

LAMPIRAN

Program Utama

```
import cv2
import numpy as np
import serial
import RPi.GPIO as GPIO

if __name__ == '__main__':
    GPIO.setwarnings(False)
    kernel = np.ones((5,5),np.uint8)
    cap = cv2.VideoCapture(0)
    panjang=320
    lebar=240
    cap.set(3,panjang)
    cap.set(4,lebar)
    in1 = 24
    in2 = 23
    en = 25
    GPIO.setmode(GPIO.BCM)
    GPIO.setup(in1,GPIO.OUT)
    GPIO.setup(in2,GPIO.OUT)
    GPIO.setup(en,GPIO.OUT)
    GPIO.output(in1,GPIO.LOW)
    GPIO.output(in2,GPIO.LOW)
    p=GPIO.PWM(en,1000)
    kondisinya=0
    p.start(100)
    final=0
    p.ChangeDutyCycle(70)
    ser = serial.Serial('/dev/ttyUSB0', 115200, timeout=0.05)
    ser.reset_input_buffer()

    def nothing(x):
        pass
    cv2.namedWindow('HueComp')
    cv2.namedWindow('SatComp')
    cv2.namedWindow('ValComp')
    cv2.namedWindow('closing')
    cv2.namedWindow('tracking')

    cv2.createTrackbar('hmin', 'HueComp',159,179,nothing)
    cv2.createTrackbar('hmax', 'HueComp',179,179,nothing)

    cv2.createTrackbar('smin', 'SatComp',101,255,nothing)
    cv2.createTrackbar('smax', 'SatComp',226,255,nothing)

    cv2.createTrackbar('vmin', 'ValComp',84,255,nothing)
    cv2.createTrackbar('vmax', 'ValComp',138,255,nothing)

    def sortSecond(val):
        return val[1]
    while True:
        nilai=""
        if(ser.in_waiting>0):
```

```

while ser.in_waiting:
    nilai = str(ser.readline())
    nilai = nilai.replace(r"\r\n", "")
    nilai = nilai.replace("b", "")
    nilai = nilai.replace("'", "")
    if (nilai != ""):
        akhir = nilai
if (int(akhir) > 200 and int(akhir) < 580):
    final = int(akhir)
print(final)

buzz = 0
GPIO.output(in1, GPIO.LOW)
GPIO.output(in2, GPIO.LOW)

ser.write(b"m\n")
_, frame = cap.read()
HSV = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
Hue, sat, val = cv2.split(HSV)
hmn = cv2.getTrackbarPos('hmin', 'HueComp')
hmx = cv2.getTrackbarPos('hmax', 'HueComp')

smn = cv2.getTrackbarPos('smin', 'SatComp')
smx = cv2.getTrackbarPos('smax', 'SatComp')

vmn = cv2.getTrackbarPos('vmin', 'ValComp')
vmx = cv2.getTrackbarPos('vmax', 'ValComp')
hthresh = cv2.inRange(np.array(Hue), np.array(hmn), np.array(hmx))
sthresh = cv2.inRange(np.array(sat), np.array(smn), np.array(smx))
vthresh = cv2.inRange(np.array(val), np.array(vmn), np.array(vmx))
tracking = cv2.bitwise_and(hthresh, cv2.bitwise_and(sthresh, vthresh))
#kebalikan = cv2.bitwise_not(hthresh, cv2.bitwise_and(sthresh, vthresh))

dilation = cv2.dilate(tracking, kernel, iterations = 1)
#dilation1 = cv2.dilate(kebalikan, kernel, iterations = 1)
closing = cv2.morphologyEx(dilation, cv2.MORPH_CLOSE, kernel)
closing = cv2.GaussianBlur(closing, (5, 5), 0)
#closing1 = cv2.morphologyEx(dilation1, cv2.MORPH_CLOSE, kernel)
#closing1 = cv2.GaussianBlur(closing1, (5, 5), 0)
contours, hierarchy = cv2.findContours(closing,
cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_TC89_L1)
result = frame.copy()
polygonelist = []
kanan = []
kiri = []
kananrev = []
kirirev = []
xkanan = 0
xkiri = 0
ykanan = 0
ykiri = 0
#print("baru")
for cntr in contours:
    perimeter = cv2.arcLength(cntr, True)
    epsilon = 0.005 * cv2.arcLength(cntr, True)
    approx = cv2.approxPolyDP(cntr, epsilon, True)
    #print(approx)
    polygonelist.append(approx)
cv2.drawContours(frame, polygonelist, -1, (0, 255, 0), 2)

```

```

for x in range(0,len(polygonelist)):
    for a in range(0,len(polygonelist[x])):
        for b in range(0,len(polygonelist[x][a])):
            if(x==0):
                kanan.append(polygonelist[x][a][b])
                kananrev.append(polygonelist[x][a][b])
            elif(x==1):
                kiri.append(polygonelist[x][a][b])
                kirirev.append(polygonelist[x][a][b])
#for k in range(0,len(kanan)):
# print(kanan[k][1])
kanan.sort(key=sortSecond)
kiri.sort(key=sortSecond)
kananrev.sort(key=sortSecond,reverse=True)
kirirev.sort(key=sortSecond,reverse=True)
for k in range (0,2):
    if(k==0):
        xkanan=kanan[k][0]
        xkiri=kiri[k][0]
        ykiri=kanan[k][1]
        ykanan=kiri[k][1]
    elif(k==1):
        if(kanan[k][0]<xkanan):
            xkananatas=kanan[k][0]
            ykananatas=kanan[k][1]
        else:
            xkananatas=xkanan
            ykananatas=ykanan
        if(kiri[k][0]>xkiri):
            xkiriatas=kiri[k][0]
            ykiriatas=kiri[k][1]
        else:
            xkiriatas=xkiri
            ykiriatas=ykiri
#print(xkananatas,ykananatas,xkiriatas,ykiriatas)
for k in range (0,2):
    if(k==0):
        xkanan=kananrev[k][0]
        xkiri=kirirev[k][0]
        ykiri=kananrev[k][1]
        ykanan=kirirev[k][1]
    elif(k==1):
        if(kananrev[k][0]<xkanan):
            xkananbawah=kananrev[k][0]
            ykananbawah=kananrev[k][1]
        else:
            xkananbawah=xkanan
            ykananbawah=ykanan
        if(kiri[k][0]>xkiri):
            xkiri bawah=kirirev[k][0]
            ykiri bawah=kirirev[k][1]
        else:
            xkiri bawah=xkiri
            ykiri bawah=ykiri
misskiri=int(panjang/2)-xkiriatas
misskanan=xkananatas-int(panjang/2)
print(misskiri,misskanan,abs(misskanan-misskiri))
posisi="Kanan"
print(kondisinya)
if(abs(misskanan-misskiri)<=100):
    posisi="Tengah"

```

```

else:
    if(misskiri>misskanan):
        posisi="Kiri"
        kondisinya=1
        GPIO.output(in1,GPIO.LOW)
        GPIO.output(in2,GPIO.HIGH)
    else:
        posisi="Kanan"
        kondisinya=2
        GPIO.output(in1,GPIO.HIGH)
        GPIO.output(in2,GPIO.LOW)
if(kondisinya==1 and posisi=="Tengah"):
    if(final>340 and final<450):
        GPIO.output(in1,GPIO.LOW)
        GPIO.output(in2,GPIO.LOW)
    else:
        GPIO.output(in1,GPIO.HIGH)
        GPIO.output(in2,GPIO.LOW)
        print("Balik arah")
elif(kondisinya==2 and posisi=="Tengah"):
    if(final>340 and final<450):
        GPIO.output(in1,GPIO.LOW)
        GPIO.output(in2,GPIO.LOW)
    else:
        GPIO.output(in1,GPIO.LOW)
        GPIO.output(in2,GPIO.HIGH)
        print("Balik arah")

cv2.putText(frame, posisi, (0,20), cv2.FONT_HERSHEY_SIMPLEX,
            0.5, (255, 0, 0), 2, cv2.LINE_AA)
cv2.circle(frame, (int(panjang/2),lebar), radius=2, color=(0, 0, 255), thickness=10)
cv2.line(frame, (int(panjang/2),0), ((int(panjang/2),lebar)), (255, 0, 0), 2)
cv2.line(frame, (xkananatas,ykananatas), ((int(panjang/2),lebar)), (0, 0, 255), 1)
cv2.line(frame, (xkiriatas,ykiriatas), ((int(panjang/2),lebar)), (0, 0, 255), 1)
cv2.line(frame, (xkananbawah,ykananbawah), ((int(panjang/2),lebar)), (0, 0, 255), 1)
cv2.line(frame, (xkiribawah,ykiribawah), ((int(panjang/2),lebar)), (0, 0, 255), 1)
#cv2.line(frame, (xkanantengan,ykanantengan), ((int(panjang/2),lebar)), (0, 0, 255), 1)
#cv2.line(frame, (xkiritengan,ykiritengan), ((int(panjang/2),lebar)), (0, 0, 255), 1)
#cv2.imshow('HueComp',hthresh)
#cv2.imshow('SatComp',sthresh)
#cv2.imshow('ValComp',vthresh)
#cv2.imshow('closing',closing)
cv2.imshow('tracking',frame)
k = cv2.waitKey(5) & 0xFF
if k == 27:
    break

cap.release()

cv2.destroyAllWindows()

```

Program Arduino Nano

```
int RPWM = 11;
int LPWM = 3;
int L_EN = 2;
int R_EN = 12;
String msg;

unsigned long interval = 1000; // the time we need to wait
unsigned long previousMillis = 0;
void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  pinMode(RPWM, OUTPUT);
  pinMode(LPWM, OUTPUT);
  pinMode(R_EN, OUTPUT);
  pinMode(L_EN, OUTPUT);

  digitalWrite(R_EN, HIGH);
  digitalWrite(L_EN, HIGH);
}

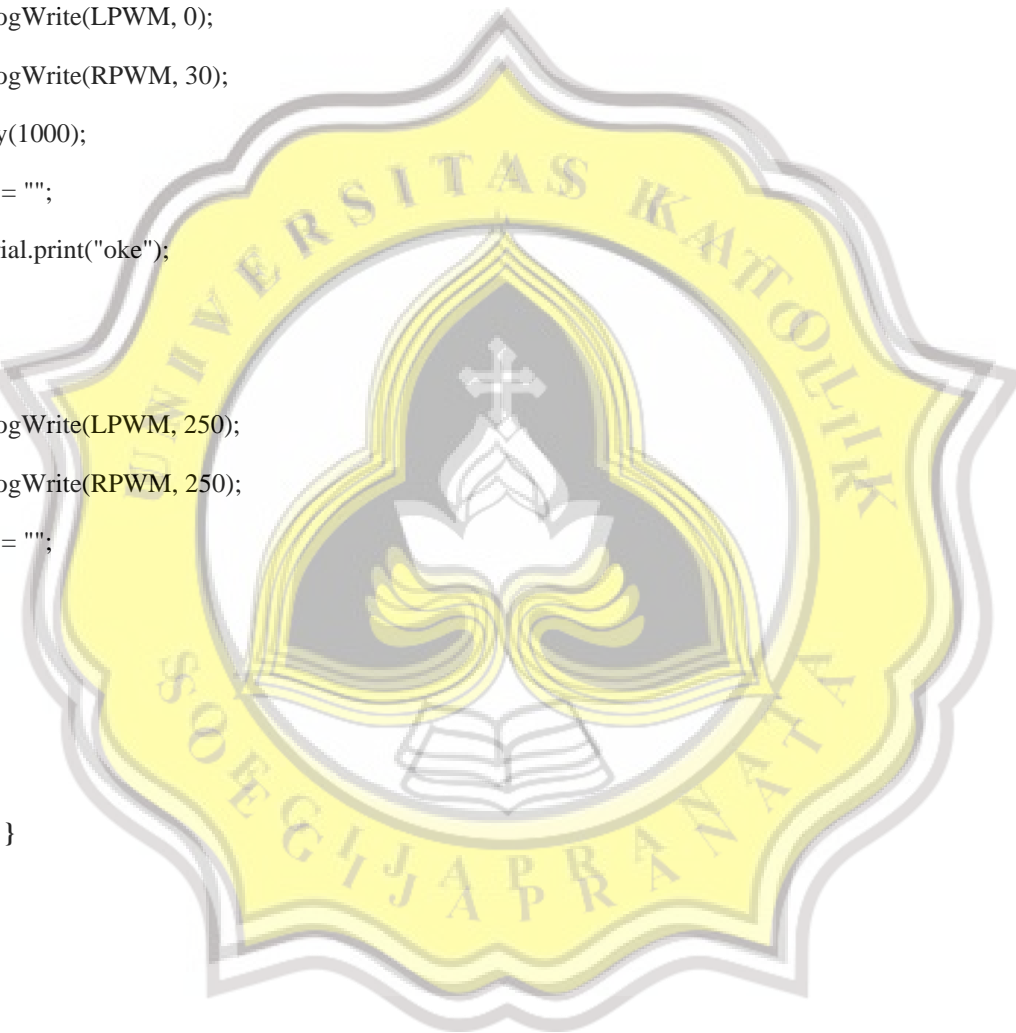
void loop() {
  // forward
  unsigned long currentMillis = millis();
  if ((unsigned long)(currentMillis - previousMillis) >= interval) {
    int nilai_sensor = analogRead(A5);
    Serial.println(nilai_sensor);
    previousMillis = millis();
  }
}
```



```
if (Serial.available() > 0) {  
  while (Serial.available() > 0) {  
    msg += (char)Serial.read();  
  }  
}
```

```
if (msg == "m") {  
  analogWrite(LPWM, 0);  
  analogWrite(RPWM, 30);  
  delay(1000);  
  msg = "";  
  // Serial.print("oke");  
}  
else {  
  analogWrite(LPWM, 250);  
  analogWrite(RPWM, 250);  
  msg = "";  
}  
}
```

```
}
```



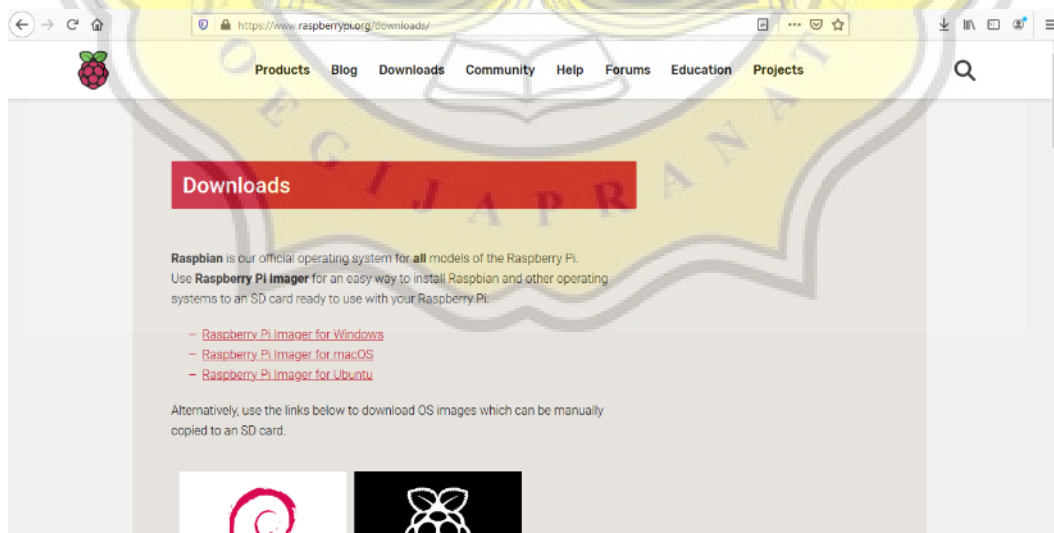
Langkah-langkah Menginstal Raspberry Pi

Di bawah ini adalah langkah-langkah untuk menginstal Raspbian Buster OS, yaitu:

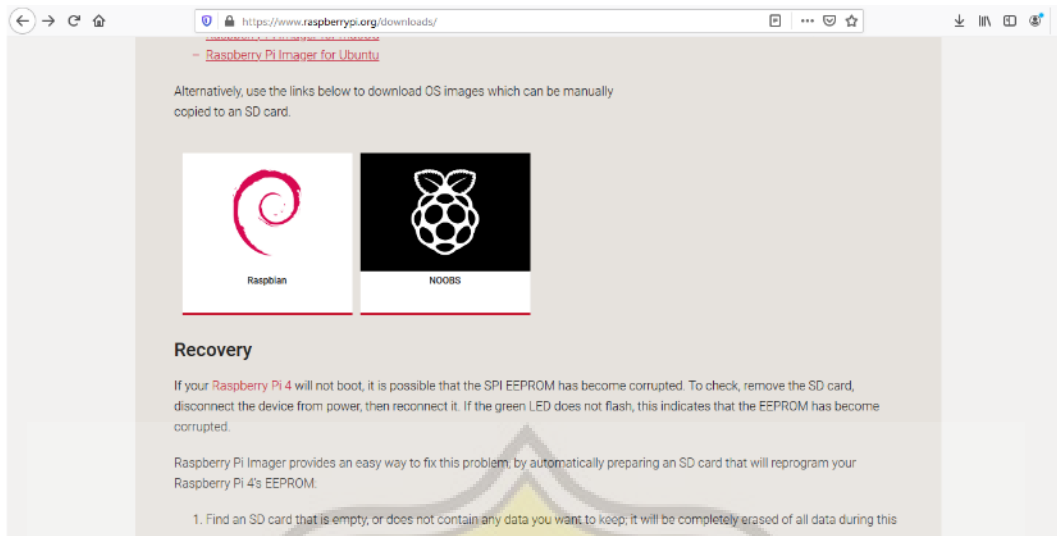
- a. Temukan dan buka situs resmi Raspberry Pi. Alamat web atau link ditunjukkan pada Gambar di bawah ini.



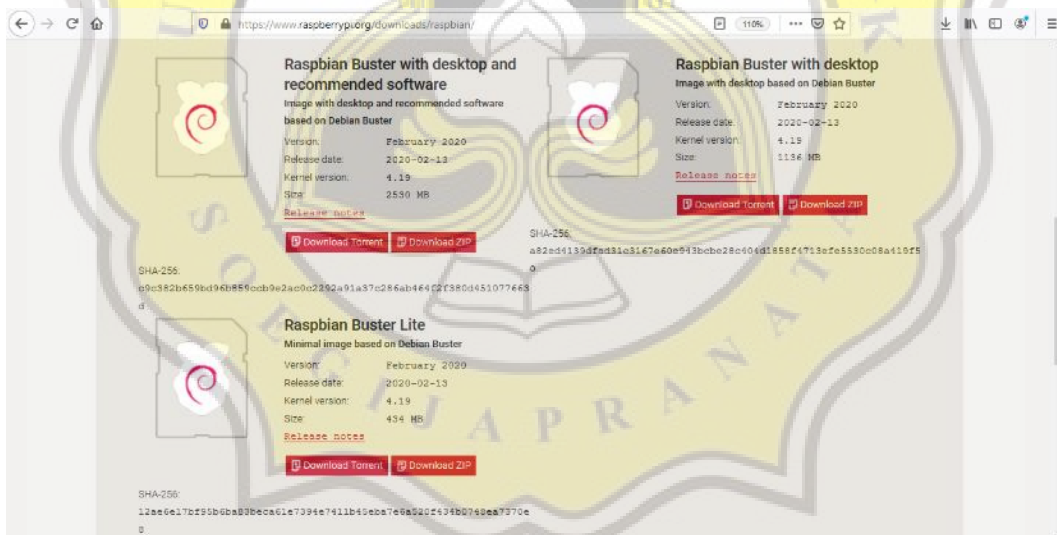
- b. Kemudian klik "Unduhan" di bilah menu yang tersedia. Kemudian muncul tampilan seperti pada gambar di bawah ini.



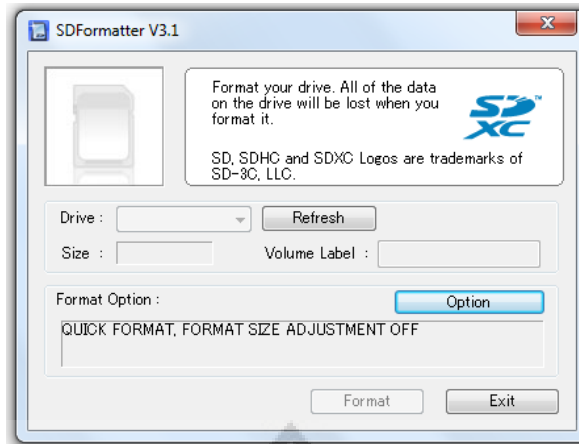
- c. Kemudian scrol melalui tampilan. Klik menu tool "Raspbian" dengan logo spiral berwarna merah, seperti terlihat pada gambar di bawah ini.



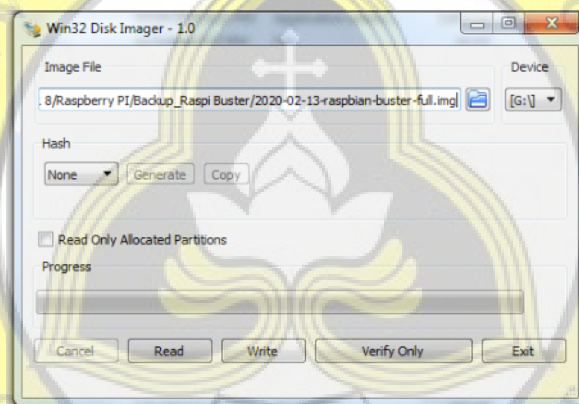
d. Saat terbuka, Anda akan melihat tiga opsi perangkat lunak dengan fungsi berbeda. Pada penelitian ini telah terinstall menu “*Raspbian Buster with Desktop and Recommended Software*”. Kemudian klik "Unduh ZIP" di menu. Yang bisa dilihat pada gambar di bawah ini.



e. Setelah diunduh, langkah selanjutnya adalah mengekstrak file di PC, yang kemudian disalin ke kartu SD 64GB. Sebelum menyalin, kartu SD harus diformat menggunakan formatter SD seperti yang ditunjukkan pada gambar di bawah ini.



Setelah memformat dan menyalin ke kartu SD di aplikasi Win32DiskImager, tulis ke file. Kemudian buka folder "raspbianbusterfull.img" di file tersebut. Kemudian klik "write" seperti terlihat pada gambar di bawah ini.



f. Saat proses write selesai, masukkan kartu SD ke dalam Raspberry Pi. Nyalakan Raspberry Pi dan sambungkan ke layar monitor, yang terhubung dengan kabel VGA ke Micro HDMI. Tujuannya untuk mengetahui apakah mekanisme operasi Raspbian Buster dapat bekerja dengan baik.

Tahapan cara Install Python versi terbaru di Raspberry

petama yang dapat dilakukan adalah mengecek apakah Python default Anda ada pada Python 3.x atau pada Python 2.x, untuk memastikan hal ini gunakan command sebagai berikut pada bagian terminal :

```
$ python --version
```

Beberapa hal perlu dilakukan untuk mengubah Python 2.7 menjadi Python 3.6. Hal pertama yang perlu diperhatikan adalah memastikan bahwa beberapa package dependencies sudah di-install pada Raspberry Pi. Pastikan dengan menggunakan command sebagai berikut:

```
# apt-get update

# apt-get install build-essential tk-dev libncurses5-dev libncursesw5-dev libreadline6-dev
libdb5.3-dev libgdbm-dev libsqlite3-dev libssl-dev libbz2-dev libexpat1-dev liblzma-dev
zlib1g-dev
```

Setelah melaksanakan beberapa program tersebut, Anda dapat langsung mengunduh source code Python 3.6 dari website resmi Python. Source code Python 3.6 yang tersedia berupa tar.xz sehingga anda perlu melakukan proses unpack. Sebelum mengunduh dengan wget, pastikan Anda sudah di direktori /tmp/ agar source code yang Anda unduh akan terhapus secara otomatis setelah reboot.

```
$ cd /tmp/
```

```
$ wget https://www.python.org/ftp/python/3.6.4/Python-3.6.4.tar.xz
```

```
$ tar xf Python-3.6.4.tar.xz
```

Setelah Anda berhasil mengekstrak seperti pada Gambar 2, Anda dapat melihat bahwa source code Python 3.6.4 sudah berada pada /tmp/Python-3.6.4. Untuk melanjutkan proses instalasi Python 3.6.4, akses direktori Python-3.6.4

```
$ cd Python-3.6.4
```

Setelah membuka direktori, ikuti proses ini untuk instalasi Python 3.6.4:

```
$ ./configure
```

```
$ make
```

```
# make altinstall
```

Setelah instalasi Python 3.6, Anda akan melihat bahwa Python default masih seperti Python sebelumnya, untuk saya Python masih 2.7.13. Untuk mengubah default Python yang digunakan perlu digunakan update-alternatives. Pertama, pastikan Anda tahu posisi instalasi Python 3.6 dan bahwa Python 3.6 yang sudah diinstal dapat diakses. Lakukan kedua command ini untuk mengetahui posisi Python 3.6

```
$ ls /usr/bin/python*
```

```
$ ls /usr/local/bin/python*
```

Setelah mengetahui posisi executable Python 3.6, lakukan update-alternatives untuk menambahkan Python 2.7.13 dan Python 3.6.4 sebagai alternatif Python. Gambar 3 menunjukkan hasil keluaran dari command untuk mengetahui posisi executable dari Python2.7, Python3.5 dan Python3.6. Tambahkan semua executable Python ini sebagai alternatif dari Python. Setelah mengetahui posisi yakni pada /usr/bin/ untuk Python2.7 dan Python3.5 dan pada /usr/local/bin/ pada Python3.6 yang baru diinstal. Lakukan proses penambahan alternatif /usr/bin/python dengan menggunakan command sebagai berikut:

```
# update-alternatives --install /usr/bin/python python /usr/bin/python2.7 1
```

```
# update-alternatives --install /usr/bin/python python /usr/bin/python3.5 2
```

```
# update-alternatives --install /usr/bin/python python /usr/local/bin/python3.6 3
```

Langkah-langkah Menginstal Library OpenCV

1. Pastikan laptop terkoneksi pada internet
2. Masukkan perintah di bawah ini satu persatu

```
sudo apt install cmake build-essential pkg-config git
sudo apt install libjpeg-dev libtiff-dev libjasper-dev libpng-dev libwebp-dev libopenexr-dev
sudo apt install libavcodec-dev libavformat-dev libswscale-dev libv4l-dev libxvidcore-dev
libx264-dev libdc1394-22-dev libgstreamer-plugins-base1.0-dev libgstreamer1.0-dev
sudo apt install libgtk-3-dev libqtgui4 libqtwebkit4 libqt4-test python3-pyqt5
sudo apt install libatlas-base-dev liblapacke-dev gfortran
sudo apt install libhdf5-dev libhdf5-103
sudo apt install python3-dev python3-pip python3-numpy
```

3. Untuk langkah selanjutnya kita harus memperbesar 'swap file' di memory card
Raspi untuk memudahkan proses instalasi dari 100 menjadi 2048.

```
Sudo nano /etc/dphys-swapfile
CONF_SWAPSIZE=2048
```

untuk menyimpan setingan di atas tekan CTRL-X, pilih Y (yes) kemudian enter.
Jangan lupa restart untuk memastikan swap file sudah berubah.

```
sudo systemctl restart dphys-swapfile
```

4. Berikut ikuti langkah di bawah ini satu persatu

```
git clone https://github.com/opencv/opencv.git
git clone https://github.com/opencv/opencv_contrib.git
mkdir ~/opencv/build
cd ~/opencv/build
cmake -D CMAKE_BUILD_TYPE=RELEASE \
-D CMAKE_INSTALL_PREFIX=/usr/local \
```

```
-D OPENCV_EXTRA_MODULES_PATH=~/.opencv_contrib/modules \  
-D ENABLE_NEON=ON \  
-D ENABLE_VFPV3=ON \  
-D BUILD_TESTS=OFF \  
-D INSTALL_PYTHON_EXAMPLES=OFF \  
-D OPENCV_ENABLE_NONFREE=ON \  
-D CMAKE_SHARED_LINKER_FLAGS=-latomic \  
-D BUILD_EXAMPLES=OFF .  
make -j$(nproc)  
sudo make install  
sudo ldconfig
```

Proses instalasi paket OpenCV sudah selesai. Anda dapat mengecek apakah instalasi berhasil dengan perintah sbb:

```
python  
import cv2  
cv2.__version__
```

Biasanya pada project computer vision dengan OpenCV membutuhkan library tambahan yakni 'opencv-contrib-python'. Silahkan ditambahkan sekalian.

```
pip install opencv-contrib-python
```

Terakhir, kembalikan swap file menjadi bernilai 100

```
sudo nano /etc/dphys-swapfile
```

```
CONF_SWAPSIZE=100
```

Restart untuk memastikan

```
sudo systemctl restart dphys-swapfile
```


Notifications

**[INFOTEL] Editor Decision**

2022-09-01 10:26 AM

Florentinus Budi Setiawan, Eric Pratama Putra, Leonardus Heru Pratomo, Slamet Riyadi:

Letter of Acceptance (LoA)

=====

On behalf of the Editor, we are pleased to inform that your submission to JURNAL INFOTEL, "Implementation of line detection self-driving car using HSV method based on raspberry pi 4" is **Accepted**

Here are some important things we would like you to do in relation to manuscript acceptance:

1. Please kindly complete the payment by below scheme
 - The cost paid by Author with Indonesia citizenship is Rp 1,500,000 (IDR)
 - The cost paid by non-Indonesia citizenship is \$100 (USD)

The payment should be transferred to the following

Bank Name : Bank BNI 46
Account Holder : Institut Teknologi Telkom Purwokerto
Holder Address : Jl. D.I. Panjaitan No. 128, Purwokerto
Account Number : 1282222128

2. Confirm your payment through our email on: infotel@st3telkom.ac.id by uploading the scan of the receipt of payment and put remarks for any details we need to concern. with Subject: PAYMENT-[INFOTEL_VOLUMEXX_NUMBERXX_YOURNAME]

3. Fill in the Copyright Transfer Form document and send the document along with scan of the receipt of payment. [Download Copyright Transfer Form](#)

4. Please send all **high-resolution figure files** of your article to email infotel@st3telkom.ac.id

We thank you a lot for your participation and again congratulate for your achievement, we are looking forward to seeing you at the next issue

Eko Fajar Cahyadi
Institut Teknologi Telkom Purwokerto
ekofajarcahyadi@ittelkom-pwt.ac.id

Notifications

**[INFOTEL] Editor Decision**

2022-09-01 10:26 AM

Florentinus Budi Setiawan, Eric Pratama Putra, Leonardus Heru Pratomo, Slamet Riyadi:

Letter of Acceptance (LoA)

On behalf of the Editor, we are pleased to inform that your submission to JURNAL INFOTEL, "Implementation of line detection self-driving car using HSV method based on raspberry pi 4" **is Accepted**

Here are some important things we would like you to do in relation to manuscript acceptance:

1. Please kindly complete the payment by below scheme
 - The cost paid by Author with Indonesia citizenship is Rp 1,500,000 (IDR)
 - The cost paid by non-Indonesia citizenship is \$100 (USD)

The payment should be transferred to the following

Bank Name : Bank BNI 46
Account Holder : Institut Teknologi Telkom Purwokerto
Holder Address : Jl. D.I. Panjaitan No. 128, Purwokerto
Account Number : 1282222128

2. Confirm your payment through our email on: infotel@st3telkom.ac.id by uploading the scan of the receipt of payment and put remarks for any details we need to concern, with Subject: PAYMENT-[INFOTEL_VOLUMEXX_NUMBERXX_YOURNAME]
3. Fill in the Copyright Transfer Form document and send the document along with scan of the receipt of payment. [Download Copyright Transfer Form](#)
4. Please send all **high-resolution figure files** of your article to email infotel@st3telkom.ac.id

We thank you a lot for your participation and again congratulate for your achievement, we are looking forward to seeing you at the next issue

Eko Fajar Cahyadi
Institut Teknologi Telkom Purwokerto
ekofajarcahyadi@ittelkom-pwt.ac.id

PAPER NAME

TA-18.F1.0033.docx

WORD COUNT

4620 Words

CHARACTER COUNT

29094 Characters

PAGE COUNT

33 Pages

FILE SIZE

64.9KB

SUBMISSION DATE

Oct 12, 2022 11:34 AM GMT+7

REPORT DATE

Oct 12, 2022 11:34 AM GMT+7

● **16% Overall Similarity**

The combined total of all matches, including overlapping sources, for each database.

- 14% Internet database
- 1% Publications database
- Crossref database
- Crossref Posted Content database
- 8% Submitted Works database



Summary

