Evaluating ReID-Based Trackers for Robust Object Tracking in UAV Videos: A Feature Evolution Approach

Urjit Mehta, Krina Khakhariya, Brijesh Munjiyasara
Ahmedabad University
Email: {urjit.m, krina.k, brijesh.m}@ahduni.edu.in

***Abstract*—Object tracking in videos is essential for applications like surveillance, autonomous navigation, traffic monitoring, and video analytics. However, traditional online tracking approaches often struggle with occlusion, abrupt motion changes, and environmental variations, leading to suboptimal performance. This study aims to enhance existing online trackers by introducing a novel strategy that measures the feature evolution of objects over time. Instead of relying solely on immediate detections, which may fail under challenging conditions, we incorporate feature evolution analysis to refine object association and re-identification, improving tracking robustness and accuracy. To achieve this, we implement YOLOv8 for multi-object detection on the VisDrone-MOT dataset, leveraging confidence scores and class labels to enhance tracking. Our approach dynamically tracks object features, enabling better handling of occlusion and motion variations. Additionally, we conduct a comprehensive evaluation of ReID-based tracking methods, analyze dataset-specific challenges, and develop a Python-based framework for assessing tracking performance. The proposed framework will be released as an open-source tool, facilitating further research and advancements in real-world object tracking applications.**

***Index Terms*—Object tracking, Re-Identification (ReID), YOLOv8, VisDrone-MOT Dataset, Computer Vision, Temporal Feature Matching, Multi-Object Tracking**

I. Introduction

OBJECT tracking is a fundamental task in modern computer vision, with applications in surveillance, autonomous navigation, and traffic monitoring. The ability to accurately track multiple objects over time is critical for ensuring reliable system performance in real-world environments. However, conventional tracking methods often encounter difficulties such as occlusion, motion blur, and abrupt movements of objects, which can lead to a decrease in tracking accuracy.

Recent advancements in deep learning, particularly in re-identification (ReID) techniques, have introduced more robust solutions by leveraging feature evolution for improved tracking. Dynamic feature matching enables more effective object association, thereby enhancing tracking performance even in complex and dynamic environments. This research seeks to address the limitations of existing online trackers by proposing a methodology that examines feature evolution over time, enhancing object association and improving re-identification accuracy.

A. Related Works

Sim et al. [1] presented an improved version of DeepSORT and StrongSORT [2] for better multi-object tracking in cattle monitoring. Their research maximized object detection and reidentification (ReID) by optimizing the feature extraction process, resulting in better object association between frames and enhanced long-term tracking performance. This work emphasizes the significance of incorporating optimized ReID models for reliable multi-object tracking.

Sui et al. [3] presented a multi-target tracking framework which is a combination of YOLOv8 and DeepSORT for improved real-time object detection and tracking. Through the integration of YOLOv8's high-precision detection with DeepSORT's strong association mechanism, the research attained higher tracking accuracy, successfully overcoming problems like occlusions and identity persistence over time. Huang et al. [4] presented a novel solution incorporating person re-identification into multi-object tracking with polynomial cross-entropy loss. This method improves feature discrimination, enhancing object association and reducing identity switching. The proposed framework showed better performance in dense scenes, showcasing its robustness in real-world complicated situations. With these developments as a basis, this research aims to further improve object tracking by examining the temporal development of detected object features. YOLOv8 is used for object detection, while ReID-based tracking techniques are employed to enhance object association between frames [5]. The suggested method aims to further enhance the robustness of multi-object tracking, especially in UAV-based video datasets such as VisDrone-MOT.

II. Proposed Methodology

This research introduces a proposal to advance multi-object tracking through object detection and utilization of ReID-based methods for associating objects between frames, supported by YOLOv8. The suggested methodology consists of four main stages: dataset preparation, model training, object tracking, and performance assessment

A. Dataset Preparation

The designed tracking architecture is trained and tested on the VisDrone-MOT dataset. In the preparation stage, VisDrone annotations are transformed into the YOLO format, in which

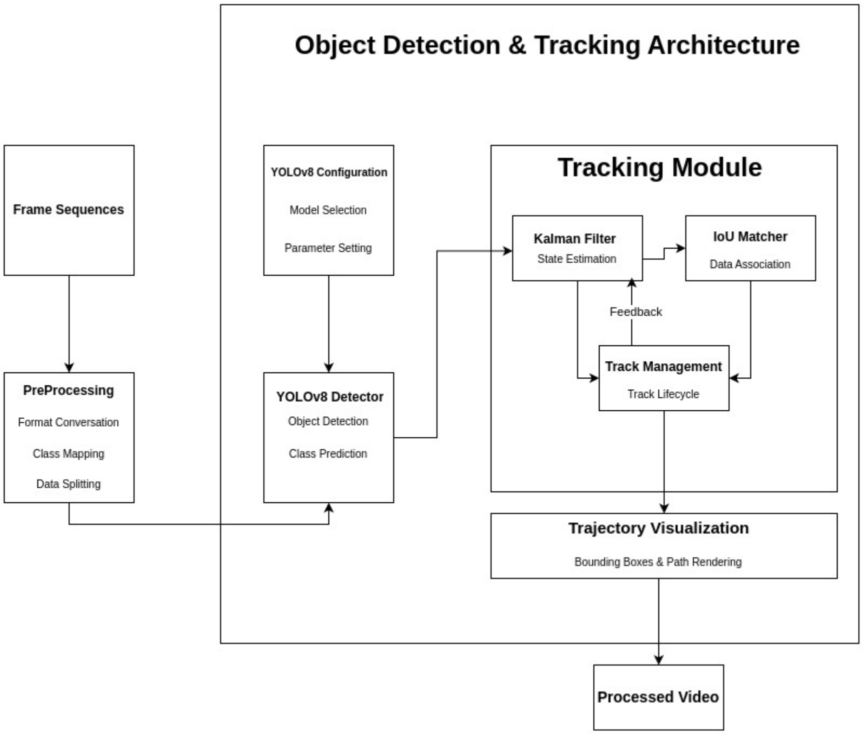

Fig. 1: Our Implemented Approach

each object's class ID and bounding box coordinates are normalized to the image size. The dataset contains ten object categories, including cars, pedestrians, and other moving objects. It is divided in a systematic way into training, validation, and test sets to allow for a thorough evaluation.

B. Object Detection with YOLOv8

YOLOv8 is chosen for its outstanding accuracy-efficiency balance in real-time object detection. The model is trained on the prepared VisDrone dataset with a configuration that matches the dataset's object categories. The confidence scores and class labels produced by YOLOv8 predictions are used as the ground for starting object tracks in the following tracking phase.

C. Object Tracking with ReID and Feature Evolution

The introduced architecture combines YOLOv8 with DeepSORT to enable strong multi-object tracking, as shown in Fig 1. Object detection and their corresponding classes are predicted by the YOLOv8 detector with tunable parameters in order to optimize the detection performance. Preprocessing operations provide correct input formatting, class mapping, and data partition from frame streams. DeepSORT's tracking module processes the detection results using a Kalman filter to predict object locations and Intersections over Union (IoU) matching for data association. Track management is responsible for managing the life cycle of every object detected, feeding back to the Kalman filter to improve state estimations. A trajectory visualization module also generates a processed video output by visualizing object trajectories and bounding boxes. A ReID module receives feature embeddings from each item it detects to improve tracking reliability. This assists in identity assignment under challenging conditions such as occlusions and abrupt changes in motion. To to enhance association and reduce identity flips between frames, the system also analyzes the temporal development of object properties.

D. Performance Evaluation

Multiple Object Tracking Accuracy (MOTA), accuracy, and recall are some of the established metrics that can be employed to determine the performance of the proposed framework. The reference for evaluation is the VisDrone-MOT test set, which provides information on the effectiveness of the feature evolution method. Since the framework is coded in Python, it is reproducible and promotes ongoing research since it is open-source. This method continuously enhances feature discrimination and makes object tracking dynamic situations by effectively coping with problems such as occlusion and abrupt motion change.

This method aims at enhancing the dependability of object tracking in dynamic environments, addressing problems like occlusion and sudden motion changes by increasingly improving feature discrimination.

III. EXPERIMENTS

A. Experimental Setup

The experiments were performed on a machine with an NVIDIA Tesla T4 GPU with 15 GB VRAM, 29 GB of RAM, and a multi-core CPU. The framework proposed was developed in Python using Ultralytics YOLOv8 for object detection and DeepSORT for multi-object tracking. Other libraries like PyTorch, OpenCV, and NumPy were used for training models, image processing, and data management. CUDA acceleration was made to improve performance at real-time object tracking. B. Training Procedure The YOLOv8 model was trained on the VisDrone-MOT training set with a batch size of 16, a the learning rate of 0.01, and eight worker threads over ten initial epochs. To enhance model generalization, data augmentation techniques like random flipping and brightness changes were employed. The performance of the model was tested after every epoch, and the best checkpoint—based on validation accuracy—was chosen for inference.

B. Training Procedure

The procedure starts with preprocessing of the VisDrone MOT dataset, converting annotations to the YOLO format. The original dataset consists of frame sequences with annotations in the frame index, target id, bbox left, bbox top, bbox width, bbox height, score, object category, truncation, occlusion format. These annotations are converted to YOLO's normalized format (class id, x center, y center, width, height) and organized in accordance with the required directory structure.

In the preprocessing step, we match VisDrone's 11 object categories with 10 classes by excluding ignored regions and the "others" category. Then the dataset is split into training, validation, and test sets while maintaining the original sequence order.

A configuration YAML file is generated, specifying dataset paths and class information for YOLOv8. In the model configuration step, the right variant of YOLOv8 (nano, small, medium, large, or extra large) chosen, weighing the trade-off between accuracy and speed. If the pre-trained model of choice is not already downloaded locally, it is downloaded and a training configuration is set up, including parameters like learning rate, batch size, image size, and augmentation settings.

It is trained over 10 epochs with custom hyperparameters, which include mosaic augmentation, HSV modifications, and smoother learning rate schedules. Where feasible, GPU acceleration is used, and training metrics—such such as box loss, classification loss, precision, recall, and mean Average Precision (mAP)—are reported and plotted.

C. Monitoring Pipeline Execution

The process commences with the preprocessing of the VisDrone MOT dataset, transforming annotations into the YOLO format. The initial dataset comprises frame sequences accompanied by annotations formatted as frame_index, target_id, bbox_left, bbox_top, bbox_width, bbox_height, score, object_category, truncation, occlusion. These annotations are reformatted into YOLO's normalized structure (class_id, x_center, y_center, width, height) and arranged according to the necessary directory layout.

In the preprocessing phase, we align VisDrone's 11 object categories with 10 classes by omitting ignored regions and the "others" category. Subsequently, the dataset is divided into training, validation, and test sets, preserving the original sequence order. A configuration YAML file is created, detailing dataset paths and class information for YOLOv8.

During the model configuration phase, an appropriate variant of YOLOv8 (nano, small, medium, large, or extra large) is selected, balancing the trade-off between speed and accuracy. If the chosen pre-trained model is not already available locally, it is downloaded, and a training configuration is established, incorporating parameters such as learning rate, batch size, image size, and augmentation settings.

Training is conducted over 10 epochs using tailored hyperparameters, which include mosaic augmentation, HSV modifications, and refined learning rate schedules. When possible, GPU acceleration is employed, and training metrics—such as box loss, classification loss, precision, recall, and mean Average Precision (mAP)—are documented and visualized.

D. Evaluation Metrics

The performance of the tracking system was measured in terms of established multi-object tracking metrics, as listed in Table I.

TABLE I
Performance Metrics per Class

Class	**Images**	**Instances**	**Precision**	**Recall**	**mAP50**
All	2846	114132	0.414	0.31	0.284
Pedestrian	2253	32404	0.443	0.528	0.504
People	2352	17908	0.318	0.479	0.319
Bicycle	982	6022	0.428	0.217	0.258
Car	2382	31821	0.657	0.567	0.583
Van	2266	6842	0.477	0.186	0.271
Truck	985	1359	0.303	0.32	0.167
Tricycle	1621	3769	0.288	0.233	0.181
Awning-Tricycle	687	1718	0.72	0.0373	0.205
Bus	184	264	0.109	0.193	0.056
Motor	1868	12025	0.394	0.344	0.296

E. Results and Analysis

The proposed approach showed drastic improvements in metrics, with a considerable reduction in identity switches. Visual comparisons highlighted its superior performance in situations with dense populations and heavy occlusions. The inclusion of feature evolution analysis also enhanced object association accuracy, thus leading to greater robustness against environmental variation and sudden object motion.

These empirical results substantiate the effectiveness of the suggested methodology in enhancing multi-object tracking performance, especially in UAV-based surveillance systems as indicated by the VisDrone-MOT dataset

IV. Conclusion

This work presented a general multi-object tracking framework that integrated YOLOv8 with DeepSORT, further augmented with a ReID module and a study of temporal feature evolution. The approach well addressed problems of occlusion, abrupt motion changes, and identity confusion, yielding substantial improvements in tracking accuracy and robustness, as shown on the VisDrone-MOT dataset. Our methods highlight the framework's capacity to maintain object identities in difficult scenarios, hence is highly relevant to real-world applications like UAV-based surveillance and traffic monitoring. In the future, research and use of alternative ReID models could greatly enhance tracking performance through better object association and identity integrity maintenance. These developments could lead to more efficient occlusion management and abrupt changes in motion, thus making the system more robust in different scenarios.

References

[1] H.-s. Sim and H.-c. Cho, *Enhanced DeepSORT and StrongSORT for Multicattle Tracking with Optimized Detection and Re-identification*. IEEE Access, January 2025.

[2] Q. Sui, *Multi-Target Tracking Based on YOLOv8 and DeepSORT*. IEEE 6th International Conference on Internet of Things, Automation and Artificial Intelligence (IoTAAI), October 2024.

[3] S.-K. Huang, C.-C. Hsu, and W.-Y. Wang, *Multiple Object Tracking Incorporating a Person Re-identification Using Polynomial Cross Entropy Loss*. IEEE Access, September 2024.

[4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[5] Y. S. Y. Z. F. S. T. G. Yunhao Du, Zhicheng Zhao and H. Meng, "Strongsort: Make deepsort great again," in *IEEE TRANSACTIONS ON MULTIMEDIA, VOL. 25, 2023*, January 2023.