

# Alat Bantu Pendeteksi Objek Untuk Tuna Netra Berbasis AI Mobilenet Pada Raspberry Pi 3B

Sofia Ariyani, Aji Brahma Nugroho, Ahmad Syarif Toyyib Mubarak

Teknik Elektro, Fakultas Teknik, Universitas Muhammadiyah Jember  
Jl. Karimata No.49 Jember 68121 Indonesia  
E-mail: syariftoyyib78@gmail.com

Naskah Masuk: 23 April 2021; Diterima: 16 Maret 2022; Terbit: 25 Maret 2022

---

## ABSTRAK

---

**Abstrak** – Saat ini teknologi alat bantu untuk para penyandang disabilitas telah berkembang pesat. Salah satunya adalah alat bantu untuk tunanetra dengan memanfaatkan kamera yang memiliki fungsi sebagai mata. Pada penelitian ini, kamera tersebut diintegrasikan dengan Raspberry Pi sebagai mikrokontroler dan juga sensor jarak HC SR04. Adanya sensor HC SR04 pada alat ini memungkinkan sistem tidak hanya dapat mendeteksi objek namun juga mengetahui jarak objek tersebut. Agar sistem dapat mendeteksi suatu objek, dibutuhkan suatu data sistem *image processing*. Model yang digunakan pada penelitian ini adalah model pra terlatih Mobilenet. Hasil penelitian ini menunjukkan sistem dapat mendeteksi objek sebanyak 80 objek. Pengujian sistem dilakukan dengan menggunakan 5 objek dengan 50 jenis varian yaitu objek buku, gelas, botol, telepon seluler, dan orang. Pada setiap jenis varian objek dilakukan pengujian sebanyak 10 kali dengan 3 variasi jarak yang berbeda-beda yaitu 100cm, jarak 150cm, dan jarak 200cm sebagai faktor penentu keakurasian. Berdasarkan pengujian pendeteksian objek yang sudah dilakukan diperoleh hasil bahwa rata-rata tingkat keberhasilan sebesar 40%. Hasil pengujian terhadap jarak didapatkan rata-rata tingkat keberhasilan sebesar 53% pada jarak uji 100 cm, 39% pada jarak uji 150 cm, dan 27% pada jarak uji 200 cm.

**Kata kunci:** Kamera, Mobilenet, Raspberry Pi, Sensor HC SR04

---

## ABSTRACT

---

**Abstract** - Currently assistive technology for people with disabilities has developed rapidly. One of them is a tool for the visually impaired by using a camera that functions as an eye. In this study, the camera was integrated with the Raspberry Pi as a microcontroller and also the HC SR04 proximity sensor. The presence of the HC SR04 sensor on this tool allows the system not only to detect objects but also to know the distance of the object. In order for the system to detect an object, a data image processing system is needed. The model used in this study is a pre-trained Mobilenet model. The results of this study indicate the system can detect as many as 80 objects. System testing is done by using 5 objects with 50 types of variants, namely book, glass, bottle, cell phone, and people. Each type of object variant was tested 10 times with 3 different distance variations, namely 100cm, 150cm distance, and 200cm distance as a determining factor for accuracy. Based on the object detection tests that have been carried out, the results show that the average success rate is 40%. In this study, the factor tested was the distance factor, which in the test obtained an average success rate of 53% at a test distance of 100 cm, 39% at a test distance of 150 cm, and 27% at a test distance of 200 cm.

**Keywords:** Camera, Mobilenet, Raspberry Pi, HC SR04 sensor

Copyright © 2022 Universitas Muhammadiyah Jember.

---

## 1. PENDAHULUAN

Saat ini para penyandang tuna netra telah memiliki beberapa alat bantu untuk melakukan mobilitas atau aktivitas. Salah satu alat bantu tersebut adalah tongkat khusus tuna netra yang sangat efisien ketika digunakan di luar rumah. Akan tetapi tongkat tersebut akan sulit digunakan bila di dalam ruangan karena dapat memecah – belah atau merusak barang yang ada diruangan tersebut. Selain itu, tongkat tersebut hanya memberikan tanda bahwa terdapat halangan untuk dihindari dan belum dapat membedakan benda yang ada didepan. Untuk membedakan benda didepan, mereka perlu bertanya kepada orang lain yang ada didekatnya menyentuh benda tersebut terlebih dahulu. Sama halnya untuk mengetahui apakah ada orang di depannya ataupun menghindari jalan raya saat mereka berjalan kaki. Adanya perkembangan teknologi yang semakin

cepat saat ini berdampak terhadap banyak terciptanya inovasi alat yang akan sangat membantu bagi tuna netra di Indonesia.

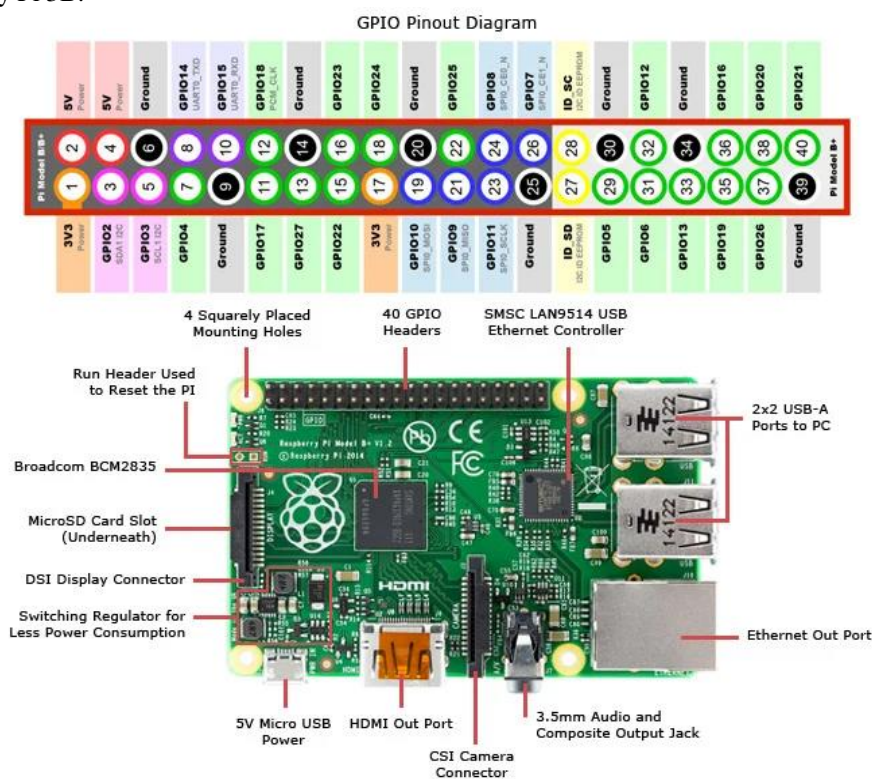
Seiring dengan dengan perkembangan teknologi yang semakin pesat, teknologi untuk alat bantu para disabilitas juga ikut berkembang pesat di seluruh dunia. Salah satunya adalah kamera yang memiliki fungsi seperti mata. Kamera ini dapat menangkap atau merekam suatu objek. Teknologi lain yang telah tercipta yaitu PC atau komputer kecil yang bisa digunakan sebagai mikrokontroler yang disebut dengan Mini PC yang bermodel Raspberry Pi. Integrasi antara kamera dan mini PC yang bermodel Raspberry Pi dapat dimanfaatkan untuk sebuah *video processing* sebagai pendeteksi suatu objek.

Tujuan dari penelitian ini adalah Mampu mengimplementasikan model pra-terlatih Mobilenet untuk mendeteksi objek dan juga mampu mengestimasi jarak objek menggunakan Sensor HC-SR04 dan menjadikannya sebagai output suara. Penelitian ini dapat dijadikan suatu refrensi untuk membangun dan merancang suata alat pendeteksi objek dan estimasi jarak menggunakan Raspberry Pi 3B dan Membantu para disabilitas tuna netra dalam menjalankan kegiatan sehari-hari dengan mudah.

## 2. KAJIAN PUSTAKA

### 2.1. Raspberry Pi

Raspberry Pi merupakan *Single Board Computer* (SBC) yang dikemas dalam sebuah *printed circuit board* (PCB) yang didalamnya terdapat microprocessor, memori serta antarmuka I/O. Raspberry Pi adalah modul *micro computer* yang mempunyai I/O digital port seperti pada papan microcontroller [1]. Sistem operasi pada Raspberry pi disebut dengan Rasbian yang merupakan bagian dari Linux. Raspberry yang digunakan pada penelitian adalah Raspberry model terbaru yaitu Raspberry Pi 3B.



Gambar 1. Raspberry Pi 3B

### 2.2. Kamera Web

Kamera web merupakan *video digital camera* yang dapat menampilkan video dalam keadaan saat ini melalui manasaja dengan penambahan aplikasi dan pemrograman. Ukuran kamera ini minimalis dapat dapat dihubungkan ke PC dengan jaringan Wifi ataupun port USB[2]. Kamera ini memiliki kualitas yang baik karena memiliki Resolusi dan Megapixel yang tinggi. Pada penelitian ini, kamera digunakan sebagai sensor utama yang mendeteksi atau menangkap citra objek didepan tuna netra. Hasil penangkapan objek selanjutnya melalui port USB akan dikirim ke Raspberry Pi. Pada penelitian ini, peletakan kamera web di jadikan sebuah kacamata dengan tujuan agar desain alat menjadi lebih

efisien dan simpel, karena nantinya alat ini akan di bawa kemana-mana, jadi pengguna akan lebih di mudahkan dengan desain kaca mata dengan kamera web ini, pada penelitian ini kamera web yang akan di pakai adalah Kamera web MTECH WB100.



Gambar 2. Kamera Web MTECH WB100

### 2.3. Sensor Ultrasonic HC SR04

Sensor ultrasonik merupakan sensor yang dapat mengubah besaran bunyi menjadi besaran listrik atau sebaliknya. Sensor ini bekerja berdasarkan prinsip dari pantulan suatu gelombang suara sehingga dapat dipakai untuk menafsirkan jarak suatu benda dengan frekuensi tertentu. Disebut sebagai sensor ultrasonik karena sensor ini menggunakan gelombang ultrasonik (bunyi) [3].



Gambar 3. Sensor Ultrasonic HC SR04

### 2.4. Earphone

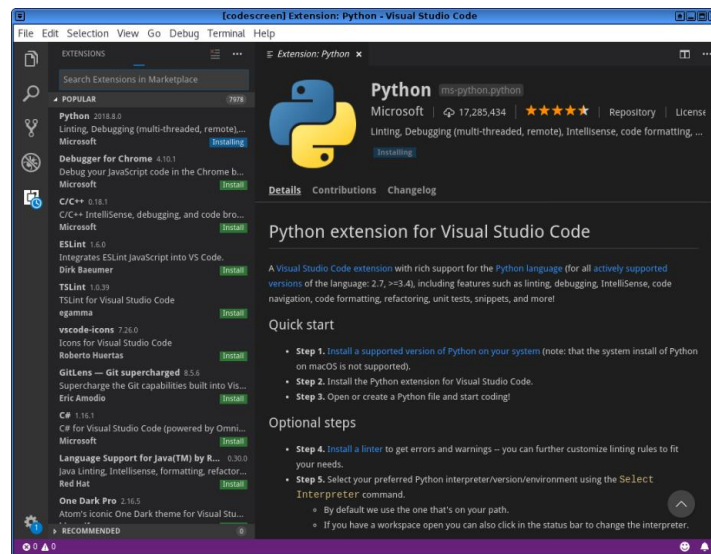
Pada *blind assistant*, *earphone* berfungsi sebagai media output (luaran) suara yang berukuran kecil yang dipasang pada telinga manusia dan dihubungkan ke mini PC melalui jek mini stereo 3.5mm. Alat ini merupakan komponen yang digunakan untuk mengeluarkan suara dari pengolahan data oleh Mini PC. *Earphone* memiliki ukuran kecil sehingga mudah untuk dibawa kemana-mana dan disimpan. *Earphone* seringkali digunakan untuk perangkat *mobile*. *Earphone* akan mengeluarkan suara hasil dari pengolahan algoritma oleh raspberry pi 3 yang akan membantu tuna netra untuk mengetahui objek disekitar. Bentuk suara yang dikeluarkan earphone akan sesuai dengan objek yang terdeteksi. *Earphone* yang akan di gunakan pada penelitian ini adalah earphone JBL.



Gambar 4. Earphone JBL

### 2.5. Bahasa Pemrograman Python

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode [4]. Python diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif. Python juga didukung oleh komunitas yang besar [5].



Gambar 5. VS code python IDE

## 2.6. OpenCV (Open Source Computer Vision Library)

*Open Source Computer Vision Library (Open CV)* merupakan sebuah pustaka perangkat lunak yang ditujukan untuk pengolahan citra dinamis secara real-time. Open CV ini dibuat oleh Intel dan sekarang didukung oleh Willow Garage dan Itseez [6]. Secara resmi, OpenCV pertama kali diluncurkan oleh Inter Research pada tahun 1999. Program ini merupakan pustaka lintas platform dan berada dalam naungan sumber terbuka dari lisensi BSD serta bersifat bebas. Program ini didedikasikan sebagian besar untuk pengolahan citra secara real-time. Jika pustaka ini menemukan pustaka Integrated Performance Primitives dari intel dalam sistem komputer, maka program ini akan menggunakan rutin ini untuk mempercepat proses kerja program ini secara otomatis.

## 2.7. Tensorflow Lite

*Tensorflow Lite* adalah seperangkat alat untuk membantu developer menjalankan model *TensorFlow* di perangkat seluler dan IoT [7]. Ini memungkinkan inferensi pembelajaran mesin di perangkat dengan latensi rendah dan ukuran biner kecil, dan ini sangat cocok apabila diterapkan di Raspberry pi 3 dimana perangkat ini tidak cukup memiliki kapasitas penyimpanan dan spesifikasi memory RAM yang kecil

## 2.8. Library eSpeak

*Library eSpeak* adalah *software open source* penyintesis suara yang ringkas untuk bahasa Inggris dan bahasa lain dan dapat di gunakan menggunakan sistem operasi Linux dan Windows. eSpeak menggunakan metode "format synthesis". Ini memungkinkan banyak bahasa disediakan dalam ukuran kecil dengan pengucapan yang jelas dan dapat digunakan dengan kecepatan tinggi, tetapi tidak sealami atau sehalus Synthesizers (Penyintesis) yang lebih besar yang didasarkan pada rekaman ucapan manusia.

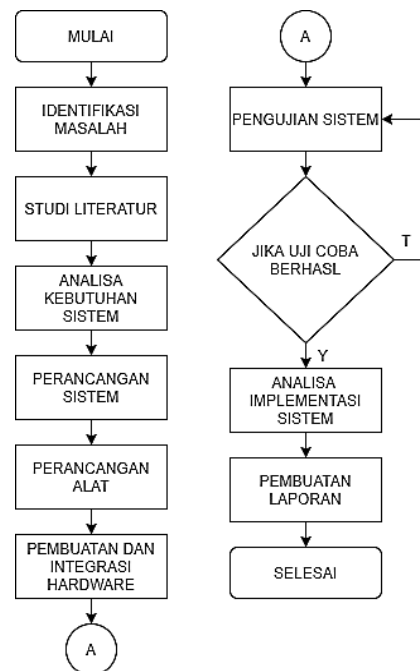
## 2.9. MobileNet model

Implementasi model *deep learning* untuk deteksi objek telah banyak digunakan di berbagai produk riset hingga produk komersil seperti self driving car, cctv cerdas, dan lain-lain. Namun sangat disayangkan, penggunaan deep learning untuk object detection ini membutuhkan perangkat yang cukup mahal seperti GPU atau spesifikasi PC yang cukup tinggi. Mobilenet model adalah salah satu arsitektur convolutional neural network (CNN) yang dapat digunakan untuk mengatasi kebutuhan akan computing resource berlebih [8]. Seperti namanya, Mobile, para peneliti dari Google membuat arsitektur CNN yang dapat digunakan untuk perangkat mobile.

Mobilenet model adalah salah satu arsitektur convolutional neural network (CNN) yang dapat digunakan untuk mengatasi kebutuhan akan computing resource berlebih. Seperti namanya, Mobile, para peneliti dari Google membuat arsitektur CNN yang dapat digunakan untuk perangkat mobile [9].

### 3. METODE PENELITIAN

Gambar 6 di bawah ini merupakan diagram alur penelitian yang berisi langkah-langkah sistematis dalam penelitian ini.



Gambar 6. Diagram Alur Penelitian

#### 3.1 Analisa Kebutuhan Sistem

Tahapan ini bertujuan untuk mendapatkan informasi kebutuhan-kebutuhan sistem yang akan digunakan dalam penelitian. Ketersediaan akan kebutuhan sistem ini akan membuat proses perancangan serta pembuatan sistem menjadi lebih mudah dan efisien.

Adapun peralatan yang digunakan pada penelitian ini antara lain:

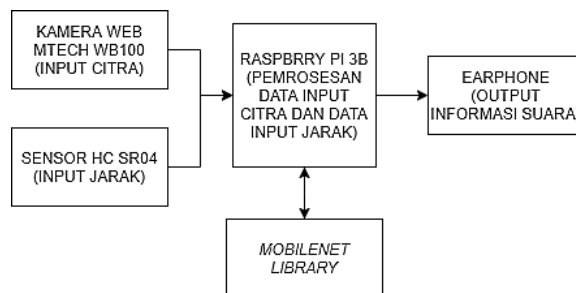
1. Kamera WEB M-TECH 100
2. Mini PC Raspberry Pi 3B
3. Earphone
4. Sensor HC SR04
5. Baterai Lithium 3.7V, 3800mAh
6. Sensor HC SR04
7. Open CV
8. Tensorflow Lite
9. Library eSpeak
10. Mobilenet Model

#### 3.2 Perencanaan Sistem

Pada perencanaan sistem dilakukan perancangan secara sistem untuk alat yang akan dibuat. Pada tahap ini juga menjelaskan pemodelan terhadap sistem yang diperlukan sebagai dasar pembuatan alat dan menerangkan cara integrasi alat dengan sistem yang akan diterapkan.

##### 3.3.1 Blog Diagram Sistem

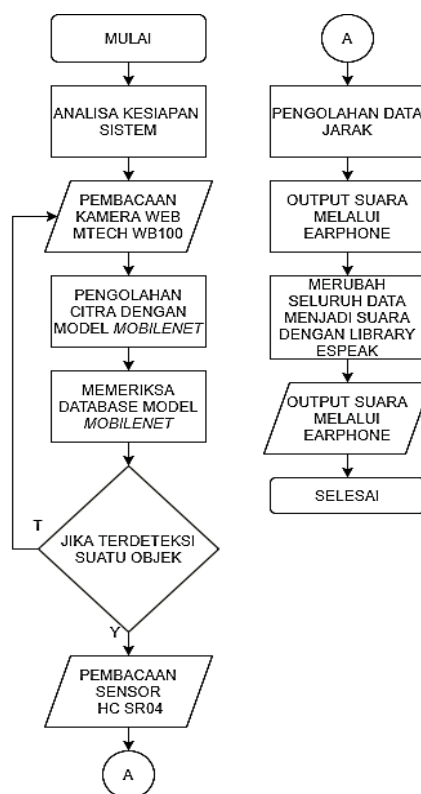
Diagram blok sistem yang dirancang dapat dilihat pada Gambar 7. Sistem dibuat dengan menggunakan sensor HC-SR04 digunakan untuk menghitung estimasi jarak dan 1 kamera sebagai input citra. Selanjutnya data dari kamera dan sensor dikirim ke mini PC dengan bantuan input sensor untuk mengetahui estimasi jarak objek dan selanjutnya akan di proses oleh algoritma pemrograman CNN yang mampu mendeteksi dan mengklasifikasi citra. Setelah berhasil mengklasifikasi, mini PC mengirim data berupa suara ke earphone agar dapat terdengar oleh tuna netra.



Gambar 7. Diagram Blok Sistem

**3.3.2 Flowchart Sistem**

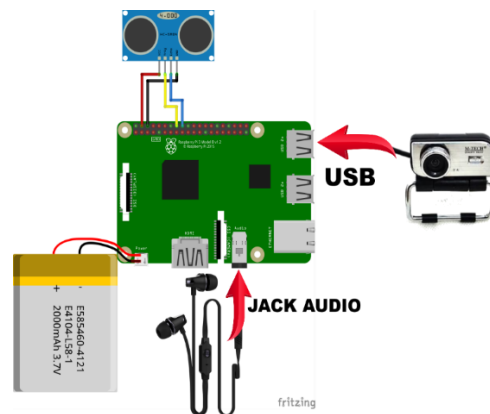
Pada flowchart sistem ini terdapat langkah-langkah untuk mengidentifikasi masalah jika terjadi kendala pada perancangan sistem. Pada gambar 8 terlihat bahwa sistem kerja alat ini dimulai dengan menganalisa kesiapan sistem. Ketika sistem sudah siap, sensor web kamera akan melakukan video processing atau video real-time. Data hasil video processing atau video real-time akan dipilah sesuai data yang berada di dalam database model Mobilenet, apabila data yang terbaca dari kamera itu sesuai dengan database yang sudah tersedia, maka akan terdeteksi suatu objek, lalu, selanjutnya sensor HC SR04 akan membaca jarak objek, seluruh data yang sudah di kumpulkan oleh kamera dan sensor jarak akan di ubah menjadik suara dengan bantuan Library eSpeak untuk selanjutnya ditampilkan melalui earphone, dan proses selesai.



Gambar 8. Flowchart Kerja Sistem

**3.3 Perencanaan Sistem Hardware**

Pada penelitian ini akan dibuat alat bantu pendeteksi objek sekitar untuk tuna netra dengan menggunakan Raspberry Pi 3B yang akan memberikan informasi objek yang ada di sekitar penyandang tuna netra dalam bentuk suara. Alat ini mudah dibawa penggunaanya kemana-mana karena sifatnya yang portable. Alat ini menggunakan sumber tegangan berupa baterai yang dapat di charge ulang.



Gambar 9. Sistem Rancangan Hardware

Gambar 9 merupakan sistem rancangan hardware atau perangkat keras pada penelitian ini. Pada proses pertama, kamera web akan memberikan masukan atau input berupa video real-time dan sensor HC-SR04 akan memberikan input data jarak ke Raspberry Pi. Sistem kerja utama alat ini berada pada Raspberry Pi tersebut karena pada Raspberry Pi terdapat sebuah system yang sudah di program untuk megolah model Mobilenet yang akan mengklasifikasi objek yang akan di deteksi.

**3.4 Rancangan Alat**

Penempatan perangkat keras atau hardware mulai dari kameara WEB M-TECH WB 100, Mini PC, earphone, dan sensor HC-SR04 dirancang sebaik mungkin agar mendapatkan respon sistem yang nyaman, efisien, dan akurat. Rancangan alat pada penelitian ini dapat dilihat pada Gambar 10 berikut.



Gambar 10. Desain Rancangan Alat

Gambar 10 merupakan desain alat pada penelitian ini. Pada desain alat ini kamera yang terpasang pada kacamata hitam yang dihubungkan bersamaan dengan earphone secara simple dan praktis, desain ini adalah desain awal dari pembuatan sistem pada penelitian ini, yang mana desain ini akan di jadikan acuan dalam mendesain alat pada sistem yang akan dibuat pada penelitian ini, kemudahan dan

kepraktisan desain sistem yang di tunjukkan pada gambar menjadi pertimbangan penting untuk menjadikan desain sistem ini sebagai acuan utama.

### 3.5 Arsitektur MobileNet

MobileNets dibangun di atas lapisan konvolusi yang dapat dipisahkan secara mendalam. Setiap lapisan konvolusi yang dapat dipisahkan secara mendalam terdiri dari depthwise convolution dan pointwise convolution. Menghitung depthwise convolution dan titik sebagai lapisan yang terpisah, MobileNet memiliki 28 lapisan. MobileNet standar memiliki 4,2 juta parameter yang dapat lebih jauh dikurangi dengan menyetel hyperparameter pengali lebar secara tepat. Ukuran gambar input adalah  $224 \times 224 \times 3$  [9].

Ada sekitar 80 objek yang dapat di deteksi dengan menggunakan model MobileNet yang di gunakan pada penelitian ini, diantaranya adalah Orang, sepeda, mobil, sepeda motor, pesawat terbang, bis, melatit, truk, perahu, lampu lalu lintas, keran, kebakaran, tanda berhenti, meteran parkir, bangku, burung, kucing, anjing, kuda, domba, lembu, gajah, beruang, kuda zebra, jerapah, ransel, payung, tas tangan, dasi, koper, frisbee, ski, papan seluncur, bola olahraga, layang-layang, tongkat pemukul baseball, sarung baseball, skateboard, papan luncur, raket tenis, botol, gelas anggur, cangkir, garpu, pisau, sendok, mangkuk, pisang, apel, sandwich, jeruk, Brokoli, wortel, Hot Dog, Pizza, donat, kue, kursi, sofa, tanaman di dalam pot, tempat tidur, meja makan, toilet, televisi, laptop, mouse, terampil, papan ketik, telepon selular, microwave, oven, pemanggang roti, wastafel, kulkas, Buku, jam, vas, gunting, boneka beruang, pengering rambut, sikat gigi. Objek deteksi dapat juga di tambah sesuai dengan kebutuhan dengan cara melakukan training objek secara mandiri. Pada penelitian ini hanya beberapa objek yang dijadikan percobaan, di antaranya objek botol, buku, gelas orang dan telepon seluler, dan setiap objek dibedakan menjadi 10 varian setiap objek.

### 3.6 Cara Kerja Alat

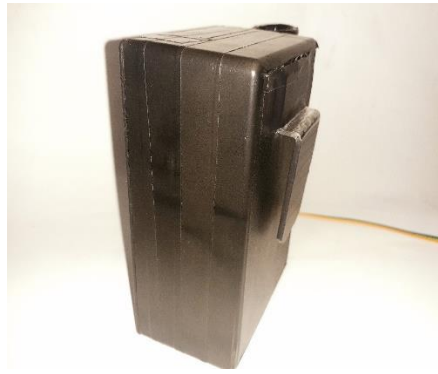
Cara kerja alat yang di buat pada penelitian ini adalah sebagai berikut :

- Mengkoneksikan kabel pada tempatnya masing-masing, kabel kamera ada pada bagian bawah dan kabel untuk earphone ada di bagian kiri
- Menekan tombol push button untuk menyalakan sistem
- Sistem akan menyala dan akan mendeteksi objek yang ada pada jangkauan kamera
- Apabila terdeteksi objek oleh kamera maka sensor HC SR04 akan mendeteksi jarak objek tersebut, dengan ketetapan objek yang terdeteksi harus lurus posisinya dengan sensor HC SR04 supaya dapat dideteksi
- Selanjutnya sistem akan mengolah data dan menjadikannya suara pada earphone
- Alat bantu ini akan terus menerus memberikan informasi objek yang terdeteksi didepannya selama objek terbaca oleh sistem, dan apabila objek sudah tidak berada di depan alat bantu ini maka sistem akan memberika informasi suara pada pengguna.



Gambar 11. Foto Tampak Depan Box Sistem





Gambar 12. Foto Tampak Samping Box Sistem



Gambar 13. Kacamata Dengan Kamera Dan Sensor HC SR04

#### 4. ANALISA & PEMBAHASAN

Pada bab ini akan dibahas tentang pengujian komponen yang akan digunakan apakah bekerja dengan baik dan analisa data yang sudah didapat. Pengujian data analisa yang dilakukan pada bab ini antara lain: Pengujian alat:

- a. Uji running Raspberry Pi 3
- b. Uji running kamera web M-TECH WB 100
- c. Pengujian sensor HC-SR04
- d. Pengujian Baterai Lithium AP77
- e. Pengujian keseluruhan sistem

##### 4.1 Uji Running Raspberry pi 3

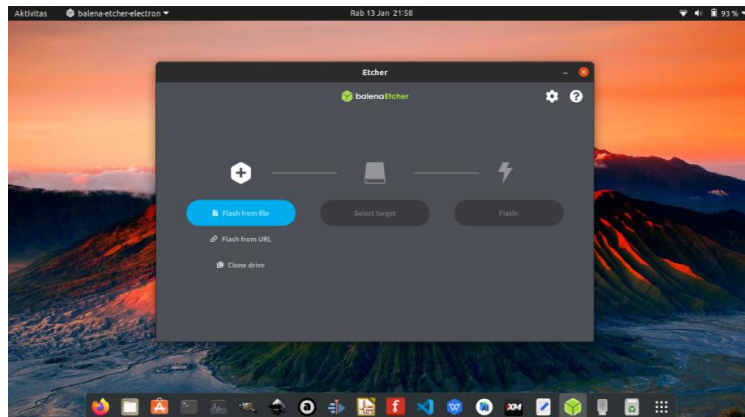
Pengujian ini bertujuan untuk menjadikan Raspberry pi 3 dapat digunakan dan berfungsi dengan baik, pengujian ini dilakukan dengan cara menginstall RASPBAN BUSTER (sistem operasi untuk Raspberry Pi 3) pada Micro SD Card lalu menjalankan Raspberry Pi 3 dengan monitor, alat yang digunakan dalam pengujian ini adalah:

1. Baterai Lithium AP77
2. Micro SD Card 64 GB
3. Komputer atau Laptop
4. Monitor
5. File installer RASPBERRY PI OS
6. Software BALENA ETCHER.

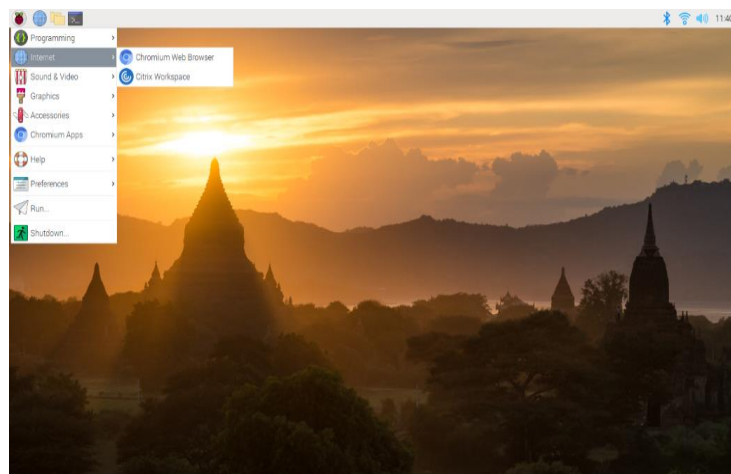
Prosedur Uji running:

- a. download terlebih dahulu file instalasi RASPBAN BUSTER pada website resminya di [www.raspberrypi.org](http://www.raspberrypi.org)
- b. download BALENA ETCHER pada website resminya di <https://www.balena.io/etcher/#downloads>
- c. Menghubungkan Micro SD Card dengan laptop yang sudah terinstall BALENA ETCHER
- d. buka BALENA ETCHER lalu pilih file instalasi RASPBERRY PI OS, pilih Micro SD Card yang akan di jadikan penyimpanan instalasi lalu tekan Flash untuk memulai instalasi

- e. Setelah selesai prosen instalasi, hubungkan Micro SD Card dengan Raspberry Pi 3 yang sudah terhubung dengan monitor.



Gambar 14. Tampilan Software Balena Etcher



Gambar 15. Tampilan Desktop OS Raspbian

Pada pengujian ini, sistem dapat booting kedalam sistem operasi yang menandakan pengujian ini berhasil.

#### 4.2 Uji *Running* Kamera Web M-TECH WB 100

Pada penelitian ini kamera di gunakan untuk mendeteksi objek yang akan di baca oleh sistem, kamera yang di gunakan pada penelitian ini adalah kamera web M-TECH WB 100, oleh karena itu dibutuhkan konfigurasi untuk menghubungkan *Raspberry Pi 3* dengan kamera web M-TECH WB 100.

Pengujian ini bertujuan untuk mengkonfigurasi kamera web M-TECH WB 100 dengan *Raspberry Pi 3* supaya sistem dapat membaca objek yang sudah di tentukan, prosedur pengujian:

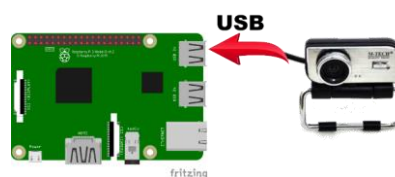
1. Hubungkan kamera web M-TECH WB 100 kepada *Raspberry Pi 3* dengan menghubungkannya kepada port USB yang sudah tersedia
2. Masuk kepada sistem konfigurasi *Raspberry Pi 3* dan aktifkan kamera
3. Masukkan *source code* dibawah ini untuk mengakases kamera pada *Raspberry Pi 3* :

```
import cv2
cap= cv2.VideoCapture(0)

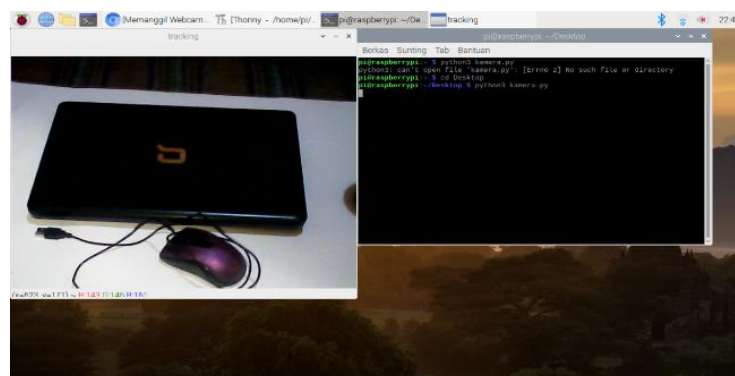
while(1):
    berhasil, frame = cap.read()
    cv2.imshow('tracking', frame)

    k = cv2.waitKey(5) & 0xFF
    if k == 27:
        cap.release()
        cv2.destroyAllWindows()
        break
```

4. Simpan *source code* yang sudah di tulis dengan *file extension .py*
5. Jalankan pada terminal *Raspberry Pi 3* dengan perintah *python3 file.py*.
6. Pada pengujian ini dilakukan 5 kali percobaan pada kamera web M-TECH WB 100 dengan *source code* yang sama.



Gambar 16. Wiring Kamera Kepada Raspberri Pi 3



Gambar 17. Screenshot Pengujian Kamera

### 4.3 Pengujian dan Running Sensor HC SR04

#### 4.3.1 Running Sensor HC SR04

Pada penelitian ini sensor yang di gunakan untuk membaca jarak objek adalah sensor HC-SR04, untuk itu diperlukan pengujian terhadap sensor HC-SR04 supaya dapat berjalan pada sistem *Raspberry Pi 3*, prosedur pengujian :

1. Hubungkan sensor HC-SR04 kepada *Raspberry Pi 3* dengan ketetapan PIN pada *Raspberry Pi 3* sebagai berikut :

Tabel 1. Konektifitas Pin Sensor HC-SR04 Pada *Raspberry Pi 3*

<b>PIN RASPBERRY PI 3</b>	<b>PIN SENSOR HC-SR04</b>
PIN 4	VCC
PIN 6	VCC
PIN 16 / GPIO 23	TRIG PIN
PIN 18 / GPIO 24	ECHO PIN

2. Tuliskan *source code* berikut untuk menjalankan sensor HC-SR04 pada *Raspberry Pi 3* :

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
GPIO_TRIGGER = 18
GPIO_ECHO = 24
GPIO.setup(GPIO_TRIGGER, GPIO.OUT)
GPIO.setup(GPIO_ECHO, GPIO.IN)

def distance():
    GPIO.output(GPIO_TRIGGER, True)
    time.sleep(0.00001)
    GPIO.output(GPIO_TRIGGER, False)
    StartTime = time.time()
    StopTime = time.time()

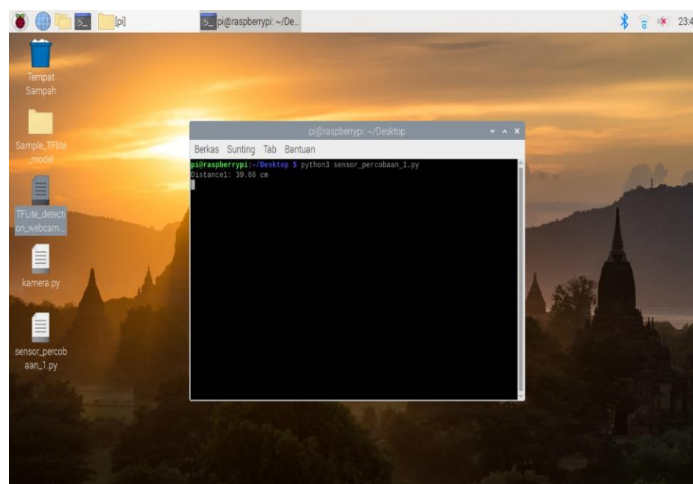
    while GPIO.input(GPIO_ECHO) == 0:
        StartTime = time.time()
    while GPIO.input(GPIO_ECHO) == 1:
        StopTime = time.time()

    TimeElapsed = StopTime - StartTime
    distance = (TimeElapsed * 34300) / 2
    return distance

if __name__ == '__main__':
    try:
        while True:
            dist = distance()
            print ("Measured Distance = %.1f cm" % dist)
            time.sleep(1)

    except KeyboardInterrupt:
        print("Measurement stopped by User")
        GPIO.cleanup()
```

3. Simpan *source code* yang sudah di tulis dengan file *extension .py*
4. Jalankan file pada terminal *Raspberry Pi 3* dengan perintah pada gambar tampilan *running*



Gambar 18. *Screenshot* Pengujian Sensor HC-SR04

**4.3.2 Uji Sensor HC SR04**

Pada pengujian ini sensor HC SR04 di uji dengan mendeteksi objek yang jaraknya berbeda, yang mana objek akan di geser dari jarak terdekat sampai jarak terjauh sensor dan penggeseran ini dilakukan dengan menggeser objek sejauh 10 cm, dan dihasilkan perolehan sebagai berikut :

Tabel 2. Hasil Pengujian Sensor HC-SR04

No	Jarak uji (cm)	Pembacaan sensor (cm)	Selisih pembacaan sensor (cm)	Nilai error	Keterangan
1.	10	10	0	0%	Berhasil
2.	20	19	1	5%	Berhasil
3.	30	29	1	3,3%	Berhasil

No	Jarak uji (cm)	Pembacaan sensor (cm)	Selisih pembacaan sensor (cm)	Nilai error	Keterangan
4.	40	39	1	2,5%	Berhasil
5.	50	49	1	2%	Berhasil
6.	60	59	1	1,7%	Berhasil
7.	70	68	2	2,8%	Berhasil
8.	80	79	1	1,2%	Berhasil
9.	90	88	2	2,2%	Berhasil
10.	100	98	2	2%	Berhasil
11.	110	109	1	0,9%	Berhasil
12.	120	118	2	1,7%	Berhasil
13.	130	129	1	0,8%	Berhasil
14.	140	138	2	1,4%	Berhasil
15.	150	149	1	0,7%	Berhasil
16.	160	158	2	1,2%	Berhasil
17.	170	169	1	0,6%	Berhasil
18.	180	178	2	1,1%	Berhasil
19.	190	188	2	1,0%	Berhasil
20.	200	197	3	1,5%	Berhasil
21.	210	208	2	0,9%	Berhasil
22.	220	216	4	1,8%	Berhasil
23.	230	227	3	1,3%	Berhasil
24.	240	238	2	0,8%	Berhasil
25.	250	247	3	1,2%	Berhasil
26.	260	258	2	0,8%	Berhasil
27.	270	267	3	1,1%	Berhasil
28.	280	276	4	1,4%	Berhasil
29.	290	286	4	1,4%	Berhasil
30.	300	297	3	1%	Berhasil
31.	310	307	3	1%	Berhasil
32.	320	318	2	0,6%	Berhasil
33.	330	326	4	1,2%	Berhasil
34.	340	336	4	1,2%	Berhasil
35.	350	346	4	1,1%	Berhasil
36.	360	356	4	1,1%	Berhasil
37.	370	367	3	0,8%	Berhasil
38.	380	377	3	0,8%	Berhasil
39.	390	385	5	1,3%	Berhasil
40.	400	395	5	1,2%	Berhasil
	Rata-Rata Error			1,4%	
	Persentase Kinerja			100 %	

Dari data yang sudah di ambil akan di tentukan nilai persentase error dan nilai persentase kinerja, dalam hal ini cara untuk menentukan nilai error adalah sebagai berikut :

$$Error = \frac{Jarak\ asli - jarak\ deteksi}{Jarak\ asli} \times 100\% \quad (1)$$

Dengan perhitungan di atas di dapat nilai rata-rata error sebesar 1,4 %, sedang untuk menentukan nilai persentase kinerja sensor HC SR04 dengan cara berikut :

$$Persentase\ Kinerja = \frac{Jumlah\ Keberhasilan}{Jumlah\ Uji} \times 100\% \quad (2)$$

Dengan menggunakan perhitungan di atas di dapatkan nilai persentase kinerja Sensor HC SR04 sebesar 100%.

#### 4.4 Perhitungan Ketahanan Baterai *Lithium AP77*

Pada perhitungan ini baterai *Lithium AP77* akan dihitung ketahanannya ketika memberikan daya pada keseluruhan sistem, dengan ketetapan perhitungan :

1. Kapasitas baterai = 3800 mAh = 3,8 AH
2. Sumber daya baterai = 3,8 AH X 3,7 Volt = 14,6 Watt/jam
3. Konsumsi daya *Raspberry Pi 3* = 4 watt
4. Konsumsi daya sensor HC-SR04 = 0,015Watt
5. Konsumsi daya kamera web M-TECH WB 100 = 0,5 Watt
6. Total keseluruhan konsumsi daya sistem = 4,5 watt
7. Daya tahan baterai = 14,6 watt/jam : 4,5 Watt = 3,2 Jam

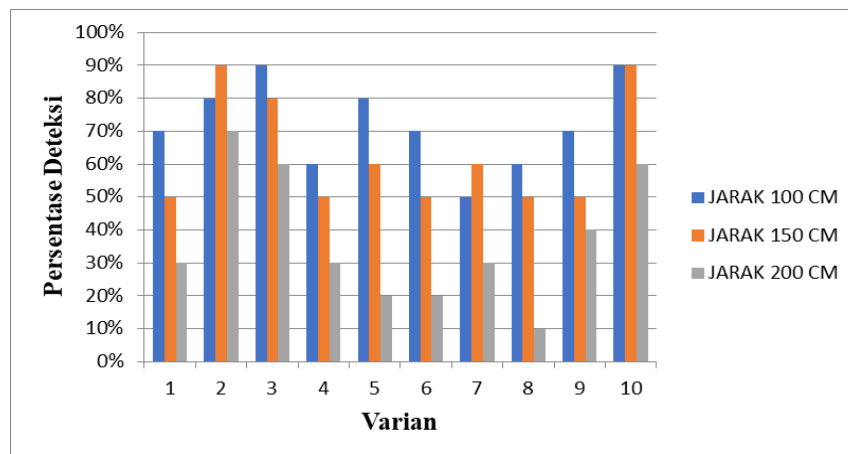
Menurut dari perhitungan di atas maka ketahanan baterai *Lithium AP77* ketika memberikan daya kepada keseluruhan sistem sekitar 3,2 jam.

#### 4.5 Pengujian Deteksi Objek

Pada pengujian ini sistem akan diuji dengan menggunakan objek-objek yang berbeda namun tetap satu jenis, pada pengujian ini akan menggunakan 5 jenis objek yang berbeda dengan 10 jenis objek berbeda disetiap pengujian. Pengambilan data pengujian deteksi objek dilakukan dengan cara meletakkan setiap objek yang akan dideteksi pada jangkauan kamera dengan 3 jarak berbeda, jarak yang dimaksud adalah jarak 100 cm, 150 cm, dan jarak 200 cm, setiap jenis objek akan ditampilkan hasil keberhasilan deteksi dan persentase deteksi objek pada setiap jarak yang sudah ditentukan.

Pendeteksian objek akan dilakukan 10 kali pada setiap jenis objek, setiap 1 kali pengujian bernilai 10%, persentase deteksi akan ditentukan dengan menghitung berapa kali dalam 10 kali pendeteksian jenis obek terdeteksi. Setelah mendapatkan nilai persentase pada setiap jenis objek pada jarak 100cm, 150cm dan 200cm angka persentase dibagikan 3 yang berarti diuji pada 3 keadaan jarak, hasil dari pembagian akan dijadikan nilai persentase pendeteksian. Apabila persentase deteksi dibawah 50% maka deteksi objek akan dinyatakan tidak berhasil, apabila persentase deteksi di atas 50% maka deteksi objek akan dinyatakan berhasil. Berikut adalah data pengujian deteksi objek:

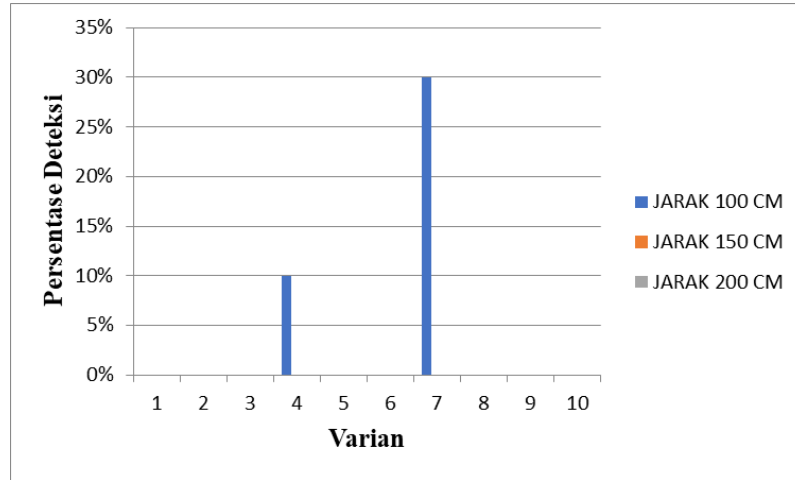
##### 4.5.1 Objek Botol



Gambar 19. Grafik Persentase Deteksi Objek Botol

Untuk hasil dari objek botol, deteksi objek pada objek varian 4, varian 6 varian 7 dan varian 8 deteksi objek dinyatakan tidak berhasil dikarenakan didapat persentase di bawah 50%, hal ini terjadi dikarenakan bentuk varian tersebut yang kecil sehingga sulit untuk sistem mendeteksinya..

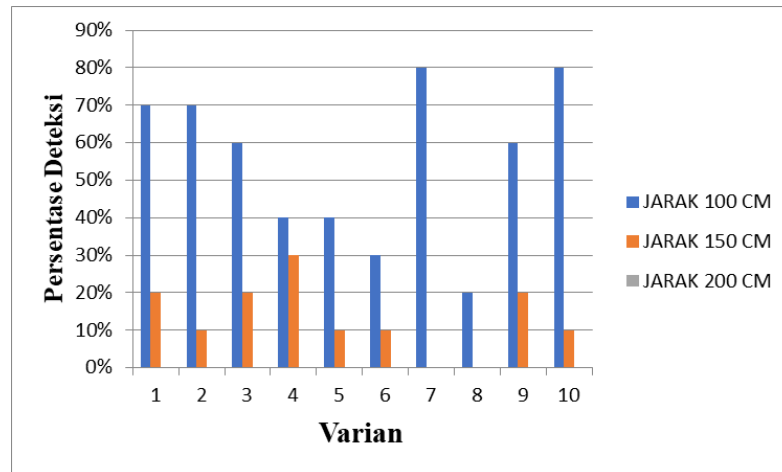
**4.5.2 Objek Buku**



Gambar 20. Grafik Persentase Deteksi Objek Buku

Pada objek buku , hasil dari pendeteksian dinyatakan tidak berhasil pada keseluruhan varian. Pada jarak 100 cm varian 7 terdeteksi 3 kali yang artinya mendapat nilai 30 % dan ini adalah pendeteksian dengan nilai tertinggi pada varian ini, hal ini terjadi karena system masih sulit mendeteksi objek dan yang terbaca adalah objek lain.

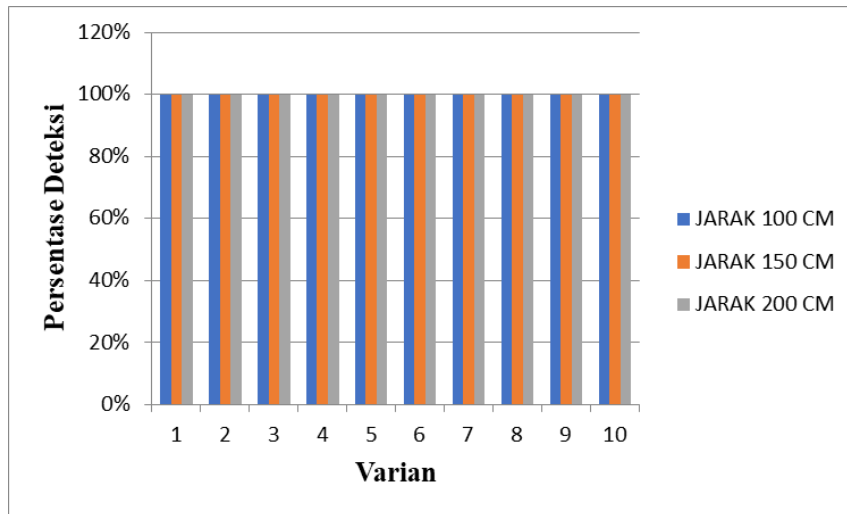
**4.5.3 Objek Gelas**



Gambar 21. Grafik Persentase Deteksi Objek Gelas

Pada objek gelas, hasil pengujian dinyatakan tidak berhasil pada setiap varian, namun pada jarak 100cm sistem dapat beberapa varian dengan baik, hal ini terjadi dikarenakan bentuk objek yang kecil sehingga pada jarak 150cm dan jarak 200cm sistem sulit untuk mengenali objek ini.

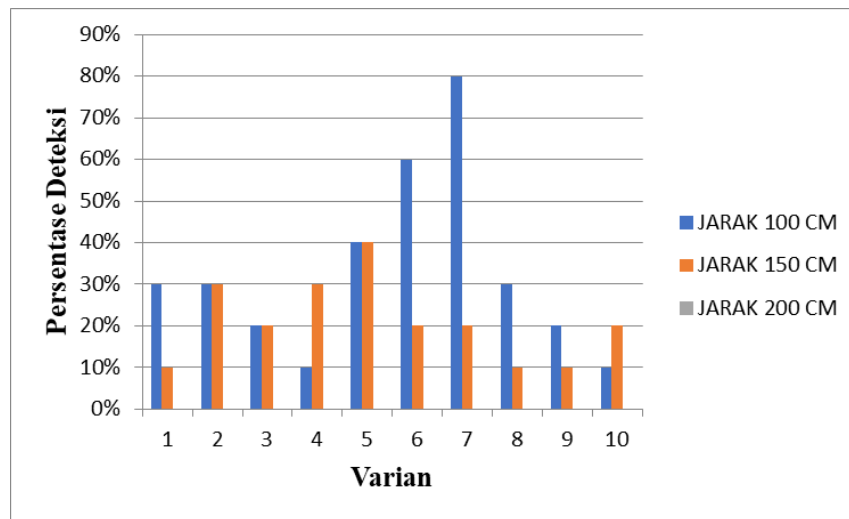
**4.5.4 Objek Orang**



Gambar 22. Grafik Persentase Deteksi Objek Orang

Pada objek orang, hasil dari pada pengujian didapatkan keberhasilan pada seluruh varian, pada jarak 100cm, 150cm dan 200cm objek dapat mendeteksi dengan akurat objek orang, hal ini terjadi dikarenakan objek orang adalah objek dengan bentuk paling besar dari pada objek yang lain, sehingga sistem dapat mendeteksi objek dengan mudah.

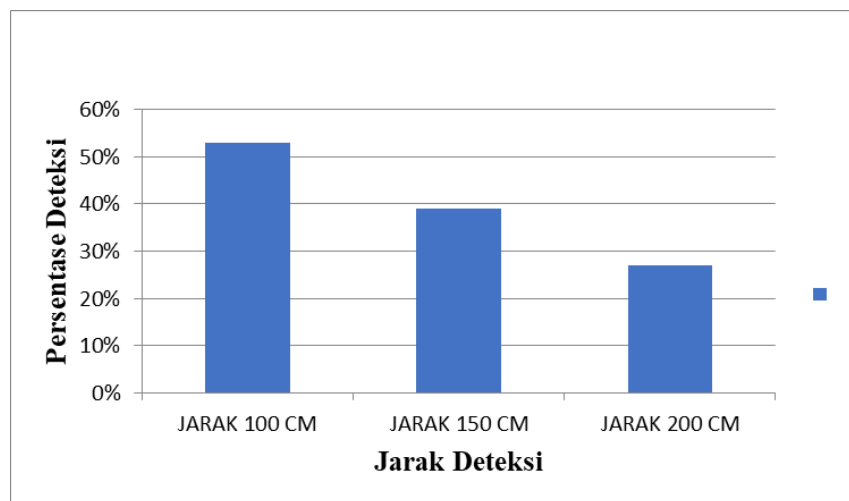
**4.5.5 Objek Telepon Seluler**



Gambar 23. Grafik Persentase Deteksi Objek Telepon seluler

Pada objek telepon seluler, sistem juga sulit untuk mendeteksi objek ini, sehingga didapatkan ketidak berhasilan pada seluruh varian objek, hal ini terjadi juga karena bentuk dari objek telepon seluler yang tidak terlalu besar sehingga objek ini sulit untuk dideteksi oleh sistem.





Gambar 24. Grafik Rata-Rata Persentase Deteksi Pada Keseluruhan Objek

Dari gambar grafik diatas dapat disimpulkan bahwa semakin jauh jarak objek deteksi maka semakin berkurang keakurasian sistem untuk mendeteksi objek. Pada dasarnya bukan hanya jarak saja yang menjadi faktor yang mempengaruhi kinerja sistem, namun ada hal lain yang juga mempengaruhi pendeteksian objek. Faktor-faktor yang menentukan keberhasilan proses deteksi dengan hasil deteksi yang akurat adalah dalam hal kualitas citra yang diinput [10]. Intensitas cahaya juga mejadi salah satu faktor keberhasilan pendeteksian objek. Pengaruh perbedaan intensitas pencahayaan terhadap kemampuan identifikasi objek telah dibuktikan. Semakin rendah tingkat pencahayaan maka semakin sulit bagi sistem untuk mengidentifikasi objek lingkaran dan sebaliknya [11]. Terkait dengan cahaya sebagai salah satu faktor yang mempengaruhi pendeteksian objek, intensitas cahaya dan jarak terhadap objek manusia mempengaruhi akurasi deteksi sistem. Pada penelitian ini, akurasi tertinggi mencapai 91,67%, dengan kondisi lingkungan memiliki intensitas cahaya 50 lux dan jarak terhadap objek 1-5 meter.

## 5. KESIMPULAN

Berdasarkan perakitan dan pengujian sistem yang dilakukan pada penelitian ini, diperoleh beberapa kesimpulan sebagai berikut:

- Sistem deteksi objek dibangun dengan Raspberry Pi 3B sebagai pusat proses deteksi objek dan model pra-terlatih Mobilnet sebagai arsitektur datanya, ada beberapa komponen pendukung yaitu kamera WEB M TECH WB 100 sebagai sensor kamera, sensor HC SR04 sebagai sensor jarak dan earphone JBL SYNCHROS S100 sebagai output, pada pengujian sensor HC SR04 rata-rata error sebesar 1,4% dan persentase kinerja 100 %, sehingga dapat disimpulkan sensor HC SR04 dapat membaca jarak objek dengan akurat, pada pengujian keseluruhan sistem, sistem dapat mendeteksi objek uji dengan mendapatkan rata-rata tingkat keberhasilan sebesar 40%, Pada perhitungan ketahanan baterai lithium ap77, baterai dapat bertahan selama kurang lebih 3,2 jam ketika memberikan daya pada keseluruhan sistem.
- Terdapat beberapa faktor yang mempengaruhi deteksi objek diantaranya yaitu jarak dan cahaya. Pada penelitian ini, faktor yang diuji adalah faktor jarak dengan hasil pengujian rata-rata tingkat keberhasilan sebesar 53% pada jarak uji 100 cm, 39% pada jarak uji 150 cm, dan 27% pada jarak uji 200cm, dengan ini dapat di simpulkan bahwa semakin jauh jarak objek deteksi semakin berkurang keakurasian sistem untuk mendeteksi objek.

**REFERENSI**

- [1] F. Sindy, "Pendeteksian Objek Manusia Secara Realtime Dengan Metode MobileNet-SSD Menggunakan Movidius Neural Stick pada Raspberry Pi," hal. 77, 2019.
- [2] J. Ali Andre, "Sistem Security Webcam Dengan Menggunakan Microsoft Visual Basic (6.0)," *Rabit J. Teknol. dan Sist. Inf. Univrab*, vol. 1, no. 2, hal. 46–58, 2016, doi: 10.36341/rabit.v1i2.23.
- [3] I. S. Walingkas *et al.*, "Perpaduan Sensor Ultrasonik Dengan Mini Computer Raspberry Pi Sebagai Pemandu Robot Beroda," vol. 8, no. 3, hal. 121–132, 2019.
- [4] I. K. S. Buana dan N. M. D. K. Putri, "Deteksi Gerakan Kepala Dan Kedipan Mata Dengan Haar Cascade Classifier Contour Dan Morfologi Dalam Pengoperasian Komputer Untuk Kaum Difable," *JURTEKSI (Jurnal Teknol. dan Sist. Informasi)*, vol. IV, no. 1, hal. 29–36, 2018, doi: 10.33330/jurtekxi.v5i1.273.
- [5] A. N. Syahrudin dan T. Kurniawan, "Input Dan Output Pada Bahasa," *J. Dasar Pemrograman Python STMIK*, no. January, hal. 1–7, 2018.
- [6] F. Martunus, "Implementasi Face Recognition Dengan Opencv Pada 'Smart Cctv' Untuk Keamanan Brankas Berbasis Iot," 2020.
- [7] S. Bramasto, "Tensorflow Lite Pada Perangkat Bergerak Guna Prediksi Waktu Panen pada Operasi Pertanian Vertikal," in *Technopex-2020*, 2020, hal. 1–9.
- [8] Hendriyana dan Y. H. Maulana, "Identifikasi Jenis Kayu Menggunakan Convolutional Neural Network Dengan Arsitektur Mobilenet," *J. RESTI(Rekayasa Sist. dan Teknol. Informasi)*, vol. 4, no. 1, hal. 70–76, 2020.
- [9] F. Evan, "Penerapan Image Classification Dengan Pre-Trained Model Mobilenet Dalam Client-Side Machine Learning Pen` Erapan Image Classification Dengan Pre-Trained Model Mobilenet Dalam Client-Side Machine Learning," 2020.
- [10] M. Zunaidi, M. Ramadhan, dan H. Winata, "Faktor Penentu Tingkat Keberhasilan Sistem Deteksi Wajah pada Citra Digital," *J. Ilm. SAINTIKOM Sains dan Komput.*, vol. 14, no. 2, hal. 121–130, 2015.
- [11] H. Rahmannuri, F. Nadziroh, dan H. Fauzy, "Pengaruh Perbedaan Intensitas Pencahayaan Terhadap Identifikasi Objek Gerak," *J. Sist. Inf. dan Bisnis Cerdas*, vol. 12, no. 2, hal. 109–116, 2019.