

BAB III

PELAKSANAAN PROYEK

3.1 Kedudukan dan Koordinasi

Penulis merupakan bagian dari divisi *High Level Programming* dan Sensor dalam proyek independen MBKM yang mengikuti perlombaan Kompetisi Robot Sepak Bola Indonesia Beroda (KRSBI Beroda). Dalam proyek ini, penulis bertanggung jawab pada pengembangan sistem penglihatan visual (*vision system*) yang mencakup *visual servoing*, kamera servo, kalibrasi kamera, serta estimasi jarak berbasis citra. Fokus utama penulis adalah memastikan robot dapat merespons objek seperti bola dan gawang secara otonom melalui umpan balik visual dari kamera.

Salah satu peran penting penulis adalah mengintegrasikan sistem deteksi objek (menggunakan YOLO) dengan sistem pengendalian gerak robot melalui pendekatan *Image-Based Visual Servoing* (IBVS). Penulis juga merancang metode pengendalian sudut pandang kamera menggunakan motor servo untuk mengatasi keterbatasan *field of view*, serta melakukan kalibrasi kamera guna memperbaiki distorsi lensa dan memungkinkan perhitungan jarak yang lebih akurat.

Struktur kedudukan dalam proyek ini melibatkan dua pembimbing utama yang mendukung penulis dari sisi akademik dan teknis:

- *Advisor* (Pembimbing Akademik)
Memberikan arahan dalam penyusunan laporan akhir, termasuk penyesuaian sistematika penulisan, tinjauan literatur, serta keterkaitan antara proyek teknis dengan capaian pembelajaran MBKM.
- *Supervisor* (Pembimbing Teknis di Lapangan)
Memberikan bimbingan terhadap pelaksanaan proyek teknis, termasuk validasi sistem deteksi visual, pengujian lapangan untuk *visual servoing*, serta pemrosesan data kalibrasi dan estimasi jarak.

Dalam koordinasi harian, penulis aktif berdiskusi dengan anggota tim lain untuk menyelaraskan hasil sistem visual dengan sistem gerak dan pengambilan keputusan robot. Komunikasi teknis dilakukan secara rutin untuk menyinkronkan pergerakan robot berdasarkan informasi visual yang diterima dari kamera. Setiap pengujian sistem visual, terutama pada pengujian kamera servo dan estimasi jarak, dilakukan secara terkoordinasi dengan tim kontrol gerak dan aktuator.

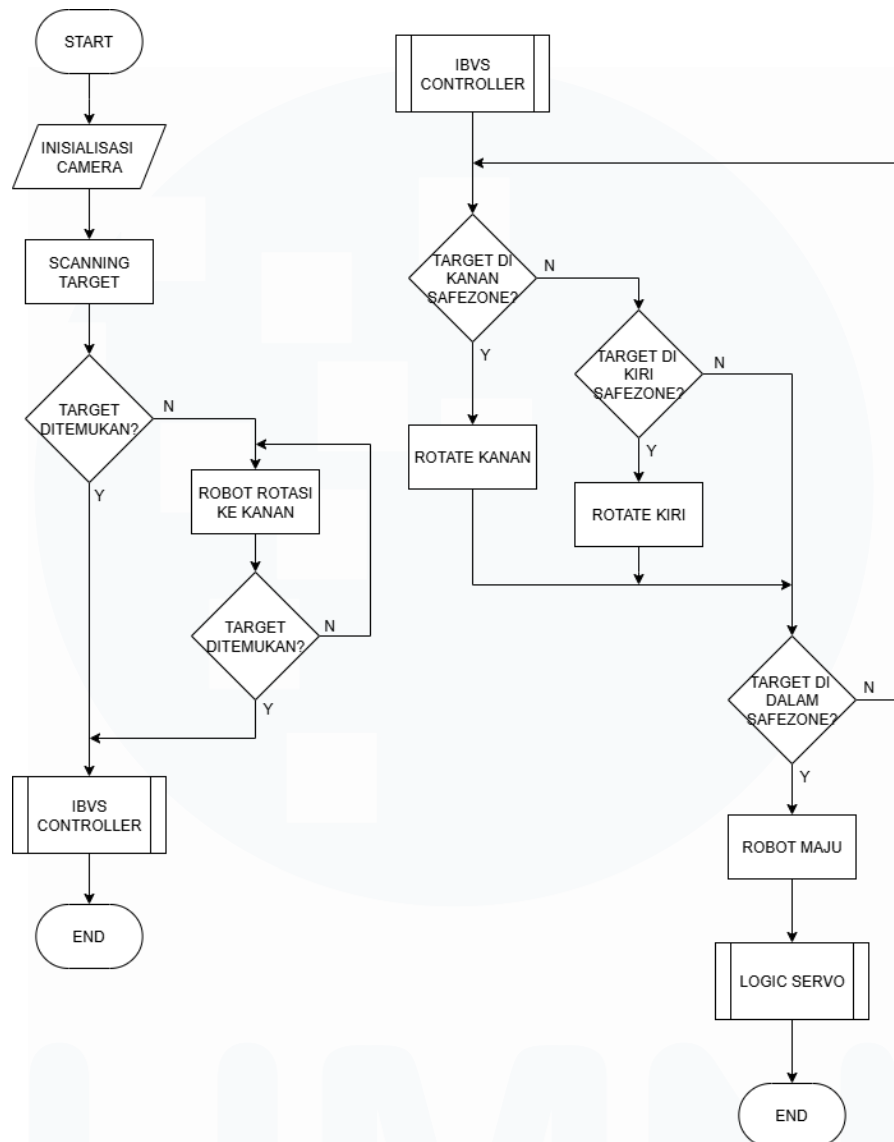
Dengan struktur koordinasi ini, penulis dapat memastikan bahwa setiap tahapan dari perancangan hingga implementasi sistem visual terintegrasi secara utuh dalam sistem robot yang dikembangkan, serta selaras dengan target kompetisi KRSBI Beroda.

3.2 Servo Visual

Servo Visual atau yang lebih dikenal dengan *Visual servoing* merupakan teknik pengendalian gerak robot yang memanfaatkan data visual dari kamera sebagai sumber utama umpan balik . Dalam pendekatan ini, keputusan terkait pergerakan robot, seperti rotasi, translasi, atau manuver lain, diambil secara langsung berdasarkan informasi yang diperoleh dari citra kamera. Teknik ini memungkinkan robot untuk berinteraksi secara adaptif dengan lingkungan yang berubah-ubah, serta merespons posisi objek target seperti bola atau gawang secara *real-time* [12].

Secara umum, *visual servoing* terbagi menjadi dua pendekatan utama, yaitu *Image-Based Visual Servoing* (IBVS) dan *Position-Based Visual Servoing* (PBVS) [13]. IBVS memanfaatkan posisi objek dalam citra 2D (koordinat *pixel*) untuk mengendalikan gerakan robot, tanpa perlu mengetahui posisi absolut objek di dunia nyata. Sebaliknya, PBVS memerlukan proses kalibrasi kamera serta estimasi posisi objek dalam ruang 3D, sehingga pergerakan robot dikendalikan berdasarkan informasi spasial secara eksplisit.

Untuk menghindari aksi yang tidak stabil akibat fluktuasi deteksi, robot dirancang hanya akan melakukan pergerakan maju apabila bola telah terdeteksi secara stabil dalam *safe zone* selama lebih dari 300 milidetik. Pendekatan ini terbukti mampu meningkatkan konsistensi dalam pengambilan keputusan gerak, sekaligus mengurangi efek delay akibat kesalahan minor dalam deteksi citra. Proses ini membentuk suatu loop pengendalian visual langsung, karena setiap aksi gerak robot dikendalikan secara *real-time* berdasarkan posisi objek dalam citra kamera tanpa perlu proses konversi ke koordinat dunia nyata.

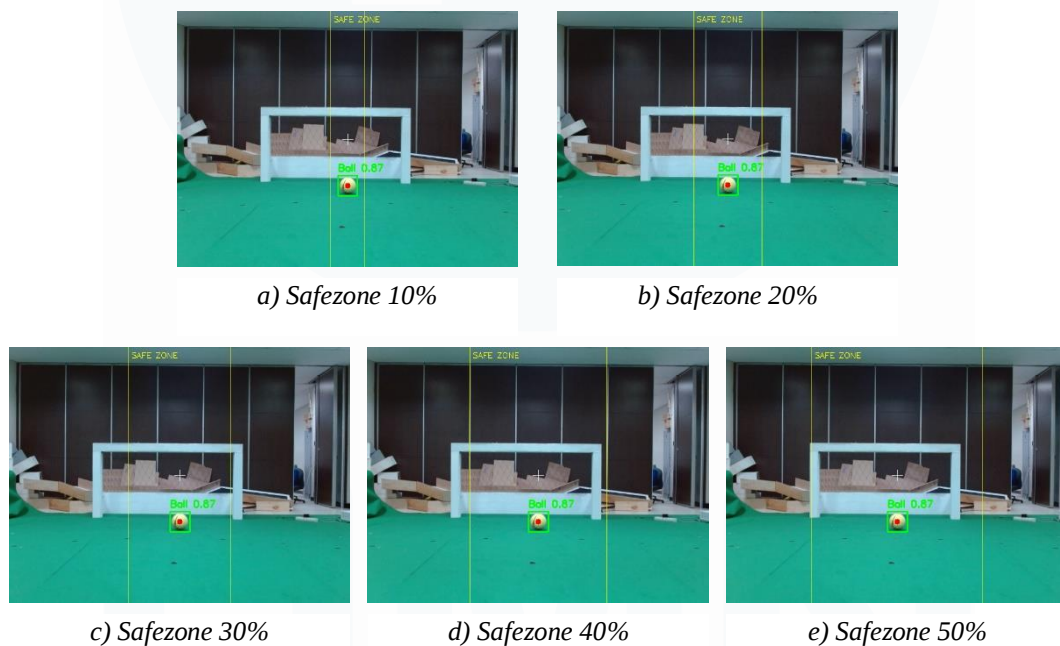


Gambar 3. 2. 1. Flowchart Alur Kerja Visual Servoing Robot KRSBI Beroda

Flowchart pada Gambar 3.2.1 menggambarkan alur kerja sistem *Visual Servoing* berbasis citra (*Image-Based Visual Servoing* – IBVS) yang diimplementasikan pada robot KRSBI Beroda. Diagram ini mencakup tahapan utama mulai dari inisialisasi sistem, proses pencarian serta deteksi objek target (bola), hingga pengambilan keputusan gerak robot berdasarkan posisi objek dalam bidang pandang kamera.

Setelah kamera aktif, sistem akan memindai lingkungan untuk mendeteksi keberadaan bola. Jika objek terdeteksi, posisi horizontal bola akan dibandingkan terhadap zona aman (*safe zone*) yang terletak di sekitar pusat citra. Apabila bola berada di luar zona ini, robot akan melakukan rotasi ke kiri atau ke kanan hingga bola masuk ke dalam *safe zone*.

Setelah bola terdeteksi secara stabil di dalam zona, robot akan bergerak maju secara perlahan untuk mendekati target. Seluruh proses ini berlangsung secara *looping* dan *real-time*, memungkinkan robot merespons secara adaptif terhadap dinamika di lapangan pertandingan.



Gambar 3. 2. 2. Perbandingan Safezone

Untuk mengoptimalkan kinerja *visual servoing*, telah dilakukan serangkaian pengujian empiris. Pengujian ini bertujuan untuk mengevaluasi dampak lebar *safe zone* terhadap stabilitas pergerakan robot dan mengukur performa sistem secara keseluruhan. Dalam pengujian ini, robot ditempatkan pada jarak 3 meter lurus di depan bola, dengan kamera robot mengarah tepat ke bola. Lima konfigurasi *safe zone* yang berbeda (10%, 20%, 30%, 40%, dan 50% dari lebar *frame* kamera) diuji

sebanyak 30 kali ulangan untuk setiap konfigurasi. Pada setiap uji coba, robot diizinkan bergerak dari posisi awal hingga menyentuh bola. Metrik "*jitter*" didefinisikan sebagai jumlah kejadian di mana robot yang sedang bergerak maju berhenti sejenak untuk melakukan rotasi (kiri atau kanan) guna menyesuaikan kembali posisi bola ke tengah *safe zone*. Data *jitter* dicatat secara otomatis. Seluruh eksperimen dilakukan di dalam ruangan dengan kondisi pencahayaan dan karakteristik lantai (karpet) yang konsisten untuk meminimalkan variabel eksternal.

Sebelum mengevaluasi waktu operasional dalam skenario *visual servoing* secara penuh, dilakukan percobaan terpisah untuk menentukan kecepatan ideal robot. Percobaan ini bertujuan untuk mengukur waktu yang dibutuhkan robot untuk menempuh jarak 3 meter dalam kondisi yang terkontrol dan optimal. Pengujian dilakukan sebanyak 30 kali ulangan, dengan robot bergerak lurus dari titik *start* menuju bola yang ditempatkan pada jarak 3 meter. Penting untuk digarisbawahi bahwa seluruh percobaan ini dilaksanakan dalam kondisi daya baterai ideal, yaitu 24V, untuk memastikan performa motor yang konsisten dan maksimal.

Tabel 3. 2. 1. Kecepatan Ideal Robot (Jarak 3 Meter, Kondisi Baterai 24V)

Waktu (s)
11.32
10.88
11.14
10.93
11.23
10.75
11.17
11.08
10.95
11.41
10.81
11.11
10.86
11.34
11.26
11.05
11.00

10.92
11.38
10.89
9.97
11.10
11.22
10.96
11.12
11.30
10.78
12.21
11.16
10.91
11.07

Untuk memperoleh nilai referensi waktu tempuh yang ideal, dilakukan percobaan terpisah dengan skenario pergerakan lurus tanpa koreksi arah. Robot diletakkan pada posisi awal sejauh 3 meter di depan bola, kemudian diarahkan untuk bergerak lurus maju hingga menyentuh bola tanpa intervensi rotasi maupun kendali visual. Percobaan ini dilakukan sebanyak 30 kali dalam kondisi tegangan ideal (baterai 24V) dan lintasan yang lurus. Hasil pengukuran menunjukkan bahwa waktu tempuh rata-rata robot untuk mencapai bola dalam kondisi ideal adalah 11,07 detik, sebagaimana dirangkum pada Tabel 3.2.1. Nilai ini kemudian digunakan sebagai waktu ideal dalam perhitungan akurasi visual servoing, yang mencerminkan seberapa dekat sistem kendali visual mampu meniru efisiensi pergerakan lurus sempurna.

Untuk mengevaluasi kinerja sistem *visual servoing* secara lebih mendalam dan mengidentifikasi konfigurasi *safe zone* yang paling optimal, dilakukan serangkaian pengujian waktu operasional robot. Pengujian ini berfokus pada pengukuran waktu yang dibutuhkan robot untuk mencapai bola, serta jumlah *jitter* (penyesuaian gerak) yang terjadi, pada berbagai persentase *safe zone*. Hasil dari kelima konfigurasi *safe zone* yang diuji (10%, 20%, 30%, 40%, dan 50%) disajikan dalam tabel-tabel berikut.

Tabel 3. 2. 2. Waktu Operasional Safezone 10%

Safezone 10%		
No	Waktu Mencapai Bola (s)	Jumlah Jitter
1	147.19	112
2	108.71	90
3	91.82	86
4	130.19	108
5	163.20	95
6	87.74	69
7	135.47	108
8	121.63	100
9	56.04	41
10	107.42	86
11	100.66	77
12	66.33	51
13	125.20	99
14	87.94	68
15	101.95	79
16	114.37	87
17	98.27	76
18	141.21	109
19	79.51	61
20	123.41	95
21	109.38	84
22	92.19	78
23	101.44	90
24	135.92	102
25	117.31	80
26	86.93	69
27	130.86	98
28	72.39	59
29	107.64	85
30	94.27	92
	107.89	84.47

Tabel 3. 2. 3. Waktu Operasional Safezone 20%

Safezone 20%		
No	Waktu Mencapai Bola (s)	Jumlah Jitter
1	51.45	44
2	11.23	2
3	30.16	20
4	28.66	19
5	29.77	22
6	11.55	0
7	30.18	20
8	16.12	6
9	30.73	21
10	31.39	20
11	40.97	32
12	32.09	19
13	31.22	21
14	22.43	13
15	27.98	19
16	20.96	13
17	11.33	2
18	37.42	25
19	15.10	4
20	38.24	28
21	26.89	18
22	33.13	24
23	19.84	8
24	28.60	21
25	41.77	29
26	15.23	5
27	35.02	25
28	12.60	1
29	30.93	22
30	23.46	17
	27.21	17.33

Tabel 3. 2. 4. Waktu Operasional
Safezone 30%

Safezone 30%		
No	Waktu Mencapai Bola (s)	Jumlah Jitter
1	17.49	8
2	18.75	10
3	16.98	8
4	18.76	9
5	12.40	2
6	17.04	7
7	17.25	7
8	11.27	0
9	19.24	9
10	14.30	3
11	25.00	16
12	10.46	12
13	21.60	10
14	17.01	7
15	10.57	1
16	14.27	5
17	9.26	0
18	19.60	9
19	12.51	2
20	19.47	10
21	19.78	11
22	9.63	13
23	16.86	8
24	16.94	6
25	12.77	2
26	9.55	0
27	17.61	8
28	18.35	9
29	18.30	8
30	15.68	6
	15.96	6.87

Tabel 3. 2. 5. Waktu Operasional
Safezone 40%

Safezone 40%		
No	Waktu Mencapai Bola (s)	Jumlah Jitter
1	13.14	4
2	11.48	1
3	10.13	0
4	10.80	0
5	13.00	2
6	13.59	4
7	10.90	1
8	13.28	3
9	16.11	5
10	13.85	2
11	13.28	2
12	12.10	1
13	11.97	3
14	11.11	0
15	12.04	1
16	14.27	3
17	12.94	2
18	16.91	5
19	12.65	2
20	12.52	1
21	14.36	1
22	15.19	4
23	11.81	2
24	12.41	3
25	13.57	2
26	14.48	4
27	14.53	2
28	13.82	2
29	18.71	5
30	14.87	3
	13.33	2.33

Tabel 3. 2. 6. Waktu Operasional Safezone 50%

Safezone 50%		
No	Waktu Mencapai Bola (s)	Jumlah Jitter
1	11.91	1
2	17.84	0
3	12.21	1
4	11.72	1
5	13.03	1
6	12.75	1
7	17.62	2
8	13.98	1
9	13.02	1
10	11.18	1
11	12.21	1
12	17.26	2
13	12.07	1
14	12.13	1
15	12.58	1
16	11.82	2
17	12.04	1
18	11.46	1
19	13.29	1
20	12.23	1
21	15.51	4
22	14.48	1
23	12.36	1
24	13.10	1
25	12.87	1
26	13.22	1
27	12.16	1
28	12.60	1
29	13.91	2
30	14.13	1
	13.22	1.20

Setelah dilakukan lima kali percobaan dengan konfigurasi *safezone* yang berbeda (10% hingga 50%), diperoleh data rata-rata waktu operasional dan jumlah *jitter* seperti ditunjukkan pada Tabel 3.2.7. Parameter waktu mencapai bola menunjukkan durasi dari awal pergerakan robot hingga menyentuh bola, sementara jumlah *jitter* mencerminkan banyaknya koreksi arah yang dilakukan robot selama proses pengambilan bola.

Tabel 3. 2. 7. Perbandingan Waktu Operasional

Safezone (%)	Waktu Mencapai Bola (s)	Rata-rata Jumlah Jitter
10%	107.89	84.47
20%	27.21	17.33
30%	15.96	6.87
40%	13.33	2.33
50%	13.22	1.20

Dari hasil tersebut, terlihat bahwa semakin sempit *safezone* (misalnya 10%), maka robot menjadi sangat sensitif terhadap perubahan posisi bola, sehingga menghasilkan *jitter* yang tinggi (rata-rata 84,47 kali koreksi) dan waktu operasional yang sangat lama. Sebaliknya, semakin lebar *safezone*, sistem menjadi lebih toleran terhadap deviasi kecil, sehingga jumlah koreksi menurun drastis dan waktu mencapai bola menjadi jauh lebih efisien.

Pada pengujian dengan tegangan ideal 24 volt, ukuran *safezone* 50% menghasilkan performa paling optimal, dengan jumlah *jitter* terendah dan waktu tempuh tercepat. Hal ini menunjukkan bahwa semakin besar daya yang tersedia, sistem *visual servoing* mampu merespons lebih cepat dan stabil, bahkan dengan lebar *safezone* yang lebih besar.

Oleh karena itu, pemilihan ukuran *safezone* yang optimal harus mempertimbangkan keseimbangan antara kecepatan (waktu operasional), stabilitas (jumlah *jitter*), serta kondisi daya operasional sistem.

Tabel 3. 2. 8. Perbandingan Akurasi dan Presisi Waktu Operasional

Safezone	Rata-rata Waktu (s)	Waktu Ideal (s)	Error (%)	Akurasi (%)	Standar Deviasi	Presisi (%)
10%	107.89	11.07	874.62	-774.62	24.76	77.05
20%	27.21	11.07	145.80	-45.80	10.06	63.05
30%	15.96	11.07	44.17	55.83	3.94	75.29
40%	13.33	11.07	20.42	79.58	1.87	85.98
50%	13.22	11.07	19.42	80.58	1.75	86.76

Selain mengevaluasi waktu operasional dan jumlah *jitter*, dilakukan pula analisis terhadap akurasi dan presisi dari sistem *visual servoing* pada tiap konfigurasi *safezone*. Akurasi dihitung berdasarkan seberapa dekat rata-rata waktu mencapai bola terhadap waktu ideal (11.07 detik), sedangkan presisi diukur dari stabilitas performa melalui standar deviasi dari waktu tempuh.

Hasil pengujian menunjukkan bahwa *safezone* 50% menghasilkan akurasi tertinggi sebesar 80.58% dan presisi tertinggi sebesar 86.76%, dengan standar deviasi waktu yang paling kecil (1.75 detik). Sebaliknya, *safezone* 10% memiliki performa terburuk dengan akurasi -774.62% dan presisi 77.05%, akibat waktu tempuh yang sangat jauh dari ideal dan fluktuasi performa yang besar antar percobaan.

Secara umum, semakin lebar *safezone*, akurasi dan presisi meningkat secara signifikan, hingga mencapai titik optimal pada konfigurasi 50%. Hal ini memperkuat temuan sebelumnya bahwa sistem *visual servoing* bekerja paling stabil dan efisien ketika diberi toleransi yang cukup terhadap deviasi posisi objek, terutama dalam kondisi suplai daya optimal.

Tabel 3. 2. 9. Perhitungan Waktu Servo

Waktu Servo
0.50
0.60
0.50
0.50
0.50
0.60
0.50
0.50
0.60
0.50
0.50
0.50
0.60
0.50
0.50
0.50
0.60
0.50
0.50
0.50
0.60
0.50
0.60
0.50
0.50
0.50
0.60
0.50
0.50
0.60
0.50
0.53

Waktu servo dihitung berdasarkan durasi aktifnya servo kamera, yaitu saat sistem menurunkan sudut pandang kamera (*Field of View*) setelah bola masuk ke zona bawah (*bottom zone*), hingga objek tidak lagi terdeteksi dan servo kembali ke posisi awal. Pengukuran dilakukan secara otomatis dalam sistem selama proses visual servoing berlangsung.

Sebanyak 30 data waktu diambil untuk merepresentasikan performa servo secara menyeluruh. Hasil pengujian menunjukkan bahwa rata-rata waktu servo aktif adalah 0.53 detik, dengan nilai minimum sebesar 0.50 detik, dan maksimum 0.60 detik. Nilai standar deviasi yang rendah, yaitu sekitar 0.03 detik, menandakan bahwa sistem servo bekerja secara konsisten dan stabil tanpa fluktuasi waktu yang signifikan.

Nilai ini juga menunjukkan bahwa waktu perhitungan dan respons servo terhadap kondisi bola cukup cepat, serta tidak menimbulkan *delay* yang berarti dalam proses pengambilan keputusan robot. Estimasi waktu ini penting untuk memastikan keseluruhan sistem tetap responsif, terutama saat menghadapi situasi cepat dalam pertandingan.

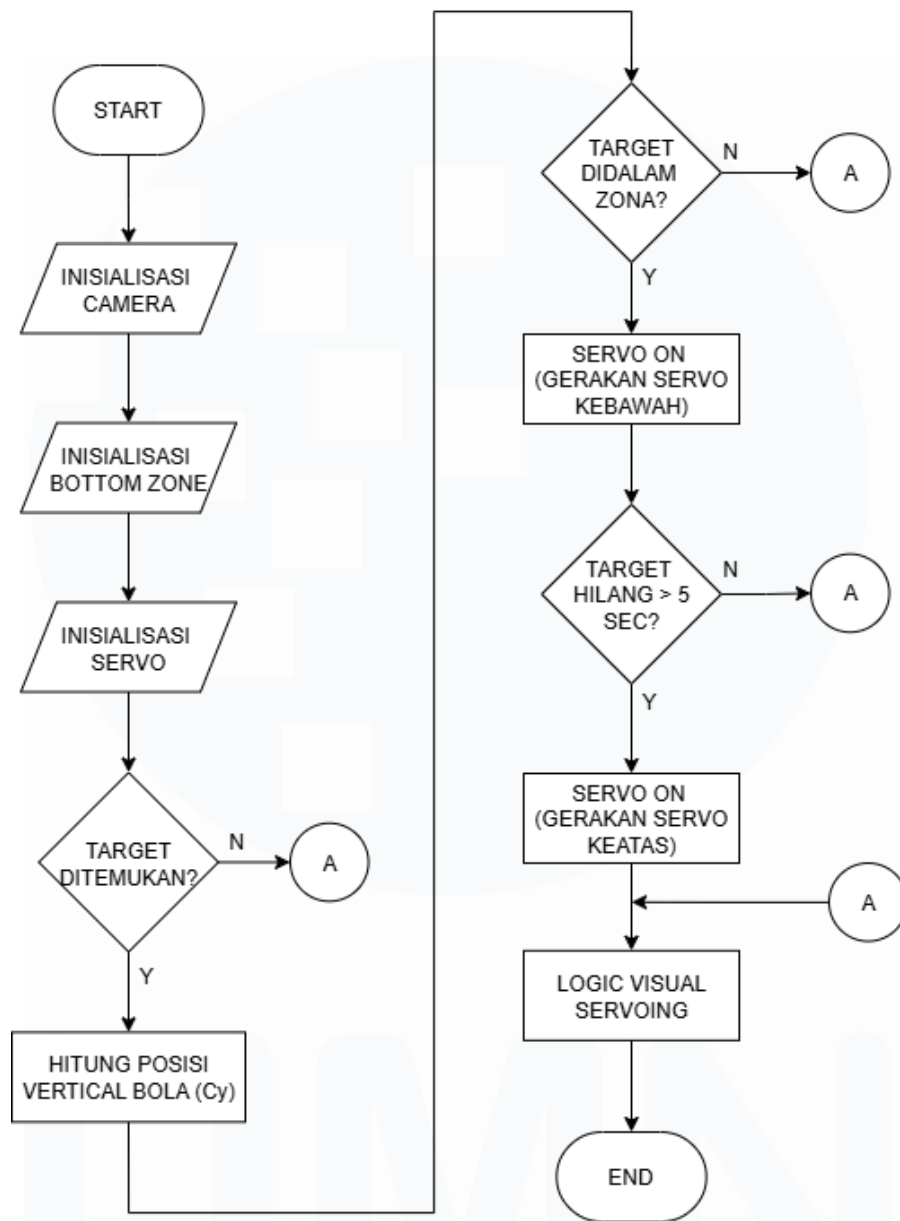
3.3 Servo Camera

Salah satu keterbatasan utama dari kamera USB yang digunakan dalam proyek ini adalah sudut pandangnya (*field of view*) yang relatif sempit. Kamera konvensional seperti Logitech C902 tidak memiliki kemampuan *wide-angle*, sehingga area yang dapat ditangkap dalam satu *frame* menjadi terbatas. Akibatnya, ketika objek seperti bola berada terlalu dekat dengan robot atau berada di posisi ekstrem atas maupun bawah dalam bidang pandang kamera, objek tersebut berisiko keluar dari *frame* dan tidak terdeteksi.

Untuk mengatasi keterbatasan ini, kamera dipasang padaudukan yang terhubung ke motor servo, membentuk sistem yang disebut *servo camera*. Sistem ini memungkinkan kamera untuk bergerak secara vertikal dan menyesuaikan sudut pandangnya secara dinamis, sehingga robot tetap dapat menjaga visibilitas terhadap objek penting meskipun terjadi perubahan posisi relatif antara robot dan objek.

Pengendalian kamera dilakukan berdasarkan posisi vertikal objek dalam citra. Ketika bola terdeteksi mendekati bagian bawah *frame* (indikasi bahwa bola sangat dekat), servo akan menurunkan sudut pandang kamera agar bola tetap berada dalam jangkauan penglihatan. Sebaliknya, jika bola tidak lagi terdeteksi selama lebih dari lima detik, sistem akan mengembalikan posisi kamera ke arah depan (posisi *default*) untuk memulai kembali proses pencarian target.

Untuk merealisasikan logika pengendalian tersebut, sistem memanfaatkan informasi posisi vertikal bola (koordinat *y* dari *bounding box* hasil deteksi) untuk menentukan kapan kamera harus menyesuaikan sudut pandangnya. Ketika bola berada di bawah ambang batas tertentu dalam *frame*, servo akan aktif untuk menurunkan posisi kamera secara bertahap. Sebaliknya, jika objek tidak lagi terdeteksi selama lebih dari lima detik, servo akan mengembalikan kamera ke posisi semula (menghadap lurus ke depan) untuk memulai proses pencarian ulang. Alur pengambilan keputusan ini dijabarkan secara sistematis dalam Gambar 3.3.1, yang menggambarkan *flowchart* pengendalian *servo camera* berdasarkan status deteksi dan posisi vertikal target.

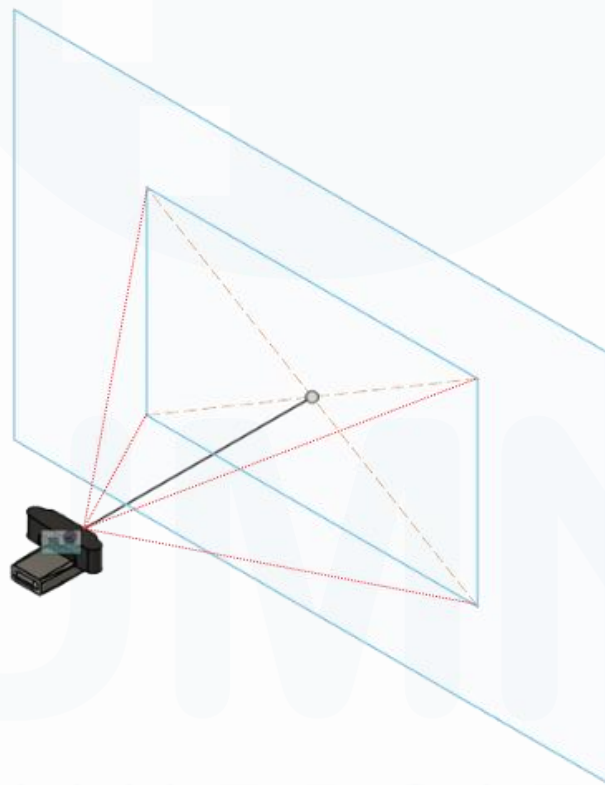


Gambar 3. 3. 1 Flowchart Alur Kerja Pergerakan Servo Kamera

Dengan mengatur orientasi kamera secara kontekstual, *servo camera* secara signifikan meningkatkan konsistensi deteksi dan memperluas cakupan persepsi robot. Pendekatan ini menjadi bagian integral dari *visual servoing* yang digunakan dalam proyek, karena memungkinkan sistem untuk terus melacak dan merespons objek target dengan lebih baik selama pertandingan berlangsung.

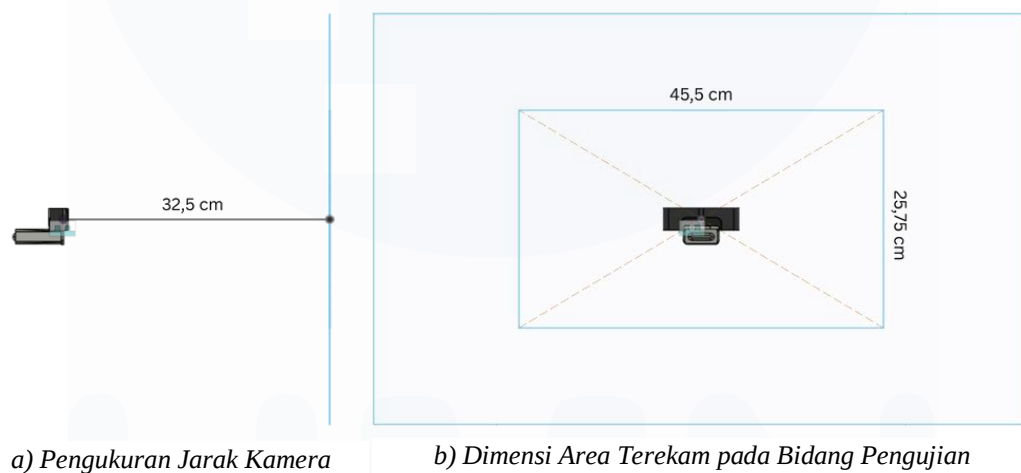
Field of View (FOV) atau sudut pandang kamera, merupakan parameter penting yang mendefinisikan seberapa luas area yang dapat ditangkap oleh kamera. FOV dinyatakan dalam satuan derajat dan bergantung pada panjang fokus lensa serta ukuran sensor kamera [14]. Dalam konteks robotika, pemahaman mengenai FOV kamera sangat krusial untuk perencanaan navigasi, deteksi objek, dan estimasi jarak, karena menentukan jangkauan efektif sistem penglihatan robot.

Pengujian FOV kamera dilakukan dengan mengukur dimensi area yang terekam oleh kamera pada jarak tertentu. Dari eksperimen ini, dimensi fisik dari area yang dapat dilihat kamera pada jarak yang diketahui dapat diukur secara empiris.



Gambar 3. 3. 2. Model Proyeksi Kamera

Untuk memperoleh data empiris yang diperlukan dalam perhitungan *Field of View* (FOV) kamera, dilakukan sebuah pengujian sederhana. Pengujian ini melibatkan penempatan sebuah bidang datar (atau objek dengan dimensi yang jelas) pada jarak tertentu di depan kamera. Kemudian, dimensi fisik dari area yang dapat terekam oleh kamera pada bidang datar tersebut diukur secara langsung. Pengukuran ini dilakukan untuk mendapatkan nilai lebar dan tinggi aktual dari frame kamera pada jarak yang telah ditentukan. Gambar 3.3.3.a dan Gambar 3.3.3.b mengilustrasikan konfigurasi pengukuran yang dilakukan untuk mendapatkan data dimensi tersebut. Data inilah yang kemudian akan digunakan dalam rumus trigonometri untuk menghitung Horizontal FOV dan Vertical FOV kamera.



Gambar 3. 3. 3. Pengukuran Bidang Pandang (Field of View) Kamera

Untuk mencari FoV horizontal dan FoV vertikal, kita gunakan rumus:

$$FoV = 2 * \arctan\left(\frac{Ukuran\ Bidang}{2 * Jarak\ ke\ Objek}\right) \quad (2)$$

Di mana:

- Ukuran Dimensi adalah lebar (untuk Horizontal FOV / HFOV) atau tinggi (untuk Vertical FOV / VFOV) dari area yang terekam.
- Jarak ke Objek adalah jarak dari pusat optik kamera ke bidang tempat dimensi diukur.

FoV Horizontal

$$\text{FoV}_{\text{hor}} = 2 \cdot \arctan\left(\frac{45.5}{2 \cdot 32.5}\right) = 2 \cdot \arctan\left(\frac{45.5}{65}\right)$$

$$\text{FoV}_{\text{hor}} = 2 \cdot \arctan(0.7) \approx 2 \cdot 34.99^\circ = 69.98^\circ$$

FoV Vertikal

$$\text{FoV}_{\text{ver}} = 2 \cdot \arctan\left(\frac{25.75}{2 \cdot 32.5}\right) = 2 \cdot \arctan\left(\frac{25.75}{65}\right)$$

$$\text{FoV}_{\text{ver}} = 2 \cdot \arctan(0.396) \approx 2 \cdot 21.65^\circ = 43.3^\circ$$

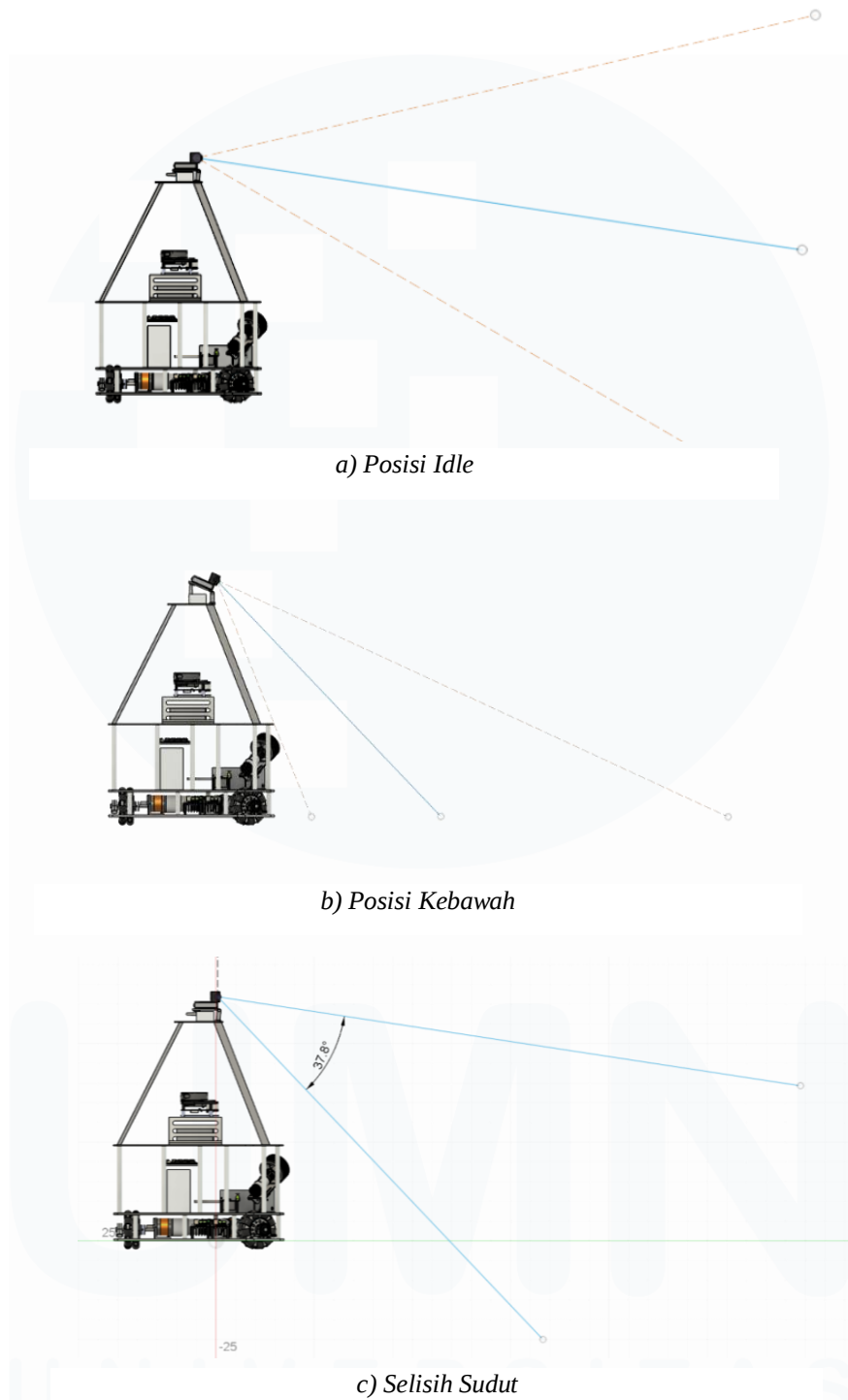
FoV Horizontal: $\approx 70^\circ$

FoV Vertikal: $\approx 43.3^\circ$

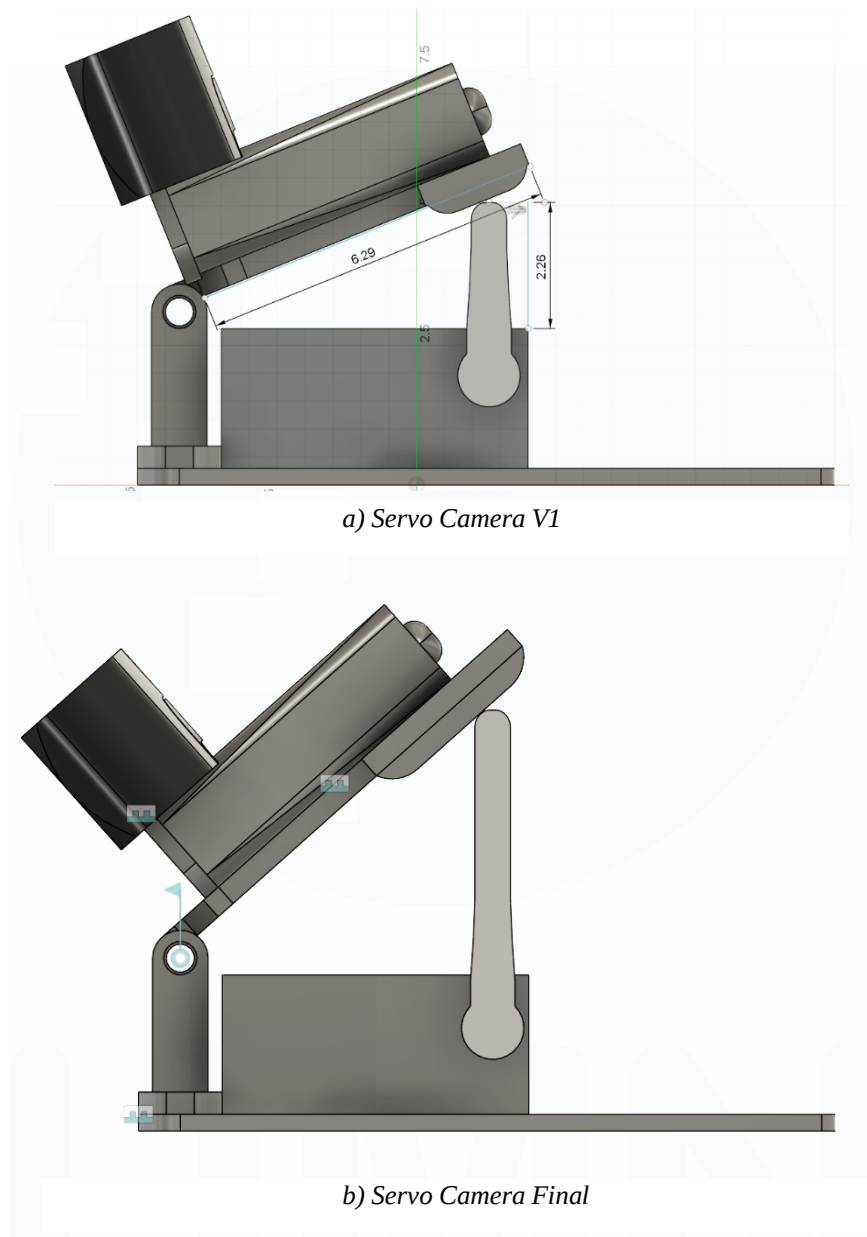
Kamera Logitech C920 yang digunakan memiliki keterbatasan pada *Vertical Field of View* (VFOV), yaitu sekitar 43,3 derajat. Dalam kondisi *default*, area tangkapan kamera hanya mencakup bidang pandang yang terbatas di depan robot. Akibatnya, ketika bola berada sangat dekat dengan dasar robot — misalnya saat akan ditendang — bola dapat berada di luar jangkauan kamera dan tidak terdeteksi.

Untuk mengatasi keterbatasan ini, kamera tidak dipasang secara horizontal sempurna, melainkan dengan kemiringan ke bawah (*tilt*) untuk menggeser *cone of vision* ke arah permukaan lantai di depan robot. Namun, pengujian awal menunjukkan bahwa agar bola tetap terdeteksi saat berada di area sangat dekat, kamera harus dimiringkan hingga ± 47 derajat ke bawah dari horizontal. Sudut sebesar ini secara mekanis dinilai kurang aman, karena melebihi batas stabil 45 derajat. Dalam kondisi robot berhenti mendadak, kamera dapat terdorong atau jatuh ke depan akibat inersia, yang berisiko merusak mekanisme kamera atau dudukan servo.

Sebagai bagian dari optimasi desain, posisi idle kamera kemudian diatur dengan sudut condong ke bawah secukupnya (sekitar 9–10 derajat) untuk mengurangi kebutuhan rotasi servo saat mengikuti bola dari posisi *idle* ke posisi ekstrem bawah. Dengan konfigurasi ini, total rentang rotasi servo untuk mencakup seluruh area penting berhasil dikurangi menjadi sekitar 37,8 derajat. Pengurangan sudut ini tidak hanya meningkatkan keamanan mekanik, tetapi juga menjaga agar deteksi bola tetap optimal pada jarak dekat tanpa mengorbankan responsivitas kamera servo selama pertandingan.



Gambar 3. 3. 4. Ilustrasi Servo Kamera

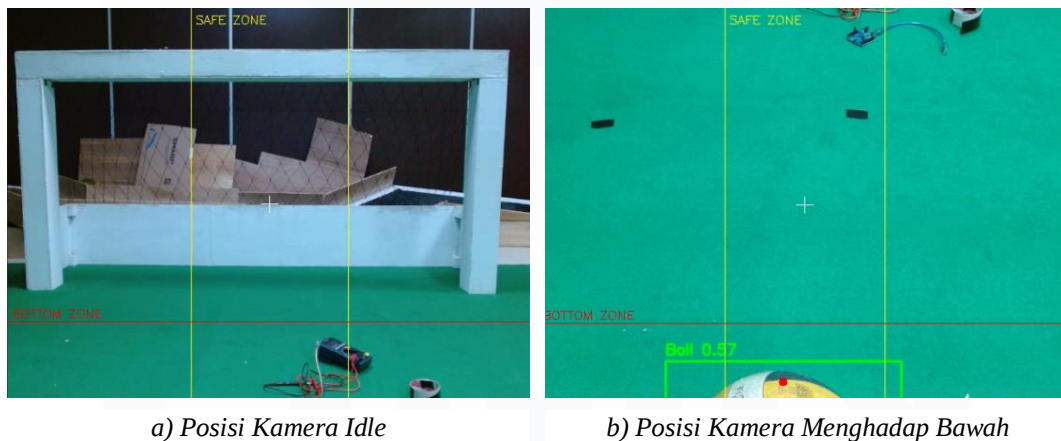


Gambar 3. 3. 5. Perbandingan Servo Camera

Setelah mengidentifikasi kebutuhan sudut *tilt* kamera sekitar 37.8 derajat untuk memastikan visibilitas bola pada jarak sangat dekat, analisis desain awal pada Gambar 3.3.5. (a. *Servo Camera V1*) menunjukkan bahwa mekanisme tersebut tidak mampu mencapai perpindahan vertikal yang diperlukan. Perhitungan sebelumnya menunjukkan bahwa untuk mencapai sudut tersebut dengan panjang horizontal lengan 6.29 cm, perpindahan vertikal harus mencapai sekitar 4.87 cm, yang melebihi kapasitas desain awal.

Oleh karena itu, dilakukan revisi dan optimasi pada desain mekanisme *servo camera* secara signifikan. Gambar 3.3.5. (b. *Servo Camera Final*) menampilkan desain mekanisme *servo camera* yang telah direvisi dan diimplementasikan. Perubahan utama terletak pada pemanjangan atau optimalisasi konfigurasi lengan pengungkit yang terhubung ke motor servo. Desain baru ini memungkinkan motor servo untuk menghasilkan perpindahan vertikal yang lebih besar padaudukan kamera.

Melalui penyesuaian desain ini, kamera kini mampu mencapai sudut tilt yang diperlukan sekitar 37.8 derajat, yang secara efektif menggeser *field of view* vertikal kamera. Hal ini memastikan bahwa robot dapat mempertahankan pandangan visual yang jelas terhadap bola, baik saat berada jauh di depan maupun saat bola sudah berada sangat dekat dengan dasar robot. Desain akhir ini merupakan solusi mekanis yang krusial untuk mendukung fungsionalitas sistem *visual servoing* secara menyeluruh.



Gambar 3. 3. 6 Servo Kamera

Dalam sistem *visual servoing* pada robot KRSBI Beroda, *Bottom Zone* merupakan area deteksi di bagian bawah *frame* kamera yang berfungsi sebagai pemicu otomatis untuk mengaktifkan pergerakan *tilt* kamera ke arah bawah menggunakan servo.

Ketika objek (bola) terdeteksi masuk ke dalam *Bottom Zone* ini, kamera secara otomatis menyesuaikan sudut pandangnya dengan menunduk agar tetap menjaga bola dalam jangkauan pandang. Hal ini penting terutama saat bola berada sangat dekat dengan dasar robot, di mana *field of view* (FOV) vertikal kamera tidak lagi mencakup area tersebut dalam posisi idle.

Pemilihan nilai *Bottom Zone* sebesar 20% dari tinggi frame tidak dilakukan secara asal-asalan, melainkan merupakan hasil kompromi antara sensitivitas sistem dan kestabilan gerakan kamera. Jika nilai ini terlalu kecil, misalnya kurang dari 10%, kamera akan terlambat mendeteksi keberadaan bola di jarak dekat. Sebaliknya, jika nilai terlalu besar, misalnya lebih dari 30%, kamera dapat terlalu sering bergerak ke bawah karena mendeteksi objek yang sebenarnya belum terlalu dekat.

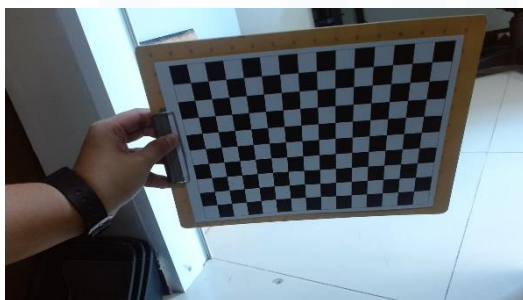
Dengan *Bottom Zone* sebesar 20% dari tinggi frame, sistem mampu mendeteksi bola yang benar-benar mendekati robot tanpa memicu gerakan kamera secara berlebihan. Sudut tilt servo yang digunakan pun telah dioptimalkan menjadi sekitar $37,8^\circ$, cukup untuk menggeser cakupan kamera ke bawah tanpa membebani struktur atau menyebabkan webcam bergeser saat robot melakukan manuver mendadak.

3.4 Kalibrasi Kamera

Dalam sistem penglihatan komputer, khususnya pada proyek robot KRSBI Beroda, kamera digunakan sebagai sumber utama informasi visual. Namun, kamera seperti Logitech C902 memiliki keterbatasan optik, termasuk distorsi pada lensa. Distorsi ini dapat menyebabkan citra objek tampak melengkung atau bergeser dari posisi aslinya, sehingga mengganggu akurasi sistem, terutama saat melakukan pelacakan objek atau estimasi posisi [15].

Kalibrasi kamera dilakukan untuk mengoreksi kesalahan-kesalahan tersebut. Tujuan utamanya adalah memperbaiki distorsi lensa dan mendapatkan parameter intrinsik kamera [16]. Dengan kalibrasi, sistem dapat menginterpretasi posisi objek dengan lebih akurat berdasarkan data citra.

Proses kalibrasi dimulai dengan akuisisi dataset gambar *checkerboard* (papan catur). Dalam proyek ini, telah dikumpulkan sebanyak 45 foto papan catur. Foto-foto ini diambil dari berbagai sudut pandang dan jarak yang berbeda-beda relatif terhadap kamera, untuk memastikan cakupan distorsi lensa yang komprehensif. Variasi sudut dan jarak ini krusial untuk memperoleh parameter kalibrasi yang *robust* dan akurat, yang dapat merepresentasikan karakteristik lensa di seluruh bidang pandang kamera [17], [18].



a) Angle Atas



b) Angle Bawah



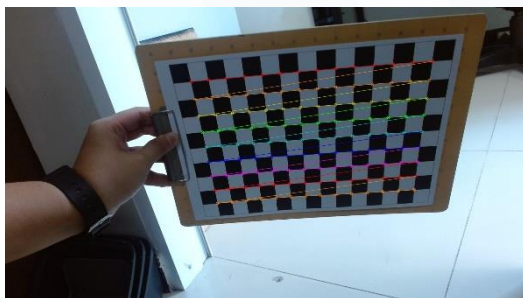
c) Angle Kanan



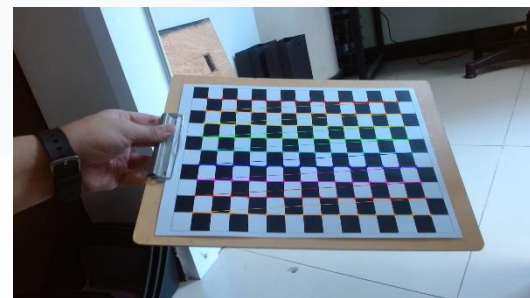
d) Angle Kiri

Gambar 3. 4. 1. Variasi Papan Catur

Sebelum proses kalibrasi dilakukan, diperlukan pengambilan data visual berupa citra *checkerboard* dari berbagai sudut dan jarak. Gambar di bawah ini menunjukkan hasil deteksi pola *checkerboard* oleh sistem pengolahan citra sebelum dilakukan proses kalibrasi.



a) Angle Atas



b) Angle Bawah



c) Angle Kanan



d) Angle Kiri

Gambar 3. 4. 2. Variasi Papan Catur yang sudah di Deteksi

Seperti yang terlihat, citra *checkerboard* mengalami distorsi, terutama pada bagian tepi (samping kiri-kanan maupun atas-bawah), di mana garis-garis yang seharusnya lurus terlihat melengkung atau tampak menyempit. Hal ini menunjukkan adanya distorsi radial dan tangensial akibat karakteristik lensa kamera. Deteksi titik-titik sudut *checkerboard* tetap dapat dilakukan, namun akan menghasilkan parameter posisi yang tidak akurat jika digunakan langsung tanpa kalibrasi.

Gambar-gambar tersebut digunakan sebagai dataset untuk proses kalibrasi kamera, yang kemudian dihitung untuk mendapatkan parameter intrinsik (*camera matrix*) dan koefisien distorsi. Setelah proses kalibrasi selesai, distorsi ini dapat dikoreksi secara matematis, menghasilkan citra yang lebih proporsional dan akurat.

Kalibrasi kamera dilakukan menggunakan metode Zhang untuk memperoleh parameter intrinsik, yaitu *camera matrix* dan *distortion coefficients* [19].

Camera Matrix:

$$\begin{bmatrix} F_x & 0 & C_x \\ 0 & F_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

Distortion Coefficients

$$[k_1, k_2, p_1, p_2, k_3] \quad (4)$$

Hasil kalibrasi ini digunakan untuk meningkatkan akurasi sistem *Visual Servoing* dan deteksi objek menggunakan YOLO pada robot KRSBI Beroda.

Camera Matrix:

$$\begin{bmatrix} 1.44343516e + 03 & 0.00000000e + 00 & 9.35708933e + 02 \\ 0.00000000e + 00 & 1.58613571e + 03 & 5.11999374e + 02 \\ 0.00000000e + 00 & 0.00000000e + 00 & 1.00000000e + 00 \end{bmatrix}$$

Matriks ini, yang sering disebut juga sebagai matriks intrinsik kamera $\mathbf{K}$, adalah representasi matematis dari bagaimana kamera memproyeksikan titik-titik 3D di dunia nyata ke *pixel-pixel* 2D pada sensornya. Komponen-komponennya memiliki makna sebagai berikut:

- $f_x = 1443.43516$ *pixel*: Menunjukkan panjang fokus kamera dalam satuan *pixel* pada sumbu horizontal (X).
- $f_y = 1586.13571$ *pixel*: Menunjukkan panjang fokus kamera dalam satuan *pixel* pada sumbu vertikal (Y). Perbedaan antara f_x dan f_y yang signifikan dapat mengindikasikan adanya aspek rasio *pixel* non-persegi atau distorsi tertentu pada sensor atau lensa.
- $c_x = 935.708933$ *pixel*: Merupakan koordinat X (horizontal) dari *principal point*, yaitu proyeksi titik pusat optik lensa ke sensor kamera. Idealnya, ini mendekati pusat gambar.
- $c_y = 511.999374$ *pixel*: Merupakan koordinat Y (vertikal) dari *principal point*.
- Elemen nol di posisi (1,2) dan (2,1) menunjukkan asumsi bahwa sumbu *pixel* adalah ortogonal (tidak miring satu sama lain), yang umum untuk sebagian besar kamera modern.

Distortion Coefficients:

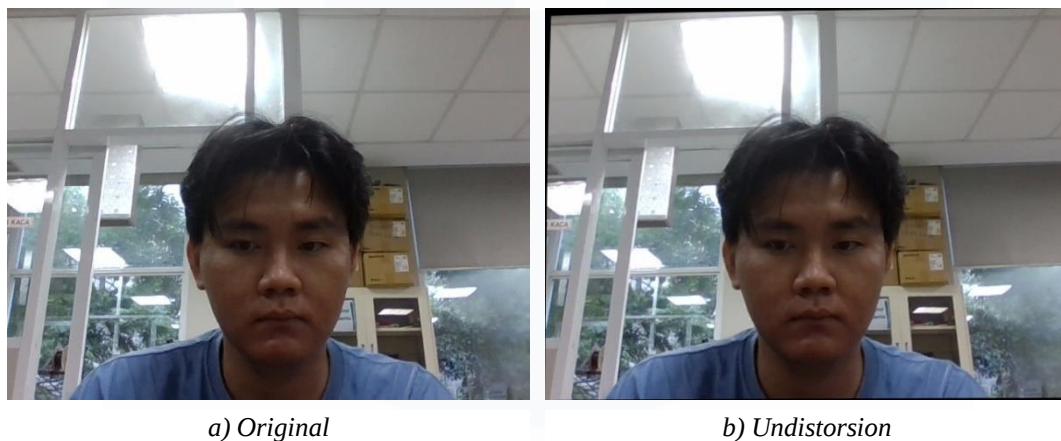
[0.09608164 -0.25137013 -0.00396017 -0.00063722 0.31532353]

Koefisien ini merepresentasikan model distorsi lensa yang digunakan untuk mengoreksi citra. Urutan koefisien ini biasanya mengikuti model distorsi Brown-Conrady, yaitu:

- $k_1 = 0.09608164$: Koefisien distorsi radial pertama. Nilai positif menunjukkan jenis distorsi bantal (*pincushion distortion*) pada tingkat tertentu.
- $k_2 = -0.25137013$: Koefisien distorsi radial kedua. Nilai negatif ini menunjukkan jenis distorsi barel (*barrel distortion*) yang lebih kuat di area tertentu, atau kombinasi yang menciptakan efek distorsi yang kompleks.

- $p_1 = -0.00396017$: Koefisien distorsi tangensial pertama. Distorsi tangensial terjadi ketika lensa tidak sejajar sempurna dengan sensor gambar.
- $p_2 = -0.00063722$: Koefisien distorsi tangensial kedua.
- $k_3 = 0.31532353$: Koefisien distorsi radial ketiga. Ini digunakan untuk model distorsi yang lebih kompleks, menangani efek yang tidak sepenuhnya dijelaskan oleh k_1 dan k_2 .

Nilai-nilai koefisien distorsi ini digunakan dalam rumus matematis untuk "memindahkan" *pixel-pixel* pada citra terdistorsi ke posisi yang seharusnya pada



Gambar 3. 4. 3. Perbandingan Gambar Original dengan Gambar yang sudah Diolah citra yang tidak terdistorsi. Semakin besar nilai absolut koefisien ini, semakin besar pula distorsi yang harus dikoreksi. Akurasi dari parameter-parameter ini sangat menentukan seberapa baik citra dapat dikoreksi dan seberapa andal sistem penglihatan robot dapat bekerja.

Pada gambar ini, yang merupakan *frame* mentah dari kamera, dapat diamati adanya distorsi lensa yang halus, terutama pada objek-objek di area pinggir citra. Perhatikan garis-garis lurus pada jendela dan rak buku di latar belakang, khususnya yang mendekati tepi *frame*. Meskipun mungkin tidak terlalu mencolok pada pandangan pertama, garis-garis tersebut cenderung terlihat sedikit melengkung ke luar (*barrel distortion*) atau ke dalam (*pincushion distortion*) akibat sifat optik lensa kamera. Efek ini dapat menyebabkan objek di tepi gambar tampak sedikit melar atau tertekan, dan posisinya sedikit bergeser dari realitas geometrisnya.

Setelah menerapkan *Camera Matrix* dan *Distortion Coefficients* hasil kalibrasi, citra di sebelah kanan menunjukkan hasil koreksi distorsi. Pada citra ini, garis-garis lurus di latar belakang (jendela, rak) yang sebelumnya mungkin terlihat melengkung kini tampak lebih lurus dan proporsional di seluruh area *frame*. Koreksi ini memastikan bahwa setiap *pixel* dalam citra yang tidak terdistorsi merepresentasikan posisi spasial yang lebih akurat di dunia nyata. Dengan citra yang tidak terdistorsi, semua perhitungan berbasis *pixel* selanjutnya, seperti penentuan pusat objek untuk *visual servoing* dan estimasi jarak, akan menjadi jauh lebih presisi dan andal. Hal ini krusial untuk memastikan bahwa robot dapat mengenali dan berinteraksi dengan lingkungan dengan akurasi yang tinggi.



Gambar 3. 4. 4. Perbandingan Gambar Original dan Gambar yang sudah Dipetakan Ulang

Gambar 3.4.4. menyajikan perbandingan visual antara citra mentah yang diakuisisi langsung dari kamera (*original webcam*) dan citra yang telah mengalami proses koreksi distorsi (*undistorted webcam*) dengan menerapkan parameter kalibrasi yang telah diperoleh.

Berdasarkan observasi visual, perbedaan antara citra terdistorsi dan tidak terdistorsi pada kasus ini tergolong subtle dan mungkin tidak secara langsung tampak signifikan pada inspeksi kasat mata. Fenomena ini dapat diatribusikan pada beberapa faktor:

- Karakteristik Optik Lensa: Kamera Logitech C920 umumnya dilengkapi dengan lensa yang memiliki distorsi intrinsik yang relatif rendah

dibandingkan dengan lensa *fish-eye* atau lensa *wide-angle* ekstrem. Kamera dengan kualitas optik yang baik secara inheren menghasilkan distorsi geometris yang minim, sehingga efek koreksi pasca-kalibrasi pun akan terlihat lebih halus.

- Penempatan Subjek pada Bidang Tengah Citra: Subjek utama dalam citra (area wajah) terpusat pada bagian tengah *frame*. Distorsi lensa, baik radial maupun tangensial, secara intrinsik memiliki efek yang paling minimal pada area pusat citra dan meningkat secara signifikan menuju tepi serta sudut *frame*. Dengan dominasi informasi visual di area sentral yang kurang terdistorsi, dampak visual dari proses undistortion pada citra keseluruhan menjadi tidak terlalu mencolok.

Meskipun demikian, penting untuk dicatat bahwa ketiadaan perbedaan visual yang drastis tidak mereduksi esensi dan urgensi kalibrasi kamera dalam aplikasi robotika presisi. Proses kalibrasi menghasilkan parameter numerik (Matriks Kamera dan Koefisien Distorsi) yang esensial bagi algoritma penglihatan komputer. Bagi algoritma, bahkan penyimpangan *pixel* yang sangat kecil akibat distorsi dapat mengakumulasi error yang signifikan dalam perhitungan spasial. Sebagai contoh:

- Akurasi Estimasi Jarak: Algoritma estimasi jarak sangat bergantung pada hubungan geometris yang akurat antara objek dalam citra 2D dan posisinya di ruang 3D. Distorsi yang tidak terkoreksi akan menyebabkan kesalahan fatal dalam perhitungan panjang fokus efektif dan posisi titik utama, sehingga menghasilkan estimasi jarak yang tidak reliabel.
- Ketepatan *Visual Servoing*: Dalam sistem *visual servoing* (IBVS), di mana kontrol robot didasarkan pada posisi *pixel* objek, distorsi dapat menyebabkan target yang seharusnya lurus atau berada pada posisi tertentu di dunia nyata, terproyeksikan secara miring atau bergeser di citra. Koreksi distorsi memastikan bahwa error yang dihitung dari citra benar-benar merepresentasikan error geometris yang akurat di dunia nyata.

Oleh karena itu, meskipun efek *undistortion* pada citra yang disajikan mungkin tampak subtil secara visual, parameter kalibrasi yang dihasilkan memiliki peran fundamental dalam memastikan integritas data visual untuk pemrosesan algoritma yang akurat. Hal ini krusial untuk mencapai performa yang stabil dan presisi pada sistem robot KRSBI Beroda, khususnya dalam tugas-tugas yang membutuhkan pemahaman spasial lingkungan dan interaksi objek.



3.5 Estimasi Jarak

Dalam kompetisi KRSBI Beroda, seluruh sistem robot harus beroperasi secara autonomus tanpa bantuan kendali eksternal seperti *remote* atau *controller*. Oleh karena itu, diperlukan mekanisme persepsi visual yang mampu memberikan informasi spasial secara *real-time* agar robot dapat mengambil keputusan secara mandiri. Salah satu aspek penting dalam proses ini adalah kemampuan robot untuk mengetahui jarak antara dirinya dengan objek target, seperti gawang. Untuk memenuhi kebutuhan tersebut, sistem penglihatan pada robot dilengkapi dengan estimasi jarak berbasis kamera, menggunakan prinsip *focal length* yang diperoleh melalui proses kalibrasi kamera. Untuk menghitung jarak antara kamera dengan objek dapat menggunakan persamaan matematis sebagai berikut :

Jarak Fokus (*Focal Length*)

$$F = \frac{P \times D}{W} \quad (5)$$

Jarak Objek

$$D = \frac{F \times W}{P}$$

Dimana :

- P adalah lebar objek dalam *pixel*
- F adalah jarak fokus kamera
- W adalah lebar sebenarnya dari objek

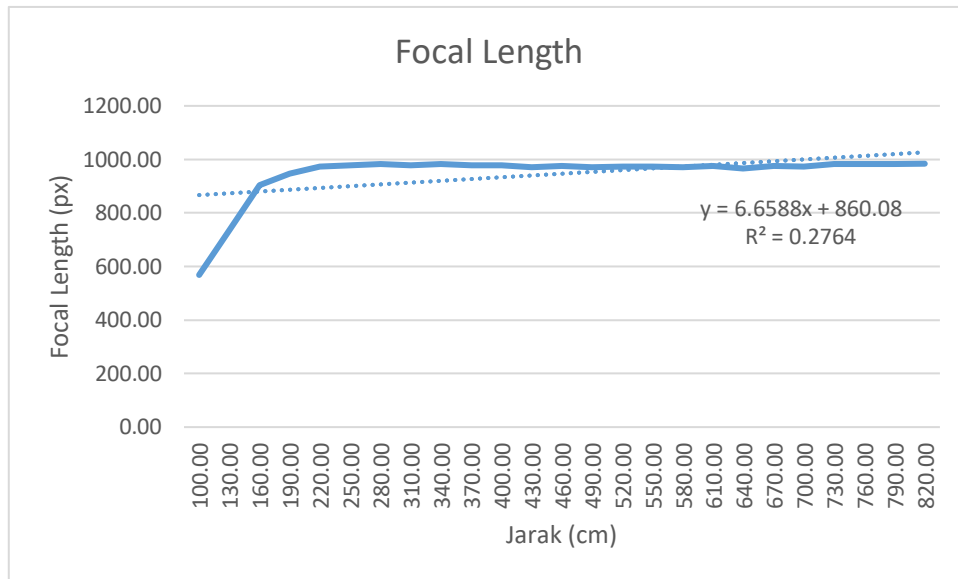
Metode yang digunakan untuk mengukur jarak objek dengan kamera adalah membandingkan objek dalam gambar dengan sebenarnya. Maka dari itu, diperlukan perhitungan jarak fokus untuk menghitung jarak sebenarnya dari objek.

Ukuran gawang pada kompetisi ini terbilang cukup besar, yaitu sekitar 2 meter × 1 meter. Ketika objek sebesar ini diamati dari jarak yang terlalu dekat, seperti 10 cm hingga 20 cm, ukuran *bounding box* hasil deteksi akan memenuhi hampir seluruh *frame* kamera. Hal ini menyebabkan ketidakseimbangan proporsi *pixel*, sehingga estimasi jarak menjadi kurang akurat dan tidak stabil. Untuk menghindari masalah tersebut, proses estimasi jarak terhadap gawang dibatasi mulai dari jarak

minimal sekitar 100 cm, dengan interval pengambilan data setiap 30 cm. Hal ini dilakukan karena pengambilan data pada jarak yang sangat dekat relatif sulit, baik dari sisi kestabilan deteksi maupun konsistensi posisi kamera. Pada jarak mulai dari 100 cm dan seterusnya, gawang masih terdeteksi dengan baik dan ukurannya tetap proporsional di dalam citra, sehingga estimasi jarak berbasis *focal length* menjadi lebih presisi dan konsisten.

Tabel 3. 5. 1. Evaluasi Akurasi Estimasi Jarak Berdasarkan Ukuran Bounding Box pada Gawang

No	Distance (cm)	Real Width (cm)	Object Width (px)	Focal Length	Estimasi Jarak (cm)	Error (%)	Sensitifitas
1	100,00	225,00	1279,00	568,44	166,53	66,53	0,00
2	130,00	225,00	1275,00	736,67	167,06	28,50	0,02
3	160,00	225,00	1270,00	903,11	167,71	4,82	0,02
4	190,00	225,00	1120,00	945,78	190,17	0,09	0,75
5	220,00	225,00	995,00	972,89	214,07	2,70	0,80
6	250,00	225,00	880,00	977,78	242,04	3,18	0,93
7	280,00	225,00	790,00	983,11	269,61	3,71	0,92
8	310,00	225,00	710,00	978,22	299,99	3,23	1,01
9	340,00	225,00	650,00	982,22	327,69	3,62	0,92
10	370,00	225,00	595,00	978,44	357,98	3,25	1,01
11	400,00	225,00	550,00	977,78	387,26	3,18	0,98
12	430,00	225,00	508,00	970,84	419,28	2,49	1,07
13	460,00	225,00	477,00	975,20	446,53	2,93	0,91
14	490,00	225,00	446,00	971,29	477,57	2,54	1,03
15	520,00	225,00	421,00	972,98	505,93	2,71	0,95
16	550,00	225,00	398,00	972,89	535,16	2,70	0,97
17	580,00	225,00	377,00	971,82	564,98	2,59	0,99
18	610,00	225,00	360,00	976,00	591,65	3,01	0,89
19	640,00	225,00	340,00	967,11	626,46	2,12	1,16
20	670,00	225,00	328,00	976,71	649,38	3,08	0,76
21	700,00	225,00	313,00	973,78	680,50	2,79	1,04
22	730,00	225,00	303,00	983,07	702,96	3,70	0,75
23	760,00	225,00	291,00	982,93	731,94	3,69	0,97
24	790,00	225,00	280,00	983,11	760,70	3,71	0,96
25	820,00	225,00	270,00	984,00	788,87	3,80	0,94
				946,65			0,83



Grafik 3. 5. 1. Variasi Focal Length Berdasarkan Ukuran Bounding Box terhadap Jarak

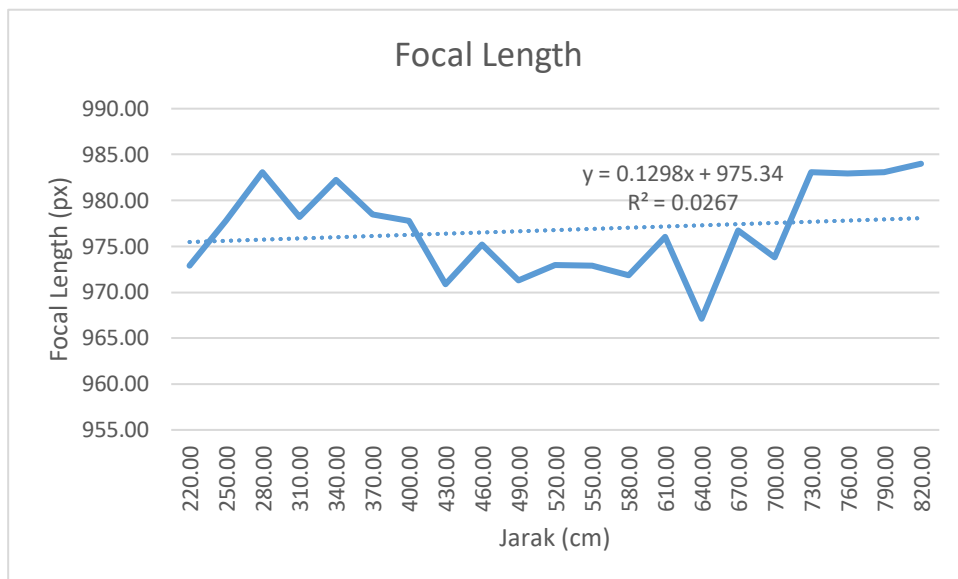
Dari hasil proses pengambilan data estimasi jarak menggunakan metode *focal length* tetap, diperoleh nilai rata-rata *focal length* sebesar 946,65 *pixel*. Namun, selama pengujian pada rentang jarak 100 cm hingga 190 cm, ditemukan adanya fluktuasi data yang cukup signifikan. Hal ini mengindikasikan adanya data abnormal, yang kemungkinan disebabkan oleh beberapa faktor seperti deteksi *bounding box* yang tidak stabil, pencahayaan yang berubah-ubah, atau sudut pandang kamera yang kurang ideal terhadap objek.

Akibat dari ketidaksempurnaan data tersebut, diperoleh nilai rata-rata galat (*error*) estimasi jarak sebesar 6,59%, yang menunjukkan bahwa pada jarak-jarak tertentu, hasil estimasi menyimpang cukup jauh dari jarak sebenarnya. Hal ini menunjukkan bahwa meskipun metode estimasi jarak dengan *focal length* dapat digunakan secara umum, akurasi pada jarak menengah (100–190 cm) masih perlu diperbaiki, misalnya dengan melakukan penyaringan data *outlier*, penambahan penyesuaian posisi kamera, atau penggunaan metode tambahan sebagai pembanding.

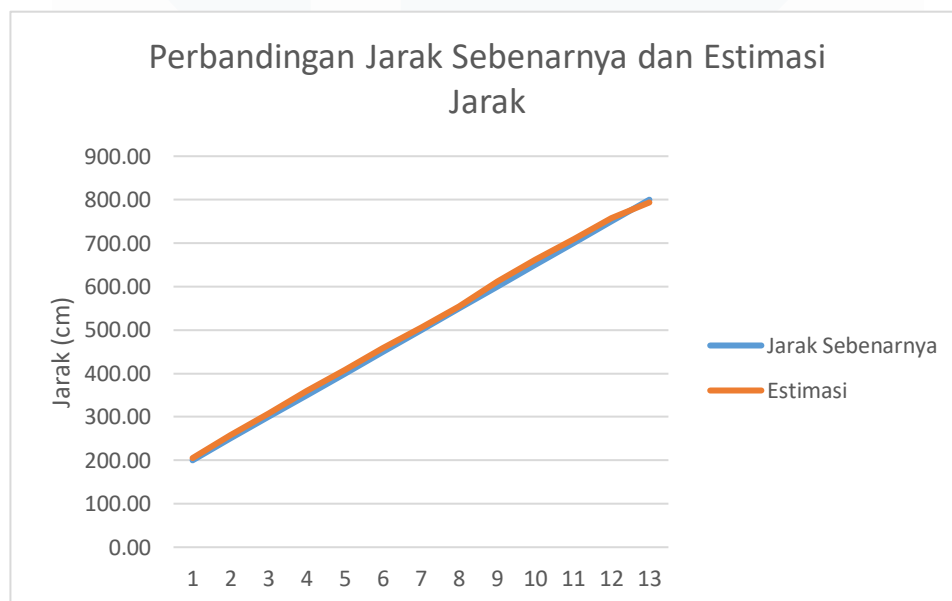
Tabel 3. 5. 2. Performa Estimasi Jarak Menggunakan Focal Length Konstan

No	Distance (cm)	Real Width (cm)	Object Width (px)	Focal Length	Estimasi Jarak (cm)	Error (%)	Sensitifitas
1	220,00	225,00	995,00	972,89	220,92	0,42	0,00
2	250,00	225,00	880,00	977,78	249,79	0,08	0,96
3	280,00	225,00	790,00	983,11	278,25	0,63	0,95
4	310,00	225,00	710,00	978,22	309,60	0,13	1,05
5	340,00	225,00	650,00	982,22	338,18	0,54	0,95
6	370,00	225,00	595,00	978,44	369,44	0,15	1,04
7	400,00	225,00	550,00	977,78	399,67	0,08	1,01
8	430,00	225,00	508,00	970,84	432,71	0,63	1,10
9	460,00	225,00	477,00	975,20	460,83	0,18	0,94
10	490,00	225,00	446,00	971,29	492,86	0,58	1,07
11	520,00	225,00	421,00	972,98	522,13	0,41	0,98
12	550,00	225,00	398,00	972,89	552,30	0,42	1,01
13	580,00	225,00	377,00	971,82	583,07	0,53	1,03
14	610,00	225,00	360,00	976,00	610,60	0,10	0,92
15	640,00	225,00	340,00	967,11	646,52	1,02	1,20
16	670,00	225,00	328,00	976,71	670,17	0,03	0,79
17	700,00	225,00	313,00	973,78	702,29	0,33	1,07
18	730,00	225,00	303,00	983,07	725,47	0,62	0,77
19	760,00	225,00	291,00	982,93	755,38	0,61	1,00
20	790,00	225,00	280,00	983,11	785,06	0,63	0,99
21	820,00	225,00	270,00	984,00	814,14	0,71	0,97
				976,96		0,42	0,94

UIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Grafik 3. 5. 2. Variasi Focal Length Berdasarkan Ukuran Bounding Box terhadap Jarak



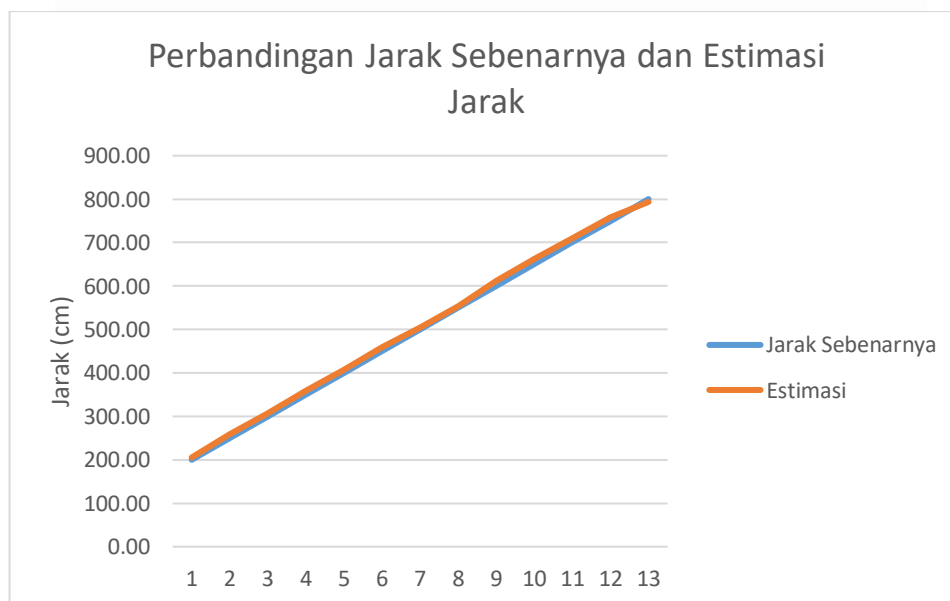
Grafik 3. 5. 3. Estimasi vs Jarak Aktual dengan Focal Length Konstan

Setelah dilakukan penyaringan terhadap data yang tidak stabil, diperoleh nilai *focal length* sebesar 976,96 *pixel* dengan rata-rata *error* sebesar 0,42% terhadap jarak sebenarnya. Hasil estimasi menunjukkan bahwa nilai jarak hasil perhitungan relatif sesuai dengan pengukuran aktual dalam kondisi pengambilan data yang terkontrol.

Untuk memastikan akurasi sistem dalam kondisi lapangan, nilai *focal length* yang telah diperoleh akan dibandingkan secara langsung dengan hasil pengukuran aktual di lingkungan pertandingan. Jika ditemukan ketidaksesuaian antara estimasi dan realita, maka nilai *focal length* akan disesuaikan guna meningkatkan keakuratan estimasi jarak selama pertandingan berlangsung.

Tabel 3. 5. 3. Perbandingan Jarak Aktual dan Estimasi

No	Distance (cm)	Estimasi (cm)	Error (%)
3	200,00	205,24	2,62
4	250,00	258,61	3,44
5	300,00	307,87	2,62
6	350,00	359,35	2,67
7	400,00	407,82	1,96
8	450,00	458,91	1,98
9	500,00	505,32	1,06
10	550,00	553,69	0,67
11	600,00	612,30	2,05
12	650,00	662,10	1,86
13	700,00	709,08	1,30
14	750,00	757,99	1,07
15	800,00	793,56	0,81
			1,85



Grafik 3. 5. 4. Perbandingan Jarak Sebenarnya dan Estimasi Jarak