1. Introduction

The accurate tracking of a vehicle or robot in the field plays an important role in agricultural automation. Currently, the Global Navigation Satellite System (GNSS) is a primary source for achieving such accuracy. However, noise, such as the loss of GNSS signals and the drafting of sensor data due to unpredictable environments, can affect the tracking accuracy. Sensor fusion aims to collaborate with different sensors in a vehicle to reduce the influence of noise. One of the most popular methods is the Kalman Filter (KF). It can estimate true values by analysis of the prediction on the system model and measurement on sensors. In this report, we evaluate the Kalman filter to improve the accuracy of location tracking for an autonomous weed-spraying vehicle developed by the JCU spinout company Autoweed.

2. Literature Review

Kalman filtering techniques have been extensively employed in a multitude of applications. Kalman presented state estimation and prediction issues, and simultaneously suggested a technique for resolving these common issues when the system model is linear [1]. In line with the specified conditions, the standard Kalman filter aims to attain maximum likelihood estimation and minimum mean squared errors. However, this is only applicable when the system model is linear, and the process and measurement noises are zero mean and completely unrelated [2]. Stanley further improve it to address filtering and prediction problems of non-linear systems name as the Extended Kalman filter (EKF) [3]. This algorithm was implemented in the Apollo program in the 1960s. The EKF addresses the issue of non-linear problems by transforming the known system model into a linear one. Nonetheless, there are noise and relationships within the system remain unknown. Therefore, it is essential to define optimal covariance matrices.

Two critical covariance matrices that significantly influence the system modelling are the process and measurement covariance matrices. The optimisation of process covariance matrix and measurement covariance can enhance the performance of the

Kalman filter [4]. In recent times, novel approaches have been proposed to estimate these values more systematically. Feng introduced the Recursive Covariance Estimation algorithm (KF-RCE), which aims to precisely determine the process noise covariance matrix [5]. This method can improve the performance of the Kalman filter provided that certain assumptions are met. The work of Akhlaghi and colleagues proposes an additive approach to estimate both process and measurement covariance. In their methodology, these covariance are dynamic and are updated in each step based on the residual covariance [6]. The author provides two case studies that demonstrate the significant enhancement in the performance of the conventional Extended Kalman Filter.

In recent years, the increased computing power has enabled us to utilize grid samples to efficiently search for the optimal values of process and measurement covariance. Matisko and teams use auto-correction function to tune these covariance extensively to find optimal matrices [7]. Furthermore, they employed the Bayesian method, which applied the Bayesian formula to comprehend the posterior probability distribution of new measurements, with the aim of selecting a predefined set of process or measurement covariance based on previous data [8]. A more recent approach combines cooperative neural networks and the Kalman filter to estimate the location of an individual when GNSS is not functioning indoors [9]. The author employed a deep neural network to estimate locations, position covariance, orientation covariance, and other sensor noises, while simultaneously applying the Kalman filter to estimate the location. This method provides robust estimation when the person is in an environment where the GNSS system is denied.

As a result, to design an optimal Kalman filter, we can focus on tuning the process and measurement covariance. If we understand the system process, we can obtain better process covariance, and if we have more measurement data in different environments available, we can obtain better measurement covariance. In this project, as a limitation of the data, we utilised a more straightforward tuning method which is similar to the method in [7]. However, we applied it in a simpler manner by adjusting each state parameter to obtain the optimal process and measurement covariance.

3. Methodology

3.1 Data

Seven recordings from same sensors were included in this project. As the Recording 7 was lost GNSS data completely, so we only will only six recordings to design our Kalman filter. And then use the Kalman Filter to estimate Recording 7. Table 1 lists the variables we used in this project, and Table 2 lists the number of observations available for our tuning process.

Table 1: variable information

Variable	Description
Eastings	relative location data (m)
Northings	relative location data (m)
Speed	vehicle moving speed (m/s)
Heading	Angle of heading, 0 represent to the north (rad)
Angular velocity z	velocity z & Change of heading (rad/s)
Time	time for each successful Recording (s)

Table 2: data key information

Data	Number of Observations	Time(s)
Recording 1	4266	213
Recording 2	1089	54
Recording 3	2762	138
Recording 4	1201	60
Recording 5	1199	60
Recording 6	1200	60

Therefore, we have approximately 11700 observation or 600s of measurement data available to support us to the design of Kalman Filter in this application. The entire data analysis process used R studio 4.3.2 [10].

3.2 Kalman Filter Design

The steps for designing an optimal Kalman filter are illustrated Figure 1. First, a system model is built to identify the prediction function. Second, this prediction function is established to calculate the position of the vehicle. Third, compute residuals between the prediction and measurement. Finally, update the state vector and its covariance P. These steps will be repeated in each Recording examination. Furthermore, we will tune the process covariance and measurement covariance to achieve an optimal minimum mean square error (MSE) among all the Recordings.

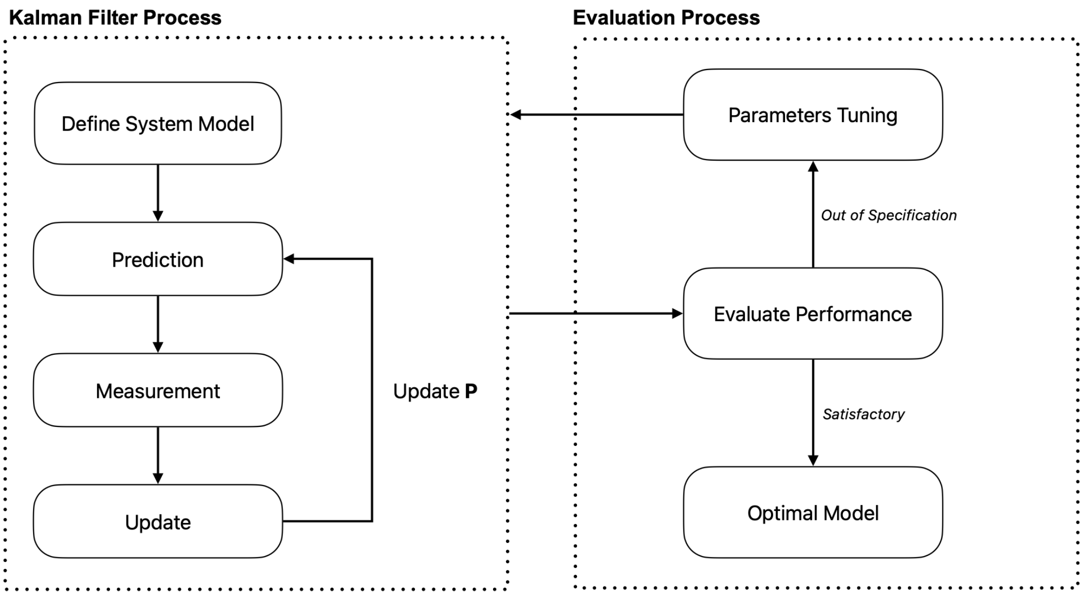

Figure 1: Kalman Filter Design Diagram

3.3 Prediction Step

First, it is essential to comprehend the state variables. This study focused on five variables. In this initial phase, the aim is to establish the relationships between each state variable. The state variable vector is given by (1).

$$\boldsymbol{x} = \begin{bmatrix} eastings \\ northings \\ speed(m/s) \\ heading(rad) \\ changeofheading(rad/s) \end{bmatrix} \quad (1)$$

From an understanding of the model, we know that these relationships are nonlinear in nature. When dealing with nonlinear relationships, we can employ the Extended Kalman

Filter (EKF) or Unscented Kalman Filter (UKF). As this report's primary objective is not to determine the superiority of one method over the other, we opted to utilise the EKF. However, additional studies can be conducted to compare the precision of the EKF and the UKF in this specific application. In EKF, the prediction function is given by (2).

$$\boldsymbol{x}_{k|k-1} = f(\boldsymbol{x}_{k-1}) + b(\boldsymbol{u}_k) \tag{2}$$

$\boldsymbol{x}_k$ is the prediction value, and $b(\boldsymbol{u}_k)$ is a function that calculates the effect of the control inputs. First, we assume that the prediction is not affected by control input, so $b(\boldsymbol{u}_k) = 0$.

We noted the time of each measurement as Δt, speed as v, heading as α, change of heading as $\dot{\alpha}$. In this step, we assume that the change of speed is zero within Δt. Therefore, we establish a state prediction vector according to mathematical relationship as (3) shows.

$$\boldsymbol{x}_{k|k-1} = \begin{bmatrix} eastings + vsin(\alpha + \Delta t\dot{\alpha})\Delta t \\ northings + vcos(\alpha + \Delta t\dot{\alpha})\Delta t \\ v \\ \alpha + \Delta t\dot{\alpha} \\ \dot{\alpha} \end{bmatrix} \tag{3}$$

The next state of covariance of the state vector $P_{k|k-1}$ is (4).

$$P_{k|k-1} = F_k P_{k-1} F_k^T + Q_k \tag{4}$$

Q_k is the process covariance. The prediction matrix F_k is the Jacobean of $\boldsymbol{x}_{k|k-1}$respect to $\boldsymbol{x}$. we use symbolic calculation package *Ryacas [11]* in R studio get the result as shown in (5).

$$F_k = \begin{bmatrix} 1 & 0 & \Delta tsin(\alpha + \Delta t\dot{\alpha}) & v\Delta tcos(\alpha + \Delta t\dot{\alpha}) & v\Delta t^2 cos(\alpha + \Delta t\dot{\alpha}) \\ 0 & 1 & \Delta tcos(\alpha + \Delta t\dot{\alpha}) & -v\Delta tsin(\alpha + \Delta t\dot{\alpha}) & -v\Delta t^2 sin(\alpha + \Delta t\dot{\alpha}) \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

Finally, we initialise the process covariance Q_k, as we assumed that each sensor is independent and that these sensors are only affected by white noise. Specifically, in this design, Δt is different in every iteration, thus Q_k is dynamic.

$$Q_k = \begin{bmatrix} \Delta t^2 & 0 & 0 & 0 & 0 \\ 0 & \Delta t^2 & 0 & 0 & 0 \\ 0 & 0 & \Delta t^2 & 0 & 0 \\ 0 & 0 & 0 & \Delta t^2 & 0 \\ 0 & 0 & 0 & 0 & \Delta t^2 \end{bmatrix} \quad (6)$$

In the first iteration, there is not any information for prediction, so P start with a 5 × 5 zero matrix as shown in (7).

$$P = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (7)$$

3.4 Measurement Step

In this step, the measurement was directly connected to each state variable. Therefore, this step is linear. In addition, during the measurement, GNSS data lost after a certain time. Thus, two different measurement matrices are defined, as shown in (8) (9).

When GNSS working

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (8)$$

When GNSS is not working

$$H_k = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

We note the measurement vector in state k as $\boldsymbol{z}_k$, and this prediction vector is calculated from (10).

$$\boldsymbol{z}_{predict} = H\boldsymbol{x}_{k|k-1} \quad (10)$$

So, the residual between measurement and prediction $\boldsymbol{y}_k$ and the covariance in the residual S_k as shown in (11) (12)

$$\boldsymbol{y}_k = \boldsymbol{z}_k - \boldsymbol{z}_{predict} \quad (11)$$

$$S_k = H_k P_{k-1} H_k^T + R_k \quad (12)$$

R_k denotes the measurement covariance. Initially, we assumed all measurements are independent; similar to H, it is different between GNSS works and fails. We define R_k

in (13) and (14). The standard deviation of each state variable was calculated from the time *Recording 1* when the vehicle was not moving. It is worth noting that, this is only a rough assignment, we will tune it to a better matrix in the tuning process.

When GNSS working

$$R_k = \begin{bmatrix} \sigma_{eas} & 0 & 0 & 0 & 0 \\ 0 & \sigma_{nor} & 0 & 0 & 0 \\ 0 & 0 & \sigma_v & 0 & 0 \\ 0 & 0 & 0 & \sigma_\alpha & 0 \\ 0 & 0 & 0 & 0 & \sigma_{\dot{\alpha}} \end{bmatrix} \tag{13}$$

When GNSS is not working

$$R_k = \begin{bmatrix} \sigma_v & 0 & 0 \\ 0 & \sigma_\alpha & 0 \\ 0 & 0 & \sigma_{\dot{\alpha}} \end{bmatrix} \tag{14}$$

3.5 Update Step

In this step, we first calculate the Kalman gain according to the result from previous steps, the equation is (15).

$$K_k = P_{k|k-1} H_k^T S_k^{-1} \tag{15}$$

Finally, we update the new measurement using Kalman gain as shown in (16) (17), I is a 5x5 Identity matrix.

$$\boldsymbol{x}_k = \boldsymbol{x}_{k|k-1} + K_k \boldsymbol{y}_k \tag{16}$$

$$P_k = (I - K_k H_K) P_{k|k-1} \tag{17}$$

3.6 Tuning

Previous studies showed that well selection and tuning on Q and R can improve the performance of the Kalman Filter. In this project, we used a simple tuning method which involves adjusting only one parameter at a time to observe the change in the mean square error from all six Recordings. Additionally, we further investigate the change in the speed and change in heading, which allows us to understand the main domination of the process noise so that we can determine an optimal process covariance matrix Q.

4. Result and Discussion

The result of using the Kalman Filter to fix the issue when GNSS was not working using the initial Q, and R is shown in Figure 2, 3 and 4. We can observe that estimations and true measurement lie almost together. It is difficult to determine the accuracy of this estimation. Therefore, we continued to investigate errors between true measurement and estimation as shown in Figure 5.

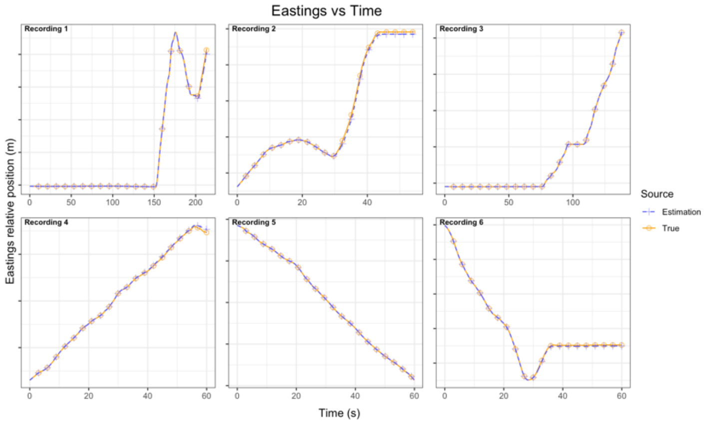

Figure 2: Kalman filter estimation and true measurement in Eastings. Only Recording 2, 4, 6 show. a slight drift in the end.

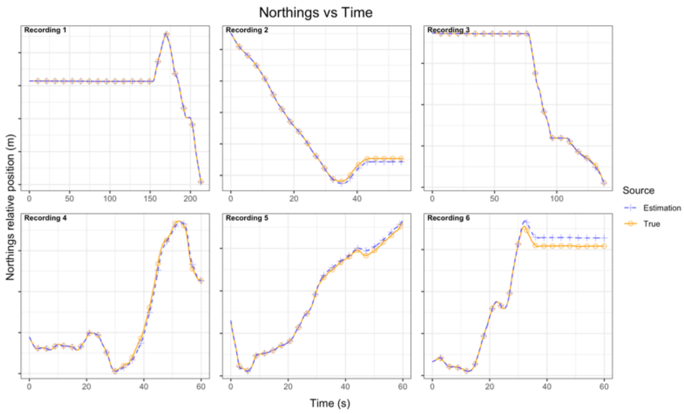

Figure 3: Kalman filter estimation and true measurement in Northings. It clearly shows that there is drift in Recording 3, 5, 6. This is more significant in Recording 6.

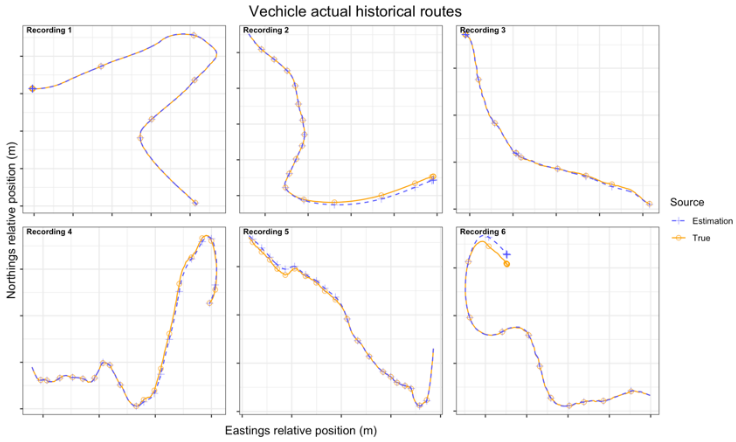

Figure 4: Kalman filter estimation and true measurement historical route. Recording 2 is underestimate the northings position while Recording 6 is overestimate. Others show good estimation.

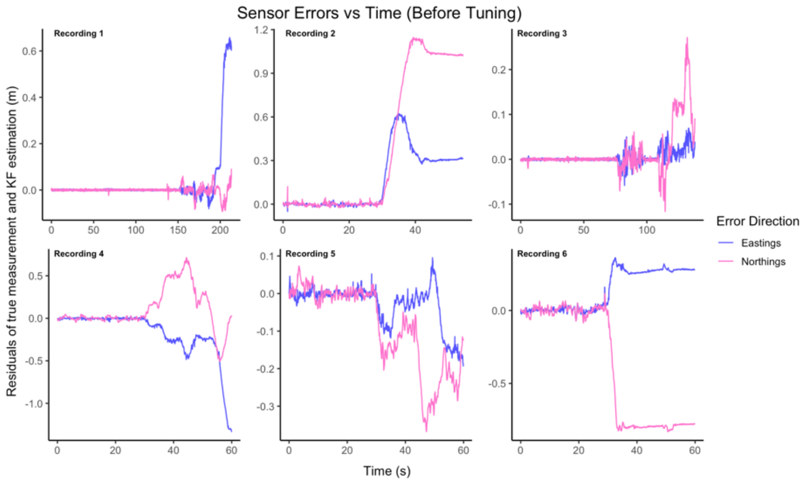

Figure 5: Error of each direction compared to true measurement before tuning. we can clearly see that error increase rapidly in some specific time such as around 180-200s in Recording 1, 30-40s in Recording 2 and 30-35s in Recording 6. Some of this rapid increase occurred only in one direction (northing or easting). We suspected that the change in heading and speed might be the reason for this rapid increase.

As Figure 6 and 7 show, either a quick change in heading or a quick change in speed results in a rapid increase in error. This increase was more significant when both the changes occurred. For example, in *Recording 2* at approximately 30-40s, there was a rapid increase and decrease in speed, and heading turn 2 rads in 10s toward the north. At the same time, the error accumulated substantially in this 10s, especially in the northern position. Similarly, *Recording 6* also increased the error when the vehicle turned from south to north. This pattern helps us to redefine the process covariance.

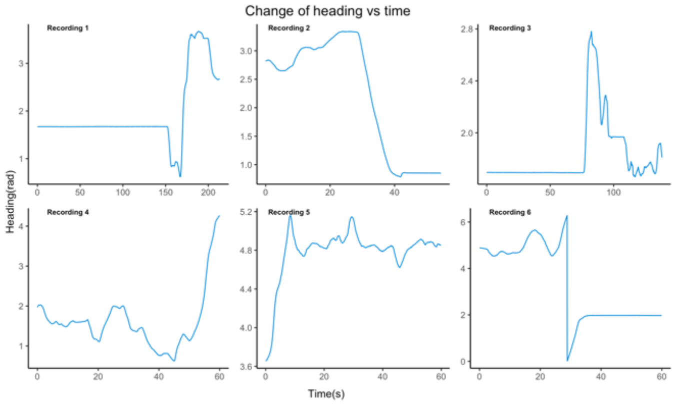

Figure 6: Change of heading respect to time, Recording 6 shows more significant change which from rad 6 to rad 0 within few seconds.

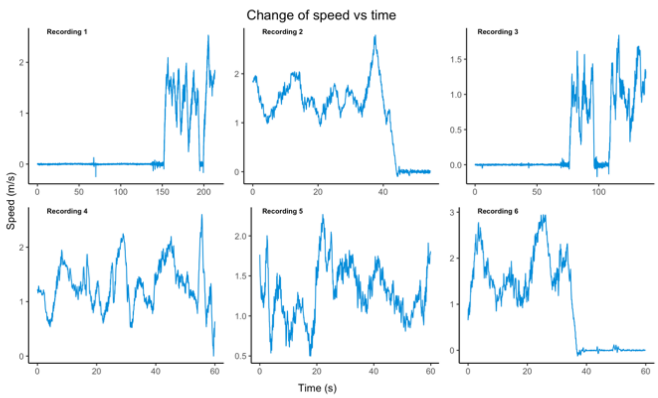

Figure 7: Change of speed respect to time, Recording 1 has large fluctuation in the last few seconds, both Recording 2 and Recording 6 encounter a quick drop in speed.

During the tuning process, we found that the adjusted R_k only resulted in a small improvement in performance, but the change in Q provided a large improvement in the overall estimation. The new process covariance considers the noise between the heading and the change in heading, and the change in speed will affect all state variables. This might be reasonable because when a vehicle accelerates, vibrations create noise in the system.

Final model of R as shown in (18) (19).

When GNSS is working

$$R_k = \begin{bmatrix} 0.009 & 0 & 0 & 0 & 0 \\ 0 & 0.00142 & 0 & 0 & 0 \\ 0 & 0 & 0.007604688 & 0 & 0 \\ 0 & 0 & 0 & 0.000632 & 0 \\ 0 & 0 & 0 & 0 & 0.001059695 \end{bmatrix} \quad (18)$$

When GNSS is not working

$$R_k = \begin{bmatrix} 0.007604688 & 0 & 0 \\ 0 & 0.000632 & 0 \\ 0 & 0 & 0.001059695 \end{bmatrix} \tag{19}$$

Final model of Q as shown in (20), standard deviation of acceleration σ_a calculated from the Recording 3 when car was not moving.

$$Q = \begin{bmatrix} \Delta t^2 + 0.05 & 0 & 0 & 0 & 0 \\ 0 & \Delta t^2 + 0.04 & 0 & 0 & 0 \\ 0 & 0 & \Delta t^2 + 0.015 & 0 & 0 \\ 0 & 0 & 0 & \Delta t^2 & \Delta t^2 \\ 0 & 0 & 0 & \Delta t^2 & \Delta t^2 \end{bmatrix} (\sigma_a - 0.044)^2 \tag{20}$$

Figure 8 shows the error after tuning, which reveals a substantial reduction in the error values. Table 3 provides the mean squared error (MSE) for each Recording both before and after tuning.

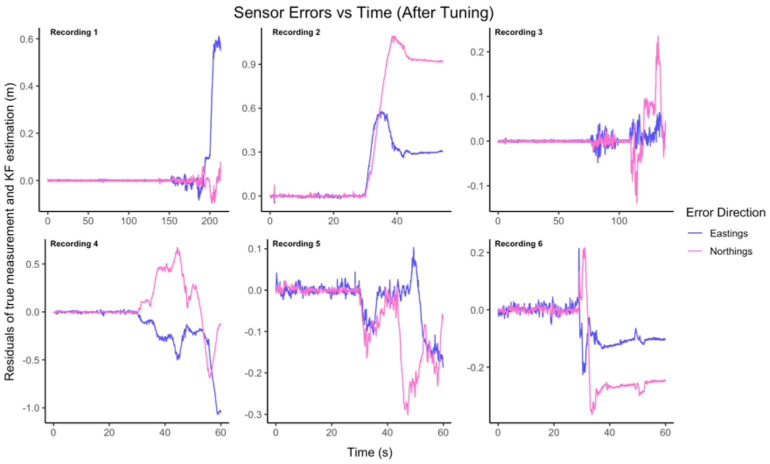

Figure 8: Error visualisation after tuning. Compare with Figure 5, all error span decreased, especially in Recording 6.

Table 3: Mean squared error before and after tuning on both directions.

Data	***Eastings (Before)***	***Eastings (After)***	***Northings (Before)***	***Northings (After)***
Recording 1	0.0185696763	0.0166320938	0.0002979653	0.0002211437
Recording 2	0.0671087769	0.0589258996	0.3942956086	0.3347245916
Recording 3	0.0002464789	0.0001867069	0.0029200967	0.0017390442
Recording 4	0.1085744049	0.0766687344	0.0702553888	0.0677975411
Recording 5	0.0038730502	0.0034412946	0.0203193837	0.0111660864
Recording 6	0.0382391678	0.0067106611	0.2925055482	0.0345793989

In the end, we applied the optimal model on Recording 7 as shown in

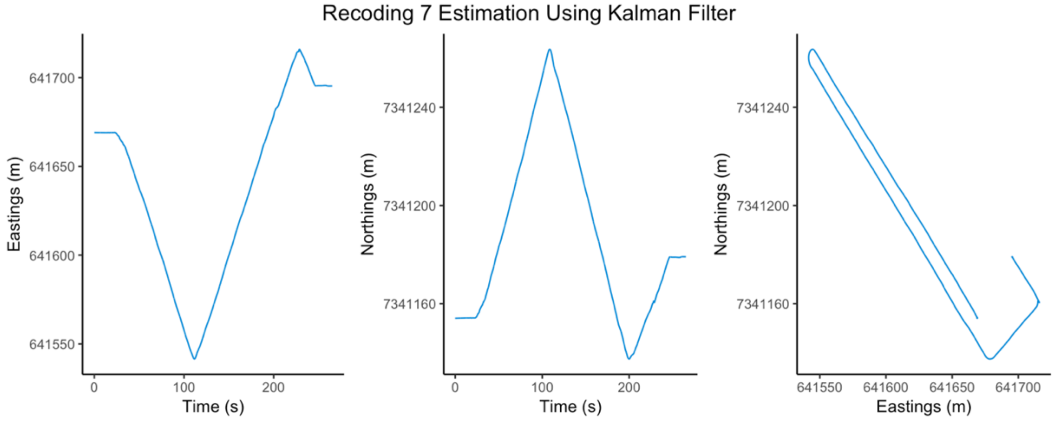

Figure 9: Visualization of Recording 7 estimation.

5. Conclusion

The objective of this report is to design a Kalman filter to estimate the position of a vehicle in the absence of GNSS. We first initialise the process and measurement covariance to obtain the result. We then analyse the results and tune both covariances accordingly. Our findings indicated that the Kalman Filter can accurately estimate the position of a vehicle based on a combination of measurements. In addition, choosing an appropriate process and measurement covariance can enhance the performance of the Kalman filter. In our project, we also found that process covariance has a greater influence on performance than measurement covariance. However, the reason of this could be we set the measurement covariance as a static matrix in this project. Some advanced methods, such as the Recursive Covariance Estimation algorithm [5], Bayesian approach [8], and deep neural networks [9], show that using a dynamic measurement covariance yields a better estimation. Future research can incorporate more states and apply additional sophisticated methods to improve the overall performance. This study has some limitations. Because we have limited available data, we cannot ensure that the results of this study can be applied in any environment. Moreover, our model works only when certain assumptions are satisfied for example, the control input noise must be very small. Further studies should be conducted when additional data are available, or assumptions are not met.

Reference

[1] R. E. Kalman, "A new approach to linear filtering and prediction problems," 1960.

[2] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," 1961.

[3] S. F. Schmidt, "The Kalman filter-Its recognition and development for aerospace applications," *Journal of Guidance and Control,* vol. 4, no. 1, pp. 4-7, 1981.

[4] M. Karasalo and X. Hu, "An optimization approach to adaptive Kalman filtering," *Automatica,* vol. 47, no. 8, pp. 1785-1793, 2011.

[5] B. Feng, M. Fu, H. Ma, Y. Xia, and B. Wang, "Kalman filter with recursive covariance estimation—Sequentially estimating process noise covariance," *IEEE Transactions on Industrial Electronics,* vol. 61, no. 11, pp. 6253-6263, 2014.

[6] S. Akhlaghi, N. Zhou, and Z. Huang, "Adaptive adjustment of noise covariance in Kalman filter for dynamic state estimation," in *2017 IEEE power & energy society general meeting*, 2017: IEEE, pp. 1-5.

[7] P. Matisko and V. Havlena, "Noise covariances estimation for Kalman filter tuning," *IFAC Proceedings Volumes,* vol. 43, no. 10, pp. 31-36, 2010.

[8] P. Matisko and V. Havlena, "Noise covariance estimation for Kalman filter tuning using Bayesian approach and Monte Carlo," *International Journal of Adaptive Control and Signal Processing,* vol. 27, no. 11, pp. 957-973, 2013.

[9] L. N. Cash, "LOCALIZATION OF PEOPLE IN GNSS-DENIED ENVIRONMENTS USING NEURAL-INERTIAL PREDICTION AND KALMAN FILTER CORRECTION," 2023.

[10] Posit team, "RStudio: Integrated Development Environment for R. Posit Software," in *Posit Software*, ed. PBC, Boston, MA, 2024.

[11] M. M. Andersen and S. Højsgaard, "Ryacas: A computer algebra system in R," vol. 4, 2019, Art no. 42. [Online]. Available: https://doi.org/10.21105/joss.01763.