

# IMPLEMENTATION OF KALMAN FILTER ON VISUAL TRACKING USING PID CONTROLLER

Abdurrahman,F.\*<sup>1</sup>, Gunawan Sugiarta\*<sup>2</sup> and Feriyonika\*<sup>3</sup>

\*Department of Electrical Engineering, Bandung State of Polytechnic, Bandung, Indonesia

## ABSTRACT

*This paper explain the design of visual control system which equip Kalman filter as an additional sub-system to predict object movement. It is a method to overcome some weakness, such as low range view of camera and low FPS (Frame per Second). It alsoassist the system to track a fast moving object. The system is implemented to 2 motor servos, which are move on horizontal and vertical axis. Digital PID (Proportional, Integral, and Derivative) controller is used in the system, and bilinear transformation is used to approximate the value of derivative in transforming the analogue to digital controller on z-transform. In conclusion, we can get the value of system responses time from both motor servos. The rise time and settling time of motor servo in horizontal axis are 0.402s and 1.63s, and vertical axis's responses are 0.38s and 1.34s.*

## KEYWORDS

*Kalman filter, PID, bilinear transformation, z-transform*

## 1. INTRODUCTION

In this digital era, robotic technologies are implementing vision-based control, such as object tracking [1], face tracking [2], and position-based control [3]. It was developed as one of the control method in robotic community. Researchers keep learning and involving the control system to transform it as close as human capability. Recent paper describes the vision-based control and computer vision as numerical theory [4], unfortunately there are lack of paper which described and design the system in a more practical way.

There are many add-on systems in vision-based control to improve accuracy, such as image filter and predictive filter. To implement such systems, which will be used on various environment with lots of noises, the method should therefore always be developed to reduce noises. So, there are many researchers in this field to improve the system's capability.

Gian Luca designed image-based visual servo using geometry [5] in mobile robot and combined it with kinematic design. This paper will explain about a visual tracking system controlled by visual control using PID (Proportional, integral, and derivative) control law to move plants in horizontal and vertical axis. It also combined with Kalman filter to predict the object movement within camera frame.

The purpose of the system is to predict the object movement, so it can make the system work in a better performance to predict the object movement while it is not detected. The system will not search the object while it was outside the range of vision of the system without using kalman filter. The control law (PID), filter, and feature extraction will be designed in one processor.

We will use 2-D in feature extraction, there are two axis which are being the feature for control system. The feature extraction method is contour in color detection, however it will not be discussed in this paper. This paper will concern to the design in development of visual control in robotics using the predictive filter.

This paper proposed on how to implement and design the system in a more practical way. According to the passages above, we can conclude that the advantages of the proposed system are:

- The digital PID algorithm will be designed using z-transform which make it easier to understand and implement.
- Kalman filter will be used to predict object movement, so it will help the system's response when tracking faster object. Kalman filter is useful on system arrangement using low resolution and low FPS camera. It will be useful if the camera frame which used in the system doesn't have enough resolution and bad fps (frame per second) [6].

This paper is organized in 5 sections. Section 2 and 3 explain about the digital control system design and Kalman filter theory. Section 4 will be discussed about the design from the control system perspective and the next section contains results and data analysis comparing data between system with Kalman filter and without Kalman filter. Finally, the last section is about the conclusion of all the system.

## 2. DIGITAL CONTROL SYSTEM DESIGN

In this section, we will review the digital control system. Z-transform can be used to design the digital control system. To transform the analogue to digital system, the system must follow these requirements [7]:

- A stable analogue system must transform to a stable digital system.
- The frequency response of the digital system must closely resemble the frequency response of the analogue system in the frequency range  $0 \rightarrow \omega_s/2$  where  $\omega_s$  is the sampling frequency.

There are many methods to transform the analogue to digital system, consist of:

### 2.1 Differencing Method

The system can be represented by a transfer function or differential equation. Numerical analysis provides standard approximations of the derivative so as to obtain the solution to a differential equation. There are 2 types of the differencing method [7], such as:

- Forward differencing  
In forward differencing the derivative approach will be found through the basic equation 2.1:

$$\dot{y}(k) = \frac{1}{T} [\dot{y}(k+1) - \dot{y}(k)] \quad (2.1)$$

After that, the equation will be transformed into z-domain, so the equation 2.2 will be produced:

$$sY(s) \rightarrow \frac{1}{T} [z - 1]Y(z) \quad (2.2)$$

Based on this method, we could find the relation between Laplace and z-transform. The relation showed on the equation 2.3, with T is a time sampling

$$s \rightarrow \frac{z-1}{T} \quad (2.3)$$

– Backward differencing

The other differencing method is backward differencing with the approximation of the derivative is showed on the equation 2.4

$$\dot{y}(k) = \frac{1}{T} [\dot{y}(k) - \dot{y}(k-1)] \quad (2.4)$$

The equation will be transformed into z domain, so we can get the relation between Laplace and z domain in equation 2.5

$$s \rightarrow \frac{z-1}{zT} \quad (2.5)$$

## 2.2 Bilinear Transform

Equation 2.6 is the relation between Laplace and z-transform [7] using bilinear transformation

$$s = c \frac{z-1}{z+1} \quad (2.6)$$

The relation can be found from the equality  $z = e^{sT}$  using the first-order approximation on the equation 2.7 where the constant  $c = 2/T$

$$s = \frac{1}{T} \ln(z) = \frac{2}{T} \left[ \frac{z-1}{z+1} \right] \quad (2.7)$$

## 3. DISCRETE KALMAN FILTER

Kalman filter is essentially a set of mathematical equations that implement a predictor-corrector type estimator that is optimal in the sense that it minimizes the estimated error covariance. There are 2 general processes to implemented, such as time update and measurement update. The time update equations can be thought of as predictor equation, while the measurement update equations can be thought of as corrector equation. Indeed the final estimation algorithm resembles that of a predictor-corrector algorithm for solving numerical problems [8] as shown in Figure 1.

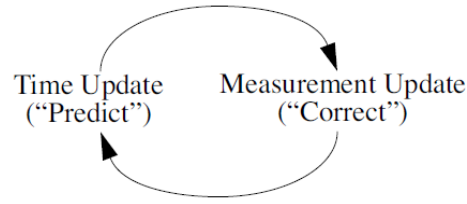


Figure 1. Process of Kalman Filter

Kalman filter can be use in visual control to track the object. Normally, a tracking algorithm would be able to locate the object anywhere within the image at any point in time. However, only a limited region of the image is searched. Visual tracking will be failed if the object was moving to outside the region of camera frame and movetoo fast. Kalman filter can predict the object movement and solve these problems [6].

To implement discrete Kalman filter [8], we need to write down these equations into algorithm:

- Time update

First task is to determine the prediction value ( $x_k$ ), followed by error covariance ( $p_k$ ). There are other variables, such as  $A$ ,  $B$ ,  $u_k$ , and  $Q$ . These variables can be assumed as constant variables.

$$x_k = Ax_{k-1} + Bu_k$$

$$P_k = AP_{k-1}A^T + Q$$

- Measurement update

The process continued to determine Kalman gain ( $k_k$ ), error covariance ( $p_k$ ), and prediction value ( $x_k$ ).  $X_k$  will be the output of Kalman filter and used as feedback in control system.  $H$  can be determined as identity matrix, furthermore  $R$  is standard deviation. The method determine variable  $R$  is various, variable can be considered as a constant value or measured using the equation.

$$K_k = P_k H^T (P_k H^T + R)^{-1}$$

$$x_k = x_k + K_k (z_k - Hx_k)$$

$$P_k = (I - K_k H) P_k$$

#### 4. SYSTEM DESIGN

The system was designed to control 2 plants, consist of servo in horizontal axis and servo in vertical axis. These were controlled using PID controller and Kalman filter as feedback to move camera. The output of camera is an image frame which is proceed by computer vision technology before proceed to Kalman filter.

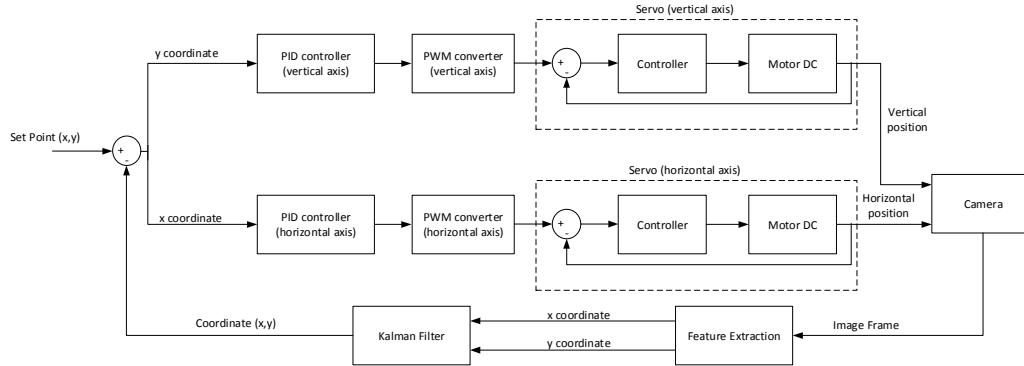


Figure 2. Block Diagram of Control System

Figure 2 is the block diagram of the control system that explained on the proceeding section. Each block in the system has their own function. Set point has 2 values, there are x and y coordinate, which are proceed by their own PID block. Then, the output PID will be converted into degree and give instructions to servo to move to certain position.

Servo has its own controller to activate the DC motor, so motor servo can determine its position. A Camera will be attached to a servo, therefore the servo's position also shows the camera frame. The object will be recognized using image processing algorithm in feature extraction and release the x and y coordinate. Figure 3 is illustrating the hardware of the system using two servos (white) and the camera (green) so the system can move the camera in 2 axis, such as horizontal and vertical axis.

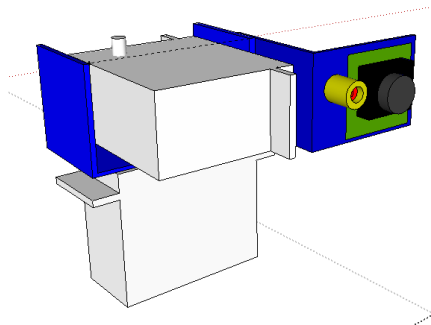


Figure 3. Horizontal and Vertical of Motor Servos

Kalman filter will proceed both coordinates in the same time, it means the coordinates will be merged. Kalman filter will be designed using the program. Finally, the output of the filter will be used as feedback in the whole system.

Subsection below will explain 2 most important components in control system, such as digital PID (Proportional, Integral, and Derivative) controller and Kalman filter.

#### 4.1 Digital PID Controller

Z-transform was implemented in the digital control system to design the algorithm. Equation 4.1 is the PID basic equation in Laplace.

$$C(s) = Kp + \frac{Ki}{s} + Kds \quad (4.1)$$

According to the equation, z-transform was applied using bilinear transform, then it will produced the equation 4.2. There is  $T_s$  variable in the equation, which is a time sampling. The time sampling has been measured with the value of 0.09second.

$$\begin{aligned} \frac{U(z)}{E(z)} &= Kp + \frac{Ki T_s}{2} \frac{z+1}{z-1} + Kd \cdot \frac{2}{T_s} \frac{z-1}{z+1} \\ \frac{U(z)}{E(z)} &= \frac{Kp(z-1)(z+1) + \frac{Ki T_s}{2}(z+1)^2 + Kd \cdot T_s \cdot 2 \cdot (z-1)^2}{(z-1)(z+1)} \rightarrow \frac{b0 + b1z^{-1} + b2z^{-2}}{a0 + a1z^{-1} + a2z^{-2}} \end{aligned} \quad (4.2)$$

Equation 4.3 is the PID equation in z domain before simplified. We can get six variables, consist of  $b0$ ,  $b1$ ,  $b2$ ,  $a0$ ,  $a1$ , and  $a2$ . Each variable will be determined using the equation 4.3 until equation 4.8.

$$b0 = Kp + \frac{Ki T_s}{2} + 2 \cdot Kd \cdot T_s \quad (4.3)$$

$$b1 = Ki \cdot T_s - 4 \cdot Kd \cdot T_s \quad (4.4)$$

$$b2 = -Kp + \frac{Ki T_s}{2} + 2 \cdot Kd \cdot T_s \quad (4.5)$$

$$a0 = 1 \quad (4.6)$$

$$a1 = 0 \quad (4.7)$$

$$a2 = -1 \quad (4.8)$$

Therefore, these equations can be substituted to be the final equation, which is equation 4.9 to implementPID (Proportional, Integral, and Derivative) algorithm in digital.

$$\begin{aligned} a0 \cdot U(z) + a1 \cdot z^{-1} \cdot U(z) + a2 \cdot z^{-2} \cdot U(z) &= b0 \cdot E(z) + b1 \cdot z^{-1} \cdot E(z) + b2 \cdot z^{-2} \cdot E(z) \\ U(z) &= \frac{-a1 \cdot z^{-1} \cdot U(z)}{a0} - \frac{a2 \cdot z^{-2} \cdot U(z)}{a0} + \frac{b0 \cdot E(z)}{a0} + \frac{b1 \cdot z^{-1} \cdot E(z)}{a0} + \frac{b2 \cdot z^{-2} \cdot E(z)}{a0} \\ u(k) &= \frac{-a1}{a0} u[k-1] - \frac{a2}{a0} u[k-2] + \frac{b0}{a0} e[k] + \frac{b1}{a0} e[k-1] + \frac{b2}{a0} e[k-2] \end{aligned} \quad (4.9)$$

The control system is a control loop, so Ziegler-Nichols type 2 was used to find the PID parameter. However, we need to tune it manually to get the best responses time. This method proposed as practical procedure to find the parameter properly.

#### 4.2 Kalman Filter

Kalman filter was designed to estimate the moving object which caught in camera frame. This filter is really beneficial in the system with a small visual angle of camera. Equations below

explain the Kalman filter process, there are 2 general process in the algorithm, such as prediction which is explained by the equation 4.10 until equation 4.11 and correction.

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}u_k \quad (4.10)$$

$$\mathbf{P}_k = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q} \quad (4.11)$$

$$\mathbf{K}_k = \mathbf{P}_k\mathbf{H}^T(\mathbf{P}_k\mathbf{H}^T + \mathbf{R})^{-1} \quad (4.12)$$

$$\mathbf{x}_k = \mathbf{x}_k + \mathbf{K}_k(\mathbf{z}_k - \mathbf{H}\mathbf{x}_k) \quad (4.13)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k\mathbf{H})\mathbf{P}_k \quad (4.14)$$

$\mathbf{x}_k$ = prediction value	$\mathbf{K}_k$ = Kalman gain
$\mathbf{P}_k$ = error prediction	$\mathbf{H}$ = transformation matrix
$u_k$ = control input	$\mathbf{R}$ = standard deviation of noise measurement
$\mathbf{A}$ = transition matrix	$\mathbf{Q}$ = process noise
$\mathbf{B}$ = system velocity	$\mathbf{z}_k$ = input kalman

There are 2 variables that can be changed to adjust the filter's responses, i.e. Q and R. Best responses will be depend on plant and the capability of the algorithm. Q will be a random value depend on the environment, however R will be found using the basic equation of standard deviation, there is:

$$\bar{x} = (x_0 + x_1 + x_2 + x_3 + x_4):5$$

$$R = \left(\frac{\sqrt{z_k - \bar{x}}}{5}\right)^2 \quad (4.15)$$

We take 5 samples to find the standard deviation. 5 samples were the most appropriate among all. If we sample more than 5, a minor change will not affect the system.

## 5. RESULT AND ANALYSIS

In this section, we present the simulation and analysed the data. There are 2 types of data which will be analysed, consist of filter data and PID parameters. MATLAB R2013a is used to monitoring the data in graphs.

Kalman filter need to analyse before the control system. According to Figure 4, Kalman filter has fast responses, however Kalman filter can search the object whether the object is not detected. Kalman filter will search the object depend on the last velocity of the object. The filter is also used to reduce noises (simulated by shaking the object), nevertheless there are no noises in system, so it will not be used.

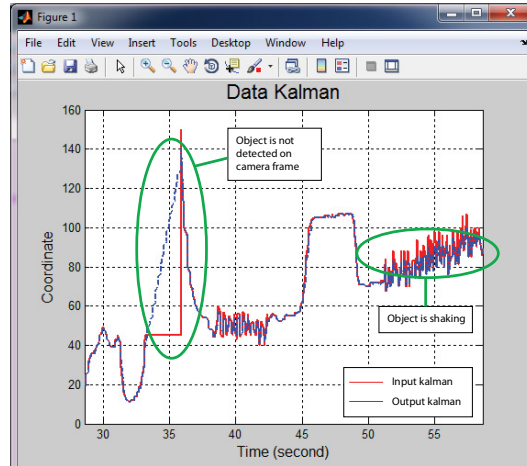


Figure 4. Kalman response (first tuning)

We determine the  $Q$  equal to 5.0 and  $R$  equal to standard deviation for each 5 sampling data. If we increase  $Q$  to 20, we will get a faster response than before as shown in Figure 5. On actual process we implemented the dynamic camera (camera is moving), a slow responses Kalman will only helpful in the static camera. Fast responses are not good enough in dynamic camera, because plant will move the camera outside the moving range harshly if the object is not detected.

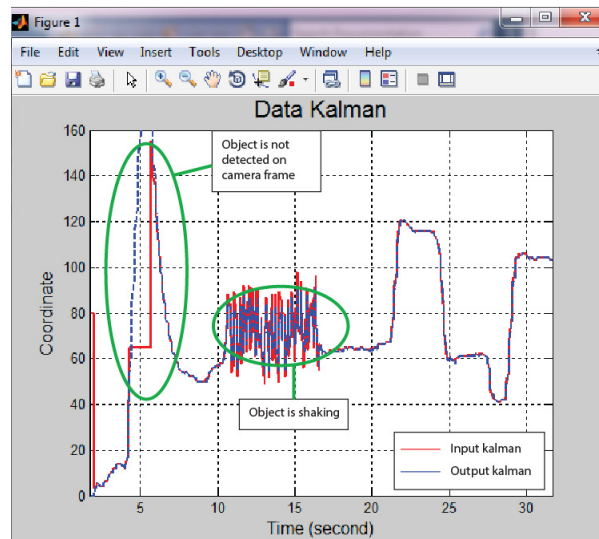


Figure 5. Kalman response (second tuning)

So, we can conclude that  $Q$  should equal to 5.0. After the value of  $Q$  has been fixed, the control system will be designed. According to the section 3, we will use a digital PID (Proportional, Integral, and Derivative) controller in bilinear transform.

First of all, the controller will be designed using Ziegler-Nichols type 2. We need to find the ultimate gain using  $P$  controller in this method, we should made the system oscillating constantly,



and then we got two variables, such as ultimate gain and Period of oscillation. After that, we will find the PID parameters using these equations:

$$Kp = 0.6 \times Ku \quad (5.1)$$

$$Ti = Pu : 2 \quad (5.2)$$

$$Td = Pu : 8 \quad (5.3)$$

According to the measurement using the equation 5.1 until equation 5.3, we got the PID parameter shown in Table 1 for horizontal axis and Table 2 for vertical axis.

Table 1. PID (Horizontal Axis) Parameter using Z-N method

KP	Ti	Td	KI	KD
2.188	1.051923	0.075938	2.08	0.318026

Table 2. PID (Vertical Axis) Parameter using Z-N method

KP	Ti	Td	KI	KD
1.902	0.36413	0.100342	5.2234	0.37146

We control motor servo, which has its own controller. After some experiments, we conclude that PD control law should be used to control the position of motor servos, so Ki is changed to 0. Control system moves 2 motor servo, such as:

- Motor servo in horizontal axis
- 

Figure 5 (A) shows the system responses using Kalman filter. There are a lot of noises because the environment condition which affect the image processing, furthermore there is a state where the object moving outside the camera frame, then Kalman filter try to find the object. Table 3 is the result of system's response in horizontal axis using Kalman filter.

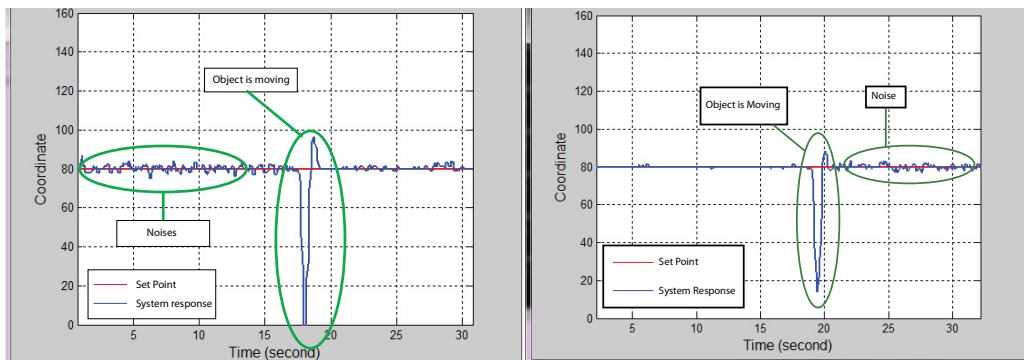


Figure 6. (A) PID Response (horizontal axis) (B) PID Response without Kalman Filter

Table 3. PID (Horizontal Axis) Parameter after Manual Tuning

Kp	Ki	Kd	Settling Time	Rise Time	Overshoot
1.188	0	0.318	1.63	0.402	16.49%

Table 4. PID (Horizontal Axis) Parameter after Manual Tuning without Kalman Filter

Kp	Ki	Kd	Settling Time	Rise Time	Overshoot
1.188	0	0.318	1.38	0.39	5.43%

Figure 6 (B) is a PID response without Kalman filter. However, the system without Kalman filter has a faster response which has smaller settling time, rise time, and overshoot.

- Motor servo in vertical axis

Figure 7(A) is the system's response of motor servo in vertical axis. According to the Table 5, system has a better result than horizontal axis with a smaller overshoot. There are also noises in the system, which has a same problem with motor servo in horizontal axis.

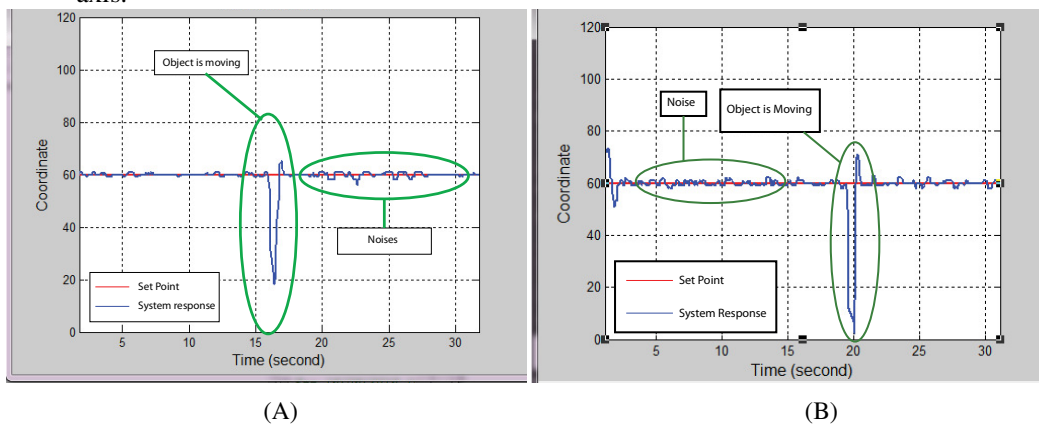


Figure 7. (A) PID Response (vertical axis) (B) PID Response without Kalman Filter

Table 5. PID (Vertical Axis) Parameter after Manual Tuning

Kp	Ki	Kd	Settling Time	Rise Time	Overshoot
1.402	0	0.274	1.34	0.38	11.9%

Table 6. PID (Vertical Axis) Parameter after Manual Tuning Without Kalman Filter

Kp	Ki	Kd	Settling Time	Rise Time	Overshoot
1.402	0	0.274	1.1700	0.47	18.9%

Meanwhile, system response without Kalman filter has been taken and it has a faster response than using Kalman filter. We can see on the Table 6 that a system without Kalman filter has a smaller settling time, but bigger overshoot.

According to the explanation, the response without Kalman filter is better than the otherwise. The reason is the feedback is not the real-time position, but it is the predictive position. It was predicted based on the object movement, therefore the response will be slower.

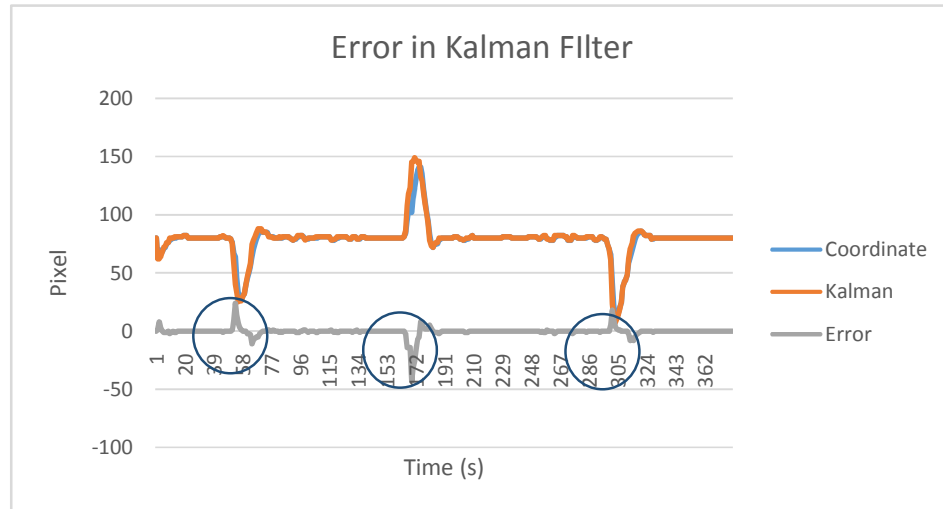


Figure 8. Error in Kalman Filter Graph

Figure 8 explain about the error between the real coordinate of the object and Kalman's output in x axis. We can see in some points, there are errors between real coordinate and Kalman's output. The error happened when the object is moving. The error graphs have an overshoot, which is signed by a blue circle, the reason is Kalman filter need to adapt with the environment. It won't happened if the camera was in static condition. It is the reason why the settling time was slower than without using Kalman filter.

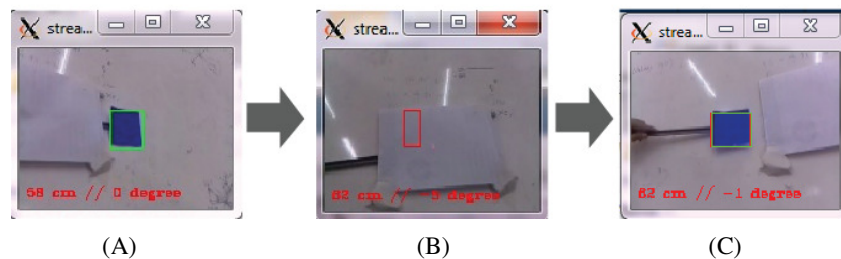


Figure 9. Object Movement (A) Initial State (B) Predict the Object (C) Final State

The benefit of Kalman filter is able to predict the object movement, which is illustrated in Figure 9. Kalman output is signed by red square, and the object is green square. On the Figure 9(B), the object was blocked by another object and the system can't detect the object, however the system still predicted the movement. Although it has no better response, kalman filter can be used in this state and the object position can be estimated while the object was blocked.

## 6. SUMMARY AND CONCLUSION

The paper proposed the designed system which is easier to implement. It was written in a more practical way and less numerical theory. The final response we can get is settling time equal to 1.34s and rise time 0.38 for vertical axis. Beside, the horizontal axis has settling time and rise time in 1.603s and 0.402s.

The result shows the difference between a system using Kalman filter and without Kalman filter. We have a faster and better result when it is not using Kalman filter. However, there are some advantages for the system using Kalman filter, such as:

- Control system can predict the object movement when the object moves to outside the camera frame.
- Kalman filter can estimate the object position when the object moves behind another object.

So, the conclusion is the system will be better using kalman filter. In the implementation, the system will be used to aim an object, and the ability to predict the object movement is really needed. It has a slower settling time than without using kalman filter, because the system need to adapt to the environment when the camera is moving.

## ACKNOWLEDGEMENTS

Thank you for all the parties involved. To our friends in Electronics Engineering of Bandung State of Polytechnic, lecturers, and my beloved parents. This paper won't finish completely without all of their help and support. I hope, this paper could be a trigger for many students and contribution to the technology development in the international world.

## REFERENCES

- [1] Garibotto, Giovanni & Buemi, Francesco (2015) *Object Detection and Tracking from fixed and mobile platforms*, ..., ....
- [2] Parmar, D.N. & Mehta, B. B. (2013) "Face Recognition Methods & Applications", International Journal Computer Technology and Applications, Vol. 4, No.1, pp84-87.
- [3] Thuiot, Benoit. Martinet, Philippe & Cordesses, Lionel (2002) *Position based visual servoing : keeping the object in the field of vision*, International Conference on Robotics & Automation, IEEE.
- [4] Burlacu, Adrian & Lazar, Corneliu (2008) *Image Based Controller for Visual Servoing Systems*, Buletinul Institutului Politehnic Din Iasi, Universitatea Tehnica „Gheorghe Asachi” din Iasi.
- [5] Mariottini, G. Luca. Oriolo, Giuseppe & Prattchizzo, Domenico (2007) "Image-Based Visual Servoing for Nonhonomic Mobile Robots Using Epipolar Geometry", *IEEE Transactions on Robotics*, Vol. 23, No.1, pp87-100.
- [6] Funk, Nathan (2003) *A Study of the Kalman Filter Applied to Visual Tracking*, Project for CMPUT 652, University of Alberta.
- [7] Fadali, M. Sami & Visioli, Antonio (2013) *Digital Control Engineering Analysis and Design Second Edition*,..., Elsevier Inc.
- [8] Welch, Greg & Bishop, Gary (2001) *An Introduction to the Kalman Filter*, University of North Carolina, ACM, Inc.

## **AUTHORS**

### **Fajar Abdurrahman**

Born in 1994 and started to study in Bandung State of Polytechnic since 2012. He worked in a business consultant from 2013 – 2015 as a part timer job, however he quitted from his job and started to focus on his education and electronic research since the middle of 2015. He will graduate from his university in 2016 of August.



### **Dr. YB Gunawan Sugiarta**

Born in 1961 and currently work as a lecture in Bandung State of Polytechnic. Granted title from Bandung Institute of Technology (ITB) as follow: Electronics Engineering (1999), Field Programmable Gate Array System Design (2003), Image Processing System Design (2009), and currently contributing in research program on image processing design



### **Feriyonika ST., M.Sc.Eng**

Born in 1985 in Sumatra, Indonesia. Studied as the electrical engineering in Gadjah Mada University. Then, he granted master degree in control engineering at National Taiwan University of Science and Technology in 2009. Currently worked and contributing in research program on instrument and control system in Bandung State of Polytechnic.

