

BAB III

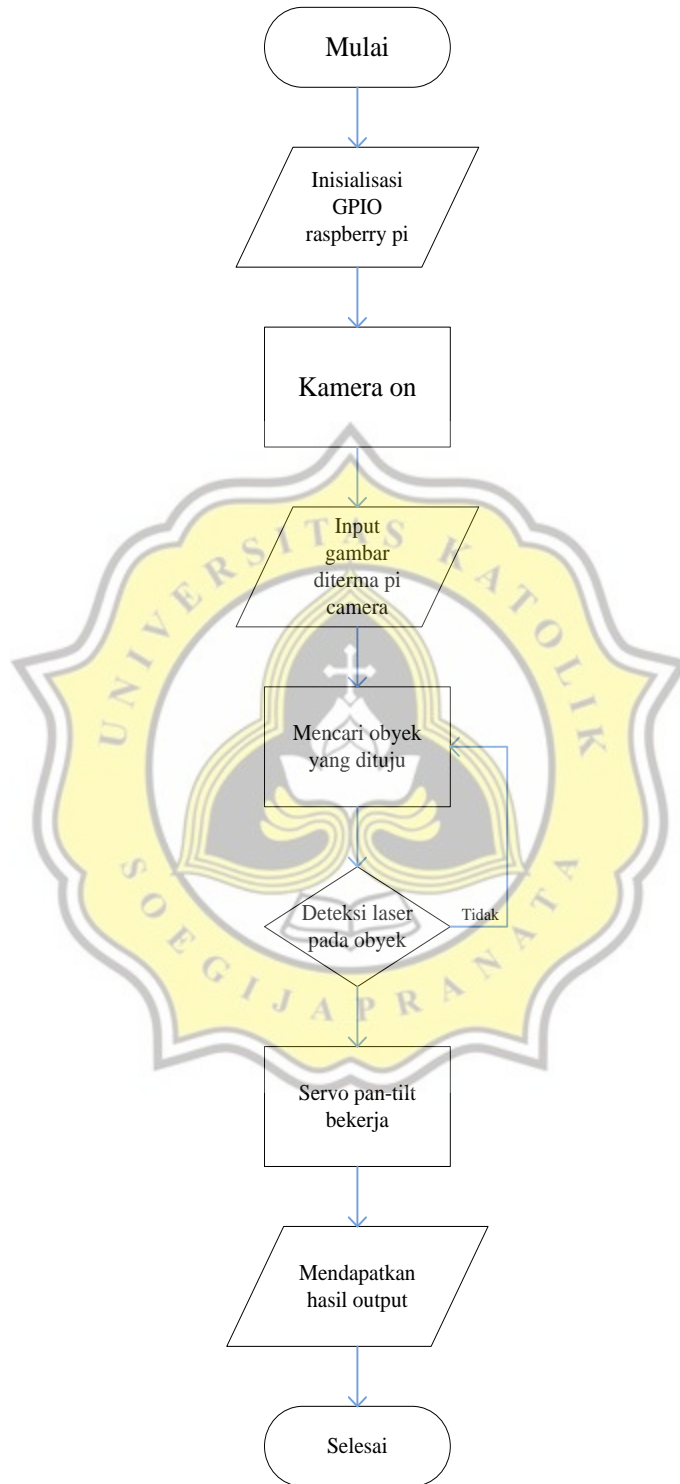
PERANCANGAN HARDWARE DAN SOFTWARE

3.1. Pendahuluan

Pada bab ini akan dijelaskan mengenai konstruksi alat deteksi perpindahan posisi berbasis *raspberry pi 4* dengan rangkaian *hardware* yang terdapat komponen-komponen didalamnya dan rangkaian digital yang meliputi program pada *raspberry pi 4* untuk menampilkan hasil pembacaan.

3.2. Rangkaian Hardware

Pada alat ini menggunakan input atau sensor utama yaitu *Raspberry Pi Camera* yang dikoneksikan langsung ke *RaspberryPi* dan output berupa dua buah servo untuk mekanisme *Pan-Tilt* yang menggerakkan laser. Berikut pada Gambar 3.1 merupakan *flowchart* cara kerja yang dibuat sebagai berikut:



Gambar 3. 1. Flowchart cara kerja

Dua servo pada *Pan-Tilt* yang terdiri dari servo pertama bergerak ke kanan atau kekiri sebanyak 180 derajat dan servo kedua bergerak ke atas dan kebawah sebanyak 90 derajat menyesuaikan dengan rangka konstruksi sistem *Pan-Tilt*.

Kedua servo dan konstruksi *pan-tilt* ini menyesuaikan dengan bobot laser 303 yang lumayan berat gambar ditunjukkan Gambar 3.2.



Gambar 3. 2. Laser dan *Pan-Tilt* Servo Motor

Seperti Gambar 3.2 perancangan alat ini menggunakan *power supply* 5V 3A untuk power servo *pan-tilt* dan juga *raspberry pi* menggunakan *powerbank*.

Dikarenakan ketika terjadi pemadaman listrik maka raspberry pi tidak langsung mati yang beresiko menyebabkan *corrupt file* pada sistem perangkat lunak *raspberry pi* akibat tidak di *shutdown* terlebih dahulu.

Tabel 2. Data Input dan Output pada Raspberry Pi

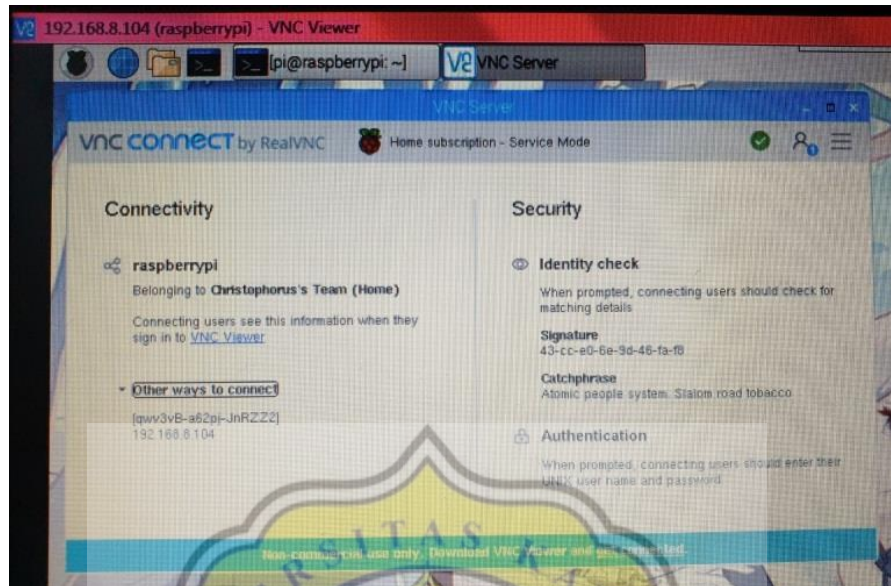
PIN GPIO	JENIS	KOMPONEN
PI 3V3	Output	VCC PCA9685
PI GND	Output	GND PCA9685
PI SCL	Output	SCL PCA9685
PI SDA	Output	SCA PCA9685
PWM PCA9685	Output	Servo orange wire
V+ PCA9685	Output	Servo red wire
GND PCA9685	Output	Servo brown wire
CSI Port	Input	Pi Camera

Tabel 2, diatas merupakan data *input* dan *output* pada komponen yang akan dipasang ke *raspberry pi*. Pada *power* dan *ground* dari 2 buah servo motor berasal dari 5V, ground, PWM servo motor dihubungkan pada pin *driver* PCA9685. Setelah seluruh komponen terpasang dengan benar maka selanjutnya mempersiapkan *software* atau perangkat lunak yang dibutuhkan oleh *raspberry pi*.

3.3. Persiapan software yang akan digunakan

Raspberry pi 4 model B merupakan perangkat *Mini Personal Computer* sehingga berbeda dari *Mikrokontroller* pada umumnya. Perangkat ini membutuhkan *Operation System (OS)* khusus yaitu *Raspbian OS* yang harus diinstall terlebih dahulu ke dalam *MicroSD Card* dengan ukuran file 7GB yang memang berperan seperti *Harddisk* dan *Personal Computer* ataupun laptop pada umumnya. Setelah dilakukan penginstallan sistem operasi selanjutnya dilakukan pengecekan dengan menghubungkan perangkat *RaspberryPi* ke layar monitor yang dapat diakses melalui kabel *HDMI (High Definition Multimedia Interface)*. Setelah dipastikan Sistem Operasi dapat berjalan dengan baik, lalu sesuaikan resolusi yang diinginkan pada *RaspberryPi Preferences* kemudian *RaspberryPi Configuration*. Pada settingan *Raspberry Pi Configuration* aktifkan kamera agar kamera dapat menampilkan *display* ketika diaktifkan. Settingan juga dapat kita rubah melalui perintah *Open LXTerminal* yang juga tersedia pada *RaspberryPi*.

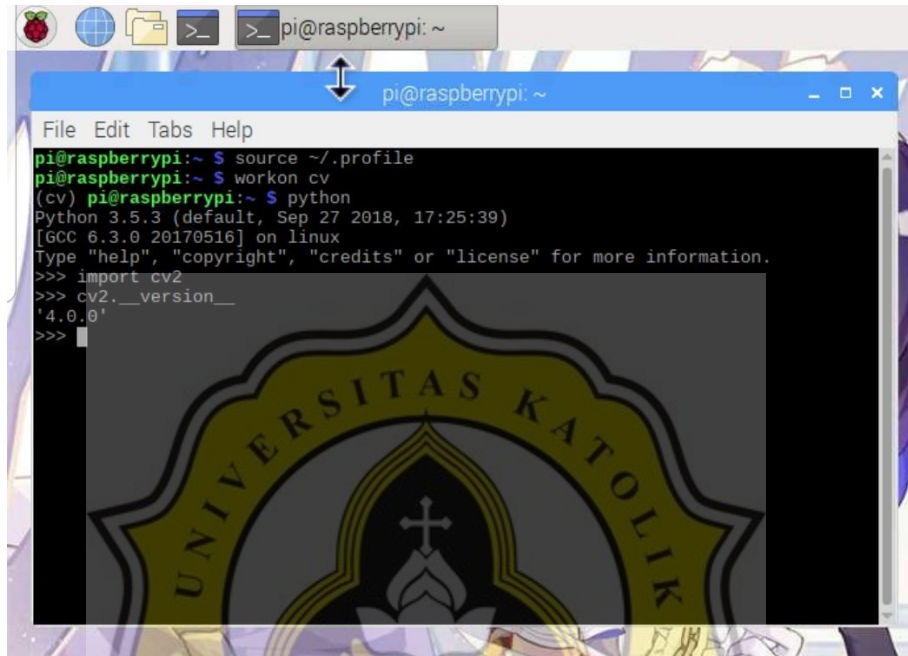
Selain akses dengan layar monitor, *Raspberry pi* juga dapat diakses menggunakan *Windows PC* dengan aplikasi bernama *VNC Viewer*. Aplikasi ini digunakan untuk mengakses antar gadget elektronik seperti *Raspberry Pi* dari jarak jauh dengan *IP Address* dan konektivitas *WiFi* yang sama sehingga tidak lagi memerlukan layar monitor. Penginstallannya cukup mudah dengan mengikuti tutorial dari *website* resminya untuk *RaspberryPi* sebagai pengganti *display* monitornya yang ditunjukkan pada Gambar 3.3.



Gambar 3. 3. VNC Viewer yang terkoneksi ke PC

Setelah akses sudah sesuai dengan yang diinginkan, saatnya melakukan penginstalan aplikasi program *Python*, *Library* *Open CV* dan *Numpy*. *Python* biasanya sudah tersedia dan siap digunakan dalam *OS Raspbian* ketika awal penginstalan. Berbeda dengan *OpenCV* dan *Numpy*, aplikasi ini harus diinstal terlebih dahulu agar dapat kompatibel dengan program *Python*. Penginstalan *OpenCV* dan *Numpy* biasanya harus dilakukan bersamaan sehingga memerlukan tata cara dan waktu yang cukup lama (hingga 5 jam), untuk langkah-langkah penginstalan dapat dicari di internet. Memastikan *OpenCV* dapat terinstall dengan baik maka kita dapat melakukan perintah pada *LXTerminal* dengan memberi perintah `source ~/.profile` untuk membuka folder dimana *library* disimpan kemudian `workoncv` untuk mengaktifkan *libraryOpenCV* setelah itu muncul huruf (cv) pada terminal.

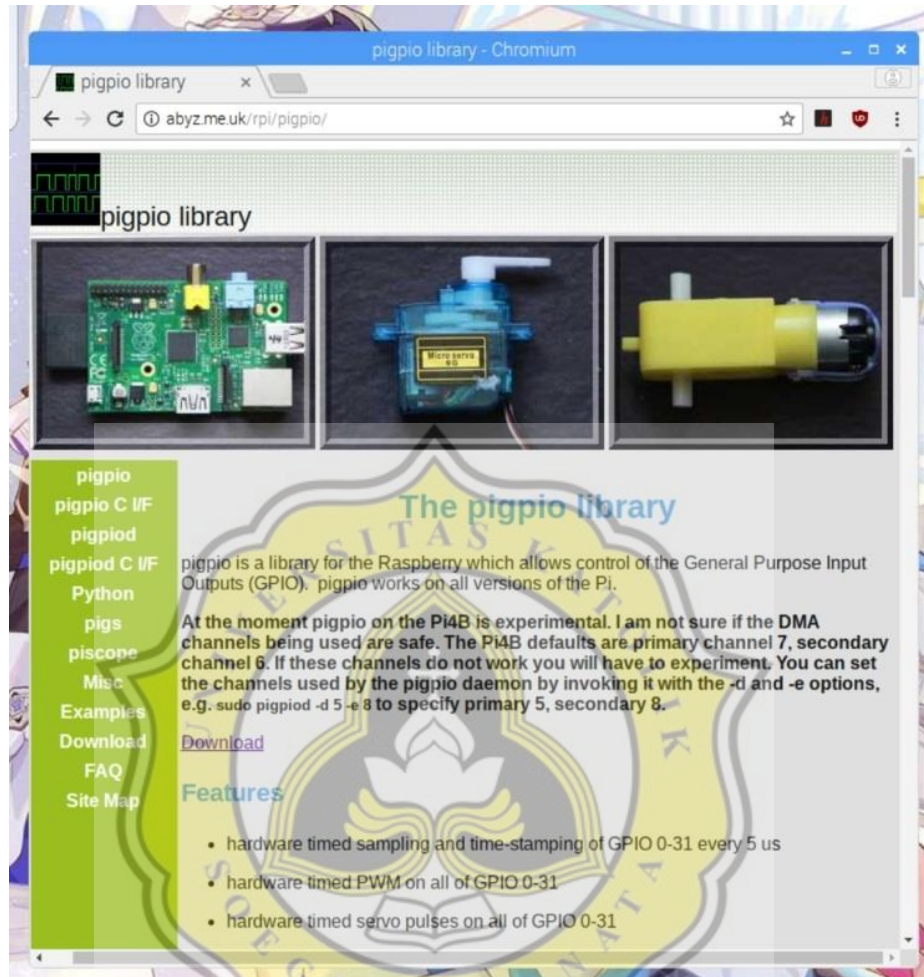
Untuk mengecek versi dari *OpenCV* dapat dicoba dengan mengetik *Python* kemudian `import cv2` dan `cv2.__version__`. Tampilan akan seperti di Gambar 3.4.



```
pi@raspberrypi:~ $ source ~/.profile
pi@raspberrypi:~ $ workon cv
(cv) pi@raspberrypi:~ $ python
Python 3.5.3 (default, Sep 27 2018, 17:25:39)
[GCC 6.3.0 20170516] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import cv2
>>> cv2.__version__
'4.0.0'
>>>
```

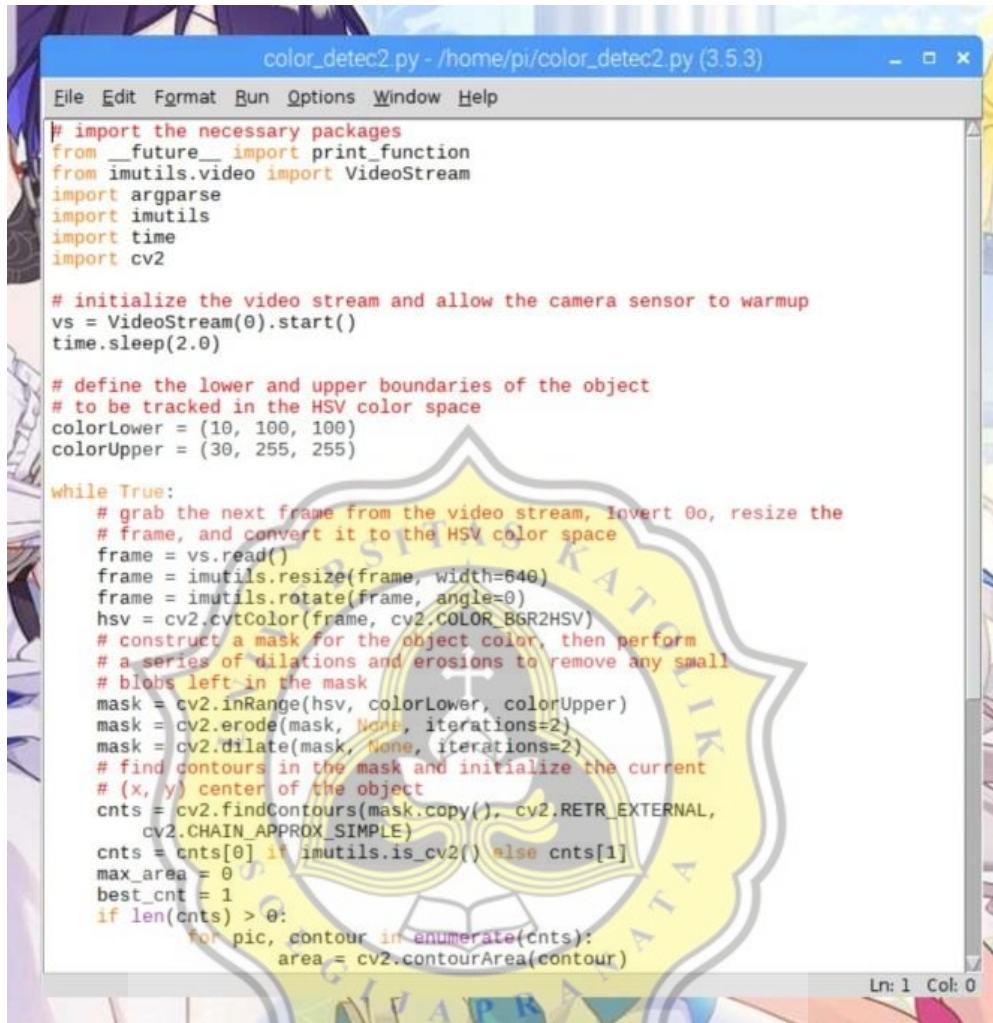
Gambar 3. 4. Perintah mengaktifkan *OpenCV*

Setelah itu tak lupa juga dilakukan penginstalan *library* khusus bernama *Pigpio.Library* ini berfungsi sebagai *filter* untuk membantu kekurangan *Raspberry Pi* di mana Pin *GPIO* pada *RaspberryPi* memiliki delay tidak seperti mikrokontroller seperti *Arduino* pada umumnya, karena *RaspberryPi* juga mengolah data pada sistem operasinya sehingga sedikit terbebani. Seperti halnya *OpenCV* ,tata cara penginstalan *Pigpio* cukup mudah dan tersedia di website resminya. Tampilan *libraryPigpio* yang dibuat khusus untuk pemrograman *Raspberry Pi* pada Gambar 3.5.



Gambar 3. 5. Website Resmi LibraryPigpio

Setelah berbagai macam *software* yang diperlukan sudah terinstall pada *Raspberry Pi*, selanjutnya pemrograman untuk bisa langsung dibuat menggunakan *softwarePython* yang sudah terinstall sebelumnya. Ditunjukkan pada Gambar 3.6 tampilan *software* pemrograman *python* pada *Raspberry Pi*.



```
color_detec2.py - /home/pi/color_detec2.py (3.5.3)
File Edit Format Run Options Window Help
# import the necessary packages
from __future__ import print_function
from imutils.video import VideoStream
import argparse
import imutils
import time
import cv2

# initialize the video stream and allow the camera sensor to warmup
vs = VideoStream(0).start()
time.sleep(2.0)

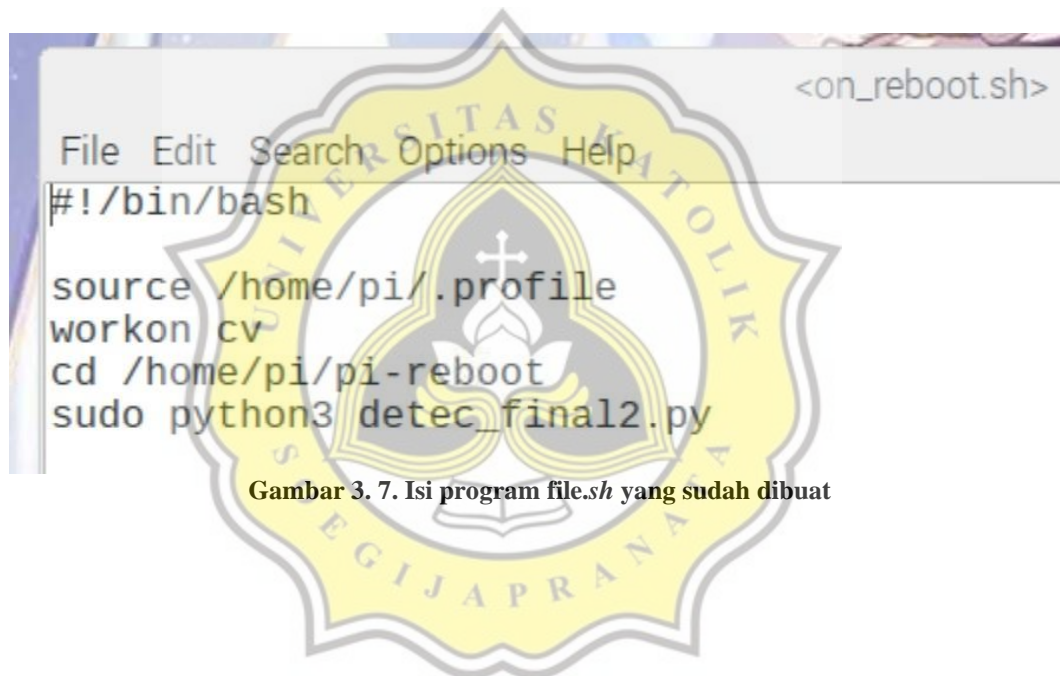
# define the lower and upper boundaries of the object
# to be tracked in the HSV color space
colorLower = (10, 100, 100)
colorUpper = (30, 255, 255)

while True:
    # grab the next frame from the video stream, invert 0o, resize the
    # frame, and convert it to the HSV color space
    frame = vs.read()
    frame = imutils.resize(frame, width=640)
    frame = imutils.rotate(frame, angle=0)
    hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
    # construct a mask for the object color, then perform
    # a series of dilations and erosions to remove any small
    # blobs left in the mask
    mask = cv2.inRange(hsv, colorLower, colorUpper)
    mask = cv2.erode(mask, None, iterations=2)
    mask = cv2.dilate(mask, None, iterations=2)
    # find contours in the mask and initialize the current
    # (x, y) center of the object
    cnts = cv2.findContours(mask.copy(), cv2.RETR_EXTERNAL,
        cv2.CHAIN_APPROX_SIMPLE)
    cnts = cnts[0] if imutils.is_cv2() else cnts[1]
    max_area = 0
    best_cnt = 1
    if len(cnts) > 0:
        for pic, contour in enumerate(cnts):
            area = cv2.contourArea(contour)
```

Gambar 3. 6. Tampilan pemrograman *Python* pada *RaspberryPi*

Setelah pembuatan program *Python* selesai dibuat, dilakukan pembuatan file *bash shell script* atau yang disebut dengan file dengan format *.sh*. File dengan format ini berisi bahasa pemrograman yang digunakan untuk menyusun perintah perintah untuk menjalankan suatu aplikasi tertentu. Seperti sebelumnya, ketika ingin menjalankan program yang berisi *libraryOpenCV*, maka harus diberi perintah `source ~/.profile` untuk membuka folder *library* dan `workon cv` untuk

mengaktifkan *library* pada *LXTerminal*. Pada kali ini, file *.sh* yang dibuat akan berisi perintah-perintah diatas secara bersamaan dengan program *Python* yang sudah dibuat sebelumnya, sehingga hanya dengan memberikan satu perintah nama file *.sh* dengan contoh `/.on_reboot.sh` tersebut pada *LXTerminal*, maka program akan dapat langsung berjalan. Ditunjukkan pada Gambar 3.7 merupakan isi program file *.sh* untuk menjalankan program dalam penelitian ini.



```
<on_reboot.sh>
File Edit Search Options Help
#!/bin/bash
source /home/pi/.profile
workon cv
cd /home/pi/pi-reboot
sudo python3 detec_final2.py
```

Gambar 3. 7. Isi program file *.sh* yang sudah dibuat

3.4. Raspberry pi 4 model B

Pada perancangan ini menggunakan *Raspberry Pi 4*. Raspberry Pi dapat berjalan pada debian berbasis GNU/Linux dan sistem operasi Raspbian. *Raspberry Pi* dilengkapi dengan berbagai macam fasilitas seperti I2C, Lan Port, HDMI Port, GPIO dan lain-lain. *Raspberry Pi 4 Model B* adalah produk terbaru dalam jajaran komputer *Raspberry Pi* yang populer. Ini menawarkan peningkatan kecepatan prosesor, kinerja multimedia, memori, dan konektivitas yang luar biasa dibandingkan dengan *Raspberry Pi 3 Model B+* generasi sebelumnya, sambil mempertahankan kompatibilitas ke belakang dan konsumsi daya yang serupa. Untuk penggunaan akhir, *Raspberry Pi 4 Model B* memberikan kinerja dekstop yang sebanding dengan sistem PC x86 *entry-level*. Fitur utama produk ini meliputi prosesor *quad-core* 64-bit kinerja tinggi, dukungan layar ganda pada resolusi hingga 4K melalui sepasang port mikro-HDMI, decode video perangkat keras hingga 4Kp60, hingga 4GB RAM, dual-band 2.4/5.0 GHz LAN nirkabel, Bluetooth 5.0, Gigabit Ethernet, USB 3.0, dan kemampuan PoE (melalui add-on PoE HAT terpisah). Berikut Gambar 3.2. merupakan *Input Output Raspberry Pi 4*.



Gambar 3. 8. Raspberry pi 4 model B

General Purpose Input Output (GPIO) adalah aspek paling dasar, namun dapat diakses dari *Raspberry Pi*. GPIO dapat memiliki arahan untuk menerima atau mengirim arus (input, output masing-masing) dan kita dapat mengontrol keadaan dan arah port/pin menggunakan bahasa pemrograman seperti Python, JavaScript, node-RED dll.

Raspberry Pi 4 memiliki pesifikasi antara lain :

Tabel 3. Tabel spesifikasi Raspberry pi 4

Processor	Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
Memory	4GB LPDDR4 (depending on model)
Connectivity	2.4 GHz and 5.0 GHz IEEE 802.11b/g/n/ac wireless, LAN, Bluetooth 5.0, BLE, Gigabit Ethernet, 2 × USB 3.0 ports,

	2 × USB 2.0 ports.
GPIO	Standard 40-pin GPIO header (fully backwards-compatible with previous boards)
Video & sound	2 × micro HDMI ports (up to 4Kp60 supported) 2-lane MIPI DSI display port 2-lane MIPI CSI camera port 4-pole stereo audio and composite video port
Multimedia	H.264 (1080p60 decode, 1080p30 encode); OpenGL ES, 3.0 graphics
Sd card support	Micro SD card slot for loading operating system and data storage
Input power	5V DC via USB-C connector (minimum 3A1) 5V DC via GPIO header (minimum 3A1) Power over Ethernet (PoE)–enabled (requires separate PoE HAT)
Product life time	The Raspberry Pi 4 Model B will remain in production until at least January 2026.