

## DETEKSI BOLA PADA KONVERSI POIN PERMAINAN TENIS MEJA BERBASIS VIDEO PROCESSING

### BALL DETECTION OF POINT CONVERSION IN TABLE TENNIS GAME BASED ON VIDEO PROCESSING

Ivan Favian<sup>1</sup>, Ratri Dwi Atmaja<sup>2</sup>, Azizah<sup>3</sup>

<sup>1,2,3</sup>Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom

[iv.favian@gmail.com](mailto:iv.favian@gmail.com)<sup>1</sup> [ratridwiatmaja@telkomuniversity.ac.id](mailto:ratridwiatmaja@telkomuniversity.ac.id)<sup>2</sup> [azizah@telkomuniveristy.co.id](mailto:azizah@telkomuniveristy.co.id)<sup>3</sup>

#### Abstrak

Pendeteksian pergerakan bola pada tenis meja dengan metode *background subtraction and estimation* adalah satu dari sekian banyak metode yang dipakai. Penentuan kondisi bola dengan *Kalman filter*, yaitu proses deteksi dan koreksi yang digunakan sangat cocok pada metode ini. Deteksi dan *tracking* bola dengan *background subtraction and estimation based on Kalman filter*, membuat penelitian tugas akhir ini memiliki keunikan dari metode yang lain.

Pada tugas akhir ini dirancang suatu program pengolahan video untuk mendeteksi pergerakan dan menentukan kondisi bola dari permainan tenis meja. Metode *background subtraction and estimation* digunakan sebagai pemisah antara *background* dan *foreground*, sehingga didapatkan objek yang akan di deteksi. Pendeteksian dan *tracking* bola dengan *Kalman filter*, sehingga sistem dapat menentukan bola saat melewati *net* dan garis.

Pada deteksi bola berwarna oranye, saat jarak 1 meter dengan luas area 250 *pixels* menghasilkan akurasi sebesar 100%. Deteksi bola melewati garis, terdapat pada luas area 500 *pixels* dan 250 *pixels* dengan akurasi 100%. Sedangkan deteksi bola melewati *net*, terdapat pada luas area 500 *pixels* dengan akurasi 81,81%.

**Kata kunci :** pengolahan video, deteksi gerakan, tenis meja, *background subtraction*, *Kalman filter*.

#### Abstract

*Detection of moving ball on table tennis with background subtraction and estimation method is one of the many methods used. Determine ball conditions with the Kalman filter, the detection and correction process is very suitable on this method. Detection and tracking of balls with subtraction background and estimation based on Kalman filter, making this final research has unique way than the other methods.*

*In this final project designed a video processing program to detect movement and determine the condition of the ball from table tennis game. Background subtraction and estimation method is used to separate between the background and foreground, so that the object will be detected. Detection and tracking of the ball with a filter calm, allowing the system to determine the ball as it passes through the net and the line.*

*In the detection of orange ball, at 1 meter distance and area 250 pixels get 100% accuracy. Ball detection when passing line, could be at area 250 pixels and 500 pixels with accuracy 100%. Ball detection when passing the net at area 500 pixels with accuracy 81,81%.*

**Keywords:** video processing, detection, table tennis, *background subtraction*, *Kalman filter*.

## 1. Pendahuluan

### 1.1 Latar Belakang

Kemajuan teknologi memicu manusia untuk terus menciptakan inovasi yang dapat memberikan kemudahan dan solusi dalam berbagai masalah. Kontroversi gol pada sejumlah pertandingan seperti liga utama Inggris, piala dunia FIFA 2010, dan Euro 2012, membuat FIFA melakukan pengujian suatu teknologi garis gawang. Teknologi ini nantinya berperan untuk membantu wasit dalam membuat keputusan akhir, teknologi ini diuji pada pertandingan piala dunia antar klub FIFA 2012 di Jepang [1].

Terkait dengan peran untuk membantu wasit dalam membuat keputusan akhir, dalam penelitian [6] dilakukan penelitian deteksi bola dan perhitungan poin pada tenis meja. Penelitian menggunakan bola berwarna hijau dan dilakukan secara *real time*, *hardware* yang dirancang untuk menunjang sistem dipakai pada penelitian tersebut. Penelitian [7] menjelaskan tentang pendeteksian bola dan pemain pada pertandingan sepak bola menggunakan *dynamic Kalman filter* dengan *velocity control*. Pendeteksian objek menggunakan *background subtraction* yang memisahkan latar depan dan latar belakang.

Tugas akhir ini terkait tentang penelitian pendeteksian bola pada permainan tenis meja. Pada permainan tenis meja dilakukan pendeteksian bola untuk kondisi bola berada di area lapangan, saat bola melewati *net*, dan saat bola melewati cakupan dari kamera. Mengandalkan *prediction and correction* dari *Kalman filter* dan metode *background subtraction* dimana *foreground* dan *background* dipisahkan. Metode pendeteksian tersebut sangat tepat untuk kasus ini yang hanya mendeteksi bola tanpa pemainnya.

Penelitian tugas akhir ini menggabungkan beberapa aspek yang berada pada kedua penelitian terkait yaitu pendeteksian bola menggunakan *computer vision* dan *Kalman filter*. Pada penelitian ini perangkat lunak yang digunakan yaitu Matlab 2016a yang berbeda dengan penelitian [6], tetapi masih ada beberapa aspek yang sama yaitu metode *background subtraction* dan *Kalman filter* dengan mengandalkan prediksi dan koreksinya sebagai penentuan perhitungan.

## 1.2 Rumusan Masalah

Permasalahan yang menjadi obyek penelitian serta pembahasan dalam tugas akhir ini adalah :

1. Bagaimana mengimplementasikan sistem agar dapat diterapkan pada studi kasus ini.
2. Bagaimana menggunakan pengolahan video dalam menangani pendeteksian bola yang bergerak.
3. Bagaimana perhitungan yang dilakukan dalam hasil akhir pendeteksian bola.

## 1.3 Tujuan

Tujuan dari Tugas Akhir ini adalah :

Mengimplementasikan sistem berbasis pengolahan video dengan kalman filter dalam pendeteksian bola dan menentukan pergerakan bola ping-pong pada permainan tenis meja maupun simulasi permainan tenis meja.

## 1.4 Metode Penelitian

Metodologi dalam proses penyelesaian penelitian ini terdiri dari beberapa tahapan yaitu :

1. Studi literatur  
Mempelajari dasar teori-teori yang digunakan dari berbagai literatur mengenai penerapan metode dipakai. Langkah-langkah yang digunakan dalam sistem ini, diantaranya *input video*, *detection and tracking*, dan terakhir yaitu perhitungan.
2. Perancangan sistem  
Pada tahap ini dilakukan perancangan untuk mengimplementasi system. video masukan yang akan diolah, dipisahkan *background* dan *foreground* menggunakan metode *background subtraction and estimation*. Proses selanjutnya yaitu *tracking* dengan *Kalman filter* agar bola dapat ditentukan prediksinya. Tahapan terakhir merupakan perhitungan terhadap deteksi pergerakan bola, bola melewati *net*, dan bola melewati batas.
3. Pengumpulan data  
Mengumpulkan data yang mendukung analisis berbasis video digital. Melakukan pengambilan video yang sesuai dengan kategori yang ditentukan untuk dijadikan bahan uji. *Dataset* yang diperoleh didapatkan dari rekaman dengan kamera digital.
4. Implementasi metode dalam sistem  
Menerapkan rancangan sesuai metode yang digunakan yang sudah dibuat kedalam perangkat sistem.
5. Pengujian sistem dalam aplikasi  
Melakukan uji coba terhadap sistem yang telah diimplementasikan menggunakan data video yang dikumpulkan, sehingga dapat mengetahui performansi sistem terhadap data uji yang mendekati kondisi asli dengan hasil yang paling memuaskan.
6. Evaluasi sistem dalam aplikasi  
Melakukan evaluasi dan analisis dari hasil pengujian terhadap sistem, apakah sistem sudah mampu bekerja sesuai dengan tujuan perancangan. Hasil evaluasi dapat menentukan langkah selanjutnya yang dilakukan terhadap sistem hingga mencapai tingkat performansi sistem yang tinggi.
7. Penarikan kesimpulan  
Menarik kesimpulan dari pengujian sistem yang sudah di lakukan.

## 2. Dasar Teori

### 2.1 Teori kalman filter [8]

Hampir tidak mungkin untuk memahami sepenuhnya makna dari *Kalman filter* mulai definisi hingga persamaan yang rumit. Persamaan dibawah ini bisa dibilang tidak serumit *Kalman filter* pada umumnya.

$$\hat{\mathbf{X}}_k = \mathbf{K}_k \cdot \mathbf{Z}_k + (\mathbf{1} - \mathbf{K}_k) \cdot \hat{\mathbf{X}}_{k-1} \quad (2.1)$$

Tujuan dari persamaan diatas adalah untuk menemukan  $\hat{\mathbf{X}}_k$ , *current estimation* dari sinyal  $\mathbf{x}$  dan menemukan setiap konsekuensi dari  $\mathbf{k}$ . Pada persamaan 2.1, diketahui  $\mathbf{k}$  adalah *discrete time intervals*,  $\mathbf{Z}_k$  adalah *measurement value*,  $\mathbf{K}_k$  disebut sebagai "*Kalman Gain*" dan  $\hat{\mathbf{X}}_{k-1}$  adalah *previous estimation* atau estimasi sinyal dari *state*

sebelumnya. Komponen yang tidak diketahui dari persamaan adalah  $K_k$  Kalman gain, komponen yang lain seperti  $Z_k$  dan  $\hat{X}_{k-1}$  sudah dimiliki.

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \quad (2.2)$$

$$z_k = Hx_k + v_k \quad (2.3)$$

Variabel acak  $w_k$  dan  $v_k$  mewakili *process noise* dan pengukuran,  $u_k$  mewakili *signal control*. Terdapat empat kategori utama dalam *Kalman filter* yang berhubungan dengan metode *Background estimation and subtraction* yaitu *time evolution*, *local structure*, *local time evolution*, dan *feedback time evolution*. Semua terfokus pada sebuah tujuan, yaitu menghasilkan gambar latar belakang yang diberikan citra itu terus-menerus dan memiliki konten latar depan. Persamaan yang digunakan berasal dari prediksi dan koreksi persamaan *Kalman filter*.

$$\hat{x}_k = A\hat{x}_{k-1} + Bu_k \quad (2.4)$$

$$\hat{P}_k = A\hat{P}_{k-1}A^T + Q \quad (2.5)$$

$$K_k = \hat{P}_k H^T (H\hat{P}_k H^T + R)^{-1} \quad (2.6)$$

$$\hat{x}_k = \hat{x}_k + K_k(z_k - H\hat{x}_k) \quad (2.7)$$

$$P_k = (1 - K_k H)\hat{P}_k \quad (2.8)$$

dimana :

A merupakan matriks penghubung *time step* sebelumnya ke *time step* saat ini.

B merupakan matriks yang menghubungkan *input* opsional ke *state*.

Q mewakili proses *noise* matriks kovarian.

H merupakan matriks yang menghubungkan *state* dengan pengukuran.

R merepresentasikan pengukuran *noise* matriks kovarian.

x adalah keadaan variabel dengan k saat ini dan k sebelumnya.

u adalah variabel kontrol dengan k saat ini.

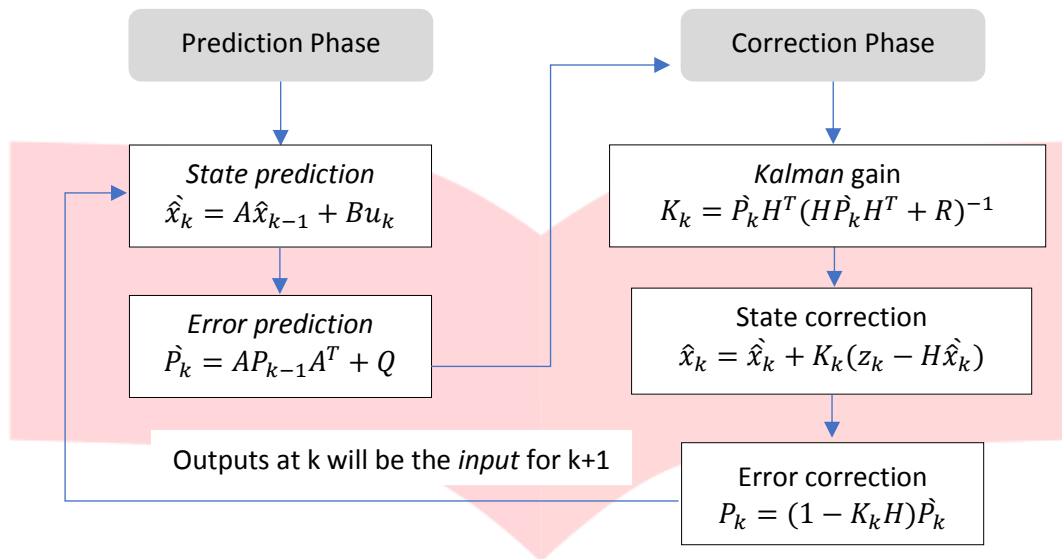
z adalah pengukuran dengan k saat ini.

Simbol A, B, dan H berada dalam matriks bentuk umum, namun dalam sebagian besar masalah pemrosesan sinyal, dapat menggunakan model sedemikian rupa sehingga simbol ini hanya nilai numerik. Berikut adalah *Time update* (prediksi) dan *Mesurement update* (koreksi) yang dimana jika persamaan diatas dikelompokan berdasarkan prediksi dan koreksinya yang bisa dilihat pada Tabel 2.1.

**Tabel 2.1** Persamaan *Time Update* dan *Measurement Update* [10]

<i>Time update</i> (prediction)	<i>Measurement update</i> (correction)
$\hat{x}_k = A\hat{x}_{k-1} + Bu_k$	$K_k = \hat{P}_k H^T (H\hat{P}_k H^T + R)^{-1}$
$\hat{P}_k = A\hat{P}_{k-1}A^T + Q$	$\hat{x}_k = \hat{x}_k + K_k(z_k - H\hat{x}_k)$
	$P_k = (1 - K_k H)\hat{P}_k$

Gambar 2.1 merupakan urutan kerja dari suatu *Kalman filter* yaitu prediksi dan koreksi. pada *prediction phase*, *input* saat k akan diolah untuk di prediksi keadaannya di *state prediction*, lalu kejanggalan saat k akan diprediksi di *error prediction*. Tahap *correction phase* dimana *input* akan dikoreksi dengan menentukan *Kalman gain* terlebih dahulu, selanjutnya *state* dan *error* akan dikoreksi dan *output* saat k akan menjadi *input* untuk k+1.



Gambar 2.1 Perhitungan Kalman Filter[10]

Berikut adalah contoh perhitungan sederhana dari Kalman filter, perhitungan menggunakan matriks 2x2 dengan nilai yang sederhana dengan ketentuan sebagai berikut.

- Kita tidak memiliki control signal dan noise jadi  $u_k$  dan  $Q$  diluar dari perhitungan.
- Misal  $\hat{x}_0 = 0$  dan  $P_0 = 1$ .
- Misal matriks  $R = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$   $A = \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix}$   $B = \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix}$   $H = \begin{bmatrix} 1 & 2 \\ 1 & 0 \end{bmatrix}$
- Saat  $k=1$  nilai  $z_k = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$

Untuk  $k=1$

$$\hat{x}_k = A\hat{x}_{k-1} = \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix} \hat{x}_0 = \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix} \cdot 0 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\hat{P}_k = AP_{k-1}A^T = \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix} P_1 \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix} \cdot 1 \cdot \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 3 \\ 3 & 5 \end{bmatrix}$$

$$K_k = \hat{P}_k H^T (H \hat{P}_k H^T + R)^{-1} = \begin{bmatrix} 2 & 3 \\ 3 & 5 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 \\ 2 & 0 \end{bmatrix} \cdot \left( \begin{bmatrix} 1 & 2 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 2 & 3 \\ 3 & 5 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 \\ 2 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \right)^{-1} = \begin{bmatrix} 8 & 2 \\ 13 & 3 \end{bmatrix} \cdot \left( \begin{bmatrix} 5 & 5 \\ 8 & 6 \end{bmatrix} \right)^{-1} = \begin{bmatrix} 8 & 2 \\ 13 & 3 \end{bmatrix} \cdot \left( \frac{1}{30-40} \cdot \begin{bmatrix} 6 & -5 \\ -8 & 5 \end{bmatrix} \right) = \begin{bmatrix} 8 & 2 \\ 13 & 3 \end{bmatrix} \cdot \begin{bmatrix} -0.6 & 0.5 \\ 0.8 & -0.5 \end{bmatrix} = \begin{bmatrix} -2.6 & 3 \\ -5.4 & 5 \end{bmatrix}$$

$$\hat{x}_k = \hat{x}_k + K_k(z_k - H\hat{x}_k) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -2.6 & 3 \\ -5.4 & 5 \end{bmatrix} \cdot \left( \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} - \begin{bmatrix} 1 & 2 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \right) = \begin{bmatrix} -2.6 & 3 \\ -5.4 & 5 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 0.4 & 0.4 \\ -0.4 & -0.4 \end{bmatrix}$$

$$P_k = (1 - K_k H) \hat{P}_k = \left( 1 - \begin{bmatrix} -2.6 & 3 \\ -5.4 & 5 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 1 & 0 \end{bmatrix} \right) \cdot \begin{bmatrix} 2 & 3 \\ 3 & 5 \end{bmatrix} = \left( \begin{bmatrix} 2 & 3 \\ 3 & 5 \end{bmatrix} - \begin{bmatrix} 0.4 & -5.2 \\ -0.4 & -10.8 \end{bmatrix} \right) = \left( \begin{bmatrix} 2 & 3 \\ 3 & 5 \end{bmatrix} - \begin{bmatrix} 0.4 & -5.2 \\ -0.4 & -10.8 \end{bmatrix} \right) = \begin{bmatrix} 1.6 & 8.2 \\ 3.4 & 15.8 \end{bmatrix}$$

## 2.2 Background [11]

### 2.2.1 Background Subtraction

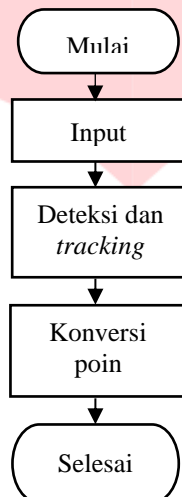
Metode standar untuk memperoleh latar belakang adalah penangkapan gambar secara manual dengan operator yang mengambil gambar tersebut dan menentukan latar belakangnya. Tetapi metode ini memiliki 3 kekurangan, yang pertama membutuhkan operator untuk pengamatan secara aktif. Kedua, operator mungkin bisa membedakan latar belakang dan latar depan mungkin juga tidak. Terakhir, jika berhasil dibedakan peluang latar depan dan latar belakang mungkin bisa tertukar dengan komponen yang lain. Pengalaman pengamatan meunjukkan latar belakang mungkin akan menangkap objek yang tidak diinginkan dari latar depan, karena pemisahan latar ini belum cukup jelas.

### 2.2.2 Background Estimation

Latar belakang dapat diperkirakan untuk membantu meringankan sebagian masalah yang muncul dari metode standar. Pendekatan paling sederhana adalah *pixel* dengan *pixel* pada skema estimasi statistik intensitas. Versi sederhana dari struktur ini adalah *windowing* memanfaatkan  $N$  *frame* sebelumnya. Meskipun sangat sederhana dan lempeng, *windowing* memiliki temporal sebelumnya yang dimana kedalaman *window* menyebabkan masalah terkait dengan sejumlah besar memori eksternal sebagai sumber daya yang diperlukan untuk mempertahankan dan mengambil seluruh *window* sebelumnya. Algoritma akan beradaptasi secara temporal mengubah latar belakang dan tidak memerlukan campur tangan pengguna tetapi masih memiliki pemisahan yang belum sempurna antara latar depan dan latar belakang.

## 3. Perancangan Sistem

### 3.1 Skema Perancangan



Gambar 1.1 Flowchart sistem pengujian

Langkah-langkah sistem pengujian dimulai dari *input* video yang sudah direkam, lalu dilakukan proses *detection and tracking* untuk mendeteksi objek, setelah itu dilakukan proses perhitungan guna mengukur akurasi dari sistem ini.

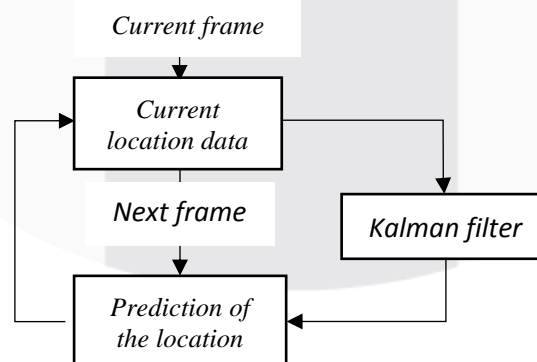
#### 3.2.1 Input Video

*Input* dari sistem adalah sebuah video berukuran maksimal 1280x720 yang sudah sedemikian rupa ditambahkan objek tertentu sehingga bisa menjadi *input* yang diinginkan.

#### 3.2.2 Detection and Tracking

*Background subtraction and estimation* adalah proses pemisahan latar belakang dengan latar depan, latar belakang merupakan bagian dari suatu *frame* yang tetap tidak ada objek yang bergerak, latar depan yang diambil berupa *frame* yang ada objek di dalamnya yaitu bola.

Gambar 3.6 merupakan blok diagram proses kerja dari *Kalman filter*. *Kalman filter* merupakan suatu *filter* dimana terdapat proses *correction and prediction* di dalam nya. Sebuah *frame* yang akan diproses dengan *Kalman filtering*, akan melalui proses *correction and prediction*. Keluaran dari proses ini akan dijadikan masukan untuk *frame* selanjutnya.



Gambar 3.2 Blok Diagram Kalman Filter [10]

### 3.2.3 Konversi Poin

Perhitungan blok diagram di atas dibagi menjadi 3 kasus yaitu:

1. Bola melewati garis

Kasus pertama, perhitungan dilakukan saat bola berhasil di deteksi ketika melewati garis. Label akan berubah menjadi warna merah dan akan diberi nilai 1 untuk mengukur akurasi. Label akan tetap berwarna hijau jika sistem gagal mendeteksi dan akan diberi nilai 0.

2. Bola melewati *net*

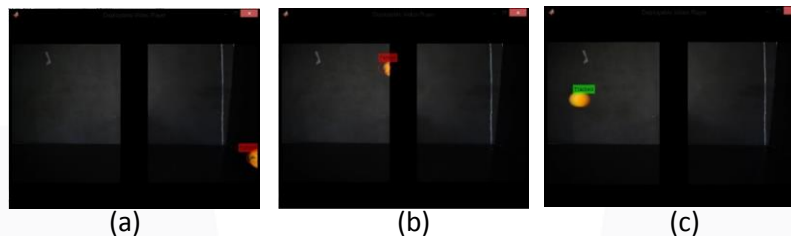
Tidak berbeda jauh dengan poin 1, pada poin ini pendeteksian hanya dilakukan pada *net* dengan perhitungan yang sama. Label akan berubah menjadi warna merah dan akan diberi nilai 1 untuk mengukur akurasi. Label akan tetap berwarna hijau jika sistem gagal mendeteksi dan akan diberi nilai 0.

3. Pergerakan bola

Penilaian terdeteksinya bola pada poin ini cukup berbeda dari poin 1 dan 2. Saat bola bergerak sistem akan mendeteksi bola dengan label berwarna hijau, jika dari awal pendeteksian hingga akhir label berada pada bola maka diberikan nilai 1. Label akan berwarna merah atau menghilang ketika gagal melakukan deteksi dan akan diberikan nilai 0.

### 4. Hasil dan Analisis

*Output* pada sistem ini berupa video yang menunjukkan bola ping-pong melewati sebuah garis yang berada pada video. *Kalman filter* digunakan pada proses ini, sehingga pada Gambar 4.1 terdapat hasil saat bola berada pada cakupan kamera dan di deteksi dengan sempurna. Gambar 4.1 (a) merupakan objek melewati garis. Gambar 4.1 (b) menunjukkan objek melewati *net*, sedangkan Gambar 4.1 (c) merupakan hasil bola terdeteksi.



**Gambar 4.1** Hasil Program (a) Bola Melewati Garis (b) Bola Melewati *Net* (c) Deteksi Bola

Percobaan deteksi bola pada permainan tenis meja dilakukan dengan mengambil 2 sampel, yaitu rekaman pertandingan dan simulasi. Simulasi dilakukan oleh diri sendiri tanpa adanya bantuan dari pemain lain, hanya mengandalkan dinding untuk membuat bola memantul. Percobaan dilakukan dengan mengganti luas daerah, agar bola dapat di *tracking* dan diketahui apakah bola terdeteksi saat melewati garis.

Percobaan pertandingan diambil pada jarak 1 meter dari meja, sedangkan simulasi diambil pada jarak 0,5 meter. Pada deteksi bola permainan tenis meja, dilakukan 2 percobaan menggunakan bola berwarna oranye dan bola berwarna putih. Deteksi bola pada simulasi tenis meja diuji dengan bola berwarna oranye. Setelah didapat nilai dari percobaan, dilakukan perhitungan akurasi dengan persamaan 4.1. Sedangkan pada konversi poin, video yang diuji adalah video permainan yang memiliki kondisi pemain mendapat poin, artinya ketika bola terdeteksi pada *net* dan garis atau garis saja akan mendapat 1 poin.

$$Akurasi = \frac{\text{jumlah percobaan berhasil}(n)}{\text{jumlah percobaan}} \times 100\% \quad (4.1)$$

#### 4.1 Deteksi Bola Permainan Tenis Meja pada Jarak 1 Meter

a. Percobaan pendeteksian bola oranye.

Deteksi sempurna diberi nilai 1, sedangkan deteksi yang belum sempurna dan tidak ada deteksi diberi nilai 0. Angka digunakan agar mempermudah perhitungan akurasi, akurasi di dapat dengan menggunakan persamaan 4.1 dan hasil akurasi dapat dilihat pada Tabel 4.1.

**Tabel 4.1** Akurasi Bola Oranye pada Jarak 1 Meter

	Garis	<i>Net</i>	Bola
Luas area 250 <i>pixels</i>	100%	54.54%	100%
Luas area 500 <i>pixels</i>	100%	81.81%	36.36%
Luas area 750 <i>pixels</i>	63.63%	36.36%	18.18%
Luas area 1000 <i>pixels</i>	18.18%	9.09%	27.27%

Percobaan pada jarak kamera 1 meter dari meja ping-pong dan didapat hasil akurasi pendeteksian. Akurasi terbaik deteksi pada garis berada di area 250 dan 500 *pixels* yaitu sebesar 100%, akurasi terbaik deteksi pada *net* berada di area 500 *pixels* sebesar 81,81%. Pada luas area 750 dan 1000 *pixels*, akurasi dikatakan buruk karena sebagian besar berada pada nilai di bawah 50%.

Akurasi dan luas area deteksi seharusnya berbanding terbalik, semakin besar luas area maka semakin kecil akurasinya. Pada akurasi bola oranye dengan jarak 1 meter, didapat hasil 54,54% untuk *net* dengan luas area 250 *pixels* dan 81,81% dengan luas area 500 *pixels*. Pada luas area 250 *pixels* saat bola melewati garis, terdapat area putih sehingga sistem mendeteksi pergerakan bolanya bukan mendeteksi bola melewati *net*. Kejadian tersebut yang membuat akurasi *net* pada luas area 500 *pixels* lebih bagus dibandingkan dengan 250 *pixels*.

b. Percobaan pendeteksian bola putih.

Pada percobaan b dilakukan pendeteksian menggunakan bola putih dengan jarak 1 meter. Deteksi sempurna diberi nilai 1, sedangkan deteksi yang belum sempurna dan tidak ada deteksi diberi nilai 0. Hasil akurasi dapat dilihat pada Tabel 4.2.

**Tabel 4.2** Akurasi Bola Putih pada Jarak 1 Meter

	Garis	Net	Bola
Luas area 250 <i>pixels</i>	72,72%	81,81%	36,36%
Luas area 500 <i>pixels</i>	81,81%	72,72%	0%
Luas area 750 <i>pixels</i>	9,09%	9,09%	0%
Luas area 1000 <i>pixels</i>	0%	9,09%	0%

Percobaan pada jarak kamera 1 meter dari meja, didapat hasil akurasi deteksi bola sebesar 36.36% dengan luas area 250 *pixels*, akurasi deteksi pada *net* didapat 81.81% dengan luas area 250 *pixels*, sedangkan akurasi terburuk didapat pada area deteksi 750 dan 1000 *pixels*.

Akurasi dan luas area deteksi seharusnya berbanding terbalik, semakin besar luas area deteksi maka semakin kecil akurasinya. Hasil akurasi bola putih pada jarak 1 meter, didapat hasil 72,72% untuk garis di luas area 250 *pixels* dan 81,81% pada luas area 500 *pixels*. Pada luas area 250 *pixels* saat bola melewati garis, terdapat area putih sehingga sistem mendeteksi pergerakan bola bukan mendeteksi bola melewati *net*. Hal tersebut yang membuat hasil akurasi garis pada area 500 *pixels* lebih bagus dari 250 *pixels*.

#### 4.2 Deteksi Bola Simulasi Tenis Meja pada Jarak 0,5 Meter

Pada kasus ini, dilakukan simulasi permainan tenis meja yang dibuat dari meja kecil sebagai lapangan. Percobaan dilakukan dengan menggunakan bola oranye saja pada jarak 0,5 meter dan mengganti luas area deteksinya. Deteksi sempurna diberi nilai 1, sedangkan deteksi yang belum sempurna dan tidak ada deteksi diberi nilai 0. Akurasi pada setiap komponen dihitung dengan menggunakan persamaan 4.1.

**Tabel 4.3** Akurasi Bola Oranye pada Jarak 0,5 Meter

	Garis	Net	Bola
Luas area 250 <i>pixels</i>	90.90%	63.63%	81.81%
Luas area 500 <i>pixels</i>	90.90%	81.81%	81.81%
Luas area 750 <i>pixels</i>	81.81%	90.90%	90.90%
Luas area 1000 <i>pixels</i>	81.81%	100%	90.90%

Jarak kamera 0,5 meter dari meja didapat hasil akurasi pendeteksian bola. Akurasi pada garis, *net* dan bola berada pada nilai diatas 80% kecuali untuk *net* pada luas area 250 *pixels*. Akurasi pada di jarak 0,5 meter dapat disimpulkan sebagai deteksi yang sangat baik.

Akurasi dan luas area deteksi pada jarak 0,5 meter seharusnya berbanding lurus, semakin besar luas area deteksi maka semakin besar juga akurasinya. Hasil akurasi bola oranye pada jarak 1 meter, didapat hasil 90,90% untuk garis pada luas area 500 *pixels* dan 81,81% pada luas area 750 *pixels*. Pada kasus ini, dikarenakan ukuran bola lebih besar, menjadikan deteksi pergerakan bola sangat bagus tetapi membuat pendeteksian pada garis menjadi turun.

#### 4.3 Perhitungan Konversi Poin

Perhitungann poin pada kasus ini dilakukan dengan menggunakan 2 kondisi, yang pertama saat bola terdeteksi pada *net* dan garis, kedua saat bola hanya terdeteksi pada garis maka akan diberi poin. Video yang diujikan adalah video permainan yang pasti mendapatkan poin, data diambil dari hasil pengujian pada permainan dan simulasi tenis meja.

a. Konversi poin bola oranye pada jarak 1 meter

**Tabel 4.4** Konversi Poin Bola Oranye pada Jarak 1 Meter

	Luas area 250 <i>pixels</i>			Luas area 500 <i>pixels</i>			Luas area 750 <i>pixels</i>			Luas area 1000 <i>pixels</i>		
	Garis	Net	Poin	Garis	Net	Poin	Garis	Net	Poin	Garis	Net	Poin
1	1	0	1	1	1	1	1	1	1	0	0	0
2	1	1	1	1	1	1	1	1	1	0	0	0
3	1	1	1	1	1	1	0	0	0	0	0	0
4	1	0	1	1	1	1	1	1	1	0	0	0
5	1	0	1	1	0	1	1	0	1	0	0	0
6	1	1	1	1	1	1	1	0	1	1	0	1
7	1	1	1	1	1	1	0	0	0	0	0	0
8	1	1	1	1	1	1	0	0	0	0	0	0
9	1	0	1	1	1	1	0	0	0	0	0	0
10	1	1	1	1	1	1	1	0	1	1	1	1
11	1	0	1	1	0	1	1	1	1	0	0	0
Jumlah	11	6	11	11	9	11	7	4	7	2	1	2

b. Konversi poin bola putih pada jarak 1 meter

**Tabel 4.5** Konversi Poin Bola Putih pada Jarak 1 Meter

	Luas area 250 <i>pixels</i>			Luas area 500 <i>pixels</i>			Luas area 750 <i>pixels</i>			Luas area 1000 <i>pixels</i>		
	Garis	Net	Poin	Garis	Net	Poin	Garis	Net	Poin	Garis	Net	Poin
1	1	1	1	1	1	1	0	0	0	0	0	0
2	1	1	1	1	1	1	0	0	0	0	0	0
3	1	1	1	1	1	1	0	0	0	0	0	0
4	1	1	1	1	1	1	0	0	0	0	0	0
5	1	0	1	1	1	1	0	0	0	0	0	0
6	1	1	1	1	0	1	1	0	1	0	0	0
7	0	1	0	0	1	0	0	1	0	0	1	0
8	0	0	0	0	1	0	0	0	0	0	0	0
9	1	1	1	1	0	1	0	0	0	0	0	0
10	0	1	0	1	1	1	0	0	0	0	0	0
11	0	1	0	0	0	0	0	0	0	0	0	0
Jumlah	6	9	6	8	8	8	0	1	1	0	1	0

c. Konversi poin bola oranye pada jarak 0,5 meter

**Tabel 4.6** Konversi Poin Bola Oranye pada Jarak 0,5 Meter

	Luas area 250 <i>pixels</i>			Luas area 500 <i>pixels</i>			Luas area 750 <i>pixels</i>			Luas area 1000 <i>pixels</i>		
	Garis	Net	Poin	Garis	Net	Poin	Garis	Net	Poin	Garis	Net	Poin
1	1	1	1	1	1	1	0	1	0	0	1	0
2	1	0	1	1	0	1	1	0	1	1	1	1
3	1	1	1	1	1	1	1	1	1	1	1	1
4	1	0	1	1	1	1	1	1	1	1	1	1
5	1	1	1	1	1	1	1	1	1	1	1	1
6	1	1	1	1	1	1	1	1	1	1	1	1
7	0	1	0	0	1	0	0	1	0	0	1	0
8	1	0	1	1	0	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1	1	1	1	1	1
11	1	0	1	1	1	1	1	1	1	1	1	1
Jumlah	10	7	10	10	9	10	9	10	9	9	11	9



Penentuan masuk tidaknya bola pada perhitungan konversi poin memiliki hasil penentuan tiap-tiap poin. Skor tertinggi yang diperoleh poin a berjumlah 11 pada luas area 250 dan 500 *pixels*, skor tertinggi yang diperoleh poin b berjumlah 8 poin pada luas area 500 *pixels*, sedangkan skor tertinggi yang diperoleh poin c berjumlah 10 pada luas area 250 dan 500 *pixels*. Konversi poin diatas mengikuti deteksi bola pada garis dan *net*, dalam keadaan *real* bisa saja bola berhasil di deteksi tapi pemain berhasil melakukan pengembalian bola.

### 5.1 Kesimpulan

Sistem dapat mengimplementasikan pendeteksian bola pada studi kasus pendeteksian bola berbasis *Kalman filter*. Pada deteksi bola permainan tenis meja dengan jarak 1 meter, bola oranye mendapat akurasi sebesar 100% di luas area 250 dan 500 *pixels* untuk garis, dan 250 *pixels* untuk bola, sedangkan untuk *net* mendapat akurasi sebesar 81,81% pada luas area 500*pixel*. Akurasi terbaik pada bola putih bedarada di luas area 250 *pixels*, yaitu 36,36% untuk bola dan 81,81% untuk *net*, sedangkan garis berada pada luas area 500 *pixels* sebesar 81,81%.

Pada deteksi bola simulasi tenis meja, dilakukan pengambilan video pada jarak 0,5 meter. Akurasi terbaik berada pada *net* dengan luas area 1000 *pixels* sebesar 100%, untuk bola sebesar 90,90% berada pada luas area 750 dan 1000 *pixels*, sedangkan akurasi pada garis bernilai 90,90% di luas area 250 dan 500 *pixels*.

Pada konversi poin, terjadi proses penentuan masuk tidaknya bola pada permainan tenis meja. Skor terbanyak diperoleh pada permainan tenis meja dengan menggunakan bola oranye pada jarak 1 meter, yaitu berjumlah 11 poin. Skor terendah diperoleh pada permainan tenis meja dengan menggunakan bola putih pada jarak 1 meter, yaitu berjumlah 8 poin.

### 5.2 Saran

Sistem pendeteksian bola pada permainan tenis meja ini masih dapat dikembangkan sehingga hasil yang diperoleh lebih akurat. Beberapa saran dari penulis yang dapat digunakan untuk mengembangkan tugas akhir ini yaitu:

1. Gunakan kain berwarna hitam pada *background* sehingga pendeteksian bola tidak terganggu dengan unsur-unsur yang ada di sekitar.
2. Menggunakan metode yang berbeda untuk pendeteksian dan *tracking*, serta penambahan sistem skor otomatis.
3. Membuat sistem *real-time*.

### DAFTAR PUSTAKA

- [1] FIFA. 2012. "*FIFA Quality Programme for Goal-Line Technology*". International Football Association Board (IFAB).
- [2] Esma Bilgin. 2009. "*Kalman filter for dummies*" <http://bilgin.esme.org/BitsAndBytes/KalmanFilterforDummies>.
- [3] Singla Nishu. 2014. "*Motion Detection Based on Frame Difference Method*", International Journal of Information & Computation Technology. Volume 4, Number 15, pp. 1559-1565.
- [4] Mao Jinzi. 2006. "*Tracking A Tennis Ball Using Image Processing Techniques*", University of Saskatchewan.
- [5] Yu, X. and Farin, D. 2005. "*Current and emerging topics in sports video processing*", IEEE International Conference on Multimedia and Expo.
- [6] Cornell University. 2015. "*Table tennis game tracker*". <http://people.ece.cornell.edu/land/courses/ece5760/FinalProjects/s2015>.
- [7] Jong-Yun Kim, Tae-Yong Kim. 2009. "*Soccer Ball Tracking using Dynamic Kalman filter with Velocity Control*", Chung-Ang University.
- [8] Pal sumit kumar, ghorai sohan. 2013 "*Moving Object Tracking System In Video With Kalman filter*", IJERT.
- [9] Greg Welch, Gary Bishop. 2006. "*An Introduction to the Kalman filter*", University of North Carolina.
- [10] M.S.Grewal, A.P. Andrews. 2001 "*Kalman filtering - Theory and Practice Using MATLAB*", Wiley.
- [11] Scott Jesse, Pusateri Michael A, Cornish Duane. 2013 "*Kalman filter Based Video Background Estimation*", Penn State University.