

---

## DESAIN ROBOT ARM BERBASIS *VISION* SEBAGAI PENGINGAT *SOCIAL DISTANCING* MENGGUNAKAN RASPBERRY PI DAN ROBOT OPERATING SYSTEM

### DESIGN OF *VISION*-BASED ARM ROBOT FOR SOCIAL DISTANCING REMINDER USING RASPBERRY PI AND ROBOT OPERATING SYSTEM

I Wayan Suparno<sup>1</sup>, Abdul Jalil<sup>2</sup>

<sup>1</sup>Teknik Informatika, STMIK Handayani Makassar

<sup>2</sup>Sistem Komputer, STMIK Handayani Makassar

<sup>1</sup>[iwayansuparno@gmail.com](mailto:iwayansuparno@gmail.com), <sup>2</sup>[abdul.jalil@handayani.ac.id](mailto:abdul.jalil@handayani.ac.id)

#### Abstrak

Tujuan penelitian ini adalah mendesain robot *arm* berbasis *vision* yang berfungsi sebagai pengingat jaga jarak (*social distancing*) di ruang publik menggunakan Raspberry Pi dan Robot Operating System 2 (ROS 2). Penelitian ini akan menggunakan kamera yang berfungsi sebagai perangkat input untuk menangkap objek gambar dalam bentuk video, Raspberry Pi yang digunakan untuk mengolah data gambar dan mengontrol gerak robot *arm*, amplifier yang berfungsi untuk mengeluarkan peringatan suara, dan motor servo yang berfungsi untuk menggerakkan sendi robot *arm*. Selanjutnya, penelitian ini akan menggunakan sistem operasi Linux Ubuntu 20.04 LTS yang terinstal pada Raspberry Pi, *library* OpenCV untuk mengolah citra gambar, serta ROS 2 Foxy Fitzroy sebagai perangkat lunak kontrol robot. Adapun metode yang akan kami gunakan untuk mendeteksi jarak antar objek manusia adalah *Binary Image Comparison* (BIC), dimana metode ini bekerja untuk membandingkan nilai citra biner master dan citra biner warna lainnya sehingga dapat diaplikasikan untuk mendeteksi jarak objek. Hasil penelitian ini adalah desain perangkat robot *arm* yang dapat digunakan untuk pengingat jaga jarak di ruang publik berbasis *vision* menggunakan Raspberry Pi dan ROS 2.

**Kata kunci :** Robot Arm, Vision, Raspberry Pi, Robot Operating System 2, Binary Image Comparison

#### Abstract

*This study aims to design a vision-based robot arm for social distancing reminder in the public spaces using Raspberry Pi and Robot Operating System 2 (ROS 2). This study will use a camera to capture image objects in the form of video, a Raspberry Pi to process the images and control the arm robot moving, an amplifier to play the sound warnings, and servo motors to move the joint of the arm robot. Furthermore, this research will use the Linux Ubuntu 20.04 LTS as an operating system installed on the Raspberry Pi, the OpenCV library for image processing tools, and ROS 2 Foxy Fitzroy as robot control software. The method we will use to detect the distance between human objects is Binary Image Comparison (BIC), which works to compare the values of the master binary image and other color binary images so that it can be applied to detect object distances. This study result is the design of the arm robot that can be used to give a keep distance warning in the public spaces based on vision using Raspberry Pi and ROS 2.*

**Keywords:** Arm Robot, Vision, Raspberry Pi, Robot Operating System 2, Binary Image Comparison

## 1. PENDAHULUAN

Covid-19 merupakan wabah penyakit yang penularannya sangat cepat dan dapat berdampak buruk bagi penderita yang terinfeksi oleh penyakit tersebut. Saat ini, penularan penyakit Covid-19 sangat sulit untuk dibendung sehingga pemerintah telah melakukan berbagai hal untuk mencegah penyebarannya, seperti penerapan PPKM, program pemberian vaksin secara gratis untuk masyarakat, serta menggalakkan pola 3M ke masyarakat, yaitu mencuci tangan, memakai masker, dan menjaga jarak (*social distancing*). Tujuan dari penelitian ini adalah membangun sebuah robot *arm* berbasis *vision* yang dapat digunakan untuk memberikan peringatan jaga jarak kepada masyarakat ketika berada di ruang publik.

Penerapan robot dalam menanggulangi penyebaran penyakit Covid-19 telah dilakukan oleh beberapa peneliti. Pincu et al telah membangun sebuah robot dengan nama *Covid-19 Officer Robot* (COR) yang bertugas untuk memberi peringatan kepada pengunjung gedung ketika tidak memakai masker [1], robot tersebut dapat memberikan peringatan berupa suara sehingga ketika pengunjung masuk ke dalam sebuah gedung tanpa masker, pengunjung tersebut wajib menggunakannya kemudian masuk kedalam gedung tersebut. Pemanfaatan *mobile robot* berbasis *vision* untuk mendeteksi suhu tubuh manusia telah diterapkan oleh Senhaji et al [2], pada penelitian tersebut, peneliti telah menggunakan kamera thermal untuk membaca suhu tubuh manusia dan telah diaplikasikan di suatu bandara. Selanjutnya, penerapan *Covid Surveillance Robot* (CS-Robot) untuk memonitoring jarak pengunjung pada sebuah ruangan telah diteliti oleh Sathyamoorthy et al [3], pada penelitian tersebut, peneliti telah menggunakan turtlebot 2 untuk menavigasi robot serta menggunakan aplikasi Yolov3 untuk memproses citra gambar.

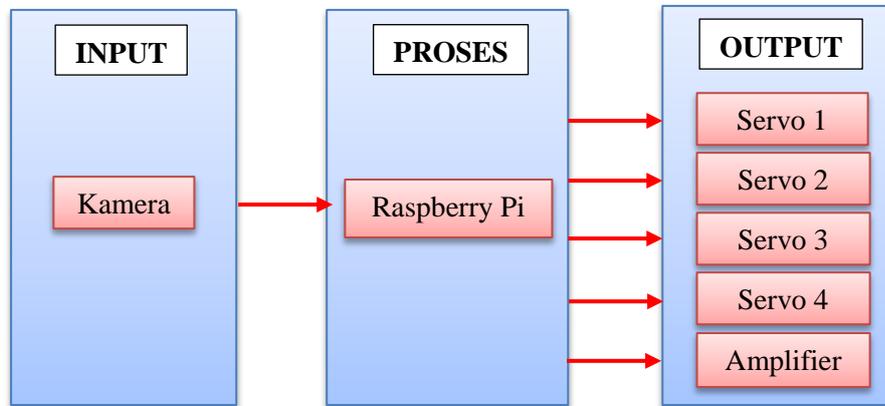
Selain itu, penelitian untuk memonitoring jarak sebagai *social distancing* menggunakan robot telah diaplikasikan oleh beberapa peneliti. Sathyamoorthy et al telah mengembangkan *mobile robot* yang dapat memonitoring *social distancing* menggunakan metode *Predestrian Tracking and Localization* [4], pada metode tersebut jarak objek antar individu dapat diukur berdasarkan estimasi jarak pixel *bounding box* yang dideteksi oleh citra gambar. Pemanfaatan *multi-agent robot* dalam memonitoring *physical distancing* telah diaplikasikan oleh Shah et al menggunakan metode *You Only Look Once* (YOLO) [5], pada penelitian tersebut jarak antar individu dapat diukur berdasarkan jarak *bounding box* dan *Fields of View* (FoV) dari objek gambar yang di deteksi. Selanjutnya Fan et al telah mengembangkan *Quadruped robot* yang dapat digunakan untuk memonitoring *social distancing* pada lingkungan perkotaan menggunakan metode *crowd-aware routing algorithm* [6], para peneliti telah menggunakan kamera dan 3D lidar untuk mengambil data objek serta menggunakan *bounding box prediction* untuk mendeteksi jarak antar individu manusia.

Pada penelitian ini, kami akan mendesain sebuah robot *arm* berbasis *vision* yang dapat bernavigasi dan mendeteksi jarak objek antar manusia hingga 360 derajat menggunakan metode *Binary Image Comparison* (BIC) [8]. Desain robot *arm* tersebut akan memberikan peringatan kepada masyarakat dalam bentuk suara ketika terdapat masyarakat yang tidak menerapkan pola jaga jarak di ruang publik sesuai dengan anjuran pemerintah. Metode BIC dalam penelitian ini akan bekerja dengan cara membandingkan titik-titik warna yang terdapat pada suatu ruang publik dengan nilai citra biner master sebagai tolak ukur dalam mendeteksi jarak objek antar manusia. Kami akan menjelaskan lebih rinci cara kerja dari metode BIC ini pada bagian metodologi penelitian ini.

## 2. METODOLOGI

Inti dari penerapan robot berbasis *vision* adalah menerapkan kamera sebagai perangkat input robot sehingga dapat diproses menggunakan teknik pengolahan citra. Robot *vision* merupakan robot yang dapat bernavigasi berdasarkan input sensor kamera dalam bentuk citra untuk mendeteksi

benda atau objek [7]. Pada desain robot *arm* yang kami bangun, terdapat tiga bagian inti arsitektur sistem, yaitu input, proses, dan output. Gambar 1 dibawah ini memperlihatkan sistem arsitektur perancangan desain robot *arm* yang akan kami bangun pada penelitian ini.



Gambar 1. Arsitektur sistem rancang bangun desain robot *arm* berbasis *vision*

Berdasarkan arsitektur perancangan sistem yang ditampilkan pada gambar 1 dapat dilihat bahwa input dari robot *vision* yang dibangun adalah menggunakan kamera. Kamera merupakan suatu perangkat input pada robot yang dapat mendeteksi suatu objek berdasarkan masukan gambar atau video yang ditangkap oleh kamera. Selanjutnya pada bagian proses, kami menggunakan Raspberry Pi model 4 untuk memproses citra gambar serta untuk mengontrol perangkat output dari robot yaitu motor servo dan amplifier. Raspberry Pi merupakan sebuah mikrokomputer yang berukuran kecil serta dapat bekerja untuk mengolah data layaknya sebuah komputer [9]. Pada Raspberry Pi terdapat seperangkat GPIO pin yang berfungsi sebagai pin penghubung antara Raspberry Pi dengan perangkat input dan output. Pada bagian output, terdapat empat buah motor servo yang berfungsi untuk menggerakkan persendian robot *arm* agar dapat bernavigasi hingga putaran 360 derajat. Motor servo merupakan sebuah perangkat output aktuator yang bergerak berdasarkan perintah sinyal pulsa yang dikirim dari perangkat kontrol, dimana pada motor servo terdapat serangkaian gear yang memiliki torsi sehingga dapat mengangkat beban suatu benda sesuai dengan kemampuan motor servo tersebut [10]. Adapun motor servo yang kami gunakan pada penelitian ini adalah motor servo yang dapat bergerak dari titik 0 hingga 360 derajat. Selanjutnya pada bagian output terdapat amplifier yang berfungsi untuk memberikan peringatan suara ketika robot *arm* mendeteksi terdapat masyarakat yang tidak menerapkan pola jaga jarak pada ruang publik.

Adapun perangkat lunak yang akan kami gunakan pada penelitian ini adalah Sistem Operasi Linux Ubuntu 20.04 LTS sebagai perangkat lunak sistem yang terinstal pada Raspberry Pi, kemudian menggunakan library OpenCV untuk mengolah citra gambar. Selanjutnya, kami juga menggunakan perangkat lunak robot yaitu Robot Operating System 2 (ROS 2) yang berfungsi untuk mengontrol perangkat input dan output robot yaitu kamera, motor servo, dan amplifier. Robot Operating System (ROS) merupakan perangkat lunak robot yang didalamnya terdapat berbagai *library*, *package* dan yang digunakan untuk mengendalikan perangkat keras robot. Pada saat ini para pengembang ROS telah mengembangkan dua jenis ROS, yaitu ROS 1 dan ROS 2. Perbedaan mendasar antara ROS 1 dan ROS 2 adalah pada protokol sistem komunikasi data yang digunakan untuk mengirim data pesan antara node publisher dan node subscriber, dimana ROS 1 menggunakan protokol TCP/IP untuk mentransfer data pesan, sedangkan ROS 2 menggunakan *Data Distribution Service* (DDS) [11]. ROS telah dibangun untuk dapat saling menghubungkan antara publisher dan subscriber, dimana publisher berfungsi untuk mengirim data pesan sedangkan

subscriber berfungsi untuk menerima data pesan. Publisher dan subscriber dapat saling terhubung dan bertukar data pesan melalui sebuah topic [12]. Adapun jenis distro ROS 2 yang kami gunakan pada penelitian ini adalah ROS 2 Foxy Fitzroy.

Pada penelitian ini, kami akan menggunakan metode BIC untuk mendeteksi jarak objek manusia pada suatu ruang publik. Teknik deteksi jarak yang digunakan adalah dengan melekatkan titik-titik warna pada dinding suatu ruang publik yang berjarak antara 50 cm. Pada metode BIC, pada saat proses deteksi citra biner mendeteksi terdapat dua titik warna memiliki nilai citra biner yang sama, maka robot mendeteksi terdapat objek manusia yang tidak menerapkan pola jaga jarak pada suatu ruang publik. Berikut adalah algoritma BIC yang akan kami terapkan pada penelitian ini.

---

#### Algoritma pada *Binary-Image Comparison*

---

**Langkah 1.** Set nilai HSV titik warna 1 (*Low* dan *High*);  
Set nilai HSV titik warna n (*Low* dan *High*);  
Set nilai HSV master (*Low* dan *High*);

**Langkah 2.** Subscribe data pesan gambar dalam bentuk video;  
*Threshold* nilai HSV titik warna 1 terhadap gambar;  
*Threshold* nilai HSV titik warna n terhadap gambar;  
*Threshold* nilai HSV master;

**Langkah 3.** **If** *threshold\_warna 1* & *threshold\_warna n* = *threshold\_master* **then**;  
Robot mendeteksi terdapat manusia yang tidak menerapkan jaga jarak;  
Amplifier aktif untuk memberikan teguran berupa suara;

**Else**  
Robot tidak mendeteksi;

**Kembali ke langkah 2;**

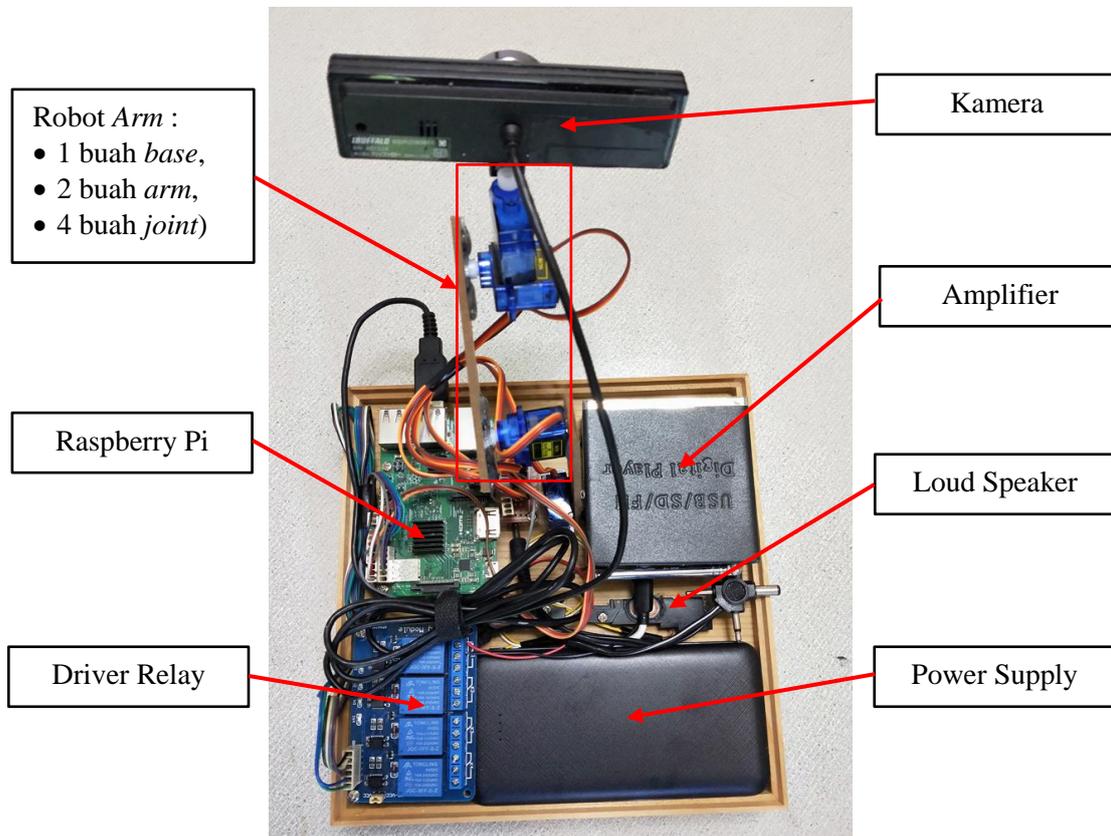
---

Pada langkah pertama BIC dapat dilihat bahwa BIC akan menentukan nilai HSV titik warna 1, titik warna n, dan master. Selanjutnya pada langkah kedua, algoritma BIC akan membaca data gambar yang dikirim dari kamera dalam bentuk video, kemudian meng-threshold nilai HSV yang telah ditetapkan pada setiap titik warna gambar tersebut. Pada langkah ketiga, metode BIC akan membandingkan antara nilai citra biner titik warna 1 dan nilai citra biner titik warna n dengan nilai biner master, jika nilai biner antara titik warna 1 dan titik warna n sama dengan nilai biner master, maka BIC mendeteksi adanya manusia yang saling berdekatan pada ruang publik, selanjutnya memerintahkan amplifier untuk aktif sehingga dapat memberikan teguran berupa suara kepada masyarakat yang tidak menerapkan pola jaga jarak, namun jika nilai threshold antara titik warna 1 dan n berbeda dengan citra biner master, maka robot tidak mendeteksi adanya manusia yang saling berdekatan pada ruang publik. Algoritma BIC ini akan terus berjalan hingga proses node ROS 2 dinonaktifkan.

### 3. PEMBAHASAN

Hasil dari penelitian ini adalah desain perangkat keras dan perangkat lunak robot *arm* berbasis *vision* yang digunakan untuk memberi peringatan jaga jarak pada ruang publik menggunakan Raspberry Pi dan ROS 2. Pada desain robot *arm* yang dibangun, terdapat beberapa bagian robot yang dibangun untuk menggerakkan kamera, yaitu 1 *base*, 2 *arm*, dan 4 *joint*. Adapun fungsi dari *base* pada robot *arm* adalah sebagai dasar robot yang dapat berputar hingga 360 derajat, selanjutnya fungsi *arm* atau lengan adalah untuk menghubungkan setiap *joint* sehingga robot *arm* dapat bernavigasi untuk bergerak naik atau turun serta menggerakkan bagian badan kamera. Sedangkan fungsi dari *joint* adalah untuk menggerakkan setiap sendi dari robot *arm* yang dibangun. Pada penelitian ini, kami menggunakan 4 buah motor servo untuk menggerakkan *base* dan *joint*

dari robot *arm*, kemudian menggunakan satu buah kamera untuk mendeteksi gambar objek dalam bentuk *video streaming*, satu buah Raspberry Pi untuk mengolah citra gambar dan mengontrol gerak robot *arm*, satu buah driver relay untuk mengontrol nyala amplifier, satu buah amplifier, dua buah loudspeaker, dan satu buah power supply. Gambar 2 berikut ini memperlihatkan hasil desain robot *arm* yang telah dibangun dalam mendeteksi gerak objek manusia pada suatu ruang publik.



Gambar 2. Hasil desain perangkat keras robot *arm* berbasis *vision*.

Pada robot *arm* yang kami bangun, seluruh perangkat input dan output terhubung ke port dan GPIO pin yang terdapat pada Raspberry Pi. Tabel 1 berikut ini memperlihatkan konfigurasi port dan GPIO pin yang digunakan untuk membaca input data kamera dan mengontrol output dari robot.

Tabel 1. Konfigurasi GPIO pin dan port yang digunakan pada Raspberry Pi.

Perangkat	GPIO pin / Port	Fungsi
Kamera	Port USB	Sebagai perangkat input yang digunakan untuk menangkap objek gambar dalam bentuk <i>video streaming</i> .
Servo 1	GPIO pin 2	Berfungsi sebagai <i>joint</i> 1.
Servo 2	GPIO pin 3	Berfungsi sebagai <i>joint</i> 2.
Servo 3	GPIO pin 4	Berfungsi sebagai <i>joint</i> 3.
Servo 4	GPIO pin 17	Berfungsi sebagai <i>joint</i> 4.
Driver relay	GPIO pin 27	Berfungsi untuk mengaktifkan dan menonaktifkan amplifier
Power Supply	Power	Berfungsi sebagai sumber tegangan dengan kapasitas daya sebesar 5 Volt DC.

Berdasarkan informasi dari tabel 1 dapat dilihat bahwa input dari perangkat kamera terhubung ke port USB Raspberry Pi, dimana fungsi dari kamera ini adalah untuk menangkap gerak objek manusia dalam bentuk *video streaming*. Adapun perangkat output motor servo 1 terhubung ke GPIO pin 2 yang berfungsi untuk menggerakkan *joint 1* pada robot *arm*, motor servo 2 terhubung ke GPIO pin 3 yang berfungsi untuk menggerakkan *joint 2*, motor servo 3 terhubung ke GPIO pin 4 yang berfungsi untuk menggerakkan *joint 3*, dan motor servo 4 terhubung ke GPIO pin 17 yang berfungsi untuk menggerakkan *joint 4*. Selanjutnya pada perangkat output driver relay terhubung ke GPIO pin 27 yang berfungsi untuk mengaktifkan dan menonaktifkan amplifier agar dapat memberikan peringatan berupa suara pada saat robot arm mendeteksi manusia yang tidak menerapkan pola *social distancing*. Adapun sumber tegangan yang digunakan untuk mengaktifkan seluruh perangkat keras adalah menggunakan catu daya baterai dengan kapasitas output sebesar 5 volt DC.

Pada pengujian sistem, kami telah melakukan ujicoba keberhasilan konektivitas perangkat input kamera dengan USB port dan konektivitas GPIO pin dalam mengontrol perangkat keras output motor servo robot *arm* dan amplifier. Tabel 2 berikut ini memperlihatkan hasil ujicoba konektivitas perangkat input kamera dengan Raspberry Pi dan GPIO pin dengan perangkat output.

Tabel 2. Hasil ujicoba perangkat keras robot *arm*.

Perangkat	GPIO pin / Port	Input / Output	Hasil
Kamera	Port USB	Video Streaming	Kamera dapat memberi masukan berupa video streaming ke Raspberry Pi melalui port USB pada saat perangkat lunak ROS 2 membaca input perangkat kamera.
Servo 1	GPIO pin 2	Pulsa 0 - 180	Motor servo 1 dapat bergerak dari titik 0 derajat hingga 180 derajat untuk menggerakkan joint 1 pada robot <i>arm</i> .
Servo 2	GPIO pin 3	Pulsa 0 - 180	Motor servo 2 dapat bergerak dari titik 0 derajat hingga 180 derajat untuk menggerakkan joint 2 pada robot <i>arm</i> .
Servo 3	GPIO pin 4	Pulsa 0 - 180	Motor servo 3 dapat bergerak dari titik 0 derajat hingga 180 derajat untuk menggerakkan joint 3 pada robot <i>arm</i> .
Servo 4	GPIO pin 17	Pulsa 0 - 180	Motor servo 4 dapat bergerak dari titik 0 derajat hingga 180 derajat untuk menggerakkan joint 4 pada robot <i>arm</i> .
Driver relay	GPIO pin 27	1	GPIO pin dapat memerintahkan driver relay untuk mengaktifkan perangkat amplifier agar dapat mengeluarkan output berupa suara.
		0	GPIO pin dapat memerintahkan driver relay untuk menonaktifkan perangkat amplifier.

Berdasarkan hasil ujicoba perangkat keras yang ditampilkan pada tabel 2 dapat dilihat bahwa perangkat lunak ROS 2 dapat membaca input kamera berupa video streaming pada saat digunakan untuk mendeteksi gerak objek manusia melalui port USB Raspberry Pi. Selanjutnya pada pengujian motor servo dapat dilihat bahwa GPIO pin Raspberry Pi dapat mengontrol gerak motor servo 1 hingga motor servo 4 dari titik 0 derajat hingga ke titik 180 derajat, hal ini menunjukkan bahwa GPIO pin 2, 3, 4, dan 17 dapat bekerja dengan baik untuk mengontrol arah gerak robot *arm* pada saat menggerakkan kamera untuk mendeteksi jarak objek manusia pada suatu ruang publik. Pada pengujian driver relay dapat dilihat bahwa ketika GPIO pin 27 diberi logika 1 maka driver relay akan aktif *on* untuk menyalakan amplifier agar dapat memberikan peringatan berupa suara agar manusia dapat menerapkan pola *social distancing* pada suatu ruang publik. Selanjutnya, pada saat

GPIO pin 27 diberi logika 0, maka driver relay akan menonaktifkan amplifier agar tidak memberi peringatan berupa suara. Berdasarkan hasil ujicoba perangkat yang telah kami lakukan pada desain robot *arm* ini dapat disimpulkan bahwa seluruh perangkat input dan output dapat bekerja dengan baik pada saat perangkat lunak ROS 2 dijalankan untuk mengontrol robot *arm*. Selanjutnya hasil dari desain robot *arm* ini akan kami kembangkan pada penelitian selanjutnya untuk dapat diterapkan dimasyarakat sehingga dapat digunakan untuk memberi peringatan kepada masyarakat yang tidak menjalankan peraturan *social distancing* ketika berada diruang publik.

#### 4. KESIMPULAN

Desain perangkat keras dan perangkat lunak untuk membangun robot *arm* yang dapat digunakan untuk memberi peringatan jaga jarak berbasis *vision* telah didesain pada penelitian ini. Kami menggunakan Raspberry Pi model 4 sebagai pusat pengolahan data robot dimana fungsi dari Raspberry Pi adalah perangkat mikrokomputer yang digunakan untuk mengolah citra gambar objek yang dideteksi oleh kamera dan mengontrol gerak robot *arm*. Selanjutnya pada desain robot *arm* ini, kami akan menggunakan perangkat lunak Robot Operating System 2 (ROS 2) dalam mengontrol perangkat input kamera dan mengontrol perangkat output robot berupa motor servo dan amplifier. Dalam mendeteksi jarak objek, kami telah mendesain perangkat lunak algoritma dengan menggunakan metode *Binary Image Comparison* (BIC), dimana metode ini digunakan untuk membandingkan jarak antara titik warna satu dan titik warna lainnya dengan citra biner master sehingga ketika nilai citra biner antara setiap titik warna dan master sama maka BIC mendeteksi terdapat objek manusia yang saling berdekatan pada ruang publik. Pada penelitian selanjutnya, kami akan menerapkan desain robot *arm* ini dalam mendeteksi jarak antar manusia pada ruang publik menggunakan sistem komunikasi data antar node ROS 2.

#### DAFTAR PUSTAKA

- [1] E.L. Pincu, A. David, V.S. Fleischmann, Y. Edan, & T.O. Gilad, "Comply with Me: Using Design Manipulations to Affect Human-Robot Interaction in a COVID-19 Officer Robot Use Case," *Multimodal Technol. Interact.* 2021. DOI: <https://doi.org/10.3390/mti5110071>.
- [2] S. Senhaji, S. Faquir, & M.A. Jamil, "Towards Robotics and Artificial Intelligence for the Prevention of Covid 19 Pandemic," *E3S Web of Conferences*, 2021. DOI: <https://doi.org/10.1051/e3sconf/202122901035>.
- [3] A.J. Sathyamoorthy, U. Patel, M. Paul, Y. Savle, & D. Manocha, "COVID Surveillance Robot: Monitoring Social Distancing Constraints in Indoor Scenarios," *PPLoS ONE* 16(12): e0259713, 2021. DOI: <https://doi.org/10.1371/journal.pone.0259713>.
- [4] A.J. Sathyamoorthy, U. Patel, M. Paul, Y.A. Savle, M. Paul, & D. Manocha, "COVID-Robot: Monitoring Social Distancing Constraints in Crowded Scenarios," *arXiv:2008.06585v2 [cs.RO]*, 2020.
- [5] S.H.H. Shah, O.H. Steignes, E.G. Gustafsson, & I.A. Hameed, "Multi-Agent Robot System to Monitor and Enforce Physical Distancing Constraints in Large Areas to Combat COVID-19 and Future Pandemics," *Appl. Sci.* 2021. DOI: <https://doi.org/10.3390/app11167200>.
- [6] T. Fan, Z. Chen, X. Zhao, J. Liang, C. Shen, D. Manocha, J. Pan, & W. Zhang, "Autonomous Social Distancing in Urban Environments using a Quadruped Robot," *arXiv:2008.08889v1 [cs.RO]*, 2020.
- [7] W. Budiharto & D. Purwanto, "Robot *Vision*: Teknik Membangun Robot Cerdas Masa Depan," Penerbit ANDI, 2015.

- [8] A. Jalil & M. Matalangi, "Object Motion Detection in Home Security System Using The Binary-Image Comparison Method Based On Robot Operating System 2 and Raspberry Pi," *ILKOM Jurnal Ilmiah*, Vol.13, No.1, 2021. DOI: <https://doi.org/10.33096/ilkom.v13i1.686.1-8>.
- [9] I.D. Wijaya, U. Nurhasan, dan M.A. Barata, "Implementasi Raspberry Pi Untuk Rancang Bangun Sistem Keamanan Pintu Ruang Server Dengan Pengenalan Wajah Menggunakan Metode Triangle Face," *Jurnal Informatika Polinema*, Vol.4, No.1, 2017.
- [10] U. Latifa, dan J.S. Saputro, "Perancangan Robot Arm Gripper Berbasis Arduino Uno Menggunakan Antarmuka Labview," *Barometer*, Vol.3, No.2, pp. 138-141, 2018.
- [11] Y. Maruyama, S. Kato, & T. Azumi, "Exploring the Performance of ROS2," *EMSOFT 16*, 2016. DOI: <http://dx.doi.org/10.1145/2968478.2968502>.
- [12] A. Jalil, "Robot Operating System (ROS) dan Gazebo Sebagai Media Pembelajaran Robot Interaktif," *ILKOM Jurnal Ilmiah*. Vol.10, No.3, 2018.