

# IMPLEMENTASI PENGOLAHAN CITRA PADA ROBOT OTONOM PENGUMPUL BOLA TENIS MEJA

Fikri Refo Julianto<sup>1</sup>, Willy Anugrah Cahyadi<sup>2</sup>, Muhammad Zakiyullah Romdlony<sup>3</sup>

<sup>1</sup>Fakultas Teknik Elektro

<sup>2</sup>Fakultas Teknik Elektro dan

The University Center of Excellence for Advanced Intelligent Communications (AICOMS), Universitas Telkom, Bandung 40257, Indonesia

<sup>3</sup>IoT Center, Universitas Telkom

<sup>1</sup>aerofo@student.telkomuniversity.ac.id, <sup>2</sup>waczze@telkomuniversity.ac.id,

<sup>3</sup>zakiyullah@telkomuniversity.ac.id

Diterima pada 27 Oktober 2020; disetujui pada 22 Agustus 2022; dan diterbitkan pada 2 September 2022.

## Abstrak

Tenis meja merupakan salah satu cabang olahraga bola kecil yang diminati banyak orang. Dalam latihan tenis meja, seorang pemain akan menggunakan banyak bola sebagai alat latihan. Perlu usaha lebih bagi para pemain yang berlatih untuk mengumpulkan bola-bola yang berserakan setelah latihan. Ada banyak alat yang diciptakan untuk mempermudah pengambilan bola-bola tersebut, namun proses pengambilannya masih manual karena menggunakan tenaga pemain itu sendiri. Untuk meringankan usaha pemain, diperlukan suatu sistem yang otomatis dalam pengumpulan bola tenis meja. Hal itu akan diwujudkan dengan perancangan robot otonom pengumpul bola tenis meja. Robot ini akan menggunakan pengolahan citra dalam pendeteksian bola tenis meja. Proses ini berguna untuk menentukan jarak robot ke bola, dan juga menjadi acuan untuk pembuatan *path planning*. Robot ini akan mengumpulkan bola tenis meja yang berserakan berdasarkan jalur yang sudah dibuat. Dengan begitu, pemain tidak perlu mengeluarkan usaha lebih untuk mengambil bola-bola yang berserakan. Tingkat akurasi pada penelitian ini dalam pendeteksian bola dan jarak mencapai 80%.

**Kata Kunci:** tenis meja, robot otonom, pengolahan citra

## Abstract

Table tennis is one of small-ball-sport branches that interests many people. In a table tennis practice, the players use a large number of balls for training and competition. Therefore, extra efforts are required for players to gather the scattered balls in every training. There are several tools developed to assist in collecting tennis table balls, however, the collection process is mostly manual and strains the players. In order to ease the effort of the players during training and competition, we design an autonomous solution, in form of a robot, for collecting the table tennis balls. The robot uses image processing for ball detection. The process determines the distances between the robot and the balls, these distances are then calculated for *path planning*. The robot moves to collect these scattered balls according to the calculated path. Thus, the players do not have to spend extra effort to gather the balls. The proposed design achieves an accuracy of up to 80% in balls collection.

**Key Words:** table tennis, autonomous robot, image processing

## 1. Pendahuluan

Tenis meja merupakan salah satu cabang olahraga bola kecil yang dimainkan dengan menggunakan raket yang dilapisi karet (bet) [1]. Ketika seorang pemain tenis meja berlatih, dibutuhkan banyak bola untuk melatih pukulan pemain tersebut. Bola hasil pukulan pemain akan berserakan di lapangan dan membuat orang tersebut perlu mengeluarkan usaha lebih untuk mengumpulkan bola.

Untuk meringankan beban pemain yang berlatih,

dibuatlah berbagai alat yang mampu membantu pemain mengumpulkan bola yang berserakan. Ada yang memiliki sistem *roller*, dan ada juga yang menggunakan alat penumbuk berongga. Namun, semua alat itu masih digunakan secara manual.

Untuk lebih meringankan proses pengumpulan bola, dirancanglah sebuah robot otonom pengumpul bola tenis meja. Robot otonom adalah sebuah robot yang dirancang untuk menangani suatu masalah tanpa adanya campur tangan manusia [2]. Penelitian sebelumnya

menggunakan sensor *ping* ultrasonik untuk mendeteksi keberadaan bola. Namun, penelitian tersebut tidak dapat menentukan posisi bola secara pasti [3]. Penelitian lainnya juga telah dirancang suatu robot otonom yang bertujuan mengumpulkan bola tenis dengan mencari bola terdekat dari robot [4]. Penelitian yang dilakukan bertujuan untuk mengumpulkan bola tenis meja. Untuk mengarahkan robot ke bola tenis meja, dibutuhkan proses pengolahan citra dua dimensi untuk melihat posisi lebih akurat dalam sebuah ruang. Pengolahan citra adalah suatu proses pengolahan sinyal dengan input berupa gambar yang akan ditransformasi sedemikian rupa dengan teknik tertentu sehingga keluarannya berupa gambar lain [5].

Berdasarkan permasalahan di atas, pada penelitian ini dibangun sebuah robot otonom pengumpul bola tenis meja berbasis pengolahan citra. Robot menggunakan roda untuk bergerak. Setelah robot mengarah pada bola, bola akan diambil menggunakan *roller* lalu diarahkan ke *slider* untuk disimpan ke dalam *storage* yang ada pada robot.

Tujuan yang ingin dicapai dalam penelitian ini adalah merancang suatu robot yang dapat mengetahui posisi bola tenis meja yang berserakan, dengan akurasi pendeteksian objek bola tenis meja dan pengukuran jarak robot terhadap objek mencapai 80

**2. Landasan Teori**

**2.1 Pengolahan Citra**

Pengolahan citra adalah suatu proses pengolahan sinyal dengan input berupa gambar/citra dua dimensi yang akan ditransformasi sedemikian rupa dengan teknik tertentu sehingga keluarannya berupa gambar lain dengan kualitas yang lebih baik [4][6].

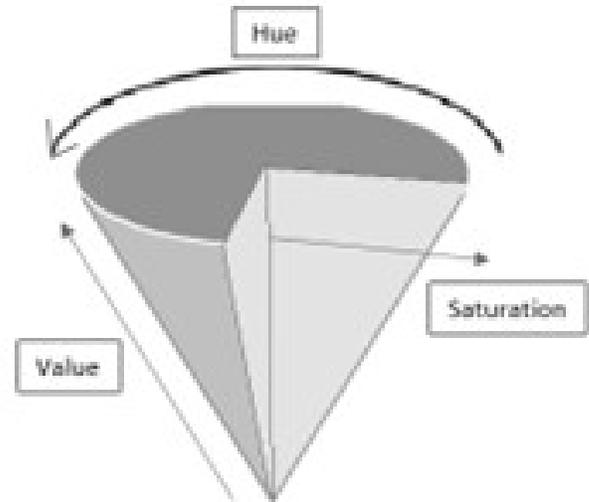
Sebuah citra dapat digambarkan seperti matriks dua dimensi  $f(x,y)$  dan terdiri dari  $M$  kolom dan  $N$  baris. Perpotongan antara kolom dan baris pada sebuah citra disebut dengan piksel yang merupakan elemen terkecil dari suatu citra [7].

$$f(x,y) = \begin{pmatrix} f(0,0) & \dots & f(0,N-1) \\ \vdots & \ddots & \vdots \\ f(M-1,0) & \dots & f(M-1,N-1) \end{pmatrix} \quad (1)$$

Ada beberapa bagian pengolahan citra yang diterapkan dalam perancangan robot otonom pengumpul bola tenis meja yaitu ruang warna *hue-saturation-value* (HSV), *thresholding*, *blob* detection, Region of Interest, dan object detection.

**2.2 Ruang Warna HVS**

Ruang warna ini memiliki prinsip yang sama dengan cara manusia memandang warna dari suatu objek. Selain itu, ruang warna HSV juga cocok untuk segmentasi warna karena memberikan informasi unik



**Gambar 1. Ruang Warna HSV, modifikasi dari [9]**

untuk setiap komponen warna yang berbeda [8]. Sebuah citra digital memiliki standar acuan ruang warna *red-green-blue* (RGB). Karena hal tersebut, kita perlu melakukan proses konversi dari citra RGB menjadi citra dalam ruang warna HSV. Ilustrasi ruang warna HSV dapat dilihat pada Gambar 1. Nilai *hue* merupakan sudut putar dari 0° hingga 360°. *Saturation* dan *value* bernilai 0 hingga 1 [9].

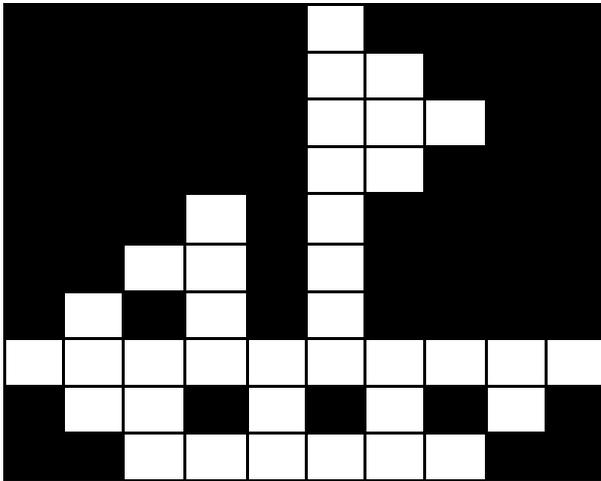
**2.3 Thresholding**

*Thresholding* merupakan suatu konsep untuk mengklasifikasikan piksel pada suatu gambar menjadi *foreground-background* piksel [8]. Pada penelitian ini, *thresholding* yang diterapkan bersifat manual dan mengacu pada rentang nilai HSV dari citra bola yang terdeteksi. Nilai *hue* yang digunakan yaitu ada di rentang 0 hingga 45, nilai *saturation* yang digunakan berada di rentang 150 hingga 255, dan *value* di rentang 171 hingga 255.

**2.4 Blob Detection**

*Blob* detection merupakan suatu teknik pengolahan citra yang bertujuan untuk menemukan dan menghitung objek yang menjadi acuan dasar pengukuran karakteristik dari objek itu sendiri. Algoritma ini membedakan objek berdasarkan pikselnya. Ada yang akan tergolong ke dalam *foreground* dan ada yang akan masuk ke dalam wilayah *background* [10].

*Blob* adalah sekumpulan piksel yang bertetangga dan memiliki nilai tertentu. Untuk menjalankan perintah *blob* detection, langkah pertama adalah menentukan warna yang akan dijadikan dasar bagi objek yang akan dideteksi. Warna akan berada dalam format RGB. Setelah itu, nilai HSV dari gambar ditentukan dengan melakukan transpose dari format RGB ke HSV. Format



Gambar 2. Citra UKuran 10x10

HSV digunakan dengan tujuan untuk mempermudah mendeteksi *blob* [11]. Setelah *blob* diperoleh, ditentukan nilai titik tengah dari *blob* tersebut menggunakan teknik *centroid* dan ditentukan luas dari *blob*. Adapun perhitungan untuk mengetahui titik tengah dari suatu *blob* adalah sebagai berikut [10].

$$X_{mid} = \frac{\max X_{mid} + \min X_{mid}}{2} \quad (2)$$

Pada persamaan tersebut terdapat  $\max X_{mid}$  yang merupakan titik paling kanan pada suatu *blob* dan  $\min X_{mid}$  yang merupakan titik paling kiri dari pada *blob*.

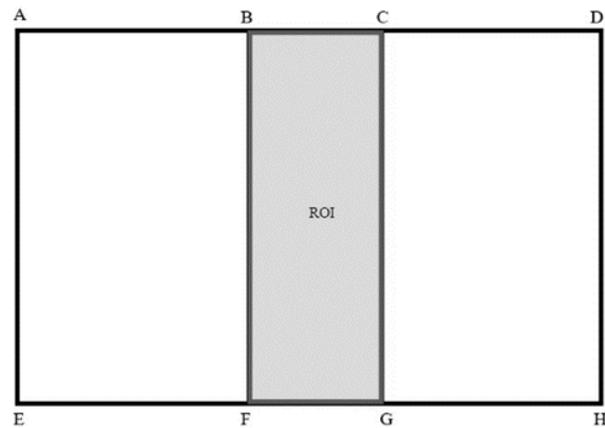
$$Y_{mid} = \frac{\max Y_{mid} + \min Y_{mid}}{2} \quad (3)$$

Pada persamaan tersebut terdapat  $\max Y_{mid}$  yang merupakan titik paling atas pada suatu *blob* dan  $\min Y_{mid}$  yang merupakan titik paling bawah dari pada *blob*.

*Centroid* akan dijadikan sebagai pusat deteksi suatu objek dan luas pada *blob* akan dijadikan tolok ukur dalam perhitungan jarak bola terhadap robot. *Centroid* memiliki peran besar dalam menentukan sudut objek terdeteksi dalam proses *scanning*.

Luas dari suatu *blob* dapat ditentukan dengan cara menghitung jumlah piksel yang menyusun *blob* tersebut [9]. Pada Gambar 2 terdapat suatu area berwarna putih. Area tersebut merupakan suatu *blob* yang terdeteksi. Jumlah area berwarna putih pada citra tersebut merupakan luas dari suatu *blob*.

Informasi posisi objek yang terdiri dari sudut objek terdeteksi ketika *scanning* dan jarak objek terdeteksi akan dijadikan acuan awal pembuatan *path planning* robot.



Gambar 3. Contoh Penerapan ROI

### 2.5 Region of Interest

Agar pengolahan lebih terfokus pada objek atau suatu daerah yang kita inginkan, kita perlu menentukan suatu daerah yang nantinya akan diolah. Daerah tersebut dinamakan *Region of Interest* (ROI) [10]. Objek yang terdeteksi pada area ROI akan diolah tanpa memisahkan kawasan ROI dari frame [14]. Gambar 3 merupakan salah satu contoh penerapan dari penggunaan ROI.

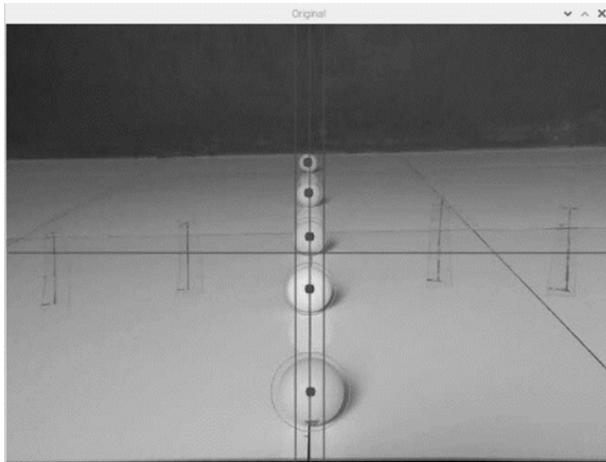
Gambar tersebut merupakan contoh citra yang sudah diberi ROI. Titik A, D, H, dan E merupakan titik paling ujung dari citra. Titik B, C, G, dan F merupakan titik paling ujung dari ROI. Area ABFE dan CDHG merupakan bagian dari citra yang tidak akan dideteksi karena ada di luar ROI. Area BCGF merupakan area yang menjadi ROI.

### 2.6 Object Detection

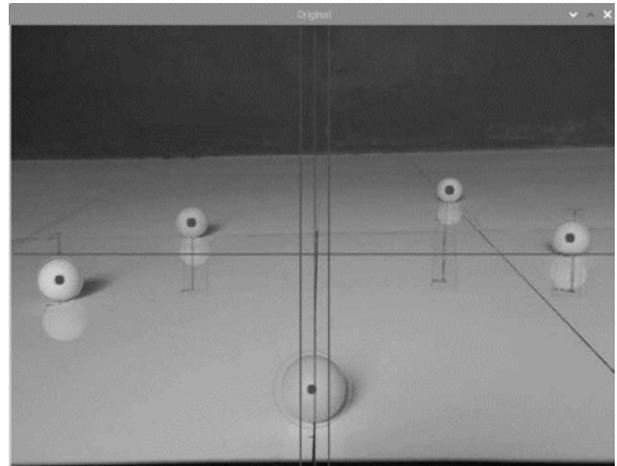
Proses deteksi objek pada sebuah *frame* menggunakan analisis dari *blob* yang terdeteksi. Ketika proses deteksi, terdapat dua keadaan yang mungkin terjadi jika dalam satu *frame* terdapat banyak *blob* yang terdeteksi. Keadaan pertama, yaitu ada lebih dari satu *blob* terdeteksi pada satu *frame* dan keadaan kedua, yaitu ada *blob* yang terdeteksi di area ROI dan ada yang tidak berada pada area ROI.

Pada Gambar 4, terdapat lima bola yang *centroid*-nya terdeteksi di area ROI. Pemberian nama bola dilakukan secara berurutan dari objek yang terdeteksi paling dekat (luas *blob* besar) hingga paling jauh (luas *blob* kecil). Bola terdekat diberi nama Bola 1, bola selanjutnya yang terdeteksi diberi nama Bola 2, dan seterusnya. Nilai luas *blob* yang terdeteksi dikonversi menjadi nilai jarak dengan persamaan yang diperoleh dari proses kalibrasi. Informasi yang diperoleh berupa nama bola, jarak, sudut bola terdeteksi, *centroid* (nilai x dan y), dan luas dari *blob* yang terdeteksi.

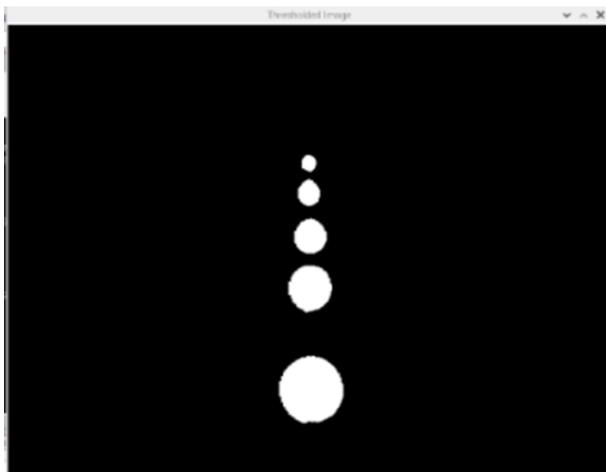
Gambar 5 merupakan tampilan citra yang sudah mengalami proses *threshold*. Objek terdeteksi diubah



**Gambar 4. Tampilan Proses Deteksi pada Keadaan Satu**



**Gambar 6. Deteksi Bola pada Keadaan Kedua**

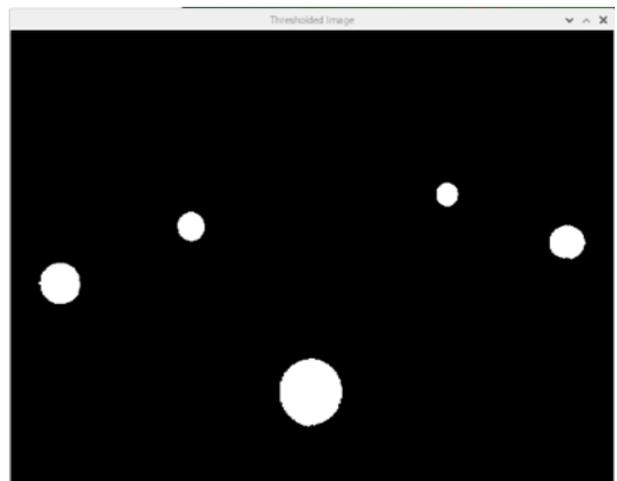


**Gambar 5. Tampilan Blob pada Keadaan Satu**

menjadi putih (*blob*) dan lingkungan sekitar objek menjadi berwarna hitam. Berikut tampilan dari citra tersebut.

Pada Gambar 6 terdapat lima bola yang berada dalam satu *frame* yang sama, namun hanya ada satu bola yang *centroid*-nya terdeteksi di area ROI. Pemberian nama bola pada kondisi ini hanya berlaku untuk bola yang berada di area ROI saja. Bola tersebut akan diberi nama Bola 1. Namun, bola tersebut akan diberi nama Bola 2 atau seterusnya jika pada proses deteksi yang sudah digabung dengan proses *scanning* terdapat bola yang terdeteksi terlebih dahulu. Bola yang berada di luar area ROI tidak akan mengalami proses pengolahan citra.

Gambar 7 merupakan tampilan citra kondisi kedua yang sudah mengalami proses *threshold*. Berikut tampilan dari citra tersebut.



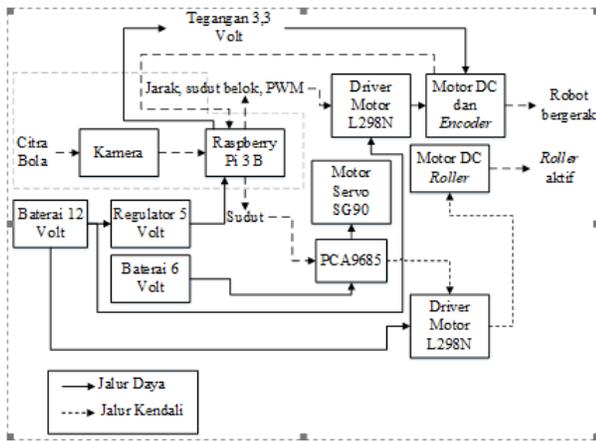
**Gambar 7. Tampilan Blob pada Keadaan Kedua**

### 3. Perancangan Sistem

#### 3.1 Desain Sistem

Robot otonom pengumpul bola tenis meja menggunakan Raspberry Pi 3 B sebagai mikrokontroler dengan sistem operasi Linux dan OpenCV sebagai *library* bagi pengolahan citra. Robot ini menggunakan sensor kamera untuk mendeteksi keberadaan objek yang nantinya akan diketahui jaraknya terhadap robot dan terhadap objek bola lainnya. Perancangan sistem yang dilakukan adalah sebagai berikut.

1. Membuat source code kamera untuk mendeteksi bola dan mengetahui jarak robot ke bola, mengendalikan motor servo untuk menggerakkan kamera, dan mengatur kecepatan motor DC.
2. Menyatukan semua perangkat keras dan menghubungkannya dengan perangkat lunak,
3. Menentukan *path planning* sebagai jalur yang akan



Gambar 8. Diagram Blok Sistem

dilalui robot,

- Mengimplementasikan metode yang digunakan untuk membuat robot berjalan melalui rute yang sudah ditentukan melalui *path planning*.

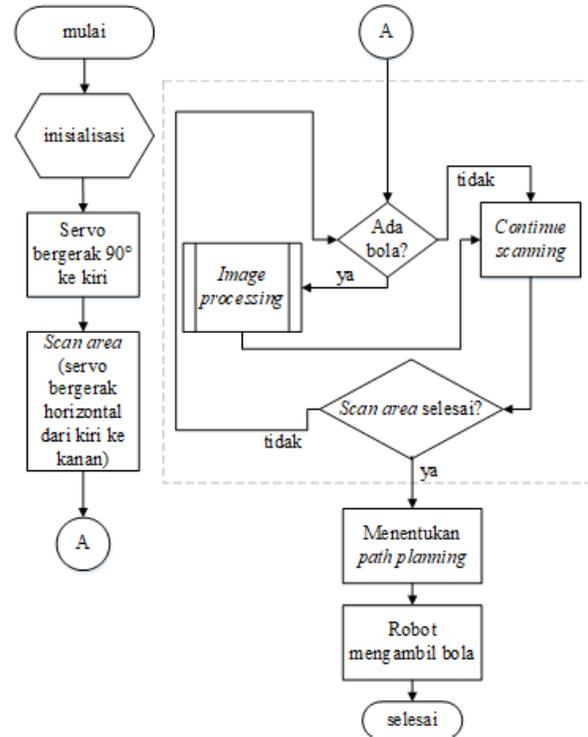
### 3.2 Diagram Blok Sistem

Pada Gambar 8, sistem ini dirancang menggunakan Raspberry Pi 3 B sebagai kontroler bagi robot. Kamera digunakan sebagai sensor untuk mendeteksi bola yang nantinya akan menghasilkan input berupa nilai jarak dan sudut lokasi bola yang didapat dari pergerakan servo. Input ini akan menjadi acuan bagi keluaran yang ada pada *motor driver* yang akan mengatur besar tegangan keluaran bagi motor DC.

### 3.3 Diagram Alir Sistem

Robot melakukan *scanning* pada area dan jika pada area tersebut terdeteksi adanya bola, maka akan dihitung jarak robot ke bola dan besar sudut dari gerakan servo. Jika tidak terdeteksi, maka servo akan bergerak sampai terdeteksi ada bola lainnya. Setelah proses *scanning* selesai, robot akan menentukan bola mana yang memiliki jarak terdekat dengan robot untuk diambil. Lalu, robot akan menentukan bola mana yang selanjutnya akan diambil. Setelah itu, sudut belok dari robot akan dihitung dan *path planning* akan dirancang menggunakan metode *nearest neighbor*, di mana informasi jarak yang diperoleh pada setiap objek akan dibandingkan dan bola terdekat akan menjadi pusat pengukuran jarak selanjutnya. Selanjutnya, robot akan berjalan mengambil bola sesuai dengan jalur yang sudah dibuat oleh *path planning*. Gambar 9 merupakan alur kerja dari sistem secara keseluruhan.

Pada sistem keseluruhan, terdapat subsistem pengolahan citra. Diagram alir subsistem tersebut terdapat pada Gambar 10. Dalam proses *scanning*, kamera mengambil satu *frame* dari tiap satu gerakan



Gambar 9. Diagram Alir Keseluruhan Sistem

servo. Pengambilan dimulai dari titik 0°. Dari *frame* yang diambil, dilakukan proses colour filtering dengan mengubah nilai RGB menjadi nilai HSV. Setelah diubah, nilai *hue*, *saturation*, dan *value* yang sesuai ditentukan dengan warna bola tenis meja. Setelah ditentukan, dilakukan proses *thresholding*. Lalu, derau dihilangkan dengan proses opening dan closing. Setelah difilter, ditentukan nilai *centroid* dan luas dari *blob* yang terdeteksi. Jika *centroid* berada di area ROI, program perhitungan jarak akan dilakukan dan sudut servo akan tersimpan di dalam database. Perhitungan jarak dilakukan dengan membandingkan nilai luas dari suatu *blob* dengan jarak sebenarnya. Semakin besar luas *blob*, maka jarak yang terukur dinilai semakin pendek.

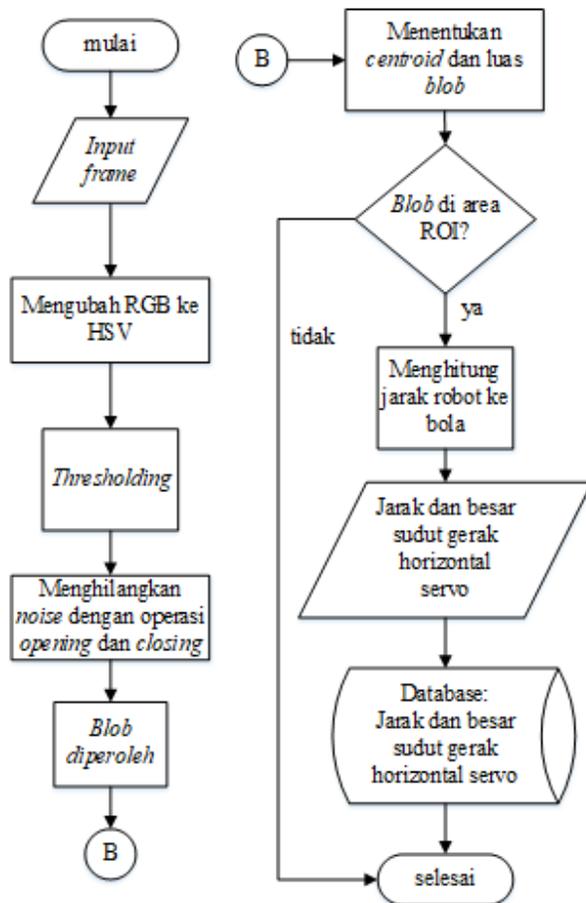
### 3.4 Mekanik Robot

Robot yang dirancang merupakan robot yang bergerak dengan dua roda penggerak dan satu roda castor. Untuk mengambil bola, robot akan menggunakan *roller* yang ada di bagian depan robot. Pada bagian belakang, terdapat pintu yang dapat dibuka untuk mengambil bola yang sudah terambil. Gambar 11 merupakan desain robot yang akan dirancang.

## 4. Hasil dan Analisis

### 4.1 Pengujian Jarak

Proses pengujian jarak bertujuan untuk membandingkan jarak sebenarnya dengan hasil yang



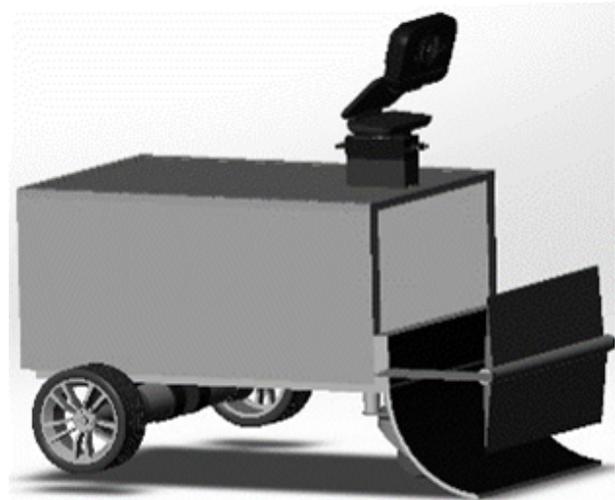
Gambar 10. Diagram Alir subsistem Image Processing

diperoleh dari perhitungan pada sistem. Sistem mengukur jarak dari poros motor servo hingga bola.

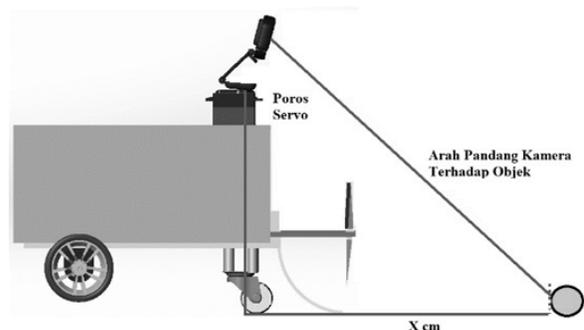
Alat ukur yang digunakan berupa media kertas yang sudah diberi nilai jarak dari 0 hingga 150 cm. Uji jarak dilakukan ketika kamera melakukan proses *scanning*. Proses pengujian dilakukan pada jarak 50 cm, 75 cm, 100 cm, 125 cm, dan 150 cm. Pengujian dilakukan sebanyak 30 kali untuk masing-masing jarak. Ilustrasi pengambilan jarak dapat dilihat pada Gambar 12. Huruf X pada ilustrasi merupakan nilai jarak terukur.

Grafik pengujian deteksi jarak bola ketika proses *scanning* dapat dilihat pada Gambar 13 di bawah ini.

Beberapa faktor yang mempengaruhi pembacaan jarak pada sistem, yaitu fokus dari kamera berubah ketika jauh, adanya keterbatasan resolusi, adanya kesalahan paralaks dari kamera, di mana ketika proses *scanning*, area deteksi mendapatkan pencahayaan yang berbeda-beda setiap berganti sudut deteksi.



Gambar 11. Desain Mekanik Robot



Gambar 12. Proses Pengambilan Data Jarak

#### 4.2 Pengujian Jumlah Bola Terdeteksi

Proses pengujian jumlah bola terdeteksi bertujuan untuk membandingkan jumlah bola di lapangan dengan jumlah bola yang dapat dideteksi oleh robot ketika proses *scanning*. Bola diletakkan di 10 titik sudut yang berbeda, dari sudut 30° hingga 120°. Selisih letak antara dua bola adalah 10°. Bola diletakkan 70 cm dari titik poros deteksi. Berikut Tabel 1 yang menunjukkan hasil pengujian jumlah bola terdeteksi.

Dari hasil pengujian, diperoleh nilai akurasi rata-rata sebesar 91,67%. Dalam proses *scanning* 10 bola, bola yang dapat terdeteksi berjumlah 7 hingga 10 bola. Penyebab bola tidak terdeteksi, yaitu karena terbatasnya area ROI, adanya *delay* antara pergerakan servo dengan proses pendeteksian oleh kamera, faktor keterbatasan resolusi dan spesifikasi perangkat keras.

#### 4.3 Pengujian Keseluruhan Sistem

Proses pengujian keseluruhan sistem dilakukan dengan sepuluh skenario yang berbeda. Berikut merupakan skenario keseluruhan sistem. Terdapat 10 skenario, di mana skenario satu sampai empat



Gambar 13. Hasil Pengukuran Jarak ketika Proses scanning

Tabel 1. Hasil Pengujian Jumlah Bola Terdeteksi

No	Sudut											Terdeteksi	Akurasi (%)	
	30°	40°	50°	60°	70°	80°	90°	100°	110°	120°	130°			
1	1	1	1	1	1	1	1	1	1	1	1	1	10	100
2	1	1	1	1	1	1	1	1	1	1	1	1	10	100
3	1	1	0	1	1	1	1	1	0	1	1	1	8	80
4	1	1	1	1	1	1	1	1	1	1	1	1	10	100
5	1	1	0	1	1	1	1	0	1	1	1	1	8	80
6	1	1	1	1	1	1	1	0	1	1	1	1	9	90
7	1	0	1	1	1	1	1	1	1	1	1	1	9	90
8	1	0	1	1	1	1	1	1	1	1	1	1	9	90
9	1	0	1	1	1	1	1	1	1	1	1	1	9	90
10	1	1	1	1	1	1	1	1	1	1	1	1	10	100
11	1	1	1	1	0	1	1	1	1	1	1	1	9	90
12	1	1	0	1	1	1	1	0	1	1	1	1	8	80
13	1	1	1	1	1	1	1	0	1	1	1	1	9	90
14	1	1	0	1	0	0	1	1	1	1	1	1	7	70
15	1	1	1	1	0	0	1	0	1	1	1	1	7	70
16	1	1	1	1	0	1	1	0	1	1	1	1	8	80
17	1	1	1	1	1	1	1	0	1	1	1	1	9	90
18	1	1	1	1	1	1	1	1	1	1	1	1	10	100
19	1	1	1	1	1	1	1	1	1	1	1	1	10	100
20	1	1	1	1	1	1	1	1	1	1	1	1	10	100
21	1	1	1	1	1	1	1	1	1	1	1	1	10	100
22	1	1	1	1	1	1	1	1	1	1	1	1	10	100
23	1	1	1	1	1	1	1	1	1	1	1	1	10	100
24	1	1	1	1	1	0	1	1	1	1	1	1	9	90
25	1	1	1	1	1	0	1	1	1	1	1	1	9	90
26	1	1	1	1	1	0	1	1	1	1	1	1	9	90
27	1	1	1	1	1	1	1	1	1	1	1	1	10	100
28	1	1	1	1	1	1	1	1	1	1	1	1	10	100
29	1	1	1	1	1	0	1	1	1	1	1	1	9	90
30	1	1	1	1	1	1	1	1	1	1	1	1	10	100
Rata-rata	1,00	0,90	0,87	1,00	0,87	0,80	1,00	0,73	1,00	1,00	9,17	91,67		

menggunakan satu bola, skenario lima sampai delapan menggunakan lima bola, dan sisanya sepuluh bola. Tabel 2 merupakan tabel skenario uji.

Tabel 3 menunjukkan pembacaan jarak pengujian keseluruhan sistem dari robot otonom pengumpul bola tenis meja.

Tabel 4 merupakan tabel keberhasilan robot dalam mengambil bola yang ada di lapangan.

Dari segi citra, terdapat beberapa kekurangan yang disebabkan oleh beberapa faktor, yaitu kontras cahaya ketika scanning, webcam yang bersifat autofocus sehingga rentan terhadap perubahan cahaya, adanya efek paralaks dari kamera yang menyebabkan pembacaan kurang akurat. Selain itu, terdapat delay pada proses akuisisi citra karena keterbatasan perangkat keras dan resolusi citra sehingga delay tersebut menyebabkan proses scanning hanya dapat melakukan pembacaan hingga sudut 160°. Keadaan bola yang tidak terdeteksi terjadi akibat dari ukuran ROI yang sudah disesuaikan

Tabel 2. Tabel Skenario Uji

Skenario	1	2	3	4	5	6	7	8	9	10
Bola 1	Jarak(cm)	70	70	70	40	70	50	40	70	90
Bola 1	Sudut(°)	90	30	60	150	90	30	30	90	0
Bola 2	Jarak(cm)					90	60	60	90	20
Bola 2	Sudut(°)					80	76	40	67	76
Bola 3	Jarak(cm)					90	90	90	90	40
Bola 3	Sudut(°)					110	79	100	87	79
Bola 4	Jarak(cm)					90	120	110	90	60
Bola 4	Sudut(°)					150	82	70	112	82
Bola 5	Jarak(cm)					90	150	130	90	80
Bola 5	Sudut(°)								135	85
Bola 6	Jarak(cm)								90	100
Bola 6	Sudut(°)								181	88
Bola 7	Jarak(cm)								90	120
Bola 7	Sudut(°)								211	91
Bola 8	Jarak(cm)								90	140
Bola 8	Sudut(°)								221	94
Bola 9	Jarak(cm)								90	160
Bola 9	Sudut(°)								230	97
Bola 10	Jarak(cm)								90	180
Bola 10	Sudut(°)								90	180

Tabel 3. Tabel Pembacaan Jarak Skenario

Skenario	1	2	3	4	5	6	7	8	9	10
Jumlah Bola	1	1	1	1	5	5	5	10	10	10
Urutan Pengambilan	1	1	1	1	1,2,3,4,5	1,2,3,4,5	1,2,4,5	1,2,3,4,5,7,8	1,2,3,4,5,6,7,9	10,4,5,2,1,6,3,7,9
Bola Terdeteksi	1	1	1	1	5	5	5	7	8	9
Bola Terambil	1	1	1	1	4	5	5	5	5	5
Jarak (cm)	1	67,71	71,16	71,47	70,36	38,96	71,02	50,75	35,25	67,68
Jarak (cm)	2	X	X	X	X	62,20	74,22	85,07	48,75	72,46
Jarak (cm)	3	X	X	X	X	80,79	77,75	41,14	65,29	78,17
Jarak (cm)	4	X	X	X	X	110,41	80,05	108,77	84,90	81,77
Jarak (cm)	5	X	X	X	X	156,66	82,84	74,58	108,98	88,04
Jarak (cm)	6	X	X	X	X	X	X	X	156,00	88,90
Jarak (cm)	7	X	X	X	X	X	X	X	143,50	89,88
Jarak (cm)	8	X	X	X	X	X	X	X	X	X
Jarak (cm)	9	X	X	X	X	X	X	X	X	92,42
Jarak (cm)	10	X	X	X	X	X	X	X	X	111,20
Waktu Tempuh (sekon)	2,47	6,18	5,04	5,47	31,98	62,10	75,17	47,95	108,42	119,50

dengan kemampuan servo.

Selain galat yang diakibatkan proses pengolahan citra, faktor lain yang memengaruhi proses pengambilan bola, yaitu posisi roller ketika akan mengambil bola. Bola tidak jarang tertabrak oleh bagian ujung dari roller atau terhempas dari roller. Lalu, area gerak robot kurang mulus sehingga menyebabkan ujung slider kadang tersangkut dan mengganggu gerak robot. Selain itu, terdapat faktor galat dari pembacaan sudut, dan pergerakan robot.

5. Kesimpulan

Berdasarkan hasil dari pengujian dan analisis pada sub-sistem dan sistem, robot mampu mengenali bola tenis meja dan mengetahui posisinya berdasarkan jarak dan sudut dengan menggunakan metode blob detection. Proses pendeteksian jumlah bola tenis meja di lapangan memiliki nilai akurasi rata-rata 91,67%. Pengukuran jarak pada rentang 50 cm hingga 150 cm saat proses scanning dari sudut 0° hingga 160° memiliki nilai akurasi rata-rata lebih dari 80%. Pada uji skenario keseluruhan sistem, proses pembacaan jarak ketika scanning memperoleh akurasi rata-rata melebihi 80%, kecuali untuk skenario pengambilan 10 bola lurus. Terdapat pembacaan yang akurasinya kurang dari 80% dikarenakan jarak bola berada di luar batas maksimal pembacaan jarak, yaitu 150 cm. Pada skenario tersebut,

**Tabel 4. Tabel Uji Keberhasilan Pengambilan Bola**

Skenario	1	2	3	4	5	6	7	8	9	10	
Jumlah Bola	1	1	1	1	5	5	5	10	10	10	
Bola Terdeteksi	1	1	1	1	5	5	5	7	8	9	
Bola Terambil	1	1	1	1	4	5	5	5	5	5	
Bola Yang Terambil	1										
	2	X	X	X	X	terlempar			terdorong	terham bat	
	3	X	X	X	X					terlempar	
	4	X	X	X	X				terdorong		
	5	X	X	X	X						
	6	X	X	X	X	X	X	X	tidak pas		
	7	X	X	X	X	X	X	X	tidak pas	terdorong	terdorong
	8	X	X	X	X	X	X	X	X		tidak pas
	9	X	X	X	X	X	X	X	X	tidak pas	terham bat
	10	X	X	X	X	X	X	X	X	X	
Waktu Tempuh (sekon)	2,47	6,18	5,04	5,47	31,98	62,1	75,17	47,95	108,42	119,5	

Terambil

Tidak

X Tidak Terdeteksi

bola kedelapan, kesembilan, dan kesepuluh memiliki tampilan blob yang menyatu sehingga pembacaan jaraknya menjadi tidak baik.

Sebagai pengembangan di masa mendatang, penggunaan mikrokomputer dengan spesifikasi lebih tinggi dianjurkan agar dapat mengurangi delay ketika proses scanning. Resolusi kamera juga dapat ditingkatkan untuk memperbaiki akurasi pengukuran jarak. Di sisi lain, motor DC pada roller juga dapat diganti dengan motor stepper yang memiliki step putaran sehingga roller dapat mengatur step putaran hingga 360° atau roller memiliki posisi default sebelum bergerak.

**Daftar Pustaka**

[1] Fang, S., Xu, L. Da, Zhu, Y., Ahati, J., Pei, H., Yan, J., & Liu, Z. (2014). An integrated system for regional environmental monitoring and management based on internet of things. *IEEE Transactions on Industrial Informatics*, 10(2), 1596–1605. <http://doi.org/10.1109/TII.2014.2302638>

[2] A. S. Taufik, “Sistem Navigasi Waypoint pada Autonomous Mobile Robot,” *Jurnal Mahasiswa TEUB*, vol. 1, no. 1, pp. 1–6, 2013.

[3] H. P. Siregar, M. N. Karim, and E. Hidayat, “ROBOT PENGAMBIL BOLA PINGPONG PADA KONTES ROBOT CERDAS INDONESIA (Divisi Battle) 2010,” vol. 2010, pp. 1–3, 2010.

[4] Vehbi Umur Çabuk, Ahmet Kubilay Şavkan, Ramazan Kahraman, Ferdi Karaduman, Okan Kiril, and Volkan Sezer, “Design and Control of a Tennis Ball Collector Robot,” 2018.

[5] H. Mulyawan, M. Z. H. Samsono, and Setiawardhana, “Identifikasi Dan Tracking Objek Berbasis Image Processing Secara Real Time,”

Jurusan Telekomunikasi, Politeknik Elektronika Negeri Surabaya, pp. 1–5, 2011.

[6] R. W. Indrawan, E. S. Ningrum, R. Y. Hakkun, and Setiawardhana, “An Image Processing System For Visual Servoing of Soccer Robot,” pp. 21–22, 2010.

[7] A. J. Al Kadafi and F. Utamingrum, “Deteksi Objek Penghalang Secara Real-Time Berbasis Mobile Bagi Penyandang Tunanetra Menggunakan Analisis Blob,” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya*, vol. 2, no. 1, pp. 423–432, 2018.

[8] Moshina Binte Asad, Fatema Jannat Sharna, Nashid Alam, and M. Jahiril Islam, “HSV and Template Matching Based Bengali Road Sign Recognition Technique,” 2016.

[9] B. Y. Budi Putranto, W. Hapsari, and K. Wijana, “Segmentasi Warna Citra Dengan Deteksi Warna Hsv Untuk Mendeteksi Objek,” *Jurnal Informatika*, vol. 6, no. 2, 2011, doi: 10.21460/inf.2010.62.81.

[10] B. N. Patro, “Design and implementation of novel image segmentation and BLOB detection algorithm for real-time video surveillance using DaVinci processor,” *Proceedings of the 2014 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2014*, pp. 1909–1915, 2014, doi: 10.1109/ICACCI.2014.6968360.

[11] M. Gupta, “Cell Identification by Blob Detection,” *Proc. of the Intl. Conf. on Advances in Computer Science and Electronic Engineering*, no. January 2012, pp. 233–236, 2012, doi: 10.3850/978-981-07-1403-1.

[12] A. Kadir and A. Susanto, *Teori dan Aplikasi Pengolahan Citra*, 1st ed. Yogyakarta: ANDI OFFSET, 2013.

[13] W. Kiadtikornthaweeeyot and A. R. L. Tatnall, “Region of interest detection based on histogram segmentation for satellite image,” *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, vol. 41, no. July, pp. 249–255, 2016, doi: 10.5194/isprsarchives-XLI-B7-249-2016.

[14] Pratomo dkk, “Implementasi Algoritma Region of Interest (ROI) Untuk Meningkatkan Performa Algoritma Deteksi Dan Klasifikasi Kendaraan,” *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. 7, no. 1, pp. 155–162, 2020, doi: 10.25126/jtiik.202071718.