Task 1: Single Object Tracking

1. Template Matching

1.1 Comparison of Template Matching Methods for Optimal Performance per Sequence

Table 1 presents a comparative analysis of six template matching (TM) methods from the OpenCV library, evaluating their performance across five sequences. Each method has its own application scenarios and strengths. For example, correlation-based methods like TM_CCOEFF are robust to global brightness changes, whereas TM_SQDIFF focuses on minimizing pixel differences, favoring exact pattern matching.

Table 1. Comparison of TM Methods for each sequence

Seq	TM_CCOEFF		TM_CCOEFF_NORMED		TM_CCORR		TM_CCORR_NORMED		TM_SQDIFF		TM_SQDIFF_NORMED	
Index	P	S	P	S	P	S	P	S	P	S	P	S
1	45.33	44.67	34.00	34.67	0.00	0.00	29.33	33.33	38.67	36.00	36.67	40.00
2	49.33	49.33	48.00	48.00	0.00	0.00	4.67	4.67	4.67	4.67	5.33	5.33
3	4.67	4.67	7.33	12.00	0.67	0.67	3.33	3.33	4.00	4.00	3.33	3.33
4	24.67	21.33	36.00	31.33	0.00	0.00	24.00	23.33	34.67	29.33	52.00	42.00
5	74.00	74.00	74.00	74.00	68.67	68.67	74.00	74.00	73.33	73.33	72.67	72.67

P (Precision): Measures positional accuracy with a threshold of 25 pixels. S (Success): Assesses overlap quality using Intersection over Union (IoU) with a threshold of 0.5. All values are presented without the percentage sign. The highest-performing TM method for each sequence is highlighted in green. The final selected method for each sequence, indicated by an underline, represents the optimal choice based on the highest score in either Precision or Success.

1.2 TM Optimized by Adaptive Template Update and Kalman Filter

Adaptive Template Update (ATU) was first applied to improve the results of the basic TM method. ATU dynamically updates the template based on the most recent accurate detections, enabling the system to adapt to gradual appearance changes in the target. This enhancement led to notable performance improvements across several sequences, as shown in Table 2.

Kalman Filtering (KF) was further integrated into the TM + ATU pipeline to refine tracking by compensating the object's current states based on its previous motion. However, as given in Table 2, KF showed minimal improvements over the TM + ATU results. This is due to the inherent limitations of TM-based detections, which lack dynamic adaptability and often result in rigid bounding boxes, limiting the KF's capacity to compensate for significant deviations or occlusions.

Despite these optimization attempts, the TM-based approaches still underperformed in most sequences. Static bounding boxes and tracking loss under occlusions, object displacement, and deformation, leads to poor accuracy and tracking continuity, particularly in dynamic environments.

2. Object Detection Algorithm with Association

The DEtection TRansformers (DETR) framework, a transformer-based architecture for end-to-end object detection, was introduced to handle the perception task. Table 2 shows that DETR achieved superior overall performance, with the most flexible bounding boxes and strong resilience against occlusions. Its ability to maintain accurate detections during partial occlusions highlights its robustness. However, Sequence 4 revealed DETR's vulnerability in specific scenarios, a classic corner case where the target was initially occluded, leading the model to misidentify and subsequently track the wrong object throughout the sequence.

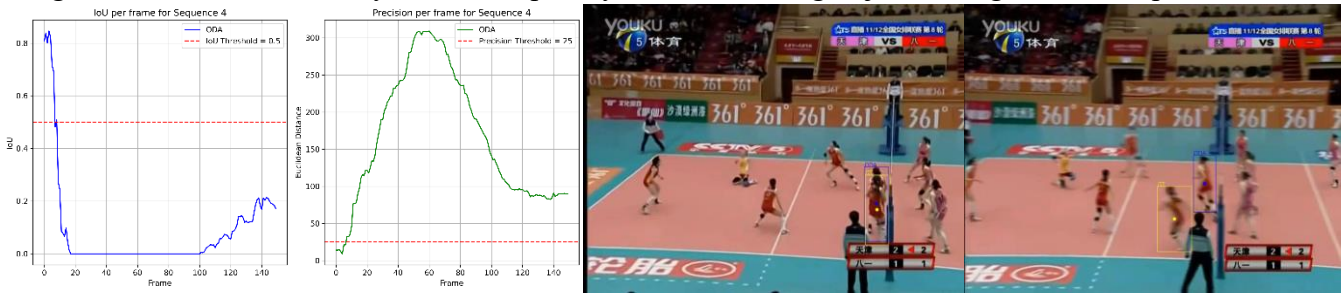


Figure 1. Sequence 4 is a corner case of occlusion.

3. An Other Method: YOLOv5 + DeepSORT

The combination of YOLOv5 for object detection and DeepSORT for tracking resulted in performance that sits between TM-based methods and DETR. This hybrid approach effectively addressed TM's rigidity by introducing bounding box flexibility and enhancing responsiveness to object movements and deformations. However, it still lagged behind DETR in overall detection precision. Similar to DETR, the YOLOv5 + DeepSORT pipeline faced challenges in occlusion scenarios, resulting in temporary tracking failures or identity switches when targets were obscured, which further impacted detection accuracy.

Table 2. Performance Comparison of Different Single Object Tracking Methods Results

Seq Index	TM		TM + ATU		TM + ATU + KF		DETR		YOLOv5 + Deep SORT	
	P	S	P	S	P	S	P	S	P	S
1	45.33%	44.67%	92.67%	90.00%	82.67%	68.00%	**96.67%**	**97.33%**	50.67%	52.00%
2	49.33%	49.33%	53.33%	40.00%	50.00%	39.33%	**90.00%**	**74.67%**	30.67%	21.33%
3	7.33%	12.00%	8.00%	12.67%	8.67%	12.67%	**31.33%**	**72.67%**	30.67%	50.00%
4	**52.00%**	**42.00%**	4.67%	4.67%	4.67%	4.67%	4.67%	5.33%	2.00%	4.00%
5	74.00%	74.00%	71.33%	73.33%	73.33%	74.00%	**81.33%**	**96.67%**	68.00%	96.67%

Using Precision (P) and Success (S) as evaluation metrics. The assessment follows the same criteria as Table 1. The highest score for each sequence is highlighted in green. The optimal results among the TM method and its improvements are highlighted in blue.

4. Evaluation

Tables 3 and 4 compare three SOT methods using Precision and Success metrics. Template Matching achieves the highest precision but has low success, indicating poor robustness, particularly under occlusion, scale variation, and viewpoint changes. Detection with Association shows the best success rate, reflecting stronger tracking stability and better handling of occlusions and scale variations, though it sacrifices precision. The Improved Method offers a balanced performance but does not surpass the best scores in either metric.

Table 3. Evaluation of SOT methods using Precision metrics (in pixels)

	Average Precision	Seq 1	Seq 2	Seq 3	Seq 4	Seq 5	Average Score
1	Template Matching	2172.47	3835.2	19038.17	27931.56	2723.86	11140.252
2	Detection with Association	602.4	1079.74	5255.25	16510.54	2045.18	**5098.622**
3	YOLOv5 + Deep SORT	14969.7	3727.82	5129.6	16800.44	2369.73	8599.458

Table 4. Evaluation of SOT methods using Success metrics

	Average Success	Seq 1	Seq 2	Seq 3	Seq 4	Seq 5	Average Score
1	Template Matching	55.41%	39.21%	12.46%	4.82%	63.70%	35.12%
2	Detection with Association	83.23%	64.17%	65.17%	8.50%	74.87%	**59.19%**
3	YOLOv5 + Deep SORT	40.95%	21.28%	46.19%	7.54%	70.95%	37.38%

5. Conclusion

Table 5 conclude the strengths and weaknesses of each method. DETR demonstrates robustness to occlusion and scale changes but struggles with varying viewpoints. YOLOv5 + DeepSORT effectively handles scale variations but remains sensitive to occlusions and viewpoint shifts. In contrast, Template Matching fails to adapt to changes in viewpoint, occlusion, and scale, which explains its instability despite high precision in certain sequences. This combined analysis emphasizes the importance of method selection based on specific environmental challenges in SOT tasks.

Table 5. Capability analysis

	TM	**DETR**	**YOLOv5 + DeepSORT**
Point view	×	×	×
Occlusion	×	√	×
Scale	×	√	√

Task 2: Multi Object Prediction

1. Prediction Method and Visualization Result

1.1 Constant Velocity Model (CVM)

CVM assumes that an object's velocity remains constant over time, leading to linear motion without changes in speed or direction. It is commonly used in trajectory prediction tasks where acceleration is negligible.

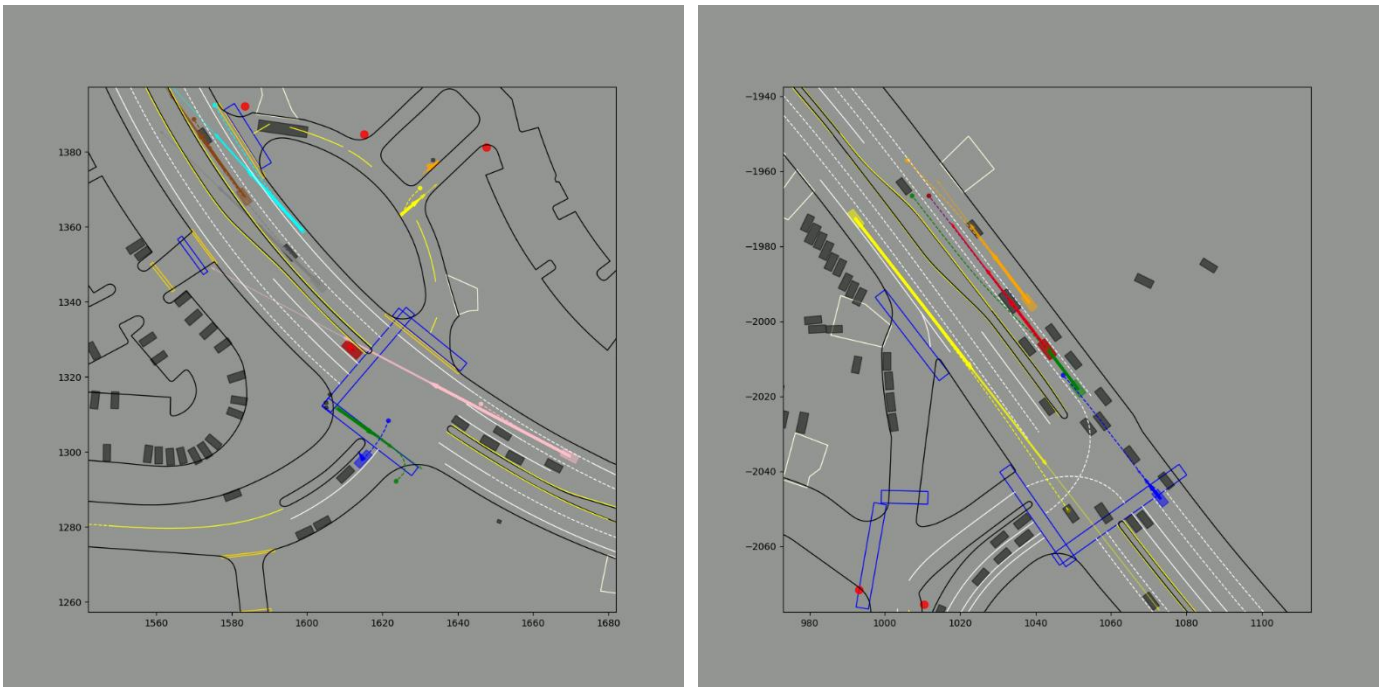


Figure 2. Scenario 1870b2208fe4ade9 (left) and ba25b5b15a9bd8ce (right) CVM visualization results

1.2 Constant Acceleration Model (CAM)

CAM assumes that an object moves with a constant acceleration, resulting in quadratic motion over time. This model captures dynamic changes in velocity and is often applied in scenarios requiring more accurate trajectory prediction.

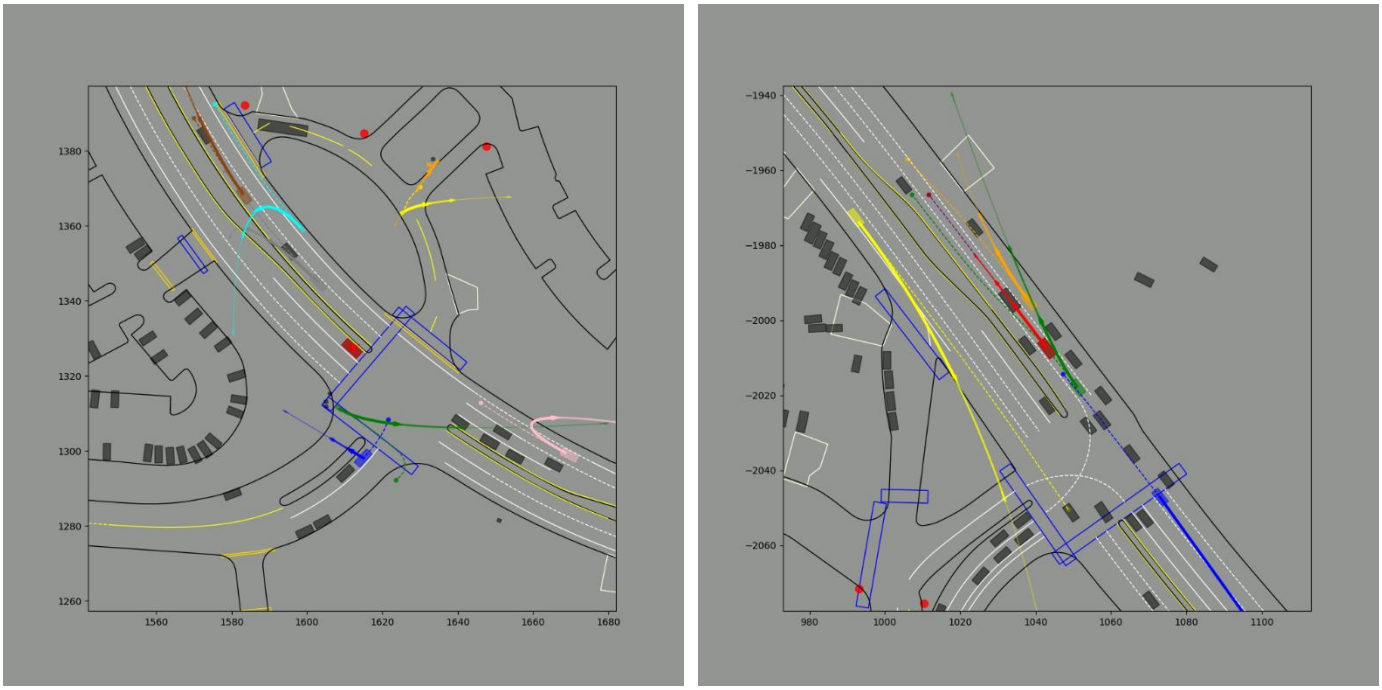


Figure 3. Scenario 1870b2208fe4ade9 (left) and ba25b5b15a9bd8ce (right) CAM visualization results

1.3 Average Displacement Error (ADE)

ADE measures the average Euclidean distance between the predicted trajectory and the ground truth trajectory over all time steps. It quantifies the overall accuracy of the predicted path.

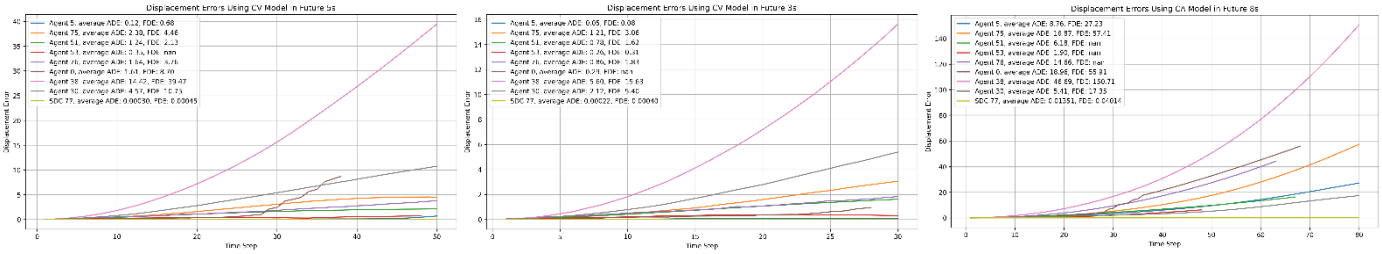


Figure 4. CVM results for scenario 1870b2208fe4ade9

1.4 Final Displacement Error (FDE)

FDE evaluates the Euclidean distance between the predicted final position and the actual final position at the last time step, focusing on the endpoint prediction accuracy.

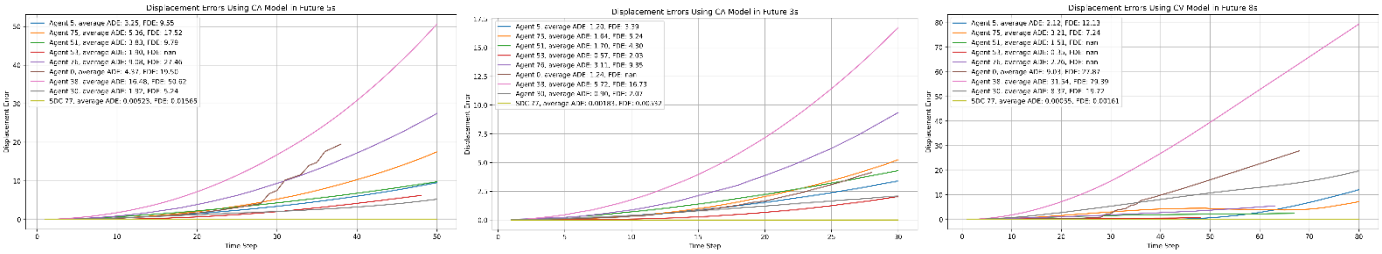


Figure 5. CAM results for scenario 1870b2208fe4ade9

2. Result Comparison

CVM and CAM were evaluated for trajectory prediction over 3, 5, and 8-second horizons using ADE and FDE as performance metrics. The CVM demonstrated reliable accuracy in short-term predictions (3 seconds), particularly in scenarios where agents maintained constant speed and direction. However, its performance declined over longer horizons (5 and 8 seconds), with increasing ADE and FDE due to its inability to account for acceleration, deceleration, or changes in direction.

In contrast, the CAM showed improved accuracy in dynamic scenarios, effectively capturing acceleration and deceleration behaviors. This resulted in lower ADE and FDE, especially in the 5 and 8-second predictions. However, the CAM's reliance on consistent acceleration assumptions made it sensitive to estimation errors, sometimes leading to larger FDE when the actual movement deviated from the model's expectations.

Overall, CVM is more suitable for short-term and linear trajectories, while CAM performs better in dynamic and longer-term predictions. However, the CAM's sensitivity to inaccurate acceleration estimates highlights the need for cautious application.

Table 6. Average ADE and FDE comparison of CVM and CAM

		Tracks (3s)	Tracks (5s)	Tracks (8s)	SDC (3s)	SDC (5s)	SDC (8s)
CVM	ADE	1.65518	4.15308	8.74233	0.915051	2.40125	5.72354
	FDE	4.47453	11.4625	25.693	2.56899	6.76678	15.9423
CAM	ADE	5.26047	14.1938	32.351	0.460007	1.59138	4.90198
	FDE	14.892	41.4711	99.5258	1.52969	5.28921	16.3305

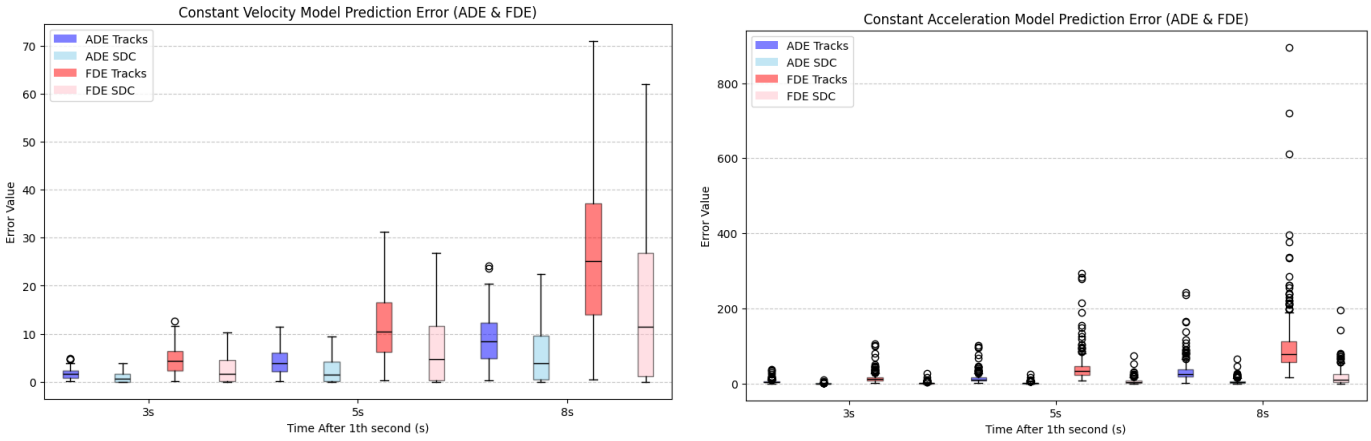


Figure 6. Total comparison for CVM (left) and CAM (right) prediction Error

Task 3: Single Object Tracking in ROS

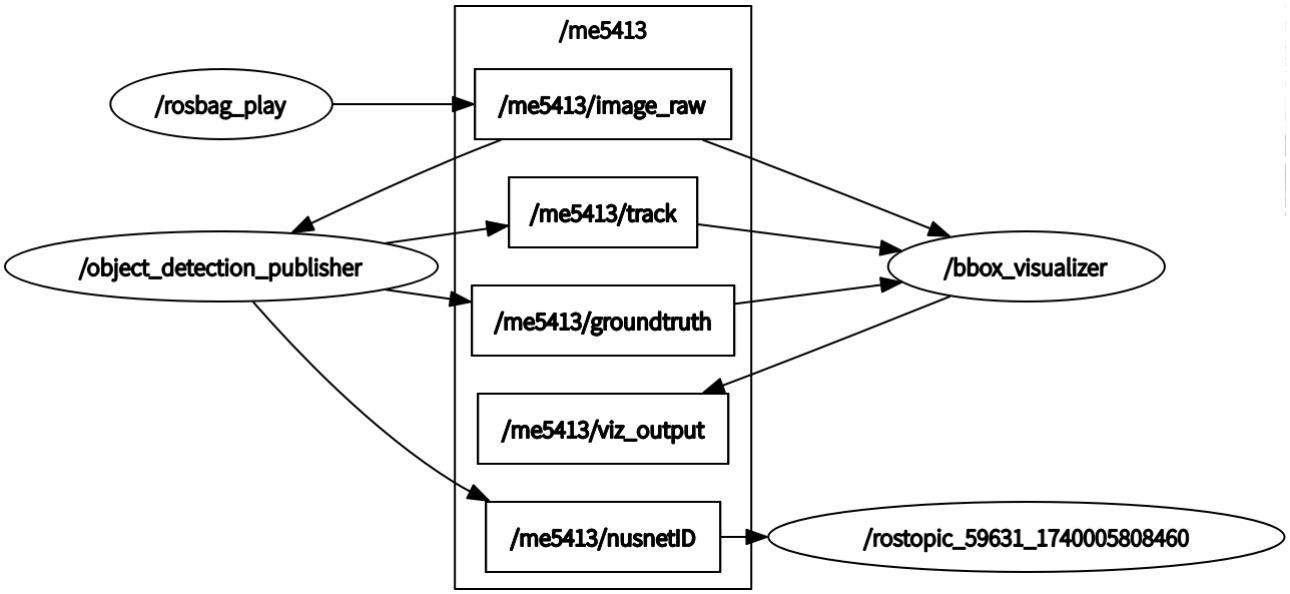


Figure 7. Ros node graph

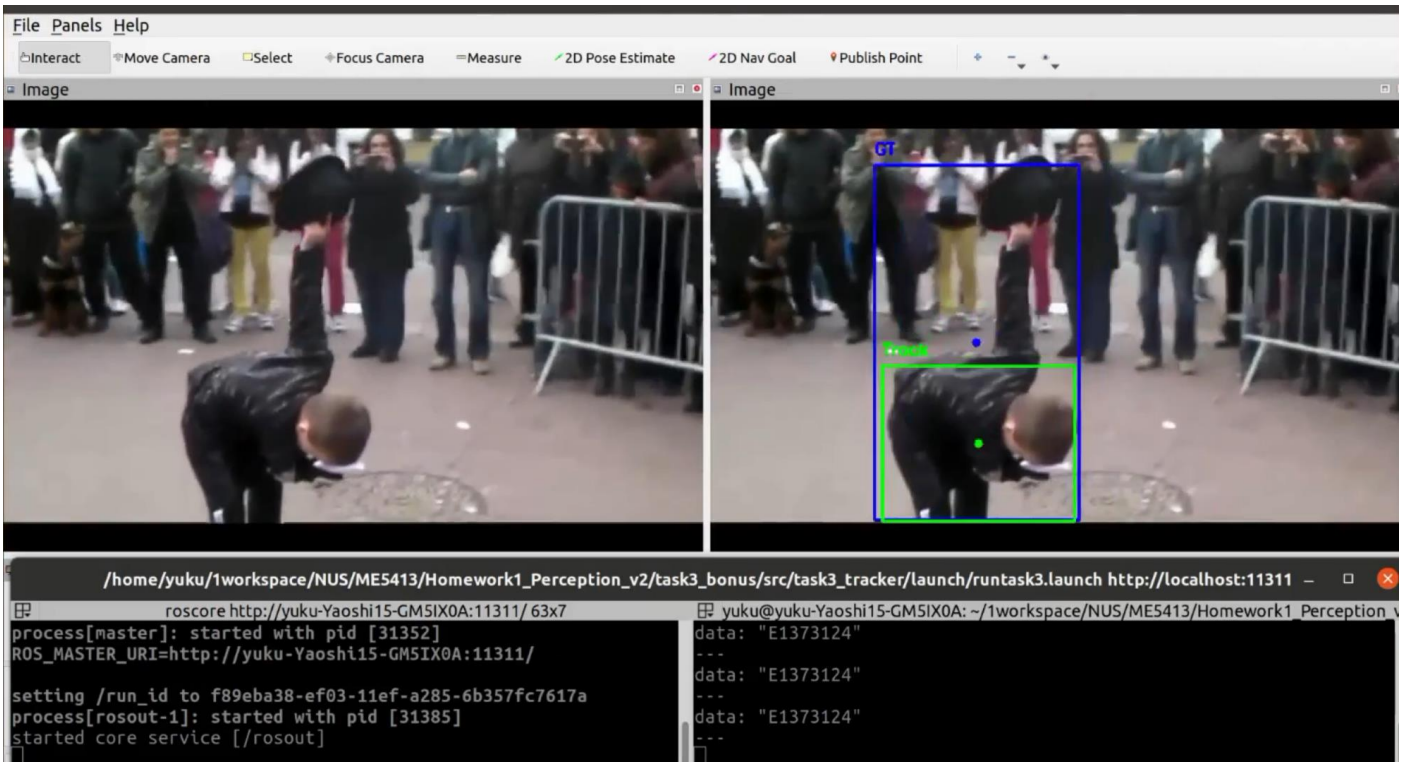


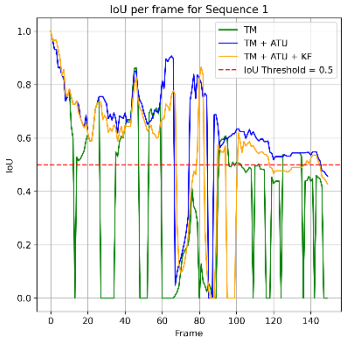
Figure 8. Rviz result.

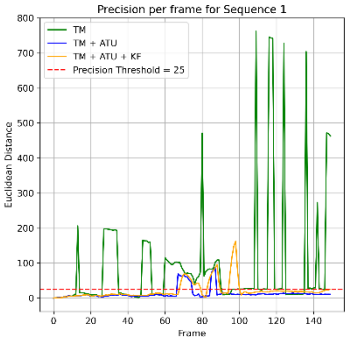
Topic Name	Message Type	Description
/me5413/image_raw	sensor_msgs/Image	Raw images from the ROS bag.
/me5413/groundtruth	vision_msgs/Detection2D	Ground truth bounding boxes.
/me5413/track	vision_msgs/Detection2D	Tracked object bounding box.
/me5413/viz_output	sensor_msgs/Image	Image with overlaid tracking results.
/me5413/nusnetID	std_msgs/String	NUS student net ID.

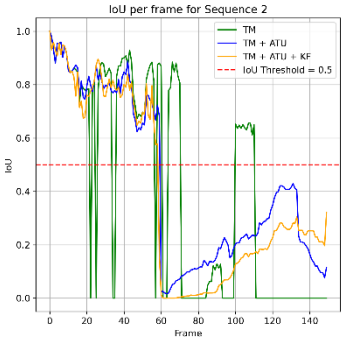
Figure 9. Published Topics

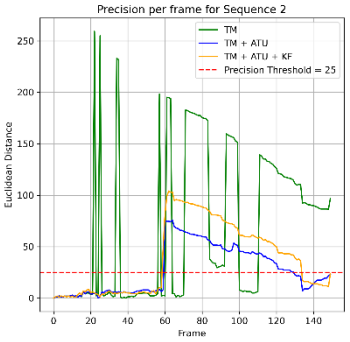
Appendix:

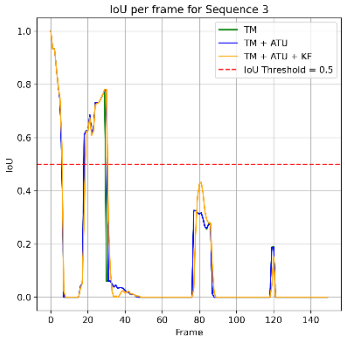
1. Template matching:

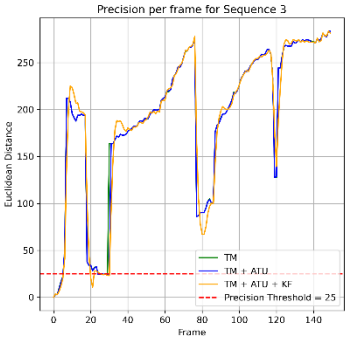


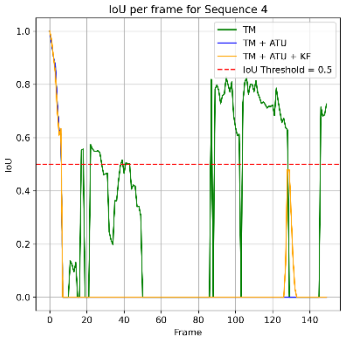


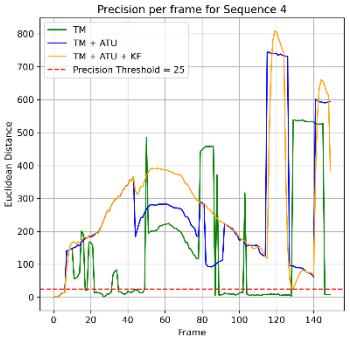


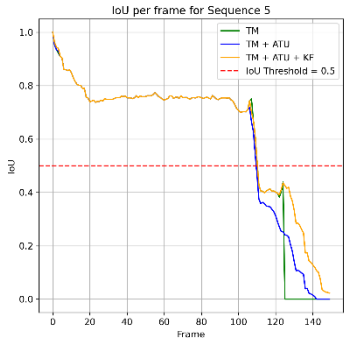


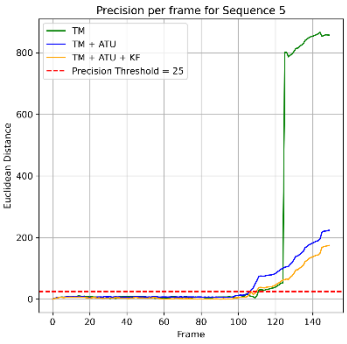




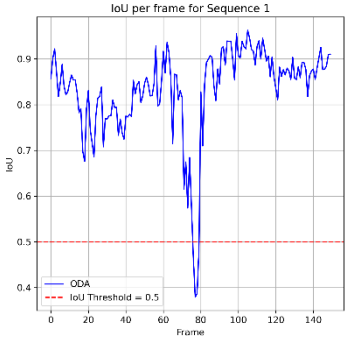


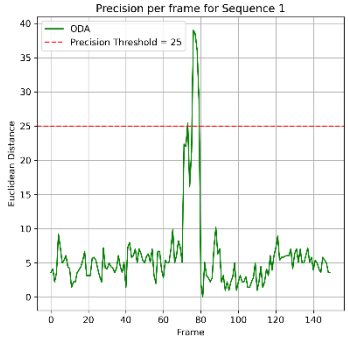


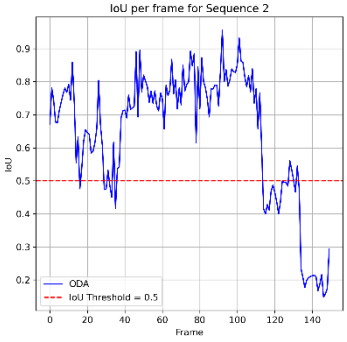


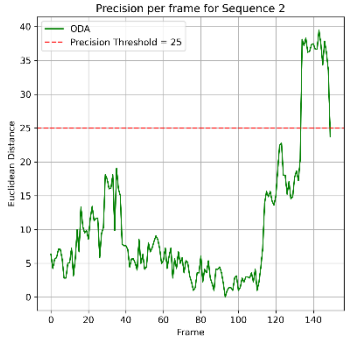


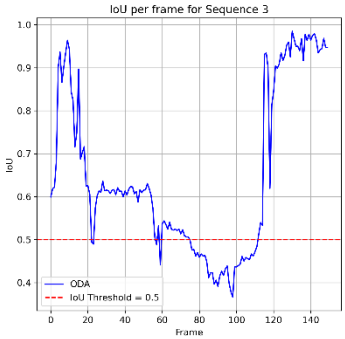
2. Object detection with association

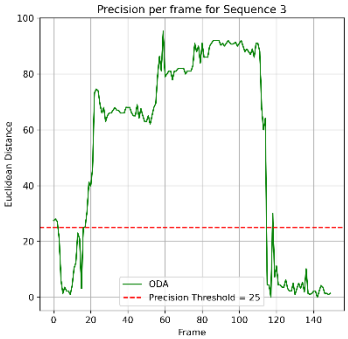


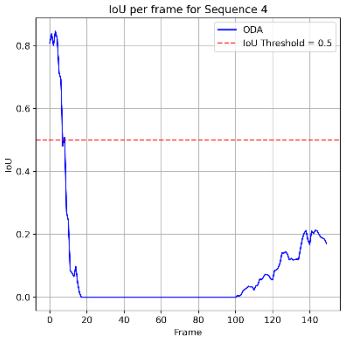


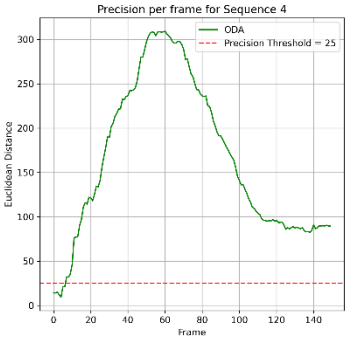


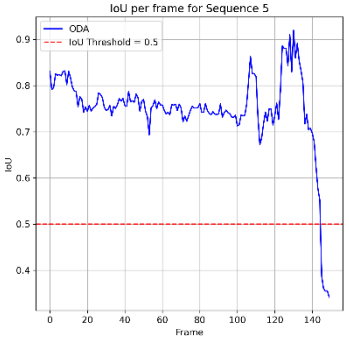


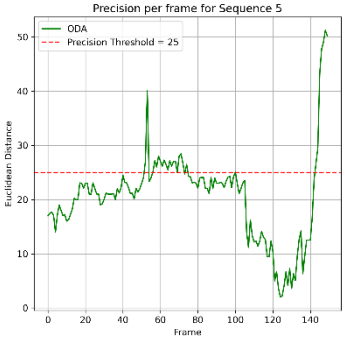












3. YOLOv5 + DeepSORT