

## **Prototipe Sistem Sentry Gun Berbasis Raspberry Pi**

**Heri Andrianto <sup>1\*)</sup>, Yohanes Satrio Bhanu Aji <sup>2</sup>, Yohana Susanthi <sup>3</sup>**

<sup>1,2,3</sup>)Program Studi Teknik Elektro  
Universitas Kristen Maranatha  
Jalan Prof. Drg. Surya Sumantri, M.P.H No. 65 Bandung - 40164

<sup>\*)</sup>Korespondensi : heri.andrianto@eng.maranatha.edu

### **Abstrak**

*Sentry gun* merupakan senjata dengan sistem bidik otomatis yang secara efisien digunakan dalam sistem keamanan suatu tempat atau daerah. *Sentry gun* dapat bergerak ke arah kiri, kanan, atas, dan bawah mengikuti pergerakan target yang sedang dibidik. Sistem *sentry gun* menggunakan deteksi gerakan melalui kamera dalam melakukan *tracking*. Sistem *sentry gun* memerlukan *library* dan *single board computer* yang mendukung agar mampu mendeteksi gerakan dan menembak sasaran dengan tepat. Pada makalah ini, *library* OpenCV dan Raspberry Pi digunakan untuk mendeteksi gerakan pada *sentry gun* dan menembak sasaran. Pelacakan pergerakan pada sistem deteksi gerak diimplementasikan menggunakan algoritma segmentasi latar depan dan latar belakang berbasis model campuran Gaussian (*background subtraction*). Metodologi penelitian dalam makalah ini terdiri dari empat tahap. Dimulai dengan kajian pustaka tentang *sentry gun* dari penelitian sebelumnya, dilanjutkan dengan perancangan dan realisasi sistem *sentry gun*, diakhiri dengan pengujian sistem *sentry gun*. Berdasarkan hasil pengujian, didapatkan persentase tertinggi tingkat keberhasilan pergerakan sebesar 100%, dan persentase tertinggi jumlah peluru mengenai target terhadap jumlah tembakan sebesar 80%. *Sentry gun* telah berhasil direalisasikan dan berfungsi dengan baik.

**Kata kunci :** *Background Subtraction, OpenCV, Raspberry Pi, Sentry Gun*

### **Abstract**

*A sentry gun is a weapon equipped with an automatic targeting system that is efficiently used in the security system of a location or area. The sentry gun can move to the left, right, up, and down following the movement of the target being aimed at. The sentry gun system uses motion detection through a camera to perform tracking. The system requires supporting libraries and a single-board computer to detect motion and accurately shoot the target. In this paper, the OpenCV library and Raspberry Pi are used to detect motion in the sentry gun system and to fire at the target. Movement tracking in the motion detection system is implemented using a foreground-background segmentation algorithm based on the Gaussian Mixture Model (background subtraction). The research methodology in this paper consists of four stages. It begins with a literature review on sentry guns from previous studies, followed by the design and implementation of the sentry gun system, and concludes with system testing. Based on the test results, the highest percentage of movement success rate achieved was 100%, and the highest percentage of bullets hitting the target relative to the total number of shots fired was 80%. The sentry gun has been successfully implemented and functions properly.*

**Keywords :** *Background Subtraction, OpenCV, Raspberry Pi, Sentry Gun*

## **I. PENDAHULUAN**

Keamanan negara terus mengalami perkembangan dari waktu ke waktu dan memerlukan sistem pengamanan yang memadai demi menjamin kesejahteraan masyarakat [1]. Dalam sistem penjagaan suatu tempat, diperlukan banyak penjaga untuk menjaga tempat-tempat penting yang harus dijaga keamanannya. Pada sistem penjagaan secara manual dibutuhkan keterampilan penjaga dalam menembak [2]. Proses manual memerlukan waktu dan tenaga [3]. Selain itu, penjaga tidak dapat terus menerus menjaga tempat-tempat yang harus dijaga karena tidak dapat menanggung beberapa efek kantuk dan kelelahan yang dapat mempengaruhi akurasi tembakan senjata [4].

---

**Info Makalah:**

Dikirim : 02-23-2026;  
Revisi 1 : 06-25-2026;  
Revisi 2 : -  
Diterima : 06-29-2026.

**Penulis Korespondensi:**

Telp : +6281809536225  
e-mail :  
[heri.andrianto@eng.maranatha](mailto:heri.andrianto@eng.maranatha).

---

*Sentry gun* merupakan senjata yang secara otomatis membidik dan menembak target yang terdeteksi oleh sensor [4][5][6][7][8]. *Sentry gun* sangat penting dalam hal pertahanan karena dapat digunakan secara otomatis tanpa harus dikendalikan secara manual oleh penggunanya. Dalam pengoperasian sistem *sentry gun* diperlukan sebuah *Single Board Computer* (SBC) yang mendukung sistem *sentry gun* agar mampu mendeteksi objek yang menjadi target tembakan, menggerakkan *sentry gun* sesuai dengan target yang terdeteksi oleh kamera dan menembakan peluru kepada target. Salah satu SBC yang dapat mendukung pengoperasian sistem *sentry gun* yaitu Raspberry Pi.

Raspberry Pi adalah sebuah SBC yang memiliki fungsi yang kompleks, Raspberry Pi dapat digunakan untuk pengolahan citra digital [9][10]. Untuk membuat program pada Raspberry Pi diperlukan bahasa pemrograman Python [11][12]. Sama seperti halnya SBC yang lain, Raspberry Pi akan mengeksekusi program yang sudah dibuat melalui sistem operasi (SO) Raspberry Pi. Selain itu, Raspberry Pi dapat mengeksekusi program yang dibuat dari luar SO Raspberry Pi seperti internet, sehingga dapat mempermudah dalam proses pembuatan program.

Pada makalah ini telah dikembangkan prototipe sistem *sentry gun* menggunakan Raspberry Pi dan kamera *webcam* yang kompatibel dengan Raspberry Pi untuk membidik target secara otomatis. Fokus pada makalah ini adalah merancang dan merealisasikan prototipe sistem *sentry gun* yang pengoperasiannya dibantu oleh Raspberry Pi, kamera, dan senjata mainan sebagai keluaran untuk membuktikan bahwa *sentry gun* dapat membidik target secara presisi. Pada makalah ini, *library* yang digunakan untuk membuat program prototipe sistem *sentry gun* berbasis Raspberry Pi yaitu *OpenCV* dan *Adafruit* untuk menjalankan beberapa *hardware* yang kompatibel dengan Raspberry Pi.

## II. KAJIAN PUSTAKA

Bagian ini menyajikan beberapa penelitian yang telah dilakukan sebelumnya oleh beberapa peneliti. Penelitian mengenai *sentry gun* telah dilakukan dengan berbagai macam cara, perangkat keras dan perangkat lunak. Yudistira dkk telah melaporkan sistem kendali senjata handgun otomatis menggunakan kamera berbasis Raspberry Pi dan ATmega 2560 [10]. Ahmad dkk telah melaporkan pengembangan *2-DOF Autonomous Sentry Gun* dengan biaya rendah, dengan pembahasan utama pada deteksi gerak dari *sentry gun* [13]. Asmoro telah melaporkan pengembangan model *autonomous sentry gun* menggunakan pemrosesan citra dan gerak [14]. Raharjo dkk telah melaporkan rancangan pendeteksi sasaran tembak yang dapat dikendalikan dari jarak jauh dengan menggunakan metode eksperimen berbasis PID [15]. Simatupang dkk telah melaporkan pengembangan sistem *tracking* pada *sentry gun* sebagai pengganti pos dengan berbasis *Internet of Things* [16]. Wijaya dkk telah melaporkan proses autentifikasi pada *sentry gun* supaya tidak digunakan sembarangan menggunakan biometrik palm vein sensor [17].

Pada penelitian-penelitian mengenai sistem *sentry gun* yang sudah dilakukan sebelumnya, penelitian-penelitian tersebut sudah menggunakan SBC namun hanya sampai pada tahap perancangan, belum ada data hasil uji coba sistem *sentry gun* dalam menembak objek yang bergerak [10][13]. Pada makalah ini, sistem *sentry gun* yang direalisasikan menggunakan Raspberry Pi, kamera *webcam* dan metode deteksi gerakan dibuat dengan menggunakan *library OpenCV* agar *sentry gun* dapat bergerak mengikuti pergerakan target yang dibidik dan menembak target sesuai sasaran.

Metode utama dalam melakukan deteksi gerakan pada makalah ini yaitu segmentasi latar depan dan latar belakang berbasis model campuran Gaussian. Metode ini dapat membedakan antara gerakan yang asli, bayangan, dan perubahan pada pencahayaan. Jika proses yang berjalan dapat memodelkan latar belakang, maka perubahan substansial dapat dipantau sehingga dapat mendeteksi target dan *sentry gun* dapat bergerak mengikuti target berdasarkan objek yang bergerak.

Pada makalah ini, proses untuk deteksi gerakan dilakukan dengan melakukan proses *background subtraction* menggunakan metode model campuran Gaussian (*Gaussian Mixture Model*). Setelah itu, proses selanjutnya yaitu mendeteksi tepi dari objek yang bergerak, lalu menentukan panjang dan lebar

dari objek, kemudian dilanjutkan dengan menggambar *Region of Interest* (ROI), dan melakukan proses *tracking* terhadap objek yang bergerak.

Dari percobaan Rosenberg C terhadap 7 metode *background subtraction*: *Basic*, *One Gaussian(1-G)*, *Min Max Inter-Frame Difference*, *Eigen Backgrounds (Eigen)*, *Codebook (CBRGB)*, *Kernel Density Estimation (KDE)*, dan *Gaussian Mixture Model (GMM)*, metode yang memiliki tingkat akurasi dengan berbagai kondisi input video adalah GMM [18][19][20]. GMM merupakan metode yang digunakan untuk menggambarkan piksel pada latar belakang (*background*) yang selalu statis dengan menggabungkan beberapa distribusi Gaussian sederhana[21][22].

Setiap komponen pada *Gaussian Mixture Model* (GMM) memiliki parameter berupa rata-rata (*mean*), variansi, dan bobot (*weight*) yang digunakan untuk memodelkan karakteristik nilai piksel dari waktu ke waktu. Pada proses *background subtraction*, GMM digunakan untuk membedakan piksel yang termasuk latar belakang (*background*) dan latar depan (*foreground*). Semakin banyak komponen Gaussian yang digunakan, semakin baik variasi latar belakang dapat direpresentasikan.

$$\{x^1, \dots, x^t\} \quad (1)$$

$$\Pr(x) : \sum_{k=1}^K \omega_k \times \eta(x; \mu_k, \sigma_k) \quad (2)$$

$$M_k(x^t) = \begin{cases} 1, & \frac{(x^t - \mu_k^t)^2}{\sigma_k^t} \leq 2.5 \\ 0, & \text{lainnya} \end{cases} \quad (3)$$

$$\omega_k^t = (1 - \alpha)\omega_k^{t-1} + \alpha \quad (4)$$

$$\mu_k^t = (1 - \rho_k)\mu_k^{t-1} + \rho_k x^t \quad (5)$$

$$\rho_k = \alpha \times \eta(x^t; \mu_k^{t-1}, \sigma_k^{t-1}) \quad (6)$$

$$(\sigma_k^t)^2 = (1 - \rho_k)(\sigma_k^{t-1})^2 + \rho_k(x^t - \mu_k^t)^2 \quad (7)$$

$$B = \text{argmin}_b (\sum_{k=1}^b \omega_k \geq T) \quad (8)$$

Persamaan (1) menunjukkan kumpulan nilai piksel yang diamati dari *frame* ke *frame* sebagai data masukan untuk membangun model. Persamaan (2) merupakan fungsi probabilitas GMM yang memodelkan distribusi nilai piksel menggunakan beberapa komponen Gaussian. Persamaan (3) digunakan untuk menentukan apakah nilai piksel saat ini sesuai dengan salah satu model Gaussian yang telah terbentuk.

Selanjutnya, Persamaan (4), (5), dan (7) digunakan untuk memperbarui parameter model berupa bobot, rata-rata, dan variansi agar dapat beradaptasi terhadap perubahan kondisi citra. Persamaan (6) menghitung faktor pembelajaran ( $\rho_k$ ) yang menentukan besarnya pengaruh data piksel baru terhadap pembaruan parameter model. Setelah parameter diperbarui, Persamaan (8) digunakan untuk menentukan komponen Gaussian yang dikategorikan sebagai latar belakang berdasarkan bobot kumulatif yang melebihi ambang batas tertentu.

Berdasarkan hasil pemodelan tersebut, piksel yang sesuai dengan model latar belakang diklasifikasikan sebagai *background* dan diberi warna hitam, sedangkan piksel yang tidak sesuai diklasifikasikan sebagai *foreground* dan diberi warna putih. Hasil segmentasi ini kemudian digunakan untuk mendeteksi objek yang bergerak pada citra video.

Setelah melakukan *background subtraction*, dilanjutkan dengan proses deteksi tepi. Deteksi tepi merupakan proses yang dilakukan untuk menentukan garis-garis tepi dari suatu objek. Deteksi tepi yang dilakukan, digunakan untuk menentukan titik koordinat terluar berupa ( $x_{\max}$ ,  $y_{\max}$ ), sehingga dapat menentukan panjang dan lebar dari objek yang diproses. Selain itu, proses untuk membuat garis *Region of Interest* (ROI) pada objek yang bergerak dan proses *tracking* pada objek yang bergerak dapat dilakukan setelah menentukan deteksi tepi objek. Untuk melakukan deteksi tepi pada suatu objek tidak rumit, yaitu dengan mengurangi nilai intensitas antar piksel yang saling bertetangga. Ketika hasil pengurangan

tersebut memiliki nilai yang cukup besar maka akan digambar garis diantara piksel yang saling bertetangga tersebut. Proses ini dilakukan oleh setiap piksel dengan nilai-nilai tertentu yang berbeda. Terdapat 2 hal yang mempengaruhi proses deteksi tepi, yaitu ukuran tetangga ( $k$ ) dan *threshold* sebagai batas dari hasil pengurangan antar piksel. Gambar 1 memperlihatkan proses pengurangan antar piksel.

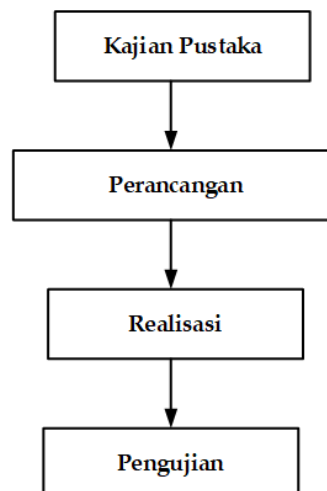
12	90	89	86	87	82
10	12	88	85	83	84
9	15	12	84	84	88
12	14	10	82	88	89
11	17	16	12	88	90
10	16	15	17	89	88

Gambar 1. Proses Pengurangan Antar Piksel

Proses *tracking* ini akan mengkonversi koordinat piksel ke koordinat langkah motor, sehingga motor yang digunakan dapat bergerak sesuai dengan langkah yang ditentukan berdasarkan koordinat langkahnya menuju koordinat langkah pada objek yang bergerak.

### III. METODE

Metodologi penelitian pada makalah ini terdiri dari 4 (empat) tahap, yaitu tahap pertama melakukan kajian pustaka mengenai penelitian yang sudah dilakukan dan deteksi gerak, tahap kedua merancang sistem dan merealisasikan sistem *sentry gun*, tahap ketiga melakukan pengujian sistem *sentry gun* dan tahap keempat melakukan analisis hasil pengujian sistem *sentry gun*. Gambar 2 memperlihatkan metodologi penelitian yang digunakan pada makalah ini.

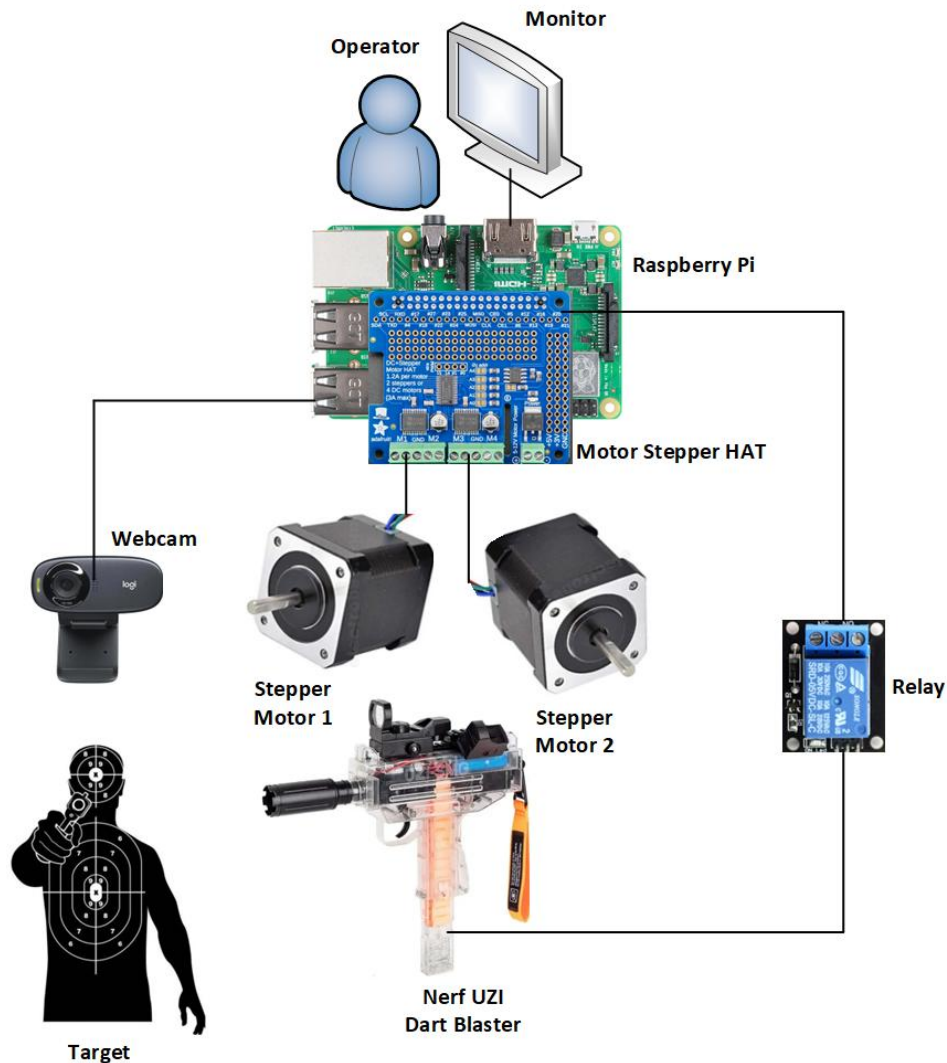


Gambar 2. Metode Penelitian

Tahap pertama berupa kajian pustaka tentang *sentry gun* mengenai penelitian yang sudah dilakukan dan deteksi gerak telah dibahas pada bagian bab kajian pustaka. Tahap kedua berupa perancangan perangkat keras dan perangkat lunak sistem *sentry gun*. Sistem *sentry gun* yang dikembangkan pada makalah ini

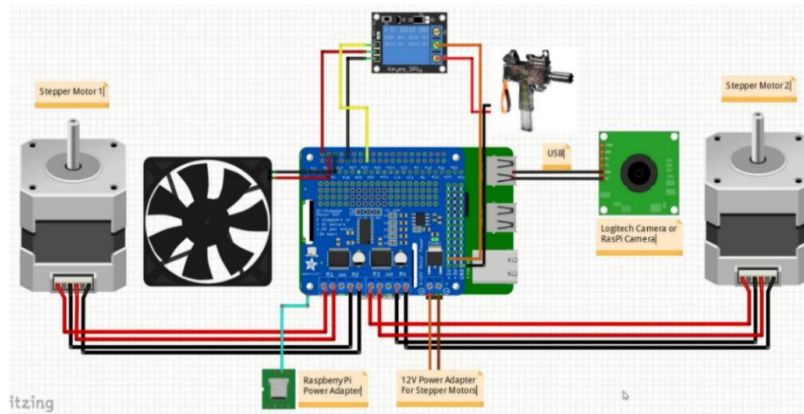
**Prototipe Sistem Sentry Gun Berbasis Raspberry Pi**  
**(Heri Andrianto, Yohanes Satrio Bhanu Aji, Yohana Susanthi : Halaman 28 - 39)**

menggunakan beberapa komponen yang terdiri dari Raspberry Pi 3 model B+, kamera webcam Logitech C310, motor stepper HAT, motor stepper, relay, dan senjata mainan Nerf UZI Dart Blaster. Sistem *sentry gun* dilengkapi juga dengan fasilitas antarmuka seperti antarmuka kamera webcam dengan Raspberry Pi menggunakan *Universal Serial Bus* (USB), antarmuka monitor dengan Raspberry Pi menggunakan HDMI, antarmuka Raspberry Pi dengan motor *Stepper* HAT melalui pin GPIO Raspberry Pi, antarmuka Pan motor dengan gerak Nerf UZI Dart Blaster menggunakan *motor stepper* 1, dan antarmuka Tilt motor dengan gerak Nerf UZI Dart Blaster menggunakan motor stepper 2. Diagram blok dari sistem *sentry gun* diperlihatkan pada Gambar 3.



Gambar 3. Diagram Blok Sistem *Sentry Gun*

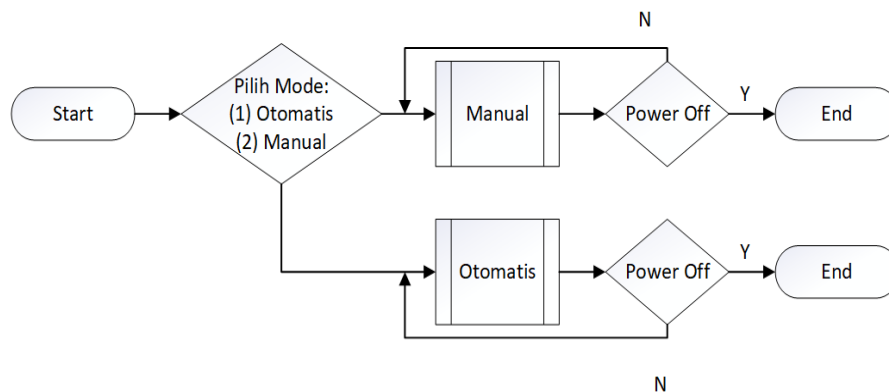
*Wiring diagram* adalah representasi skematik yang menggambarkan hubungan antar komponen elektronik melalui sambungan kabel agar dapat saling terintegrasi. Pada motor *stepper* HAT dihubungkan dengan beberapa pin GPIO yang sesuai dengan Raspberry Pi. Kemudian *relay* tunggal dan kedua motor *stepper* yang digunakan sebagai penggerak *sentry gun*, dihubungkan dengan motor *stepper* HAT (motor *controller*) dan Raspberry Pi melalui pin GPIO yang sesuai. Kemudian *relay* tunggal ini dihubungkan dengan senjata mainan Nerf UZI *Dart Blaster*, sehingga dapat dijadikan sebagai *switch on-off* untuk senjata mainan Nerf UZI *Dart Blaster*. Gambar 4 memperlihatkan *wiring diagram* dari *sentry gun* yang dibuat.



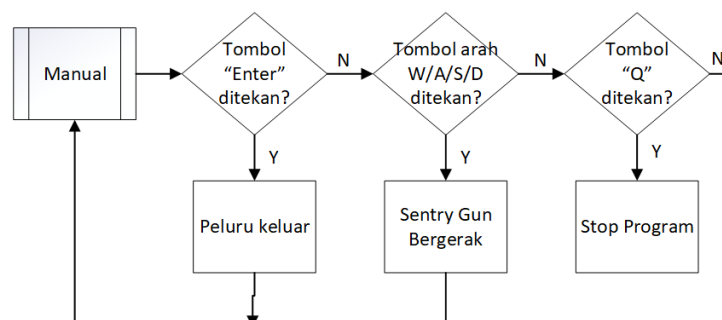
Gambar 4. Wiring Diagram *Sentry Gun*

Hubungan antar komponen pada sistem sentry gun adalah sebagai berikut: Pin M1 dan M2 dihubungkan ke Stepper Motor 1, Pin M3 dan M4 dihubungkan ke Stepper Motor 2, Pin GPIO 22 dihubungkan ke Pin S (Relay 5V), Pin 5V dihubungkan ke Pin + (Relay 5V), Pin GND dihubungkan Pin - (Relay 5V), Pin 5V dihubungkan ke pin + (Kipas DC 5V), Pin GND dihubungkan ke pin - (Kipas DC 5V) dan Port USB Raspberry Pi dihubungkan ke Logitech C310 Webcam.

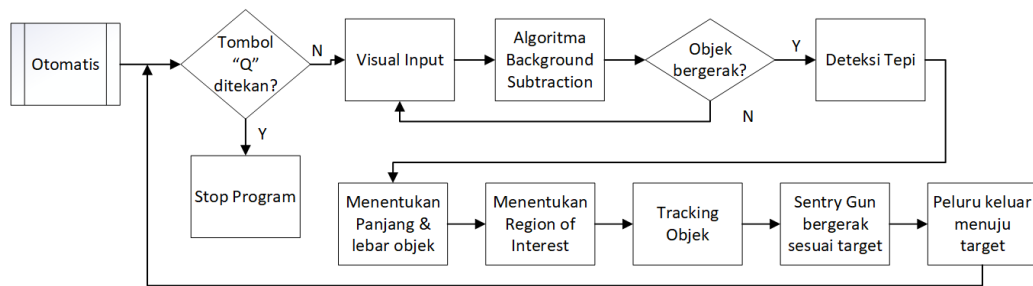
Alur operasi perangkat lunak sistem ini merupakan penjelasan alur ketika user mengoperasikan *sentry gun*. Program utama (Gambar 5) akan dimulai dengan pilihan mode manual dan otomatis, jika memilih mode manual maka *sentry gun* akan beroperasi secara manual (Gambar 6), jika memilih mode otomatis maka *sentry gun* akan beroperasi secara otomatis (Gambar 7). Cara kerja sistem *sentry gun* pada mode otomatis yaitu diawali dengan pendeteksian target yang bergerak oleh kamera webcam, kemudian target tersebut akan diikuti pergerakannya oleh *sentry gun* dengan menggerakkan motor arah horizontal dan motor arah vertikal. Lalu *sentry gun* akan menembak target tersebut ketika target tersebut terdeteksi diam.



Gambar 5. Flowchart Program Utama



Gambar 6. Flowchart Sub Program Manual



Gambar 7. Flowchart Subprogram Otomatis

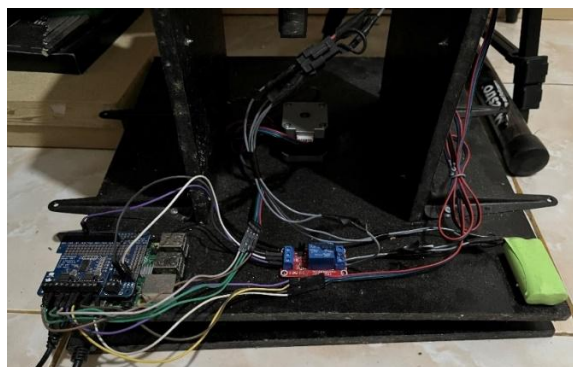
Realisasi alat ini merupakan langkah penerapan dari rancangan desain dan *wiring diagram sentry gun* sebelumnya. Realisasi ini terdiri dari realisasi mekanik (Gambar 8) dan realisasi elektronik (Gambar 9 dan Gambar 10). Realisasi mekanik ini adalah realisasi desain *sentry gun* menggunakan komponen pendukung selain komponen elektronik. Pada realisasi mekanik ini, perlu merancang atau memikirkan segala ukuran dengan tepat agar apabila rusak atau salah implementasi, tidak sulit untuk memperbaikinya. Dengan merancang ukuran yang tepat untuk konstruksi *sentry gun*, maka *sentry gun* dapat berfungsi dengan baik. Sedangkan realisasi elektronik ini merupakan implementasi desain *sentry gun* dari bagian yang menggunakan komponen elektronik. Beberapa komponen elektronik yang sudah disiapkan dilakukan *wiring* terhadap pin GPIO pada Raspberry Pi dan setiap komponennya dihubungkan menggunakan kabel.



Gambar 8. Realisasi Mekanik Sentry Gun



Gambar 9. Realisasi Elektronik Sentry Gun



Gambar 10. Realisasi Elektronik Sentry Gun

Perangkat lunak yang digunakan untuk menjalankan program pada sistem *sentry gun* dijalankan melalui terminal Raspberry Pi dengan sebuah tampilan antarmuka bagi pengguna untuk mengontrol *sentry gun* secara manual maupun otomatis. Tampilan dari antarmuka ini dibuat dengan sederhana agar dapat mudah

dimengerti bagi penggunaannya. Antarmuka pengguna dirancang untuk melakukan perintah kalibrasi, mode manual atau otomatis, dan menampilkan video secara langsung. Antarmuka pengguna ini dirancang menggunakan pemrograman Python. Gambar 11 memperlihatkan tampilan antarmuka pengguna.

```
=====
SENTRY GUN
(Yohanes Satrio BA/1922009)
=====
Mode:
(1) Otomatis
(2) Manual
Pilih Mode: 1

Kalibrasi arah vertikal sentry gun:
(W) Atas, (S) Bawah.
Tekan (enter) untuk selesai.

.....

Kalibrasi arah horizontal sentry gun:
(A) Kiri, (D) Kanan.
Tekan (enter) untuk selesai.

.....

Kalibrasi Selesai.

Beroperasi dengan Video? (y, n): y

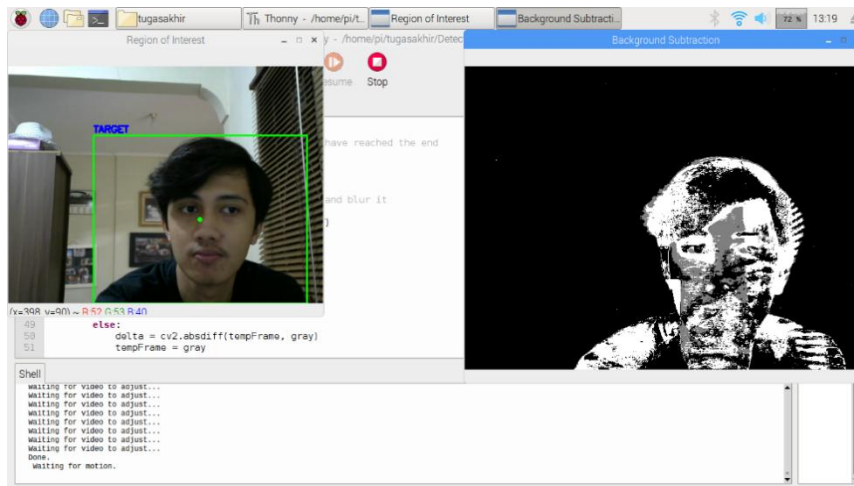
.....
.....Sentry Gun Beroperasi dengan Video
```

Gambar 11. Tampilan Antarmuka untuk Pengguna

#### IV. HASIL DAN DISKUSI

Bagian ini memaparkan data pengamatan dan analisis data dari hasil percobaan terhadap *sentry gun* yang berupa hasil pergerakan dari *sentry gun*, peluru mengenai target, dan jumlah peluru mengenai target terhadap jumlah tembakan. Percobaan pada *sentry gun* dilakukan sebanyak 10 kali pada dua kondisi, yaitu kondisi terang dengan intensitas cahaya sebesar 140 lux, dan kondisi gelap dengan intensitas cahaya sebesar 15 lux. Dari ketiga kondisi tersebut, percobaan dilakukan dengan jarak tertentu, yaitu dengan jarak 1 meter, 2 meter, dan 3 meter karena keterbatasan peluru yang dilontarkan oleh *sentry gun* hanya mampu mencapai jarak 4 meter - 5 meter. Untuk kecepatan target yang bergerak diatur sebesar 0.1 m/s – 1 m/s, karena dengan kecepatan 0.1 m/s objek yang bergerak sudah sangat lambat, dan dengan kecepatan 1 m/s objek yang bergerak sudah cukup cepat dengan respon *delay* yang sama yaitu 6 detik. Batas kecepatan pada objek menggunakan interval, dikarenakan objek yang bergerak tidak bergerak secara konstan.

Gambar 12 memperlihatkan salah satu percobaan proses deteksi gerak. Deteksi gerakan yang dipakai pada makalah ini menggunakan algoritma *background subtraction*. Proses perancangan dari deteksi gerakan ini diawali dengan *background subtraction* untuk pemisahan latar depan dan latar belakang. Kemudian melakukan deteksi tepi dari objek yang bergerak dan mencari nilai panjang dan lebar dari objek tersebut. Setelah itu, objek yang telah dicari panjang dan lebarnya akan digambarkan kotak mengelilingi objek tersebut sesuai dengan tinggi dan lebarnya (*Region of Interest*). Nama target dan titik tengah pada objek yang digunakan sebagai titik target digambar berdasarkan nilai panjang dan lebarnya objek tersebut.



Gambar 12. Salah Satu Percobaan Proses Deteksi Gerak

Dari hasil percobaan dilakukan proses penghitungan untuk menentukan tingkat keberhasilan pergerakan, peluru mengenai target, dan jumlah peluru mengenai target terhadap jumlah tembakan dari *sentry gun*. Dari ketiga data tersebut dalam berbagai kondisi tertentu, data ini dipengaruhi oleh faktor lain seperti kesalahan manusia (*human error*), kualitas perangkat keras, dan kualitas perangkat lunak. Dengan adanya faktor tersebut, persentase *error* dapat dihitung menggunakan rumus (9). Persentase tingkat keberhasilan pergerakan, peluru mengenai target, dan jumlah peluru mengenai target terhadap jumlah tembakan memiliki perhitungan yang berbeda pada setiap datanya.

$$\% \text{ error} = 100\% - \% (\text{keberhasilan}) \quad (9)$$

#### A. PERGERAKAN SENTRY GUN

Dari data pengamatan pergerakan *sentry gun*, analisis dilakukan untuk menentukan tingkat keberhasilan pergerakan *sentry gun* dalam mengikuti pergerakan target. Penghitungan tingkat keberhasilan pergerakan *sentry gun* menggunakan rumus (10). Tabel 1 memperlihatkan analisis hasil pengujian pergerakan *sentry gun*.

$$\% \text{ keberhasilan pergerakan} = \frac{\% \text{ gerak horizontal} + \% \text{ gerak vertikal}}{2} \quad (10)$$

Tabel 1. Pergerakan *Sentry Gun*

Faktor Kondisi	Jumlah Pergerakan yang berhasil	Keberhasilan Pergerakan (%)	Error (%)
Jarak 1 meter, kondisi terang (140 lux)	Horizontal kiri (8); Horizontal kanan (9) Vertikal atas (10); Vertikal bawah (10)	92.5	7.5
Jarak 2 meter, kondisi terang (140 lux)	Horizontal kiri (10); Horizontal kanan (10) Vertikal atas (10); Vertikal bawah (10)	100	0
Jarak 3 meter, kondisi terang (140 lux)	Horizontal kiri (7); Horizontal kanan (10) Vertikal atas (10); Vertikal bawah (10)	92.5	7.5
Jarak 1 meter, kondisi gelap (15 lux)	Horizontal kiri (9); Horizontal kanan (9) Vertikal atas (10); Vertikal bawah (10)	95	5
Jarak 2 meter, kondisi gelap (15 lux)	Horizontal kiri (10); Horizontal kanan (10) Vertikal atas (10); Vertikal bawah (10)	100	0
Jarak 3 meter, kondisi gelap (15 lux)	Horizontal kiri (8); Horizontal kanan (6) Vertikal atas (10); Vertikal bawah (10)	85	15

**B. PELURU MENGENAI TARGET**

Pada peluru mengenai target ini, analisis data dilakukan untuk menghitung tingkat keberhasilan peluru mengenai target dalam berbagai kondisi tertentu. Dalam melakukan perhitungan pada data peluru mengenai target, rumus yang digunakan adalah rumus (11). Masing-masing faktor kondisi dilakukan percobaan sebanyak 10 kali. Tabel 2 memperlihatkan hasil analisis peluru mengenai target.

$$\% \text{ keberhasilan peluru mengenai target} = \frac{\text{jumlah percobaan berhasil}}{\text{banyak percobaan}} \times 100\% \quad (11)$$

Tabel 2. Peluru Mengenai Target dari 10 kali percobaan

Faktor Kondisi	Peluru Mengenai target	Peluru Mengenai Target(%)	Error(%)
Jarak 1 meter, kondisi terang (140 lux)	7	70	30
Jarak 2 meter, kondisi terang (140 lux)	7	70	30
Jarak 3 meter, kondisi terang (140 lux)	8	80	20
Jarak 1 meter, kondisi gelap (15 lux)	8	80	20
Jarak 2 meter, kondisi gelap (15 lux)	7	70	30
Jarak 3 meter, kondisi gelap (15 lux)	8	80	20

**C. JUMLAH PELURU MENGENAI TARGET TERHADAP JUMLAH TEMBAKAN (90 TEMBAKAN)**

Pada data jumlah peluru mengenai target terhadap jumlah tembakan ini, analisis yang dilakukan berbeda dengan data sebelumnya. Analisis yang dilakukan pada data ini menggunakan parameter jumlah tembakan dan jumlah peluru yang mengenai target. Untuk menentukan persentase jumlah peluru mengenai target terhadap jumlah tembakan dalam berbagai kondisi tertentu menggunakan rumus (12). Tabel 3 memperlihatkan hasil analisis jumlah peluru mengenai target terhadap jumlah tembakan.

$$\% \text{ jumlah peluru mengenai target} = \frac{\text{jumlah peluru mengenai target}}{\text{jumlah tembakan}} \times 100\% \quad (12)$$

Tabel 3. Jumlah Peluru Mengenai Target Terhadap Jumlah Tembakan (90 Tembakan)

Faktor Kondisi	Peluru Mengenai Target	Peluru Mengenai Target (%)	Error (%)
Jarak 1 meter, kondisi terang (140 lux)	72	80	20
Jarak 2 meter, kondisi terang (140 lux)	58	64.4	35.6
Jarak 3 meter, kondisi terang (140 lux)	54	60	40
Jarak 1 meter, kondisi gelap (15 lux)	72	80	20
Jarak 2 meter, kondisi gelap (15 lux)	67	74.4	25.6
Jarak 3 meter, kondisi gelap (15 lux)	51	56.6	43.4

**V. KESIMPULAN**

Prototipe Sistem *Sentry gun* telah berhasil dirancang dan direalisasikan menggunakan Raspberry Pi dengan metode deteksi gerakan melalui algoritma segmentasi latar depan dan latar belakang berbasis model campuran Gaussian (*background subtraction*). *Sentry gun* dapat direalisasikan dengan baik menggunakan alat dan bahan yang sudah ditentukan sebelumnya. *Sentry gun* dapat beroperasi sesuai dengan rancangan yang sudah dibuat. *Sentry gun* dapat mengikuti pergerakan target dengan sangat baik. Tingkat keberhasilan pergerakan secara horizontal maupun vertikal tertinggi adalah 100% pada faktor kondisi jarak 2 meter dalam cahaya terang (140 lux) maupun gelap (15 lux). Tingkat keberhasilan peluru mengenai target tertinggi adalah 80% pada faktor kondisi jarak 3 meter dengan cahaya terang (140 lux), faktor kondisi jarak 1 meter dengan cahaya gelap (15 lux) dan faktor kondisi jarak 3 meter dengan

cahaya gelap (15 lux). Sedangkan *sentry gun* dengan persentase jumlah peluru mengenai target terhadap jumlah tembakan (90 tembakan) tertinggi adalah 80% pada faktor kondisi jarak 1 meter dengan cahaya terang (140 lux) maupun gelap (15 lux). Prototipe system *sentry gun* yang telah direalisasikan dapat menembak target sesuai dengan sasaran tembak.

#### UCAPAN TERIMAKASIH

Ucapan terima kasih disampaikan kepada Universitas Kristen Maranatha atas dukungan yang telah diberikan.

#### DAFTAR PUSTAKA

- [1] F. S. Saro, S. R. U. A. Sompie, and E. K. Allo, "Rancang Bangun Alat Simulasi Latihan Menembak Berbasis Arduino Uno," *J. Tek. Elektro dan Komput.*, vol. 7, no. 3, pp. 251–258, 2018.
- [2] S. H. Sabila, Pp. D. P. Adi, M. Baidlowi, and S. Subairi, "Sistem Otomatis Penggerak Lesan Tembak Untuk Latihan Teknik Tempur Cepat Dengan Metode PID," *J. Apl. Sains, Informasi, Elektron. dan Komput.*, vol. 3, no. 2, pp. 63–70, 2022.
- [3] H. Andrianto, Y. Susanthi, V. Jonathan, and F. Teknik, "Platform Sistem Pemantauan Penggunaan Energi Listrik Berbasis IoT," *Techné J. Ilm. Elektrotek.*, vol. 23, no. 2, pp. 199–212, 2024.
- [4] N. Gupta, G. Jaguste, S. Jaiswal, N. Jamadar, and J. Dange, "Sentry Gun," in *International Conference on Innovative and Advanced Technologies in Engineering*, 2018, pp. 74–78.
- [5] M. Kashif, M. Arslan, R. Chakma, F. Banoori, A. Al Mamun, and G. L. Chakma, "Design and Implementation of Image Capture Sentry Gun Robot," in *MATEC Web of Conferences*, 2018, vol. 160, pp. 1–5.
- [6] Y. Naik, H. Kolte, A. Gupta, A. Yashwantrao, and M. D. Nikam, "Portable Sentry Gun," *Int. Res. J. Eng. Technol.*, vol. 06, no. 04, pp. 3130–3132, 2019.
- [7] E. A. Z. Hamidi, M. R. Effendi, and F. F. Asmoro, "Design and Implementation of Model Autonomous Sentry Gun Using Image Processing and Motion Tracking," in *2019 IEEE 5th International Conference on Wireless and Telematics (ICWT)*, 2019, pp. 1–5.
- [8] R. R. Alcala, Z. G. Arceo, J. N. Baterisna, J. O. Morada, J. O. D. Ramirez, and R. E. Tolentino, "Design and Implementation of Body Wearable Device and Oculus Rift Controlled Panning and Aiming Sentry Gun Turret," in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)*, 2020, pp. 871–876.
- [9] I. D. Wijaya, U. Nurhasan, and M. A. Barata, "Implementasi Raspberry Pi untuk Rancang Bangun Sistem Keamanan Pintu Ruang Server dengan Pengenalan Wajah Menggunakan Metode Triangle Face," *J. Inform. Polinema*, vol. 4, no. 1, pp. 9–16, 2017.
- [10] A. A. N. Yudistira, F. Y. Limpraptono, and S. Sotyohadi, "Sistem Kendali Senjata Handgun Otomatis Menggunakan Kamera Berbasis Atmega 2560," *Magнетika*, vol. 07, no. 02, pp. 56–62, 2023.
- [11] S. Karthikeyan, R. A. Raj, M. V. Cruz, L. Chen, J. L. A. Vishal, and V. S. Rohith, "A Systematic Analysis on Raspberry Pi Prototyping: Uses, Challenges, Benefits, and Drawbacks," *IEEE Internet Things J.*, vol. 10, no. 16, pp. 14397–14417, 2023.
- [12] J. Á. Ariza and S. G. Gil, "RaspyLab: A Low-Cost Remote Laboratory to Learn Programming and Physical Computing Through Python and Raspberry Pi," *IEEE Rev. Iberoam. Tecnol. del Aprendiz.*, vol. 17, no. 2, pp. 140–149, 2022.
- [13] H. Ahmad, S. M. Ali, U. A. Sheikh, Z. Murtaza, and M. Rizwan, "Design and manufacturing of a low-cost 2-DOF autonomous sentry gun," in *2016 2nd International Conference on Robotics and Artificial Intelligence (ICRAI)*, 2016.
- [14] F. F. Asmoro, *Rancang bangun model autonomous sentry gun menggunakan image processing dan motion processing*. Bandung: Diploma thesis, UIN Sunan Gunung Djati, 2018.
- [15] A. Raharjo, E. Kuncoro, and I. Azhar, "Rancang Bangun Tracking Arah Tembakan Menggunakan Sensor Posisi Berbasis PID," *Telka*, vol. 7, no. 1, pp. 43–48, 2021.
- [16] C. Simatupang, M. Anwar, and R. Budi, "Telekomunikasih Militer Rancang Bangun Sistem Tracking pada

- Sentry Gun Sebagai Pengganti Pos dengar Berbasis Internet of Things,” *J. Telkommil*, vol. 3, no. Mei, pp. 84–92, 2022.
- [17] M. A. Wijaya, P. Choirina, and V. A. Yoda, “Telekomunikasi Militer Implementasi Palm Vein Sensor Dengan Metode Biometric Pattern Pada Sistem Sentry Gun Berbasis Fitur Local Binary Pattern (LBP),” *J. Telkommil*, vol. 3, no. Mei, pp. 30–40, 2022.
- [18] M. Monika, K. Ganapathy, and M. F. Zia, “Moving Object Detection Using Eigen Background In Comparison with Gaussian Mixture Model to measure F-Score,” in *2024 IEEE 9th International Conference on Engineering Technologies and Applied Sciences (ICETAS)*, 2024, pp. 1–7.
- [19] S. Bariko, A. Klilou, A. Abounada, and A. Arsalane, “Parallel Implementation of Gaussian Mixture Model Background Subtraction on Jetson Nano,” in *2024 World Conference on Complex Systems (WCCS)*, 2024, pp. 1–6.
- [20] J. Xie, Y. Liu, Y. Zhang, and Z. Wang, “Research on Video Moving Object Detection Based on Improved Gaussian Mixture Model,” in *2023 3rd International Symposium on Computer Technology and Information Science (ISCTIS)*, 2023, pp. 863–866.
- [21] X. Lu and C. Xu, “Novel Gaussian mixture model background subtraction method for detecting moving objects X,” in *2018 IEEE International Conference of Safety Produce Informatization (IICSPI)*, 2018, pp. 6–10.
- [22] S. T. Ali, K. Goyal, and J. Singhai, “Moving object detection using self adaptive Gaussian Mixture Model for real time applications,” in *2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE)*, 2018, pp. 153–156.