# Single Camera Multi-Object Tracking

Vivian Pan
University of Washington
1410 NE Campus Pkwy, Seattle, WA 98195
vivianvp@uw.edu

Anthony Chu
University of Washington
1410 NE Campus Pkwy, Seattle, WA 98195
lijiechu@uw.edu

## 1. Introduction

Multi-Object Tracking (MOT) is a fundamental task in computer vision that aims to track multiple objects across a sequence of video frames. By assigning consistent identifiers to detected objects, MOT enables the analysis of dynamic scenes, making it an essential technology in applications such as video surveillance, autonomous driving, sports analytics, and human behavior analysis. The primary objective of MOT is to maintain object identities across frames while handling challenges such as object occlusion, overlapping, and abrupt motion changes. At the core of most MOT systems lies a two-stage process: object detection and data association. Object detection identifies objects in each frame, producing bounding boxes that localize the objects and confidence scores that quantify the reliability of the detections. Data association then matches these detected objects across frames into distinct tracks, assigning unique identifiers (IDs) to each track or object. This process ensures that objects retain consistent identities as they move through the scene.

Our work will be focusing on single-camera MOT, which involves tracking multiple objects with a single camera. We found a MOT model that uses purely object detection with track association through calculating the Intersection over Union (IoU) between detections (or bounding boxes). While this model works effectively for a simple video, its performance when occlusions occur or when objects leave and re-enter the frame suffers terribly. This is due to the method primarily relying on predicting that an object remains the same across frames based on how much their bounding boxes overlap. Thus, we will implement an algorithm influenced by the BoTSORT algorithm to improve the original model's deficiencies. The new approach involves implementing re-identification (ReID), Kalman Filters, and a high-low confidence hierarchy during track association. This hierarchical matching strategy, combined with feature identification through ReID and Kalman Filter motion predictions, significantly reduces ID switches and enhances tracking robustness. Unlike traditional methods that rely solely on predicting where objects will move based on their motion, this approach leverages ReID to match objects to their unique appearance features. These features can be used to re-identify objects even after mix-ups, such as occlusions or objects leaving and re-entering the frame, making the tracking process more reliable in challenging environments like crowded scenes. By adopting this approach, we aim to achieve a more efficient and accurate single-camera MOT pipeline, capable of handling real-world complexities with improved tracking accuracy and fewer ID switches, even with challenges like occlusion and objects leaving the frame.

## 2. Related Work

The field of Multi-Object Tracking (MOT) has evolved significantly over the past few decades, with advancements driven by both theoretical insights and technological progress. The concept of tracking multiple objects simultaneously was first formalized by Zenon Pylyshyn and Ronald W. Storm in 1988 [4]. They introduced the idea of a parallel tracking mechanism capable of monitoring multiple independent targets, laying the groundwork for subsequent research in MOT. With the advent of robust object detectors, the tracking-by-detection framework became prominent. This approach involves detecting objects in each frame and then associating these detections across frames to form trajectories. The Simple Online and Realtime Tracking (SORT) algorithm, introduced by Bewley et al. in 2016, exemplifies this paradigm by combining Kalman filtering for motion prediction with the Hungarian algorithm for data association [2]. To address challenges such as occlusions and similar-looking objects, researchers began incorporating appearance information into tracking systems. The Deep SORT algorithm, proposed by Wojke et al. in 2017, extended SORT by integrating deep appearance descriptors, enhancing the tracker's ability to distinguish between objects with similar motion patterns. Recent developments have focused on improving data association techniques. The ByteTrack algorithm, introduced by Zhang et al. in 2021, emphasizes the importance of low-confidence detections, demonstrating that incorporating these can significantly enhance tracking performance

[5]. Building upon previous advancements, BoT-SORT combines hierarchical matching strategies with appearance-based re-identification (ReID) features. This approach first matches high-confidence detections using motion information and then utilizes ReID embeddings for remaining associations, effectively reducing ID switches and improving robustness in complex scenarios.

## 3. Methodology

Our single-camera multi-object tracking (MOT) pipeline improves upon a baseline system by integrating advanced detection, feature extraction, and tracking components. The proposed architecture is designed to address challenges such as occlusions, ID switches, and fragmented tracks, which are common in complex scenes. The pipeline consists of four main stages: detection, ReID-based feature extraction, tracking with Kalman filters and hierarchical matching, and post-processing with clustering. In figure 1 we have visualized a high level outline of our new architecture with our implementations that were added on top of the baseline highlighted in yellow.

### 3.1. Dataset

The dataset used in our approach is a portion of the NVIDIA City Challenge 2023 dataset [1]. This dataset contains urban scenes recorded at 30 frames per second, offering a rich source of training and evaluation data for object tracking tasks. Annotations for each frame include the camera ID, tracking ID, frame ID, and bounding box coordinates (xmin, ymin, width, height). The axis-aligned rectangular bounding box of the detected object is denoted by its pixel-valued coordinates within the image canvas–xmin, ymin, width, height–computed from the top-left corner of the image. All of these values are integers. These bounding boxes represent the location and size of each object in the scene. The testing phase mandates consistent tracking IDs across frames, making robust tracking algorithms crucial for success. This dataset's diversity in urban environments makes it an excellent benchmark for evaluating single-camera MOT systems. We divide the dataset into three splits: training, validation, and testing. The training set comprises of six sequences, each lasting 60 seconds with 1800 frames and annotated ground truth labels for all objects. The validation set consists of three sequences of the same length, also including ground truth labels for offline performance evaluation. The testing set, containing three longer sequences of 3600 frames each, excludes ground truth labels and requires participants to submit predictions in the specified format for evaluation.

### 3.2. Original Baseline Model

Our baseline model implements a straightforward approach to object tracking using YOLO for detection, a simple IoU-based tracker for maintaining object identities, and clustering for post processing to match tracklets together. It employs the YOLOv8x model and was initialized with pre-trained weights from the COCO dataset to leverage its extensive training on diverse objects, including humans. The training process involved reorganizing the NVIDIA City Challenge dataset to fit the YOLO engine requirements, extracting video frames, resizing them to match YOLOv8's input format, and fine-tuning the model over 100 epochs. This fine-tuning step enabled the model to adapt specifically to the dataset, improving its performance in segmenting humans accurately. Once trained, each frame of the video was

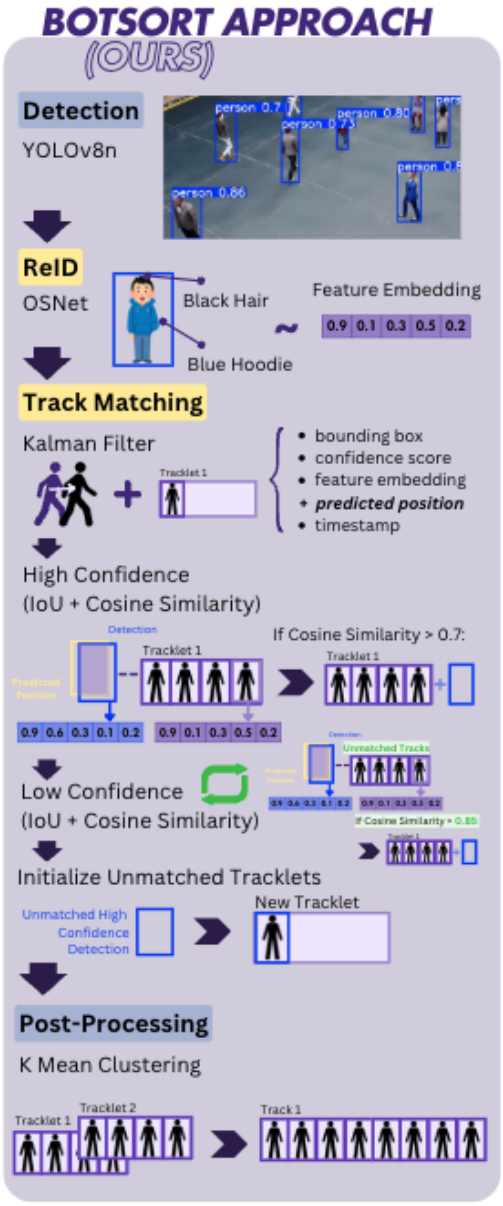


Figure 1. High Level Diagram of Our Architecture

passed through the YOLOv8x model to detect and localize humans. For each detection, the model recorded bounding boxes, confidence scores, and associated metadata. The outputs were saved in a .txt file, with each line structured as follows: [cam_id, person_id, frame_id, x, y, w, h, confidence, -1].

The tracking component of the baseline model uses an IoU-based mechanism to associate detections with existing tracks. For each frame, the tracker calculates the Intersection over Union (IoU) between the bounding boxes of detected objects and the bounding boxes of active tracks (between 2 detections). The Hungarian algorithm is applied to solve the assignment problem, ensuring optimal matching by minimizing the cost $1 - \text{IoU}$. Matches with the lowest costs are selected to associate detections with tracks. If no match is found for a detection, a new track is initialized with a unique ID. Conversely, if no detection matches an existing track, the track is marked as inactive and removed from the list of active tracks. Optionally, the system can calculate average features for tracks after processing all frames, supporting appearance-based association in future extensions.

The baseline model also incorporates post-processing with KMeans clustering to merge fragmented tracklets caused by objects leaving and re-entering the frame. Each tracklet maintains a history of bounding boxes and appearance features across frames. Once tracking is complete, the average appearance features for all tracklets are computed. These features are clustered using KMeans, where each cluster corresponds to a unique object identity. This step ensures that fragmented tracks are consolidated, reducing ID switches and improving overall tracking consistency.

This baseline performs well in simple scenarios where objects do not overlap significantly and follow predictable motion paths. However, it struggles in crowded scenes where objects with similar motion patterns may lead to ID switches or when objects are occluded or reappear after leaving the frame. The reliance solely on IoU during the tracking phase limits its ability to differentiate objects with similar spatial positions, creating issues in complex environments. We aim to enhance this baseline system by integrating appearance-based matching and hierarchical data association methods, which will improve the consistency of identification and tracking results. The baseline will serve as a foundation to evaluate the effectiveness of the enhancements introduced in our proposed pipeline. The detection (YoloV8n model) and post processing (KMeans) will remain the same in our new model.

### 3.3. New Architecture: ReID

To address challenges involving distinguishing objects with similar motion patterns, handling occlusions, and managing cases where objects leave and re-enter the frame, we incorporate a Re-Identification (ReID) model. After detecting objects with YOLO, their cropped bounding boxes are passed to OSNet (Omni-Scale Network), a ReID model specifically designed for extracting discriminative appearance embeddings. OSNet operates by capturing multi-scale features, effectively encoding both fine-grained details (e.g., texture, patterns) and broader characteristics (e.g., color distributions, shapes). This capability is achieved through its novel omni-scale feature learning architecture, which processes information at multiple receptive fields simultaneously [7]. At its core, OSNet introduces lightweight residual blocks that aggregate features from various scales. These blocks enable the network to efficiently focus on both local and global cues, making it particularly well-suited for ReID tasks where objects may vary in scale, pose, or perspective. The model normalizes extracted embeddings to ensure consistency across frames, making it robust to environmental changes such as variations in lighting or camera angles. These embeddings are high-dimensional feature vectors that uniquely represent each object, encoding characteristics like color, texture, and shape. By maintaining this distinctiveness across frames, OSNet allows the system to effectively re-identify objects even after they have been temporarily occluded or missed by the detector. To implement OSNet, we leverage TorchReID, a robust library for ReID tasks. After training, OSNet generates embeddings for each detected object, which are combined with bounding box information from YOLO. These combined features provide a powerful mechanism for associating objects across frames, significantly reducing ID switches and fragmented tracks. This integration ensures that the tracking system is robust, accurate, and capable of handling the complexities of real-world scenarios.

### 3.4. New Architecture - Improved Tracker

The tracking component of our pipeline integrates Kalman Filters, IoU-based matching, and a high-confidence-first hierarchy to associate detections with existing tracks and maintain consistent object identities across frames. This multi-step process ensures robust performance in dynamic and cluttered environments.

To predict the motion of tracked objects and ensure continuity in tracking, especially in cases of missed detections or occlusions, we integrate Kalman Filters as part of our motion prediction strategy. The filter predicts the next position and dimensions of an object's bounding box based on its current state, which includes the object's position (x, y), velocity ($v_x$, $v_y$), and bounding box dimensions (w, h).

The Kalman Filter operates in two main phases: prediction and update [3]. During the prediction phase, the filter uses a linear motion model to estimate the state of the object in the next frame. This model assumes constant velocity, predicting the object's position by extrapolating its motion based on the previously estimated state and velocity. This step produces a prior estimate, which serves as the predicted state for the object before any new detection is observed. In the update phase, the Kalman Filter refines its prediction using a measurement update. When a new detection becomes available, the filter compares the observed position of the object (from the detector) to the predicted position and computes a residual, which is the difference between these two values. This residual is used to correct the prior estimate, producing a more accurate posterior estimate of the object's state. The correction is weighted by the Kalman Gain, which determines how much influence the new measurement should have on the updated state. This integration of Kalman Filters provides our pipeline with a strong foundation for motion prediction, ensuring that tracks remain accurate and consistent across frames, even in the face of uncertainty or partial observations.

In our tracking system, we employ a high-low hierarchy in the track matching process to match detections from the object detector to the predicted positions of existing tracklets (between a detection and the track's predicted position). This approach combines spatial and appearance-based metrics [6]. First, we perform high-confidence matching to prioritize detections that are more likely to be accurate. Detections with confidence scores above a 0.6 are considered for this stage. These high-confidence detections are matched to existing tracklets based on the IoU metric, which quantifies the spatial overlap between bounding boxes. IoU is calculated as the ratio of the overlapping area between two bounding boxes to their union area, providing a measure of how closely a detection aligns with a predicted tracklets. To facilitate this matching, we construct a cost matrix, where each entry represents the IoU score (inverted to a cost value) between a detection and a tracklets. This matrix is passed to the Hungarian algorithm, assigning detections to tracklets in a manner that minimizes the total cost. This creates temporary matches between detections and tracklets. The feature embeddings of these matches get compared by calculating the cosine similarity between the tracklet's most recent detection and the currently considered detection. Higher cosine similarity values indicate a closer match between the visual features of two objects. If the cosine similarity is greater that 0.7. the match is confirmed and the detection gets added to the tracklet. Next, we repeat the above process but with all the low confidence detections and unmatched tracklets. However, the cosine similarity threshold is increased from 0.7 to 0.85. Lastly, we initialize the remaining unmatched high confidence detections as new tracklets.

## 4. Experiments

Before the track association portion we also found that filtering out low detection scores less than 0.3 increased final result metrics, so we implemented that prior to the track logic. The core of our experiment focused on optimizing the hierarchical data association logic. A key component of our experimental evaluation was determining the optimal cosine similarity thresholds. For high-confidence detections, we conducted a series of controlled experiments where the threshold was varied across 0.65, 0.7, 0.75, and 0.8. Each threshold setting was evaluated on the validation split, and we tracked changes in Multiple Object Tracking Accuracy (MOTA), ID-F1, and ID-P. The threshold of 0.7 emerged as the most balanced choice, minimizing both ID switches and false associations while improving the overall stability of the tracklets. We found that at 0.65, the system allowed too many questionable matches, ID-P; at 0.8, the system became overly restrictive, failing to properly re-link objects that were indeed the same individual. By contrast, a 0.7 threshold struck an effective compromise, ensuring that visually similar objects were correctly re-identified without overly penalizing minor variations in appearance due to pose changes or partial occlusions.

Similarly, for the second stage of matching, which considered low-confidence detections, we repeated the threshold-tuning process with a stricter range of 0.75, 0.8, 0.85, and 0.9. The rationale behind a higher threshold here was that low-confidence detections are inherently riskier candidates for identity association, necessitating a more stringent similarity check to prevent incorrect matches. After analyzing performance metrics for each candidate value, we selected a threshold of 0.85. This provided a good balance between maintaining continuity for objects that momentarily dipped in detection confidence and preventing the introduction of spurious new identities.

## 5. Results

Our proposed approach, influenced by the BoTSORT algorithm, delivered notable improvements over the baseline system across a range of standard MOT performance metrics. As previously discussed, the key metrics included:

- ID-F1: The harmonic mean of ID Precision (ID-P) and ID Recall (ID-R). A higher ID-F1 indicates that the tracker excels in maintaining stable and correct object identities throughout the sequence. An improvement here suggests fewer ID switches and more consistent identity assignments over time.
- ID-P (ID Precision): Measures the proportion of correctly identified IDs among all assigned IDs. High ID-

P means the tracker rarely introduces new, incorrect IDs for objects, reducing identity fragmentation and confusion.

- ID-R (ID Recall): Reflects the proportion of true persistent identities successfully retained by the tracker. Improved ID-R indicates that the system rarely loses track of an object's identity once established.
- MOTA (Multiple Object Tracking Accuracy): Aggregates false positives, missed detections, and ID switches into a single measure, offering a comprehensive overview of tracking performance.
- Recall: The fraction of ground-truth objects that are detected and tracked. Higher recall indicates fewer missed targets.
- Precision: The fraction of predicted detections that correspond to true objects. High precision ensures the system maintains high-quality detections.

Table 1. Performance Results

| | BoTSORT | Baseline |
|---|---|---|
| ID-F1 | 79.88 | 56.36 |
| ID-P | 86.99 | 66.80 |
| ID-R | 73.84 | 48.75 |
| MOTA | 85.94 | 62.91 |
| Recall | 83.08 | 68.22 |
| Precision | 97.87 | 93.48 |

As seen in the resultd above, compared to the baseline, our system showed substantial gains. ID-F1 improved from 56.36 to 79.88, underlining a marked reduction in ID fragmentation and identity switches. Both ID-P and ID-R also saw significant increases, signaling that our combination of appearance features (ReID) and hierarchical association stabilized object identities and minimized mismatches. MOTA leaped from 62.91 to 85.94, confirming that the integrated approach effectively tackled various error sources—missed detections, false alarms, and ID switches. Together, the boosts in Recall (68.22 to 83.08) and Precision (93.48 to 97.87) reflect that our system not only tracked more objects but also did so with greater confidence and fewer incorrect detections.

The qualitative examples illustrated in figure 2 underscore the real-world impact of our approach. In the baseline scenario, a tracked individual originally assigned ID "18" crossed paths with another object. Due to the close proximity and reliance solely on spatial overlap, the baseline system lost track of the individual's identity and reassigned it as "46" in the subsequent frames. This ID switch reflects the baseline's susceptibility to occlusions and overlapping objects, where simple IoU-based tracking is insufficient to maintain correct identities. In contrast, our enhanced pipeline, leveraging OSNet embeddings and hierarchical data association, retained the correct identity for the same individual—referred to as "15" in our output—throughout the entire interaction. The appearance embeddings allowed our system to confirm that the individual before and after the overlap was the same person, despite changes in the bounding box position and partial occlusion. The Kalman Filter's prediction of motion continuity further assisted in maintaining the track, while the carefully tuned cosine similarity thresholds prevented the system from prematurely assigning a new ID. This qualitative evidence directly aligns with our hypothesis: by integrating appearance-based features, motion modeling, and a two-tiered matching scheme, the system can robustly handle occlusions, re-entries, and heavily overlapping objects without succumbing to identity confusion.

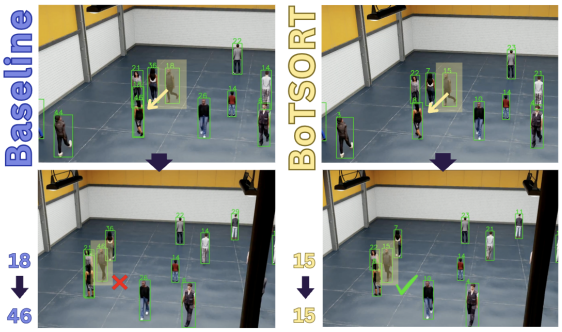


Figure 2. BoTSort model prevented ID swap from two object's path intersection.

## 6. Limitations and Future Work

Despite these encouraging results, certain limitations remain. Our reliance on a pre-trained OSNet model, while beneficial for rapid deployment, may not optimally capture domain-specific characteristics of the NVIDIA City Challenge dataset's urban environments. Fine-tuning the ReID model or training it on a more extensive, domain-specific dataset could potentially yield even stronger identity retention and reduce residual ID switches in highly crowded or visually ambiguous scenarios. Additionally, while the chosen cosine similarity thresholds performed well overall, they were derived empirically. Future work could explore adaptive thresholding strategies that dynamically adjust similarity requirements based on scene complexity, object densities, or environmental factors like lighting conditions. Another avenue of improvement involves the integration of temporal cues beyond the Kalman Filter's linear motion model, such as leveraging more advanced motion models or incorporating scene-specific priors. Lastly, while our model delivered substantial gains in ID consistency and MOTA, fine-grained improvements—like improving long-

term re-identification after very long occlusions—could further enhance system robustness. Exploring more sophisticated clustering techniques, multi-camera fusion, or transformer-based attention mechanisms for data association may address these remaining challenges. In summary, our enhancements effectively addressed the baseline system's weaknesses by improving tracking performance during occlusions, ensuring correct identity reassignments when objects left and re-entered the frame, and stabilizing identity maintenance even when multiple objects overlapped heavily. While there are still areas for refinement, our current results show that the integrated approach significantly pushes the state-of-the-art in single-camera MOT for complex real-world scenarios.

## References


[1] AI City Challenge. 2023 challenge tracks. https://www.aicitychallenge.org/2023-challenge-tracks/, 2023. Accessed: 2024-11-19. 2

[2] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3464–3468, 2016. 1

[3] R. E. Kalman. A new approach to linear filtering and prediction problems. *ASME Journal of Basic Engineering*, 82(1):35–45, March 1960. 4

[4] Zenon W Pylyshyn and Ron W Storm. Tracking multiple independent targets: Evidence for a parallel tracking mechanism. *Spatial vision*, 3(3):179–197, 1988. 1

[5] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3645–3649, 2017. 2

[6] F. Zheng, H. Wang, and M. Li. A survey on kalman filter algorithms for dynamic systems. *Algorithms*, 13(4):80, April 2020. 4

[7] Kaiyang Zhou, Yongxin Yang, Andrea Cavallaro, and Tao Xiang. Omni-scale feature learning for person re-identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 3