

# Implementation of Sensor Filters and Altitude Estimation of Unmanned Aerial Vehicle using Kalman Filter

\*<sup>1</sup>Mertcan Akin, <sup>1</sup>Aytaç Gören and <sup>2</sup>Ahmed Rachid

<sup>1</sup>Department of Mechanical Engineering, Automatic Control and Robotics Labs., Dokuz Eylul University, Izmir, Turkey

<sup>2</sup>Laboratory of Innovative Technologies, University of Picardie Jules Verne, Amiens, France

## Article history

Received: 23-12-2020

Revised: 28-01-2021

Accepted: 30-01-2021

Corresponding Author:

Mertcan Akin

Department of Mechanical

Engineering, Automatic Control

and Robotics Labs., Dokuz

Eylul University, Izmir, Turkey

Email: mertcanakin95@gmail.com

**Abstract:** Sensor outputs are used when finding the position of an Autonomous Aerial Vehicle (UAV). Outputs such as air pressure, vehicle acceleration or magnetic field help the vehicle decide its attitude and altitude. However, it must also predict the next movement of the vehicle. The Kalman Filter is used to assist observing and predicting the state of a system. To achieve this, sensors and a model that can predict the future movement of the vehicle are needed. In practice, the sensors and models are not perfect. There are always uncertainties like weather conditions. Also the data from the sensors are often noisy. Getting less noisy data for an unmanned aerial vehicle is important for stable flight. This uncertainty is reduced with the Kalman Filter. The mentioned uncertainties can be reduced using the Kalman Filter and more stable data can be obtained.

**Keywords:** Kalman Filter, Noise, Altitude, UAV

## Introduction

This article focuses on the sensor noise filtering of the altitude of an UAV. Many different types of sensors can be used to measure height in UAVs. Some of these sensors are barometer, ultrasonic and laser distance sensors. These sensors have advantages and disadvantages among others. Laser sensors have more resolution and quicker response time than ultrasonic sensors (Sasidharan *et al.*, 2016). But ultrasonic sensors are cheaper and have wider field of view. Above 5-10 m altitude, it is better to use a barometer. But barometer sensors are usually not very sensitive and are too noisy due to factors such as temperature and wind. Using a UAV with raw data from the sensors used is not a healthy method. Due to incorrect altitude data along with unstable flight, there may be an accident at the time of landing. It is necessary to make the data more noise-free by using a number of filters. In this study, performance comparison was made by examining different filters. As mentioned, the sensor data is noisy. They can vary depending on various factors such as temperature, density, wind. Equation 1 and 2 are also given the height measurement equations of ultrasonic and barometer sensors, respectively:

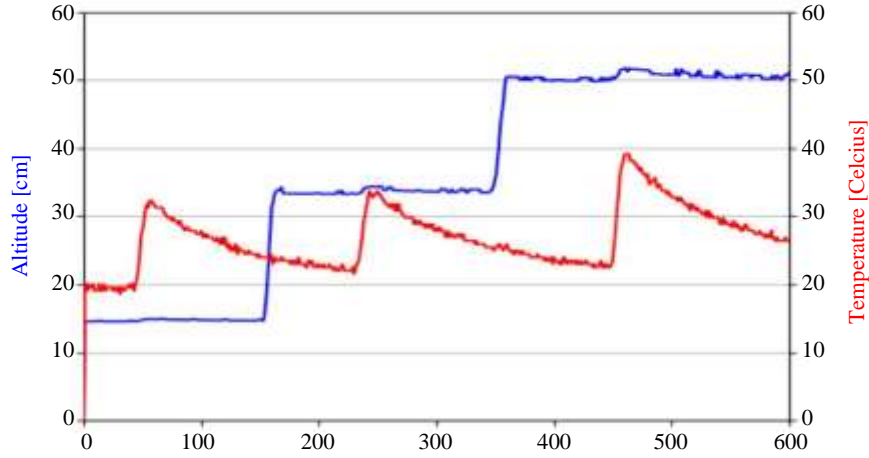
$$h_{ultrasonic} = (331.3 + 0.66xT)t / 2 \quad (1)$$

$$h_{barometer} = \left( \frac{p}{p_0} \frac{RL}{gM} - 1 \right) \cdot \frac{T_0}{L} \quad (2)$$

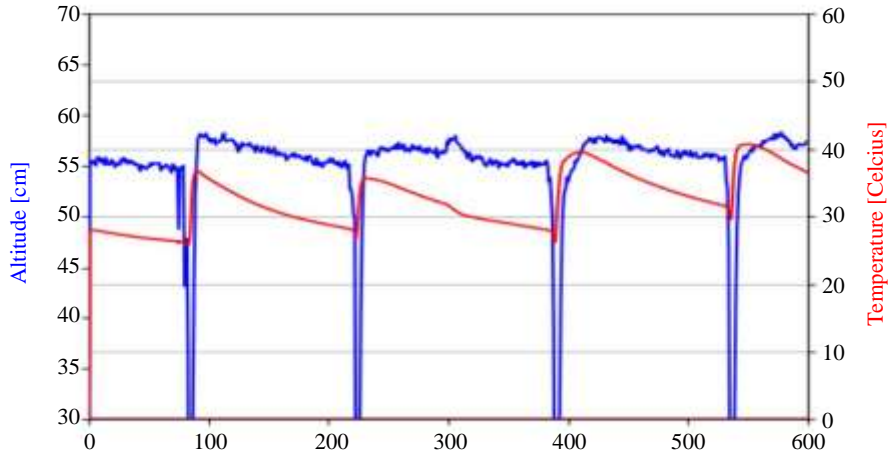
where the coefficients are as:  $T$  current temperature,  $t$  time,  $p$  atmospheric pressure at defined level,  $p_0$  sea level standard atmospheric pressure,  $L$  temperature lapse rate,  $T_0$  sea level standard temperature,  $g$  Earth-surface gravitational acceleration,  $M$  molar mass of dry air,  $R$  universal gas constant (Gaşior *et al.*, 2014). When these equations are examined, they appear to vary depending on temperature.

Figure 1 and 2 show changes in ultrasonic and barometer height data depending on temperature. As can be seen from the figures, the barometer sensor is more noisy when temperature changes.

This study was written to examine the Kalman filter as well as various filters. Kalman filter was published by (Kalman, 1960) as a paper titled "A New Approach to Linear Filtering and Prediction Problems". The Kalman filter is used, especially in the guidance and control of aircraft, spacecraft and autonomous vehicles. Kalman filter works on linear models, as will be mentioned later. Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF) algorithms have been developed for nonlinear models (Rhudy *et al.*, 2013). In this study, the Kalman filter will be examined. This paper will be helpful for previous (Kurttay *et al.*, 2018) and upcoming projects on UAVs in the Automatic Control and Robotics Laboratory. EKF and UKF will be examined in later studies.



**Fig. 1:** Altitude-temperature graph (Ultrasonic sensor)



**Fig. 2:** Altitude-temperature graph (Barometric sensor)

## Materials and Methods

In this Section, some filters used to reduce sensor noise and the Kalman filter will be examined. Getting average is a simple way for filtering. Distance value,  $h_t$  observes the altitude of the UAV. Average is calculating with below equation:

$$\hat{x}_t = \left( \sum_0^n h_t \right) / n \quad (3)$$

$\hat{x}_t$  means estimate of the true state  $x_t$ . But getting average requires some memory. Because all values must be kept since the measurement is taken. As shown in the Fig. 3, noise of the sensor is reduced.

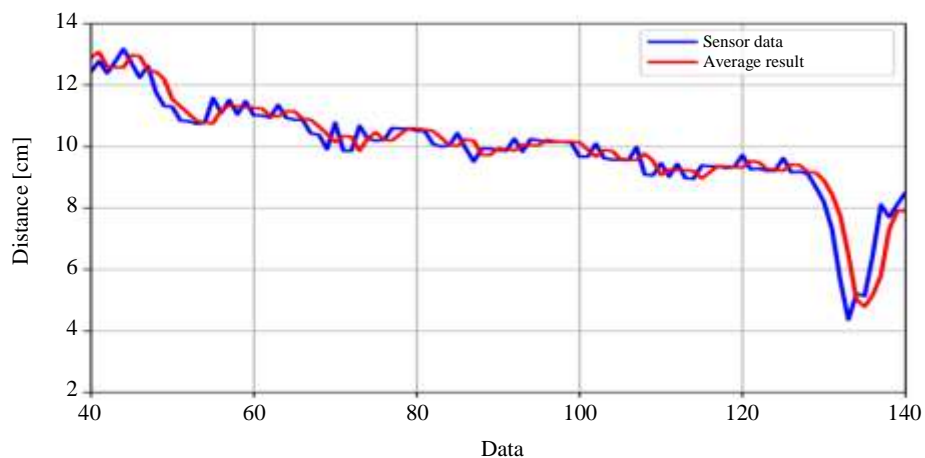
However, there is still a lot of noise. It would be more efficient to recursively estimate rather than take a direct mean:

$$\hat{x}_t = [x_{t-1} \times (t-1) + h_t] / t \quad (4)$$

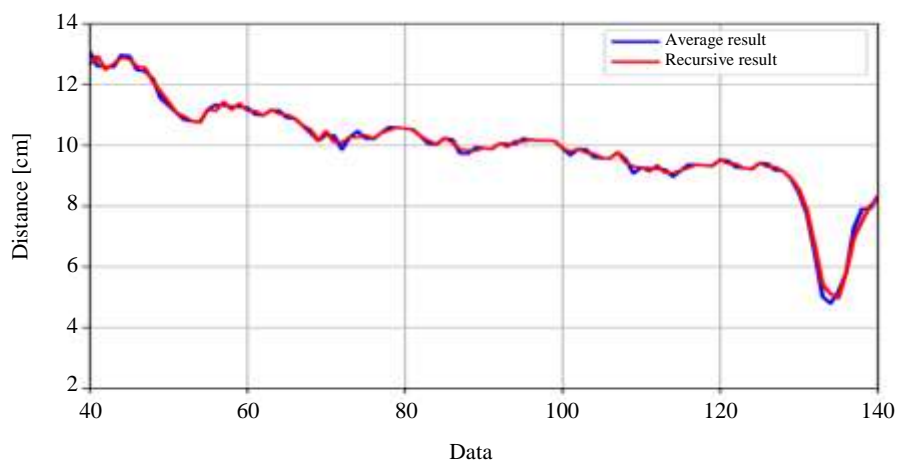
Using recursive average, we can perform the computation more efficient without storing all the data (Tellex *et al.*, 2018). As shown in Fig. 4, sensor noise is slightly reduced compared to the normal average.

The disadvantage of the average filter is the time it takes to process the data. An alternative to this method is the **moving average** method. Instead of averaging all the data at once, the averages are added to the general average at certain intervals (Raudys and Pabarškaitė, 2018).

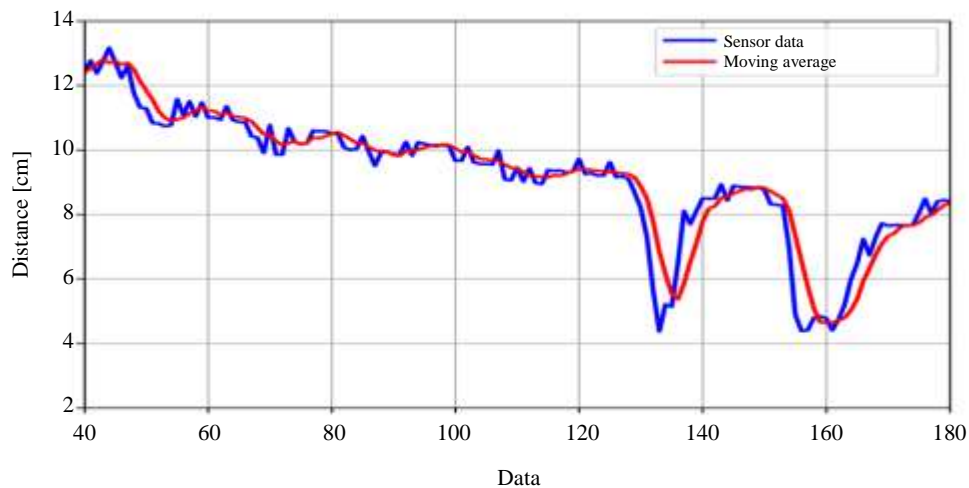
As shown in the Fig. 5, there is some delay because the operation is performed at certain periods. As the calculation period is reduced, the delay is reduced. Delay is advantageous in terms of spent power. However, memory usage is still high because it is necessary to store previous data as well.



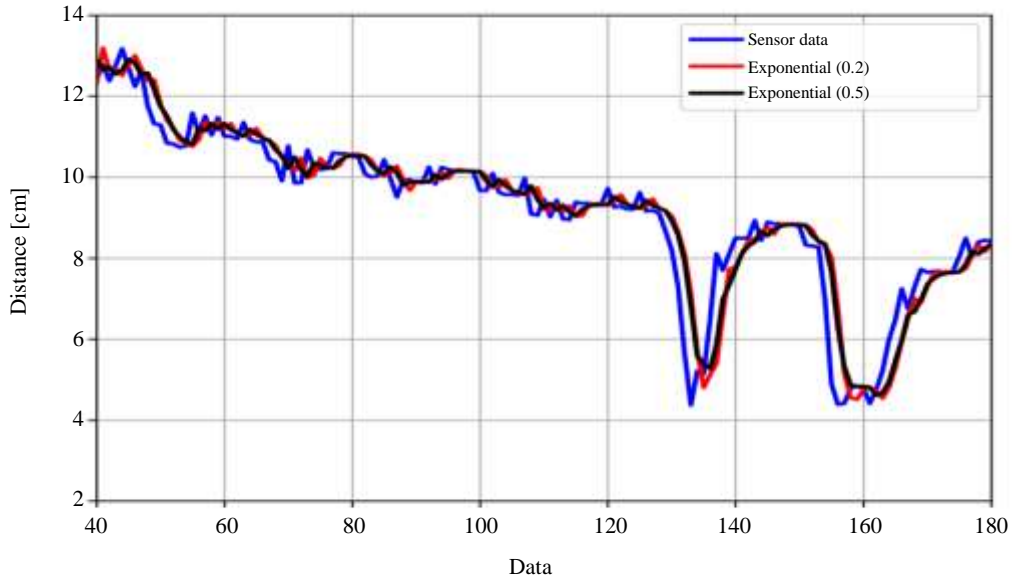
**Fig. 3:** Average result



**Fig. 4:** Recursive form result



**Fig. 5:** Moving average result



**Fig. 6:** Exponential smoothing result

**Exponential filter** is a recursive filter and creates new optimized data using the latest optimized data and new data. The mathematical model is as shown below (Ostertagová and Ostertag, 2011):

$$\hat{y}_{i+1} = \alpha y_i + (1 - \alpha) \hat{y}_i \quad (5)$$

$y_i$  new measurement,  $\hat{y}_i$  last smoothed value,  $\hat{y}_{i+1}$  new smoothed value and  $\alpha$  represents smoothing constant.

In Fig. 6, exponential smoothing filter with different smoothing constant is examined. Smoothing constant greatly changes the characteristic of the filter. If the smoothing constant is small, the filter responds a little later, but returns a smoother result. If the smoothing constant is large, the filter will respond to changes sooner, but the results will not be very smooth.

Exponential filter does not need much memory, as only the final measurement must be stored. Another advantage is that it does not consume much power because it does not take too much measurement.

**Low-pass and High-pass filters** are one of the most popular filters today. It is often preferred because its application is easy and fast. These filters can be examined in two classes: Active and Passive. Active elements such as transistor or operational amplifiers are used in active filters, passive elements such as inductors, capacitors and resistors are used in passive filters. Active elements require external source for their work and can provide power gain. Passive elements, on the other hand, do not need an external source for their work and do not have an effect on power gain (Lacanette, 1991; Khan *et al.*, 2016).

Low/High Pass filters are digital filters consisting of circuits with passive elements. A low pass filter does not affect low frequencies and rejects high frequencies. In this study, a software was developed that would coincide with a low-pass filter without adding an external circuit.

A first order low-pass filter can be implemented with RC circuit and according to Kirchoff's Law. To apply the equation to the algorithm, the equation is written as:

$$y(n) = \alpha x[n] + (1 - \alpha)y[n-1] \quad (6)$$

$$\alpha = \frac{\Delta t}{\Omega + \Delta t} \quad (7)$$

$\Omega$  in the equation is a time constant depending on the resistance and capacitance values of the circuit and  $\alpha$  is a smoothing factor. As shown in the Fig. 7, sensor noise was reduced using the low pass filter.

The High pass filter does not allow low frequencies while passing high frequencies, as opposed to the low-pass filter. The extraction of the algorithm is similar to low-pass filter:

$$y[n] = \alpha(y[n-1] + (x[n] - x[n-1])) \quad (8)$$

The main difference between these two filters is that the high pass passes higher frequencies than the cutoff frequency, while the low pass filter passes lower frequencies than the cutoff frequency.

In addition to the filters described above, there is also **the Kalman Filter**, which is used in autonomous vehicles today. Basically, the Kalman Filter is a tool that predict

values using mathematical expressions and equations and works with Gaussian distribution and linear models (Nonami *et al.*, 2010; Welch and Bishop, 2006). Take position estimation as an example. For KF, it is primarily assumed that our position has a Gaussian distribution. If we expect to see our vehicle at position  $x$ , it means that at the highest probability it is at Point  $x$  and the farther away it is from Point  $x$ , it is less likely to see our vehicle at that point. In general, any linear combination of independent normal deviations is a normal deviation. So KF equations must be linear. Unlike the Kalman filter, Extended Kalman Filter (EKF) and Unscented Kalman Filter are also used.

The Kalman Filter consists of two main parts: "Prediction" and "Correction" (Kalman, 1960; Rhudy *et al.*, 2017). Figure 8 shows the principle of the Kalman filter as a diagram. In the first part, the state vectors are calculated according to the data from the last iteration. Also the covariance matrix is updated:

$$\hat{X}_{(k|k-1)} = A\hat{x}_{k-1} + Bu_k \quad (9)$$

$\hat{x}_{(k|k-1)}$  is the predicted state vector,  $\hat{x}_{(k-1)}$  is the previous estimated state vector,  $u$  is the input vector,  $A$  and  $B$  are matrices that define the system dynamics. In the next step, the state error covariance matrix will be predicted using Equation (10):

$$P_{(k|k-1)} = AP_{k-1}A^T + Q \quad (10)$$

Covariance is a measure of how two random variables change together. If one of the variables increases or decreases while the other changes in the same way, the covariance is positive. If the variables change in a way that is opposite to each other the covariance is negative. Covariance is zero when the variables are independent of each other.

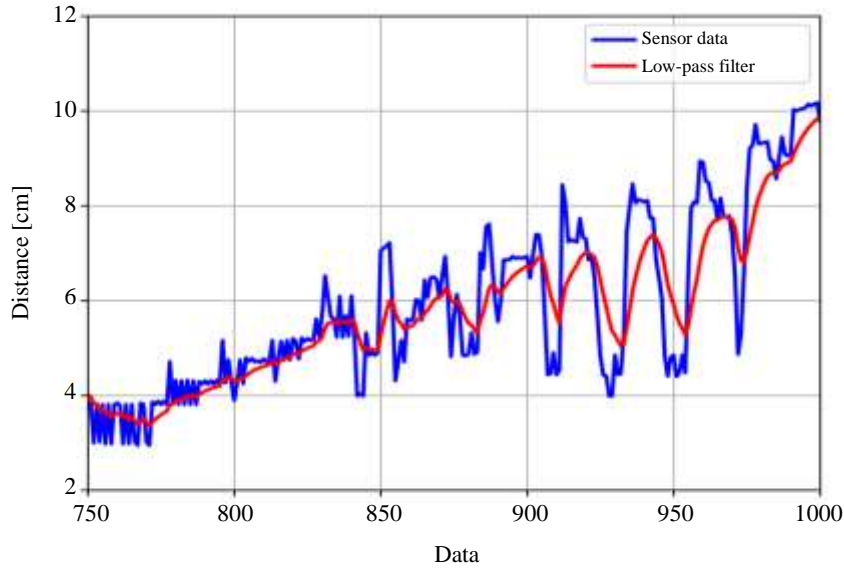


Fig. 7: Low Pass Filter Result

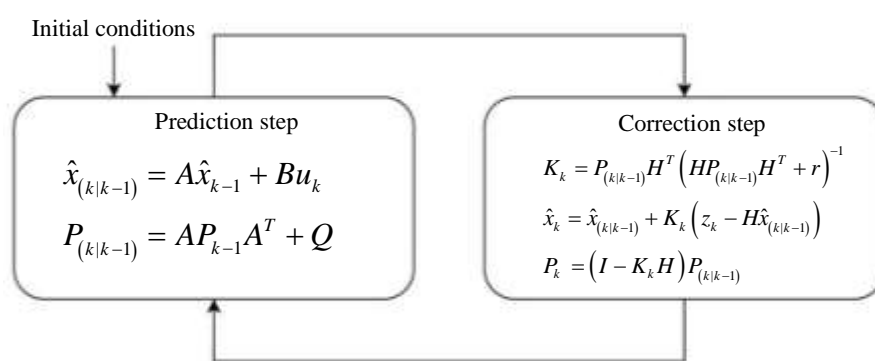
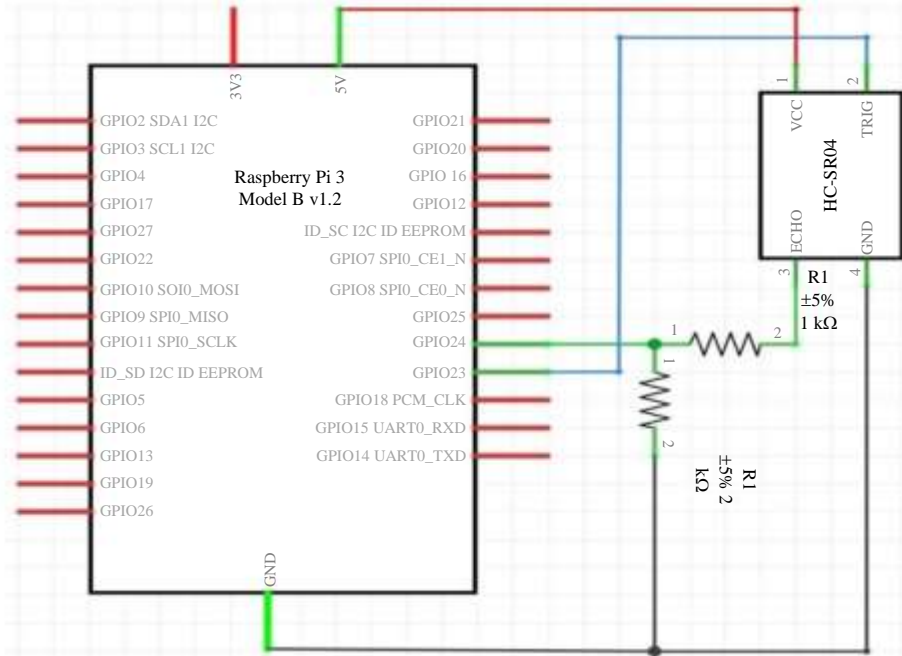


Fig. 8: Kalman filter diagram



**Fig. 9:** Circuit diagram

Process noise covariance “ $q$ ” is taken as arbitrary according to (Al Tahtawi, 2018). Measurement noise covariance “ $r$ ” value varies from sensor to sensor. To find this value, a group of raw data is first taken from the sensor. The covariance of these data is found with the “cov” command in MATLAB. Results are found using  $r = 0.0324$  and  $q = [1e-3, 1e-4, 1e-5]$ .

In the second part (correction), an update is made according to the estimates and measurements. Once the predicted values are obtained, Kalman Gain ( $K$ ) is calculated by Equation (11):

$$K_k = P_{(k|k-1)} H^T (H P_{(k|k-1)} H^T + r)^{-1} \quad (11)$$

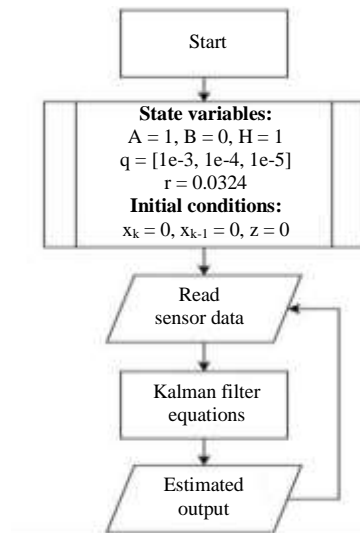
The model estimate is compared with the observation and this difference is scaled by a value known as the Kalman Gain ( $K$ ). Then Kalman Gain, feedback as an input to the model to improve the next estimates. Measurements are followed more closely if high gain is used but the noise cannot be fully removed. If low gain is used, model estimations are more closely followed. The noise is completely removed, but reaction rate decreases.

$H$  is a matrix that necessary to define the output equation.  $r$  is the measurement noise covariance:

$$\hat{x}_k = \hat{x}_{(k|k-1)} + K_k (z_k - H \hat{x}_{(k|k-1)}) \quad (12)$$

**Table 1:** Variable names

Variable	Description
$x$	State vector
$z$	Measurement vector
$A$	Process matrix
$B$	Control matrix
$H$	Observation matrix
$u$	Input vector
$P$	Covariance matrix
$K$	Kalman gain
$q$	Process noise covariance
$r$	Measurement noise covariance



**Fig. 10:** Algorithm flowchart

The Kalman filter receives its information from errors, noise and uncertainties. The purpose of the filter is to process this imperfect information and reduce uncertainty and noise:

$$P_k = (I - K_k H) P_{(k|k-1)} \quad (13)$$

Table 1 shows the variables used in the Kalman filter. While preparing the algorithm flowchart in Fig. 10, some assumptions were made. As mentioned earlier, the values  $A$  and  $B$  are related to the dynamics of the system.  $A = [1]$  has been adopted to obtain a scalable value according to the measure. Since there is no control input,  $B = [0]$ . With the assumption of  $H = [1]$ , it is ensured that the output depends on the observation data.

EKF is used for nonlinear models (Kerimoğlu, 2011). Linearization is performed using the Taylor expansion and the best estimate for the Gaussian mean is made using the HJ Jacobian matrix instead of the H Matrix (Angrisani *et al.*, 2009). UKF offers a different solution. Instead of linearizing the transformation function, it makes an approximation for next step. We know that the distribution after the nonlinear transformation is not Gaussian, but we assume it as Gaussian and try to approach the best real distribution. Sigma points are used to find this approach. A few points are taken around the expected position and nonlinear transformation is applied (Tellex *et al.*, 2018).

EKF works very well in nonlinear transformations, but UKF works better in high-level nonlinear transformations.

## Results

Experiments were performed using Raspberry Pi 3 and an ultrasonic distance sensor, as in the circuit shown in Fig. 9. Ultrasonic sensors can measure

distance using sound waves. Microcontroller sends a signal to trigger pin of the sensor and then after the signal is sent, the sensor sends an ultrasonic wave and starts counting. After the wave hits the obstacle, it returns and the echo pin is triggered. Then the counter stops and the time difference is calculated. Distance can be obtained because the speed of the sound wave and the duration of the wave are known. HC-SR04 ultrasonic sensor used in this study. This sensor provides ultrasonic waves with a frequency of 40 kHz and it can measure up to 400 cm distance with a resolution of 3 mm.

The effect of the Kalman Filter was observed by changing the distance. As shown in the Fig. 13, the noisy data received from the sensor has been transformed into a more stable shape with the Kalman Filter. Tests are made with different  $q$  covariance values. By taking  $q = 1e-3$ , the response time was very close to the measurement speed, but a negligible amount of noise measurement was taken due to sensor noise. By taking  $q = 1e-4$  and  $q = 1e-5$ , sensor noise has been eliminated, but response time has increased. In summary, the response rate slows down as the noise variance decreases.

Figure 14 shows measurements made with an ultrasonic sensor attached to the UAV. The vehicle was lifted 1.8 m above the ground and then landed as shown in the Fig. 11. As can be seen from the graph, as the distance increases, the sensor sensitivity decreases and more noisy data occurs. If these data are used in the control system, there may be instability in the system and may cause an accident.

Kalman filter and low pass filter were compared as shown in the Fig. 15.  $q = 1e-4$  in this the graph and more accurate data is obtained with the Kalman filter. A low pass filter filters out frequencies above a passband. But a Kalman filter is suitable when there is a dynamic model that you can use to predict the value of a signal in the future.

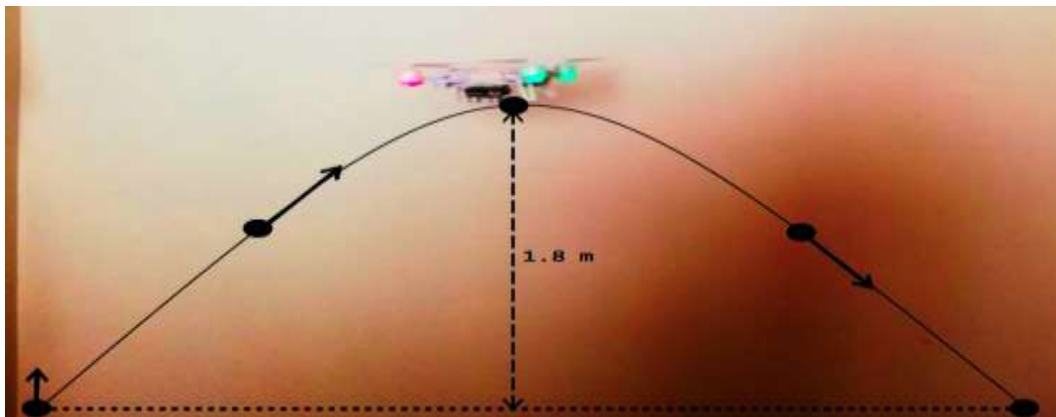
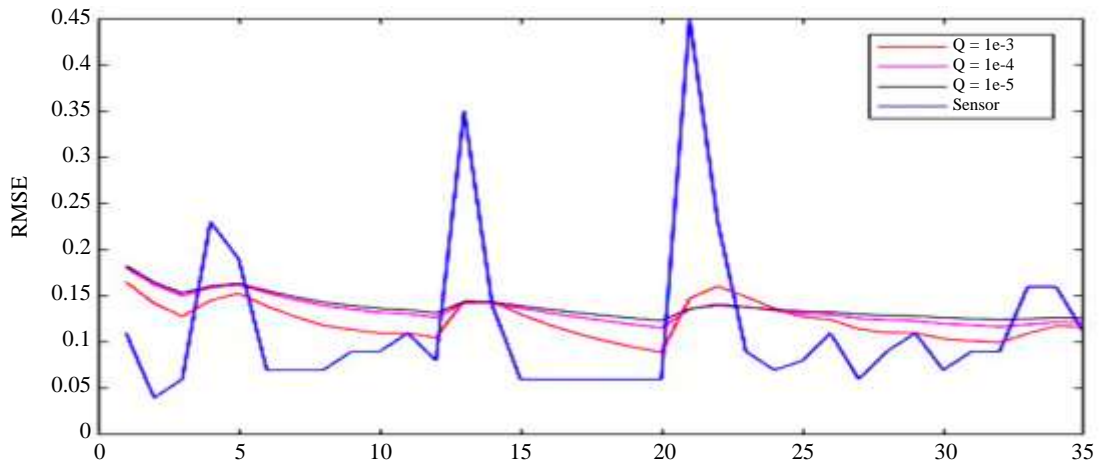
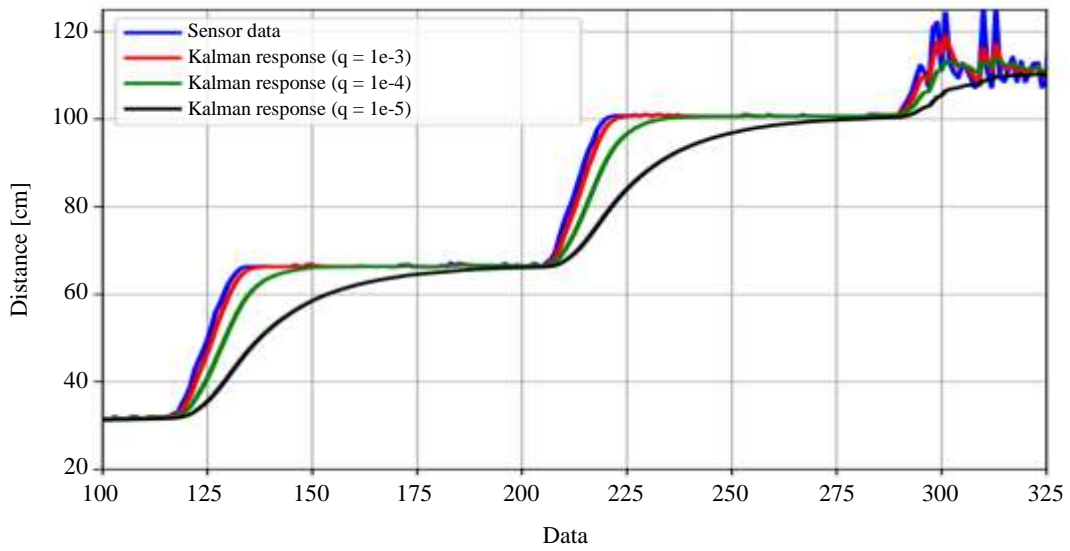


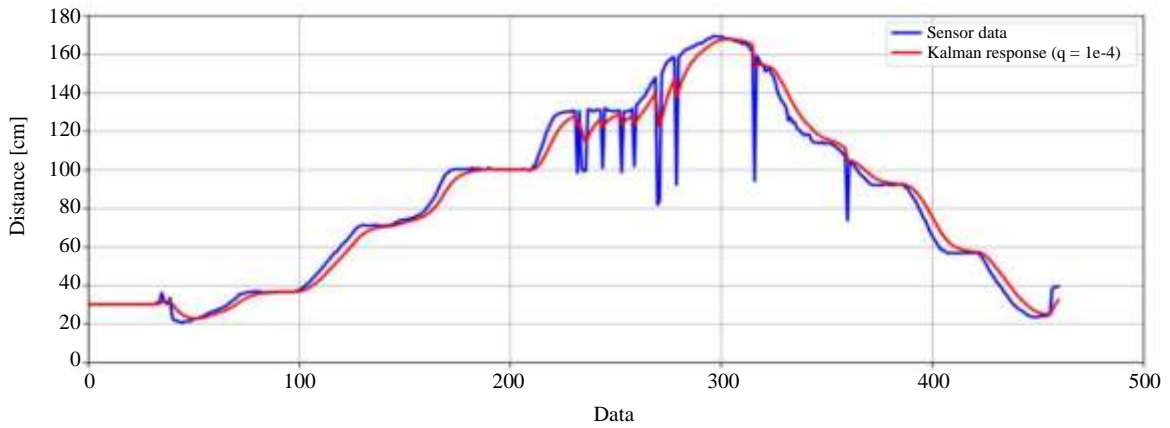
Fig. 11: Test platform



**Fig 12:** RMSE graph



**Fig 13:** Responses with Different Covariance



**Fig 14:** Flight result



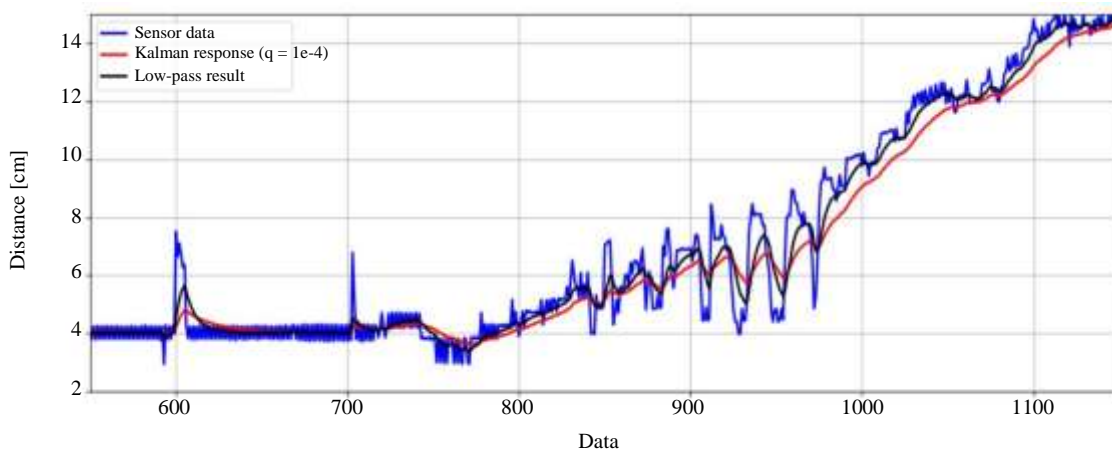


Fig 15: Kalman filter and low pass filter comparison

## Discussion

Root Mean Squared Error (RMSE) is used to determine the error rate between measurement values and model estimates (Chai and Draxler, 2014). The RMSE value is asked to approach zero. It is expressed mathematically as follows:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{n}} \quad (14)$$

As shown in the Fig. 12, with  $q = 1e-4$  covariance, the system would be more useful. The error rate has been reduced with the Kalman Filter. If rapid reactions are desired in the system in which the filter will be used, it is more efficient to have a high covariance.

## Conclusion

The Kalman filter has been successfully applied to the ultrasonic distance sensor. In fact, noisy data obtained from the sensor has been made more stable using the Kalman Filter. Less noisy altitude data provided more stable altitude control of the vehicle. In addition, the results of the Kalman filter with different covariance values were examined. With smaller covariance, the amount of noise decreases but the time response increases.

Basically, the Kalman Filter is a tool which estimates and predicts values using model equations under the assumptions that noise has a Gaussian distribution. The model in this paper is assumed to be linear but it will become non-linear as the angle of the vehicle changes. For future work, the Extended Kalman Filter (EKF) could be implemented for more efficient filtering as it can be applied to non-linear models. EKF linearizes the non-linear model around the Gaussian mean using the Taylor Series and Jacobian Matrix. At this point, EKF should be used to get the altitude data more efficiently.

## Acknowledgment

Authors would like to express their gratitude to lab crew of Automatic Control and Robotics Laboratories of the Mechanical Engineering Department of Dokuz Eylul University for their technical support and Engineering Faculty of DEU for their support to lab project entitled as "Design and Production of a Fire-Fighting Octocopter".

## Author's Contributions

**Mertcan Akin:** The author of the article and took part in the creation and implementation of algorithms.

**Aytac Goren:** Advisor of the study and provided technical support.

**Ahmed Rachid:** Helped to correct errors and reviewing critically the article.

## Ethics

This article is original and contains unpublished material. Author declares that are not ethical issues and no conflict of interest that may arise after the publication of this manuscript.

## References

- Al Tahtawi, A. R. (2018). Kalman filter algorithm design for hc-sr04 ultrasonic sensor data acquisition system. *IJITEE (International Journal of Information Technology and Electrical Engineering)*, 2(1), 15-19.
- Angrisani, L., Baccigalupi, A., & Moriello, R. S. L. (2009). Ultrasonic-based distance measurement through discrete extended kalman filter. *Kalman Filter Recent Advances and Applications*, 269-296.
- Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)?—Arguments against avoiding RMSE in the literature. *Geoscientific model development*, 7(3), 1247-1250.

- Gaşior, P., Gardecki, S., Gośliński, J., & Giernacki, W. (2014). Estimation of altitude and vertical velocity for multicopter aerial vehicle using Kalman filter. In *Recent Advances in Automation, Robotics and Measuring Techniques* (pp. 377-385). Springer, Cham.
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems.
- Kerimoğlu, K. (2011). Sabit kanatlı bir insansız hava aracı için düşük bütçeli otopilot sistemi tasarımı (Master's thesis, TOBB Ekonomi ve Teknoloji Üniversitesi Fen Bilimleri Enstitüsü).
- Khan, M. D., Alam, M. D. M., Masud, M. D. A., & Amin, A. A. (2016). Importance of High Order High Pass and Low Pass Filter. *World Applied Sciences Journal*, 34(9), 1261-1268.
- Kurttay, B., Ceyhun, H.E., Yüksel, T., 2018. Yangına Müdahale Drone Tasarımı ve İmalatı, Dokuz Eylül University, Izmir, Turkey
- Lacanette, K. (1991). A basic introduction to filters-active, passive and switched-capacitor. National Semiconductor Corporation, <http://www.swarthmore.edu/NatSci/echeeve1/Ref/D ataSheet/Inttofilters.pdf>
- Nonami, K., Kendoul, F., Suzuki, S., Wang, W., & Nakazawa, D. (2010). *Autonomous flying robots: unmanned aerial vehicles and micro aerial vehicles*. Springer Science & Business Media.
- Ostertagová, E., & Ostertag, O. (2011, September). The simple exponential smoothing model. In *The 4th International Conference on Modelling of Mechanical and Mechatronic Systems*, Technical University of Košice, Slovak Republic, Proceedings of conference (pp. 380-384).
- Raudys, A., & Pabarškaitė, Ž. (2018). Optimising the smoothness and accuracy of moving average for stock price data. *Technological and Economic Development of Economy*, 24(3), 984-1003.
- Rhudy, M. B., Salguero, R. A., & Holappa, K. (2017). A kalman filtering tutorial for undergraduate students. *International Journal of Computer Science & Engineering Survey*, 8(1), 1-9.
- Rhudy, M. B., Gu, Y., Gross, J., Gururajan, S., & Napolitano, M., (2013). Sensitivity Analysis of Extended and Unscented Kalman Filters for Attitude Estimation. *AIAA Journal of Aerospace Information Systems*. 10. 10.2514/1.54899.
- Sasidharan, A., Salahuddin, M. K., Bose, D., & Ramachandran, K. (2016). Performance comparison of Infrared and Ultrasonic sensors for obstacles of different materials in vehicle/ robot navigation applications. *IOP Conference Series: Materials Science and Engineering*. 149, 012141.
- Tellex, S., Brown, A., & Lupashin, S. (2018). Estimation for Quadrotors. arXiv preprint arXiv:1809.00037.
- Welch, G., & Bishop, G. (2006). *An introduction to the Kalman filter*, UNC-Chapel Hill (p. 16). TR 95-041.