ME5413 HomeWork1 Report

Cao Chenyu
e1192847@u.nus.edu
A0285295N

Task 1: Single Object Tracking

Template Matching (Original Approach)

OpenCV offers 6 template matching methods, so I first tested all of them on the given 5 sequences and got the average IoU results shown in Table 1 to determine which method was best suited for a particular sequence.

Seq Index	SQDIFF	SQDIFF NORMED	CCORR	CCORR NORMED	CCOEFF	CCOEFF NORMED
1	0.008	0.019	0	0.474	0.450	**0.501**
2	0.214	**0.426**	0.001	0.227	0.387	0.284
3	0	0	0	**0.419**	0.385	0.337
4	0.043	0.154	0.272	0.176	0.390	**0.398**
5	0	0	0	0.154	**0.446**	0.285

Table 1: Average IoU Results On Different Templates On The Given Sequences

Template Matching with Adjusted Search Region

In the initial approach, I only chose a preferred template but did not restrict the search region, so in this section, I set a search region on a given sequence and adjusted it for different sequences in order to obtain better performance (higher average IoU and lower average distance). Figure 1 shows the end frame of the template matching results and the adjusted search region, with green rectangles for the predictions and blue rectangles for the search region.

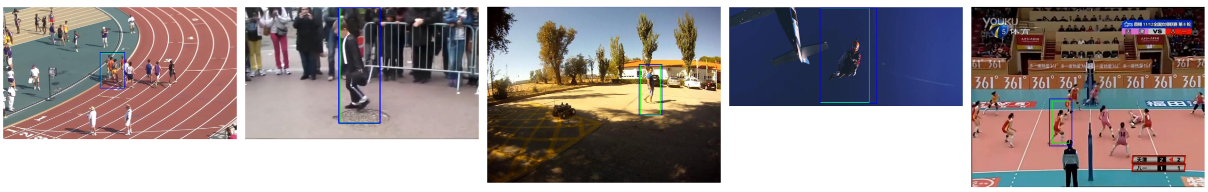

Figure 1: Template Matching With Adjusted Search Region

Kalman Filter Based on Template Matching Measurements

As for the design of the Kalman filter, I set the states of the Kalman filter to x, y, v_x, v_y, which represent the center position and center velocity of the tracked object, respectively, and $x + \frac{w}{2}, y + \frac{h}{2}$ as the measurement values of the Kalman filter. I adjusted the process noise covariance matrix and the measurement noise covariance matrix to improve on all sequences.

Results Analysis

From Table 2, we can that the integration of adjusted search regions and Kalman filtering significantly enhances the robustness and accuracy of template matching in video tracking. These methods may be further applied to real-time tracking systems where precision and computational efficiency are paramount.

Seq Index	TM Method	Average IoU (Success)			Average Distance (Precision)		
		Original	+SR	+SR+KF	Original	+SR	+SR+KF
1	CCOEFF NORMED	0.501	**0.665**	0.662	115	12.0	**11.5**
2	SQDIFF NORMED	0.426	0.613	**0.724**	59.4	26.7	**9.95**
3	CCORR NORMED	0.419	**0.736**	0.726	54.5	**9.26**	9.34
4	CCOEFF NORMED	0.398	**0.464**	0.463	78.6	43.8	**43.6**
5	CCOEFF	0.446	0.692	**0.698**	90.2	18.2	**18.1**

Table 2: Template Matching Results

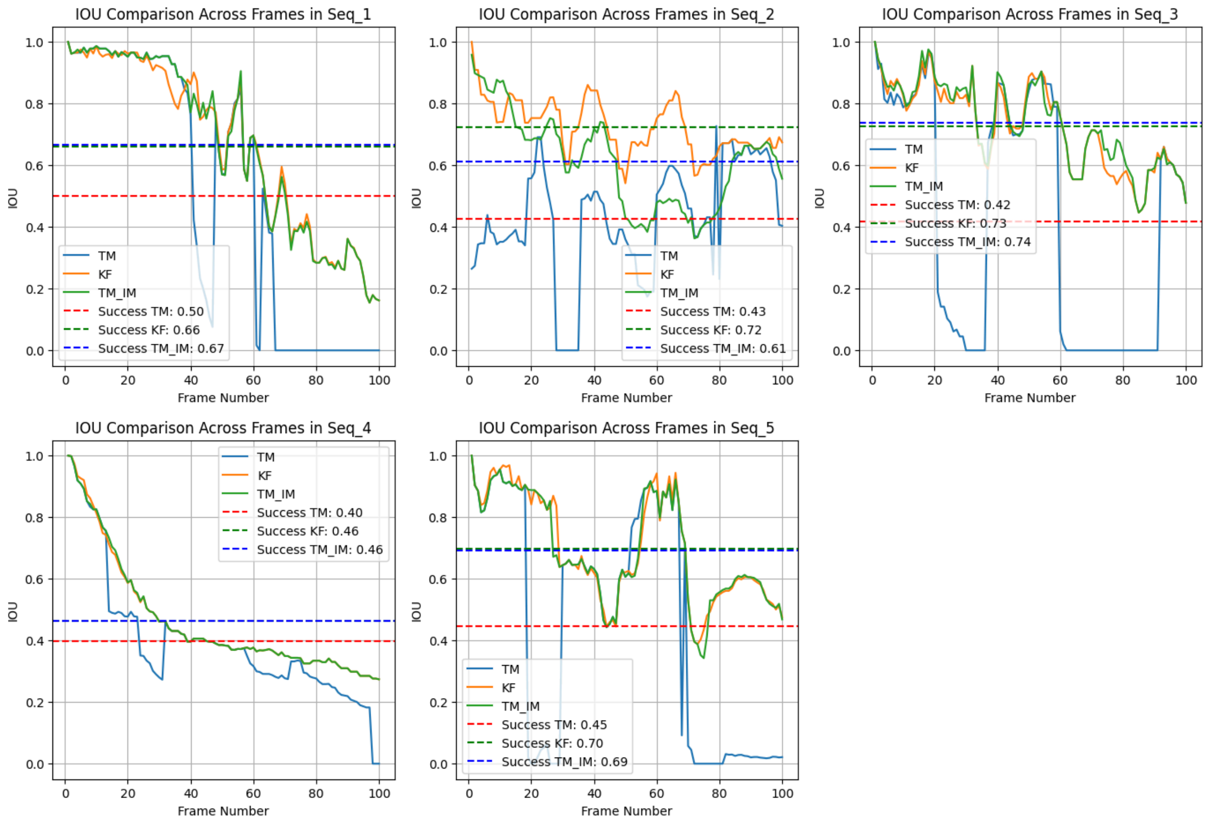

Figure 2: IoU (Success) Comparison Results Across Frames

Figures referenced as Figure 2 and Figure 3 present a comparative analysis of Success and Precision metrics across frames for the five sequences studied.

- The implementation of an adjusted search region has proven to be significantly effective. This strategy not only mitigates drifting issues but also economizes on computation time by confining the matching process to a smaller, current window. Consequently, this focused approach ensures that template matching is not adversely influenced by extraneous objects with similar appearances that are not the intended targets of detection.
- Moreover, incorporating Kalman filtering has incrementally enhanced the accuracy of target tracking. This enhancement is particularly noticeable in sequences characterized by regular motion patterns, such as in 'seq 2'. By integrating velocity into the prediction equations, Kalman filtering extends the assessment criteria beyond the immediate template matching results to include the dynamic parameters of the target. This addition allows subsequent frame predictions to be informed by both the spatial measurements and the velocity trends observed in preceding frames, thus yielding a more coherent and continuous tracking trajectory

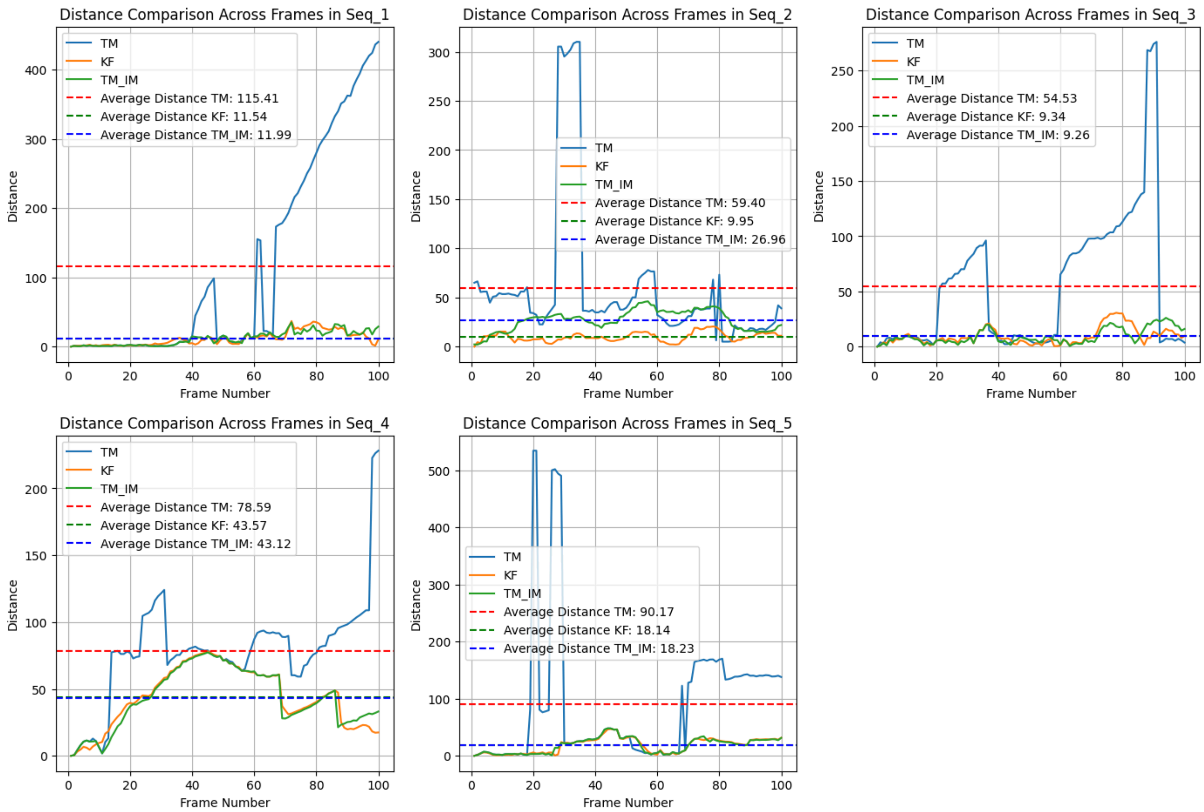

Figure 3: Distance (Precision) Comparison Across Frames

Task 2: Multi Object Prediction

Evaluation Metrics for Multi-Object Prediction

- Average Displacement Error (ADE): This metric quantifies the prediction accuracy by calculating the mean Euclidean distance (L2 norm) between the predicted and the actual positions of multiple objects across all evaluated time steps.
- Final Displacement Error (FDE): This metric measures the prediction precision at the most critical juncture—the final time step. It computes the Euclidean distance (L2 norm) between the predicted and the actual positions of the objects at this concluding time step."

Using constant velocity model

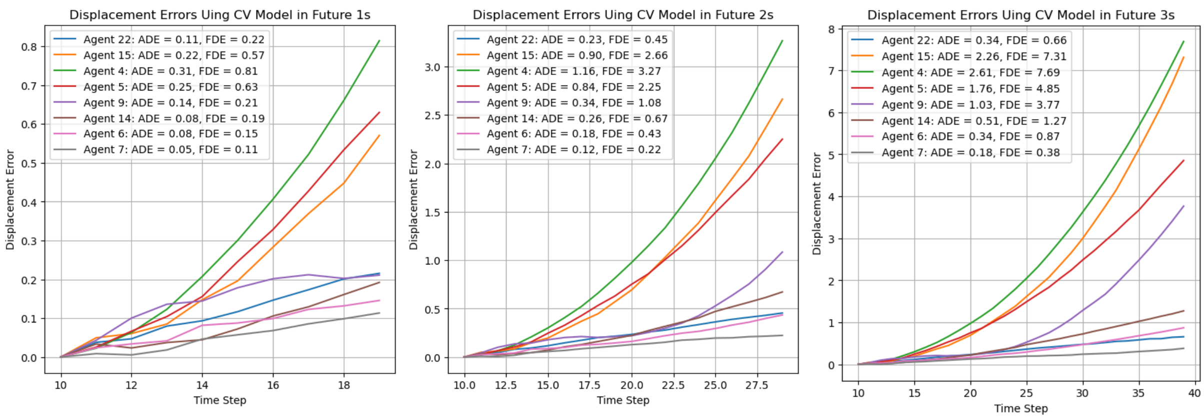

Figure 4: Displacement Errors of the predicted future 1s, 2s, and 3s trajectories using CV model

Figure 4 illustrates the displacement errors encountered when employing a constant velocity (CV) model to predict future trajectories of multiple agents over 1-second, 2-second, and 3-second time horizons. The CV model assumes that each agent maintains a steady speed and direction of travel throughout the prediction period.

Using constant acceleration model

Figure 5 demonstrates the displacement errors resulting from the application of a constant acceleration (CA) model for predicting the future 1-second, 2-second, and 3-second trajectories of various agents. The CA model presupposes that each agent's acceleration remains unchanged throughout the prediction interval. This model considers both the velocity and the acceleration of the agent at the last observed time step to forecast future positions, thereby accounting for changes in speed over time.

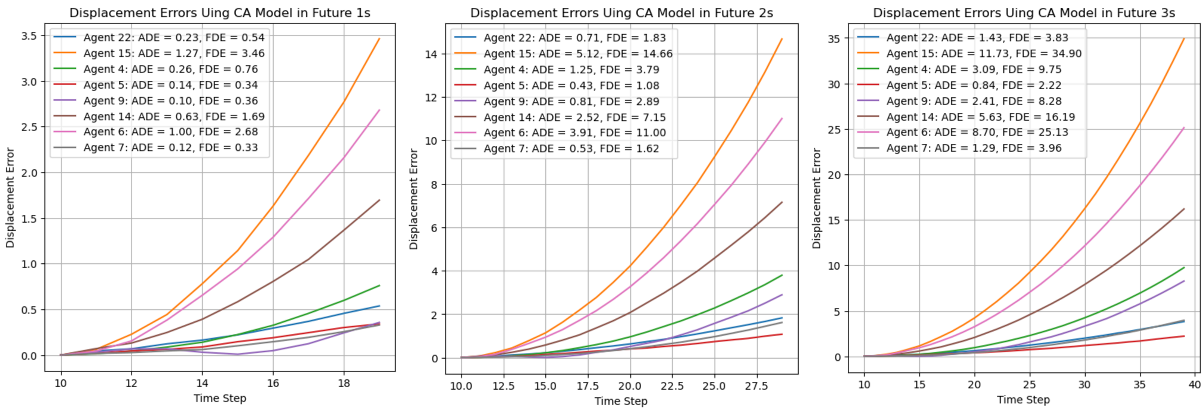

Figure 5: Displacement Errors of the predicted future 1s, 2s, and 3s trajectories using CA model

Using Constant Turn Rate and Velocity (CTRV) model

Figure 6 demonstrates the displacement errors resulting from the application of a constant acceleration (CA) model for predicting the future 1-second, 2-second, and 3-second trajectories of various agents. The CA model presupposes that each agent's acceleration remains unchanged throughout the prediction interval. This model considers both the velocity and the acceleration of the agent at the last observed time step to forecast future positions, thereby accounting for changes in speed over time.

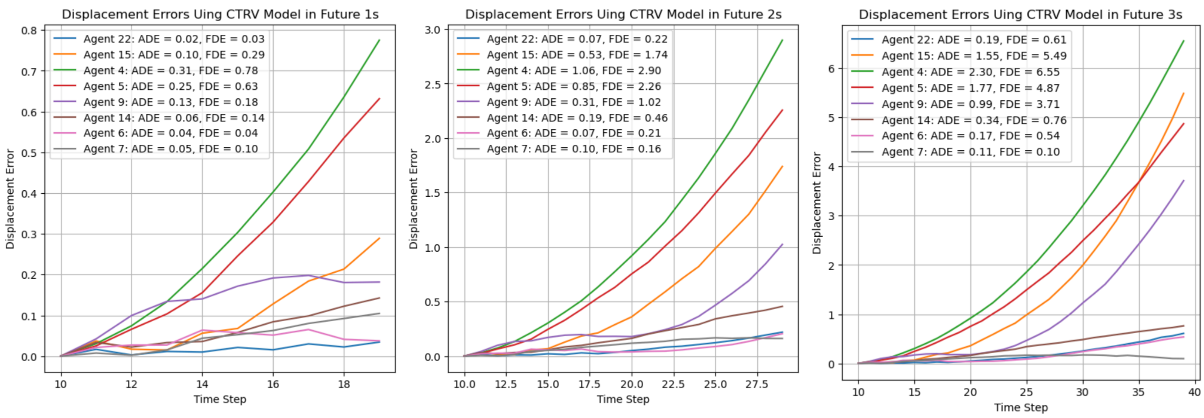

Figure 6: Displacement Errors of the predicted future 1s, 2s, and 3s trajectories using CTRV model

Roadmap Visualization

Figure 7 showcases a comprehensive roadmap visualization, contrasting the predicted trajectories with the actual paths (ground truth) for multiple agents over a future 3-second time frame. The visualizations compare the trajectories forecasted by three different models: constant velocity (CV), constant acceleration (CA), and constant turn rate and velocity (CTRV).

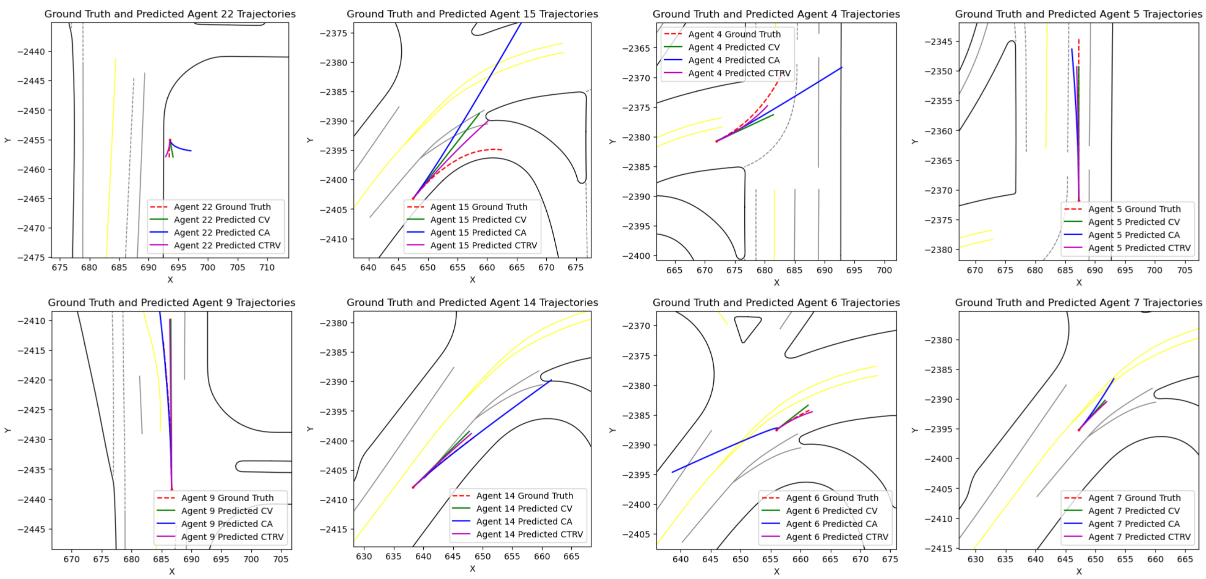

Figure 7: roadmap visualization in the future 3s

Results Analysis

In Table 3, the Constant Turn Rate and Velocity (CTRV) model consistently exhibits the lowest Average Displacement Error (ADE) and Final Displacement Error (FDE) across all time frames, indicating superior predictive accuracy. The Constant Velocity (CV) model performs well in the short term but its accuracy declines over longer intervals. The Constant Acceleration (CA) model, while intuitive, results in higher errors, suggesting less suitability for these prediction scenarios. Overall, the CTRV model's strong performance suggests it is the most reliable for predicting agent trajectories in this context.

Model	ADE			FDE		
	1s	2s	3s	1s	2s	3s
CV	0.156	0.504	1.130	0.362	1.381	3.350
CA	0.470	1.909	4.391	1.267	5.502	13.031
CTRV	0.120	0.397	0.927	0.275	1.121	2.829

Table 3: ADE and FDE in the future 1s, 2s, and 3s using different models

Analyzing the results in conjunction with Table 3 and Figure 7, it is evident that the CTRV model demonstrates enhanced robustness and predictive accuracy over the CV and CA models.

- Agents 7, 6, and 22 show the most accurate trajectory predictions with the lowest ADE and FDE when using the CTRV model. This suggests that incorporating turn rate into the model is crucial for capturing the nuances of their movements.
- Agent 5's movement is best predicted by the CA model, indicating a consistent acceleration pattern. However, the high ADE and FDE values for Agent 15 and 6 with the CA model may indicate erratic behavior or a sudden change in dynamics, which the model fails to capture accurately.
- The predictive success of the CTRV model for Agent 7, 6, and 22 underscores its suitability for scenarios involving angular velocity components, affirming its application in complex dynamic systems where direction and speed are variable.

Bonus Task

In Bonus Task, I am required to apply the best-matched method in Task 1, and publish the tracked object, the ground truth objects, and my matric number. The ros graph is shown in Figure 8, which includes the following nodes:

- `/ground_truth_seq_1`
- `/ground_truth_seq_2`
- `/matric_number`
- `/detection_seq_1`
- `/detection_seq_2`
- `/track_pub_p`
- `/track_sub_p`

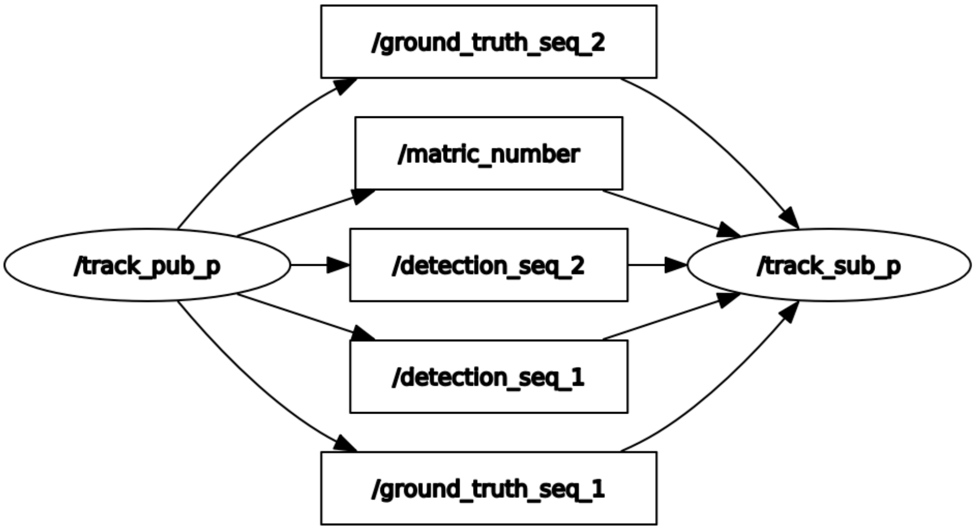

Figure 8: Ros Graph in Bonus Task

I have completed the Publisher and Subsriber files in Python:

- `track_pub_p.py`: This script sets up ROS publishers, processes image data from the bags, and utilizes the Kalman filter to predict and correct object positions, finally publishing the results to specific ROS topics.
- `track_sub_p.py`: This script implements a subscriber node named 'track_sub_p' that listens to the required topics. Then it uses threading to handle incoming messages safely and records them into a ROS bag file.