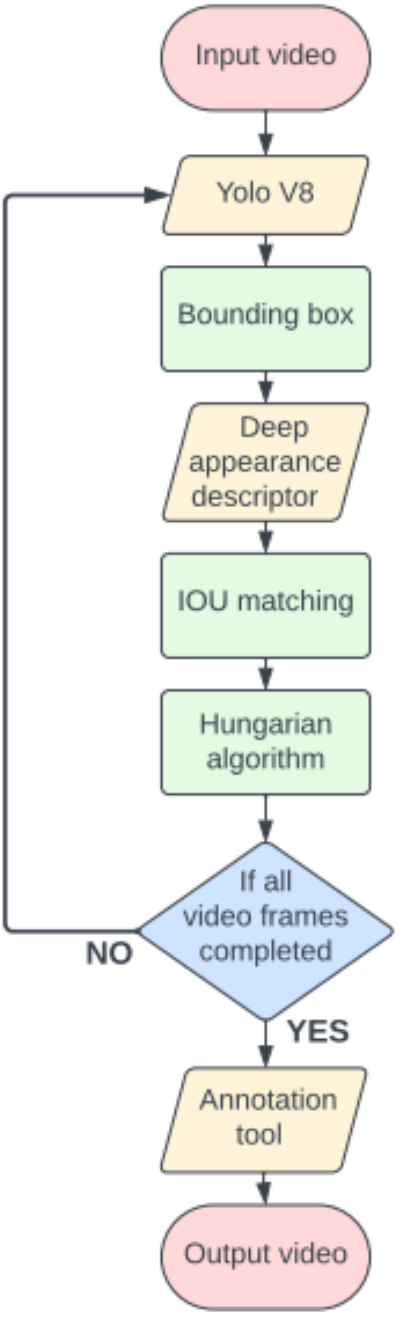

Figure 3.4: Flowchart

Chapter 4

Results and discussions

4.1 Datasets

The COCO dataset is a large-scale dataset designed for object detection, segmentation, and captioning tasks. It contains images from a wide range of categories, and it is widely used in the computer vision community for benchmarking and training purposes. The dataset is labeled with bounding boxes around object instances, and each instance is associated with a specific category. The COCO dataset has 80 object categories, including common objects such as person, car, dog, cat, and more.

The COCO dataset is well-suited for training object detection models like YOLO because it provides a diverse set of images with multiple objects in various contexts. This diversity helps the model generalize well to different scenarios and object appearances.

4.2 Evaluation metrics

Intersection Over Union (IoU)

IoU is used to measure the overlap between predicted and ground-truth bounding boxes.

$$IoU = \frac{AreaofOverlap}{AreaofUnion} \tag{4.1}$$

It's used in both training (to calculate localization loss) and NMS.

4.3 Experimental Setup

Training Configuration

- The model is trained for 100 epochs using Stochastic Gradient Descent (SGD) as the optimizer.
- Specific settings include a batch size of 16, image size of 640x640 pixels, and a OneCycleLR learning rate schedule.

Early Stopping

- Patience for early stopping is set to 50 epochs, allowing the training process to halt if no improvement is observed during this period.

Augmentation Techniques

- Various image augmentations, including HSV adjustments, rotation, and translation, are applied during training to enhance the model's robustness.

Object Detection Model

- YOLOv8 is employed for real-time object detection in video frames, providing efficient and accurate results.

Annotation Process

- The GUI allows users to manually annotate frames with bounding boxes through the `AnnotationEditor` class, providing a user-friendly interface for refining YOLOv8 detections.

Video Processing

- OpenCV is utilized for video processing, frame extraction, and display, ensuring efficient handling of video files.

Threading for Concurrency

– Threading is implemented to maintain concurrent execution of video playback and GUI updates, enhancing responsiveness by processing video frames in the background.

Annotations Storage

– Annotations, represented as bounding boxes, are stored in JSON format, facilitating easy retrieval and sharing of annotated data.

Export Options

– The model can be exported in TorchScript format, and there's consideration for mobile deployment with the ability to optimize the model for such use cases.

Integration and Visualization

– The GUI integrates YOLOv8 results by displaying both the original video and the video with detection results, providing users with a comprehensive view of the object detection performance. Threading ensures a responsive design, enabling smooth interaction even during video processing.

4.4 Experimental Results

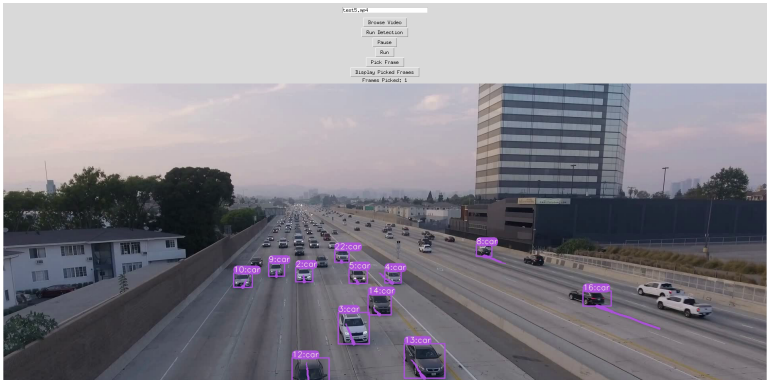

Figure 4.1: Annotation Tool Main Window

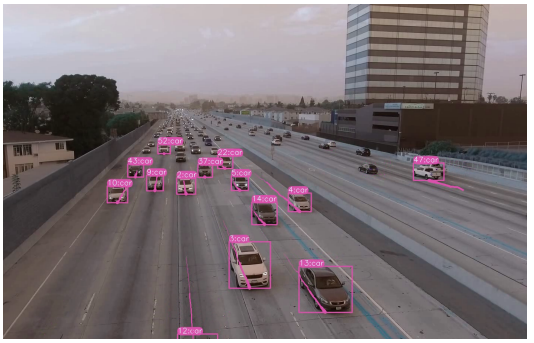

Figure 4.2: Before Annotation

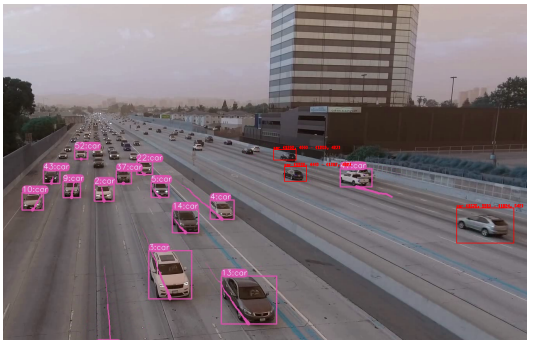

Figure 4.3: After Annotation

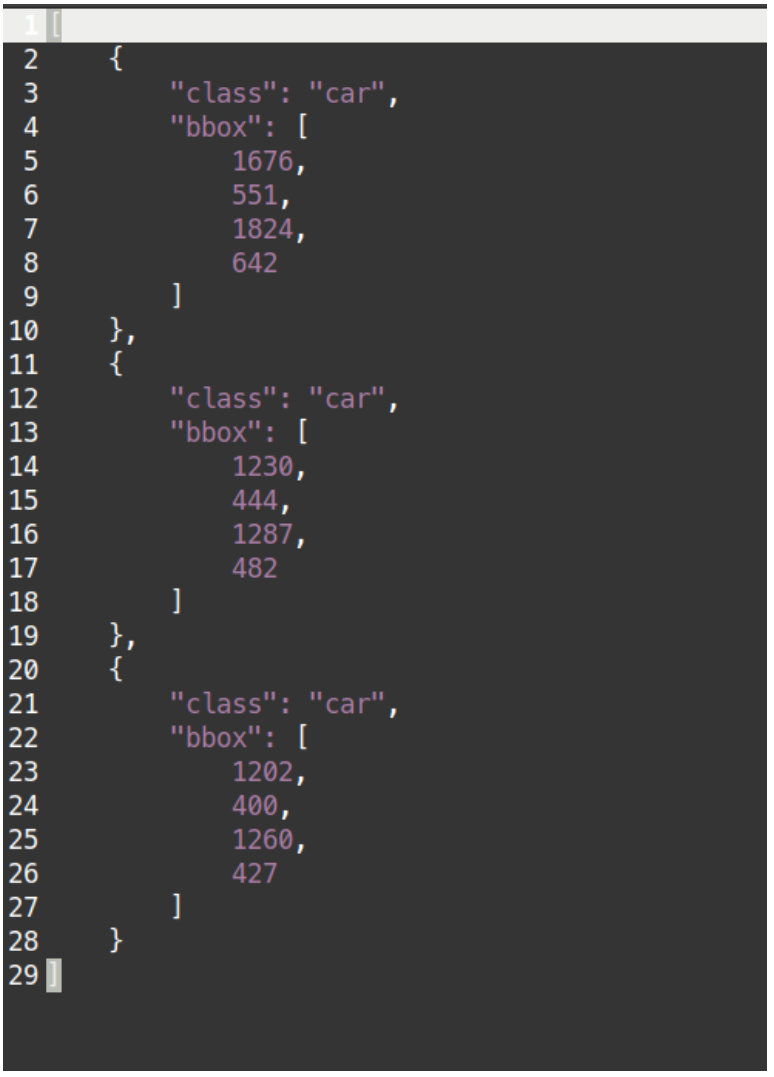

Figure 4.4: Co-ordinates and Class of newly ceated Bounding Box

Chapter 5

Conclusions and future scope

5.1 Conclusions

One of the most rapidly developing disciplines of technology, AI and machine learning are bringing about incredible advancements that benefit numerous industries worldwide.The YOLO algorithm efficiently detects objects by dividing images into grid cells and predicting bounding boxes. The deep appearance descriptor encodes object information for similarity comparison, using cosine distance matrices. IOU matching and the Hungarian algorithm enable effective object tracking. An annotation tool assists in refining model accuracy

5.2 Future scope

In the market data annotation is anticipated to expand significantly. This increase in demand has been influenced by the use of AI-based services across many industries. Businesses have started a variety of initiatives aimed at creating content assets and improving user experience.

Additionally, it has produced worthwhile growth chances. With such a huge demand of data, there will be shortage of annotators, hence a semi automatic annotator which will increase annotation many folds need to be implemented. There will be lot of time saved which can be utilized for better data generation and tuning models for better efficiency.

Bibliography

[1] Al-Shakarji, Noor M., et al. "Semi-automatic system for rapid annotation of moving objects in surveillance videos using deep detection and multi-object tracking techniques." 2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR). IEEE, 2020.

[2] Gu, Chuang, and Ming-Chieh Lee. "Semiautomatic segmentation and tracking of semantic video objects." IEEE Transactions on Circuits and Systems for Video Technology 8.5 (1998): 572-584.

[3] Al-Shakarji, Noor M., et al. "Semi-automatic system for rapid annotation of moving objects in surveillance videos using deep detection and multi-object tracking techniques." 2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR). IEEE, 2020.

[4] Bewley, Alex, et al. "Simple online and realtime tracking." 2016 IEEE international conference on image processing (ICIP). IEEE, 2016.

[5] Du, Yunhao, et al. "Strongsort: Make deepsort great again." IEEE Transactions on Multimedia (2023).

report

ORIGINALITY REPORT

11%	10%	3%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.slideshare.net Internet Source	4%
2	www.coursehero.com Internet Source	1%
3	github.com Internet Source	1%
4	S. Bhaggiaraj, M. Priyadharsini, K. Karuppasamy, R Snegha. "Deep Learning Based Self Driving Cars Using Computer Vision", 2023 International Conference on Networking and Communications (ICNWC), 2023 Publication	1%
5	Pragati Sathia, Swarnalatha P, Shravan Prakash. "Sign Language Interpreter via Gesture Detection", 2023 Third International Conference on Smart Technologies, Communication and Robotics (STCR), 2023 Publication	1%
6	catalog.uttyler.edu Internet Source	1%

7	www.gnedenko.net Internet Source	1 %
8	unsworks.unsw.edu.au Internet Source	1 %
9	www.collectionscanada.ca Internet Source	<1 %
10	docs.lib.purdue.edu Internet Source	<1 %
11	gtusitecirculars.s3.amazonaws.com Internet Source	<1 %
12	Tobias Fleck, Svetlana Pavlitska, Sven Nitzsche, Brian Pachideh et al. "Low-Power Traffic Surveillance using Multiple RGB and Event Cameras: A Survey", 2023 IEEE International Smart Cities Conference (ISC2), 2023 Publication	<1 %

Exclude quotes	On	Exclude matches	< 5 words
Exclude bibliography	On		