A Minor Project Synopsis on

**Object Detection and tracking in video image**

Submitted to Manipal University Jaipur

Towards the partial fulfillment for the Award of the Degree of

**BACHELORS OF TECHNOLOGY**

In Computers Science and Engineering

2021-2025

By

Sumit Kumar and Vaibhav Kr. Goel

219301597, 219301152



Under the guidance of

Dr. Sandeep Chaurasia

**Department of Computer Science and Engineering**

**School of Computer Science and Engineering**

**Manipal University Jaipur**

**Jaipur, Rajasthan**

**Introduction**Our everyday lives are dotted with video cameras, which generate a vast amount of visual data that has the potential to be used for several different purposes. Object detection and tracking within video footage are crucial advancements in computer vision, enabling tasks like:

1. Surveillance and security: Detecting and tracking suspicious activity or individuals.
2. Traffic monitoring and analysis: Counting vehicles, identifying traffic violations, and optimizing traffic flow.
3. Sports analytics: Tracking player movements and performance metric
4. Human-computer interaction: Creating more intuitive and interactive applications with real-time object recognition.
5. Autonomous robotics: Enabling robots to navigate environments and interact with objects safely.

While various object detection and tracking methods exist, You Only Look Once (YOLO) has emerged as a popular choice due to its:

1. Real-time performance: Offers fast inference speeds, making it suitable for real-world applications.
2. Single-stage architecture: Simplifies implementation and training compared to two-stage approaches.
3. Accuracy and flexibility: Provides good accuracy while being adaptable to different tasks and datasets.

This minor project aims to explore the capabilities of YOLO for object detection and tracking in video images, paving the way for practical applications and further research in this domain.

**Motivation**

The motivation for this project stems from the following key aspects:

1. Addressing Limitations of Existing Methods:

* Traditional object detection and tracking techniques often encounter challenges such as slow processing times, accuracy issues, and complexity.
* YOLO (You Only Look Once) offers a promising solution to overcome these limitations by providing faster processing, improved accuracy, and simplified implementation.

1. Exploring YOLO's Potential in Practical Applications:

* By leveraging the advantages of YOLO, we can develop real-world applications across various domains, including surveillance, traffic analysis, safety systems, and human-computer interaction.
* This project serves as a platform to explore and harness the practical implications of YOLO in diverse settings, paving the way for innovative solutions and advancements in these areas.

1. Gaining Hands-On Experience with a State-of-the-Art Model:

* Engaging with YOLO offers invaluable hands-on experience in implementing and utilizing a cutting-edge object detection and tracking model.
* Through practical experimentation and application, participants can enhance their understanding of computer vision techniques and their practical applications in real-world scenarios.

1. Contributing to the Growing Field of Deep Learning:

* By delving into YOLO and its functionalities, this project contributes to the broader landscape of deep learning.
* Through research, experimentation, and analysis, participants contribute to the advancement of object detection and tracking algorithms, fostering growth and innovation in the field of computer vision.

Overall, the motivation behind this project lies in the opportunity to address existing limitations, explore practical applications, gain hands-on experience with state-of-the-art technology, and contribute to the evolving field of deep learning. It represents a journey towards innovation, discovery, and advancement in the realm of object detection and tracking.

**Project Objective**

The primary objective of this minor project is:

* To Implement and evaluate YOLO for object detection and tracking in video images.
* To Explore different YOLO variants and training strategies to optimize performance for specific datasets and applications.
* To Visualize and analyze the results to assess the effectiveness of YOLO for this task.
* To Compare YOLO's performance with other object detection and tracking methods (if applicable), highlighting its advantages and limitations.
* To Present the findings and potential applications of the project in a clear and concise manner.

Pros and Cons of Existing Methods

|  |  |  |
| --- | --- | --- |
| Method | Pros | Cons |
| R-CNN (Region-based CNN) | High accuracy, flexible | Slow, computationally expensive |
| SSD (Single Shot MultiBox Detector) | Fast, good for large datasets | Not as accurate as R-CNN, less flexible |
| RNN (Recurrent Neural Network) | Excellent tracking, handles complex motion | Slow, data-dependent, limited detection |
| YOLO (You Only Look Once) | Real-time speed, good accuracy, simple architecture | Not as accurate as R-CNN, less flexible for specific tasks |

**Methodology/ Planning of work:**

1. Object Detection and Tracking Fundamentals: Will Grasp core concepts of object detection and tracking, focusing on YOLOv8 algorithm comprehension and its Python implementation for video image analysis.
2. Data Preparation: Will Compile diverse video image datasets and standardize properties to meet YOLOv8 model requirements, ensuring consistency and accuracy in object detection and tracking.
3. Model Training: Will Configure and fine-tune YOLOv8 model parameters using collected datasets, optimizing performance for accurate object detection and tracking in video images.
4. Integration with Python: Will Develop Python scripts for seamless integration of YOLOv8 model functionalities into video processing applications, facilitating real-time object detection and tracking using OpenCV and YOLOv8 algorithms.
5. Testing and Documentation: Will Conduct comprehensive testing to ensure robust system functionality and address any encountered errors, while documenting methodologies, implementations, and outcomes for thorough project documentation.

**Facilities required for proposed work:**

This project leveraged the computational resources and software environment provided by Google Colab for the development and execution of YOLO-based object detection and tracking in video images. The following facilities were utilized:

1. Software:

* Deep Learning Frameworks: Either TensorFlow or PyTorch, chosen based on project requirements and personal preference. These frameworks served as the foundation for building and training the YOLO model. YOLO implementations (Darknet, YOLOv8)
* YOLO Implementation: An open-source implementation of YOLO, such as Darknet, You Only Look Once PyTorch (YOLOv8), was selected based on the desired version and ease of integration.
* Data Preprocessing Tools: Libraries like OpenCV were used for data preparation tasks such as image resizing, normalization, and augmentation.
* Data Visualization Tools: Tools like Matplotlib or Seaborn were employed to visualize detection results and analyze model performance through graphs and charts.

1. Hardware:

* Google Colab GPUs: The project benefited from the free access to T4 GPUs provided by Google Colab. This hardware acceleration significantly enhanced the training and inference speeds of the YOLO model.

Benefits of Utilizing Google Colab:

* No local setup required: Eliminates the need for local software installation, making it accessible to users with limited hardware resources.
* Accessibility and scalability: Operates within a web browser, accessible from any device with an internet connection. Offers different GPU configurations for scaling computational power.
* Simplified resource management: Handles all hardware and software maintenance, allowing focus on development and execution.

**Bibliography**:

* Joseph Redmon et al. (2016): YOLO: Unified, Real-Time Object Detection (CVPR) (ieeexplore.ieee.org/document/7780460: <https://ieeexplore.ieee.org/document/7780460>)
* Alexey Bochkovskiy et al. (2022): YOLOv8: Trainable 640x640 @ 60 FPS for Object Detection(github.com/WongKinYiu/YOLOv8:<https://github.com/WongKinYiu/YOLOv8>)
* Google Colab: (https://colab.research.google.com/: <https://colab.research.google.com/>)
* Hugging Face Hub: (https://huggingface.co/models: <https://huggingface.co/models>)
* Roboflow: (https://roboflow.com/: <https://roboflow.com/>)
* Machine Learning Crash Course: (https://developers.google.com/machine-learning/crash-course: https://developers.google.com/machine-learning/crash-course)