

# Sistem Pengaturan Lampu Lalu Lintas Menggunakan Pengolahan Citra Digital

Fariz Muhammad Rachmatullah<sup>1</sup>, Gita Indah Hapsari<sup>2</sup>, Lisda Meisaroh<sup>3</sup>

1,2,3 Prodi D3 Teknologi Komputer, Fakultas Ilmu Terapan, Universitas Telkom

farizmuhammadr@student.telkomuniversity.ac.id, 2 gitaindahhapsari@tass.telkomuniversity.ac.id 3

lisdameisaroh@tass.telkomuniversity.ac.id

## Abstrak

Prototipe ini bertujuan untuk membuat sistem pendeteksi antrian kendaraan yang menimbulkan terjadinya kemacetan lalu lintas di suatu perkotaan. Sistem berfungsi untuk mengatur lampu lalu lintas berdasarkan jumlah kendaraan. dalam prototipe yang dirancang dapat menyelesaikan masalah kemacetan lalu lintas di perkotaan. Pengendalian kemacetan lalu lintas yang baik dan optimal tidak mungkin dilakukan tanpa menggunakan sistem pengelolaan data yang cerdas dan realtime. Melalui prototipe ini, kita mampu meningkatkan kualitas pengendalian kemacetan baik dalam aspek strategis maupun operasional. Prototipe dilakukan dengan menggunakan metode pengolahan citra digital.

**Kata Kunci:** CCTV, Raspberry Pi, Pengaturan lalu lintas, kemacetan lalu lintas.

## Abstract

*This prototype aims to create a vehicle queue detection system that causes traffic jams in an urban area. The system functions to read traffic lights based on the number of vehicles. in a prototype designed to solve the problem of traffic jams in urban areas. Good and optimal traffic jam control cannot be done without using an intelligent and realtime data management system. Through the prototype, we are able to improve the quality of congestion control in both strategic and operational aspects. The prototype was carried out using digital image processing methods.*

**Keywords:** CCTV, Raspberry Pi, traffic management, traffic jam.

## BAB 1. PENDAHULUAN

### 1.1 Latar Belakang

Lampu lalu lintas merupakan rambu terpenting yang ada di persimpangan jalan raya. Tanpa keberadaannya,

kendaraan yang berbagai arah akan berpotensi kecelakaan di satu titik tertentu. Melansir laman carscoops, seperti dikutip Rabu 16 Oktober 2019, kota yang pertama kali menggunakan lampu lalu lintas adalah London, Inggris. Kala itu, di awal tahun 1960-an, kawasan tersebut ramai dengan kereta kuda dan orang yang melintas. Lampu ini yang mengendalikan arus lalu lintas yang terpasang di persimpangan jalan, lampu ini menentukan kapan kendaraan harus berjalan dan kapan harus berhenti secara bergantian[1].

Lampu lalu lintas (menurut UU no 22/2009 tentang lalu lintas dan angkutan jalan) alat pemberi isyarat lalu lintas atau APILL. Saat ini telah banyak jenis sistem lampu lalu lintas salah satunya yaitu digerakkan oleh manusia (Manual Actuated Traffic Light Controller) dan ada juga kontrol dengan waktu tetap (Pre-timed) kontrol ini mempunyai pengaturan waktu hijau yang sudah ditentukan[2].

Alat ini sebagai salah satu teknologi *monitoring* dan mengurangi kemacetan dengan menggunakan sebuah *mikrokontroler* dengan pemrograman sebuah aplikasi *python* dan sistem otomatis penggunaan pada lampu lalu lintas untuk mengurangi kepadatan di persimpangan lampu lalu lintas tersebut.

### 1.2 Rumusan Masalah

Berdasarkan uraian di atas, maka rumusan masalah untuk proyek akhir ini adalah sebagai berikut:

1. Bagaimana membuat sistem deteksi antrian dengan menggunakan pengolahan citra?
2. Bagaimana mengatur sistem lampu lalu lintas berdasarkan panjangannya antrian kendaraan?

### 1.3 Tujuan

Adapun tujuan dari proyek akhir ini adalah:

1. Membuat prototipe sistem pendeteksi antrian kendaraan dengan menggunakan pengolahan citra.
2. Mengatur sistem lampu lalu lintas berdasarkan jumlah kendaraan terbanyak.

### 1.4 Batasan Masalah

Batasan masalah yang ada adalah sebagai berikut:

1. Kendaraan yang diujikan yaitu kendaraan roda empat.
2. Alat masih berupa prototipe yang menggunakan python, raspberry pi 4, raspberry pi camera v2.
3. Informasi yang diberikan jumlah antrian yang ada di empat simpang.
4. Penyalan lampu hijau 1 per 1 pada jalan.

## BAB 2. TINJAUAN PUSTAKA

### 2.1 Penelitian Sebelumnya

1. Dari Muhammad Taufiq Hidayat (Alauddin Makassar, 2016) menjelaskan sebuah prototipe lampu lalu lintas yang menggunakan metode white box testing. White box testing adalah rancangan pengujian menggunakan struktur kontrol yang salah satunya caranya adalah Cyclomatic Complexity, yaitu suatu matrix software yang menetapkan ukuran kompleksitas logika

program yang dapat terjamin secara keseluruhan independent path yang ada di dalam modul dikerjakan minimal satu kali.

2. Oleh Dewi Putri Suryani (Telkom University, 2016) , merancang sebuah sistem kontrol lampu rumah menggunakan raspberry pi berbasis web atau biasa disebut web server, dimana web tersebut nantinya akan difungsikan sebagai system kontrol atau *remote*. Dengan fungsi yang sama sebagai otomatis. Dengan menggunakan perangkat mobile seperti handphone yang dapat aplikasi browser sehingga dapat membuka web server tersebut[5].

3. TESLA, Jurnal Teknik Elektro (2007) merancang sistem pengaturan lampu merah lalu lintas secara sentral dan dikontrol dari jarak jauh keseluruhan.
4. Reza Sulistyowati, Giva Andriana Mutiara, periyadi (Telkom University, 2016) membuat sistem keamanan parkir berdasarkan IP address pada kendaraan yang terparkir.

### 2.2 Pengutipan Teori dari Daftar Pustaka

#### 2.2.1 webcam

Webcam adalah perangkat yang berupa kamera digital yang dihubungkan ke laptop/komputer.

Webcam dapat mengirimkan gambar-gambar secara live dari manapun ia dipasangkan dengan bantuan internet. Webcam dapat dilihat pada Gambar 2.1.



Gambar 2.1 Webcam

Spesifikasi dari Webcam dapat dilihat pada Tabel 2.1.

Tabel 2.1 Spesifikasi Webcam

Frame Rate	30FPS
Lens	Full HD Glass
Resolution	1920x1080 (1080p)
Lens feature	Auto Expsure White Balance
Frame rate	640 x 480 pixels / up to 15fps (VGA), 320 x 240 pixels / up to 30fps (CIF)
Picture control	Auto Color Compensation, Auto Brightness Adjustment
Focus type	Auto Focus
Camera Rod	Full 360 Degree Rotation

### 2.2.1 Raspberry Pi Type 3

Raspberry Pi merupakan komputer *single-board* yang dikembangkan oleh *Raspberry Pi Foundation*. Yang bisa juga digunakan untuk menjalankan program perkantoran, *games* pada komputer dan juga pemutar video media yang bisa beresolusi tinggi. Raspberry dibagi menjadi 2 tipe yaitu *Raspberry* model A dan *Raspberry* model B. Yang membedakan dari 2 tipe tersebut adalah kapasitas penyimpanan RAM. Pada *Raspberry* model A hanya terdapat 256MB dan *Raspberry* model B 512MB. *Raspberry Pi Type 3* dapat dilihat pada Gambar 2.2.



Gambar 2. 2 Raspberry Pi 3

Spesifikasi dari *Raspberry Pi 3* dapat dilihat pada Tabel 2.2.

Tabel 2.2 Spesifikasi *Raspberry Pi 3*

SoC	Broadcom BCM2837
CPU	4x ARM Cortex-A53, 1.2GHz
GPU	Broadcom VideoCore IV
RAM	1GB LPDDR2 (900 MHz)
Networking	10/100 Ethernet, 2.4GHz 802.11n wireless

Bluetooth	Bluetooth 4.1 Classic, Bluetooth Low Energy
GPIO	40-pin header, populated
Ports	HDMI, 3.5mm analogue audio-video jack, 4x USB 2.0, Ethernet, Camera Serial Interface (CSI), Display Serial Interface (DSI)



Gambar 2.1 Arduino Nano[5]

**2.2.1 Lampu Lalu Lintas**

adalah dioda yang dapat memancarkan cahaya pada saat mendapat arus bias maju (forward bias). LED (Light Emitting Dioda) dapat memancarkan cahaya karena menggunakan doping galium, arsenic dan fosforus. Jenis doping yang berbeda diata dapat menghasilkan cahaya dengan warna yang berbeda[6]. LED dapat dilihat pada Gambar 2.3.



Gambar 2. 3 LED

**2.2.1 Python**

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. Python diklaim sebagai

bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif[7]. Python juga didukung oleh komunitas yang besar. Python dapat dilihat pada Gambar 2.4.



Gambar 2. 4 Python

**2.2.1 LAN**

LAN adalah berfungsi sebagai menyambungkan antara laptop dengan raspberry pi.

**2.2.2 Jumper Female to Felame**

Jumper adalah berfungsi sebagai menyambungkan antara LED dengan Raspberry pi. Jumper dapat dilihat pada Gambar 2.5.



Gambar 2.5 Jumper

**2.2.3 Adaptor**

Adaptor adalah sebuah alat yang berfungsi untuk mengubang tegangan AC tinggi menjadi DC rendah. Adaptor banyak digunakan sebagai

catu daya seperti pada radio,amplifier dan perangkat elektronik lainnya. Adaptor dapat dilihat pada Gambar 2.6.



Gambar 2. 6 Adaptor

### 2.2.4 OpenCV

OpenCV adalah sebuah library sederhana terkait citra digital yang mempunyai fitur, antara lain : pengenalan wajah, pelacakan wajah, digunakan untuk mengolah gambar dan video hingga kita mampu meng-extrak informasi didalamnya[8]. *OpenCV* dapat dilihat pada Gambar 2.7.



Gambar 2. 7 OpenCV

### 2.2.5 Traffic Surveillance

Lalu lintas dengan menghitung dan mengklasifikasikan kendaraan yang diperoleh dari inputan video menggunakan image processing dengan metode Counting and

Classification of Vehicles from video using Image Processing.

### 2.2.6 Putty

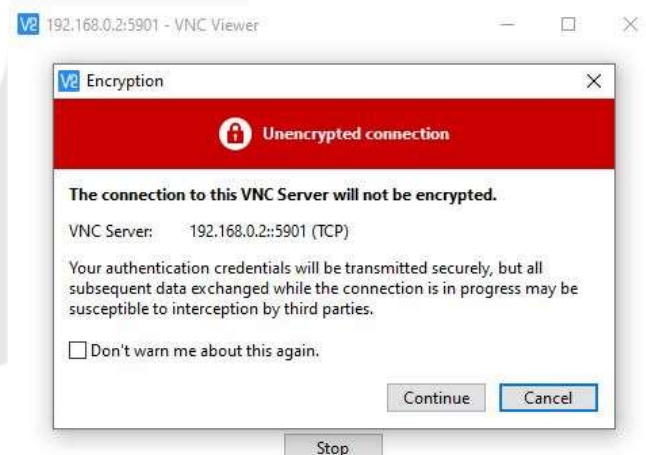
Putty adalah aplikasi remote access untuk menyambungkan raspberry pi ke VNC server. Putty dapat dilihat pada Gambar 2.8.



Gambar 2. 8 Putty

### 2.2.7 VNCserver

VNC server adalah sebagai client untuk membuka sistem operasi raspberry pi. VNCserver dapat dilihat pada Gambar 2.9.

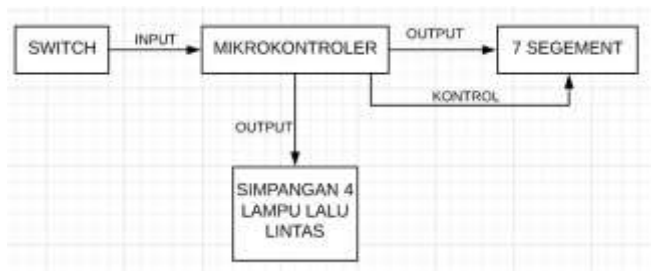


Gambar 2. 9 VNCserver

**BAB 3. ANALISIS DAN PERANCANGAN**

**3.1 Gambaran Sistem Saat Ini (atau Produk)**

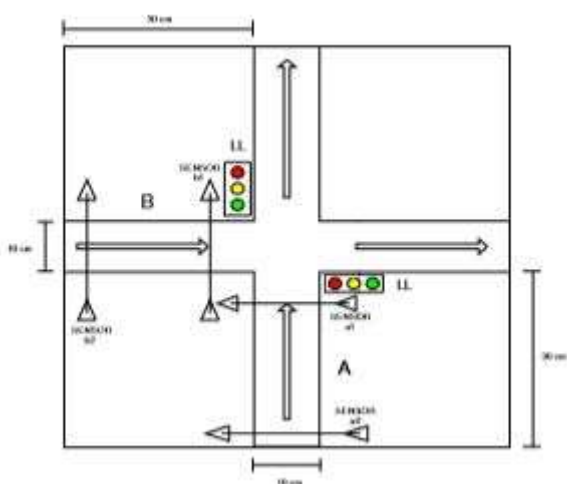
Metode logika fuzzy digunakan untuk menentukan lamanya waktu lampu lalu lintas menyala sesuai dengan volume kendaraan yang sedang mengantri pada sebuah persimpangan[2]. Gambar 3.1.1 merupakan sistem CCTV yang digunakan pada sistem saat ini.



Gambar 3.1.1 Sistem Lampu Lalu Lintas yang digunakan saat ini

**3.1.1 Blok Diagram / Topologi Sistem**

pada bagian input alat yang sudah dinyalakan akan secara otomatis maka alat akan mencari panjangnya antrian kendaraan yang padat untuk diurutkan jalur mana yang akan pertama kali lampu hijau dan diurutkan dari yang terpanjang.



Gambar 3.1.2 Blok Diagram Sistem Saat Ini

**3.1.2 Cara Kerja Sistem**

Pengaturan waktu lampu lalu lintas tetap(Fixed Time Operation) dalam pengoperasiannya menggunakan waktu siklus dan Panjang fase yang diatur terlebih dahulu dan dipertahankan untuk suatu periode tertentu. Panjang siklus dan fase adalah tetap selama interval tertentu. Sehingga tipe ini merupakan bentuk pengendalian lampu lalu lintas yang paling murah dan sederhana.

**3.1.3 Analisis Kebutuhan Sistem (atau Produk)**

Analisa kebutuhan fungsional dapat dilihat pada Tabel 3.1.1 dan analisa kebutuhan non fungsional dapat dilihat pada Tabel 3.1.2

Tabel 3.1 Kebutuhan Fungsional

No.	Kebutuhan Fungsional
1	Sistem dapat merekam dalam keadaan pagi maupun malam hari.
2	Sistem dapat menghitung kendaraan.

Tabel 3.2 Kebutuhan Non Fungsional

No.	Kebutuhan Non Fungsional
1	Resolusi video pada sistem maksimal sampai 1080p.
2	Sistem harus dapat memastikan bahwa data yang digunakan dalam sistem harus terlindungi dari akses yang tidak berwenang.

### 3.1 Perancangan

#### 3.2.1 Gambaran Sistem Usulan

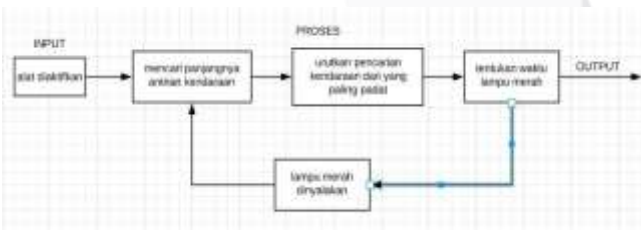
user untuk mengaktifkan alatnya agar camera bisa menangkap panjangnya antrian yang akan diproses oleh raspberry pi dengan halis keluarannya LED dan timer. Gambar 3.2.1 merupakan sistem usulan Raspberry Pi sebagai server.



Gambar 3.2.1 Gambar Sistem Usulan

#### 3.2.2 Blok Diagram / Topologi Sistem

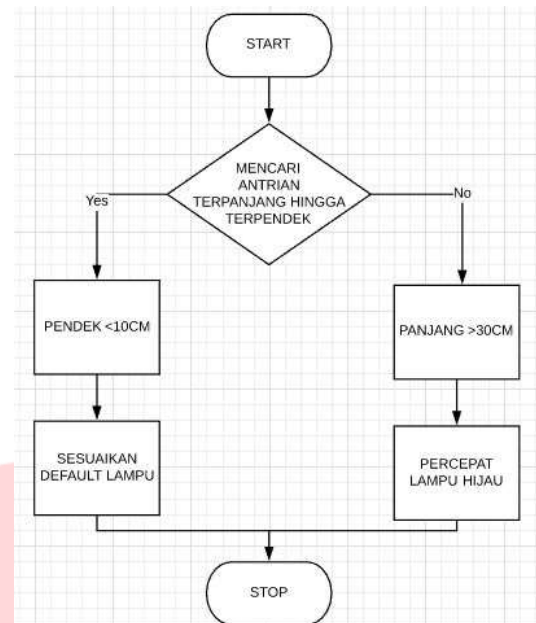
Topologi Sistem Lampu Lalu Lintas *Raspberry* sebagai server. Webcam untuk disambungkan langsung ke *raspberry* lalu ditampilkan hasil kendaraan yang diubah. Gambar 3.2.2 merupakan topologi sistem usulan.



Gambar 3.2.2 Topologi Sistem Usulan

#### 3.2.3 Flowchart Sistem Usulan

Flowchart sistem usulan dapat di lihat pada Gambar 3.2.3.



Gambar 3.2.3 Flowchart Sistem Usulan saat ini

#### 3.2.4 Cara Kerja

Camera yang sudah terhubung pada raspberry pi berfungsi sebagai alat yang dapat mendeteksi kendaraan. Python yang merancang sistem otomatisnya agar diurutkan. Raspberry pi akan berperan sebagai server dan media penyimpanan. Dan lampu lalu lintas sebagai pemberitahuan berhenti dan jalannya kendaraan.

#### 3.2.5 Spesifikasi Sistem

##### 3.2.5.1 Perangkat Keras

Pada Tabel 3.2.4 adalah daftar kebutuhan perangkat keras.

Tabel 3.2.4 Kebutuhan Perangkat Keras

No	Perangkat keras	Spesifikasi
1	Webcam	<ul style="list-style-type: none"> <li>• 1080p</li> <li>• 30FPS</li> <li>• Auto Focus</li> <li>• Video Light Correction</li> </ul>

		<ul style="list-style-type: none"> <li>• Auto Exposure White Balance</li> </ul>			<ul style="list-style-type: none"> <li>• 40-pin connector</li> </ul>
2	Raspberry Pi 3	<ul style="list-style-type: none"> <li>• Broadcom BCM2711 processor (1.5 GHz quad-core Arm Cortex-A72 CPU)</li> <li>• VideoCore 6 GPU</li> <li>• 1GB, 2GB, or 4GB of RAM</li> <li>• microSD card reader for storage</li> <li>• 2 x USB 3.0 ports</li> <li>• 2 x USB 2.0 ports</li> <li>• 1 x USB Type-C port (for power)</li> <li>• 2 x micro HDMI 2.0 ports</li> <li>• Gigabit Ethernet</li> <li>• 3.5mm audio jack</li> <li>• 802.11ac WiFi</li> <li>• Bluetooth 5.0</li> </ul>	4	LED	<ul style="list-style-type: none"> <li>• <i>Led merah</i></li> <li>• <i>Led kuning</i></li> <li>• <i>Led hijau</i></li> </ul>
			5	Laptop	<ul style="list-style-type: none"> <li>• <i>Intel Core i5 6200U 2.3Ghz up to 2.8GHz (3MB Cache)</i></li> <li>• <i>4GB On Board &amp; 8GB RAM DDR4 (upgrade)</i></li> <li>• <i>1000GB HDD 5400RPM</i></li> <li>• <i>Intel HD Graphics 520 &amp; Nvidia GeForce GT930MX 2GB DDR3</i></li> <li>• <i>Windows 10 Home SL 64bit</i></li> <li>• <i>14" inch HD LED Display</i></li> <li>• <i>DVD RW SuperMulti Drive</i></li> <li>• <i>Sonic Master</i></li> <li>• <i>HD Webcam, Card Reader,</i></li> </ul>



		<p><i>WiFi, HDMI, Bluetooth V4.0</i></p> <ul style="list-style-type: none"> <li>• <i>Gigabit LAN, Headphone Jack, Microphone,</i></li> </ul>
--	--	----------------------------------------------------------------------------------------------------------------------------------------------

		<p><i>USB 3.0 &amp; USB Type C 3.1</i></p> <ul style="list-style-type: none"> <li>• <i>2.0kg included 4 Cells Battery</i></li> </ul>
--	--	--------------------------------------------------------------------------------------------------------------------------------------

### 3.2.5.2 Perangkat Lunak

Pada Tabel 3.2.5 adalah daftar kebutuhan perangkat lunak.

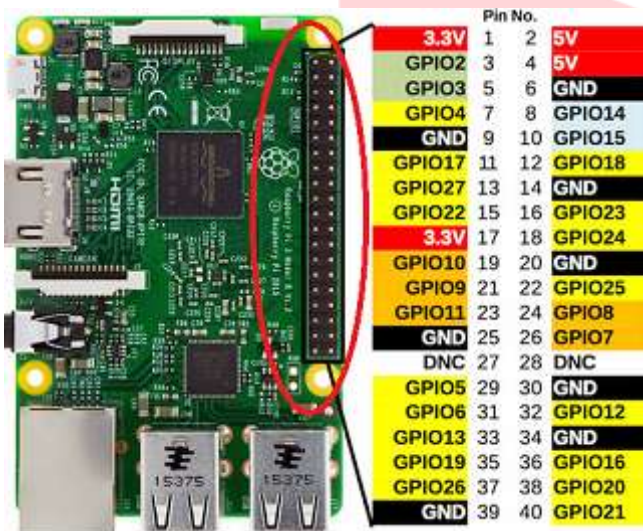
Tabel 3.2.5 Kebutuhan Perangkat Lunak.

No	Perangkat lunak	Spesifikasi
1	Putty	<ul style="list-style-type: none"> <li>• Sebagai remote atau menghubungkan koneksi internet dari raspberry pi ke laptop.</li> </ul>
2	VNC sever	<ul style="list-style-type: none"> <li>• Sebagai membuka sistem operasi raspberry pi di windows.</li> </ul>
3	Python	<ul style="list-style-type: none"> <li>• <i>Texts editor</i> yang mendukung Bahasa pemrograman Jed, Vim, Nano dan Bahasa pemrograman lainnya.</li> <li>• Bersifat <i>open source</i>.</li> </ul>
4	OpenCV	<ul style="list-style-type: none"> <li>• <i>Library</i></li> </ul>

## BAB 4 IMPLEMENTASI DAN PENGUJIAN

### 4.1 Implementasi

Implementasi sistem dibuat berdasarkan topologi sistem usulan pada gambar 3.2.2 bahan yang diperlukan untuk merancang proyek akhir ini dengan mendownload dan menginstall terlebih dahulu putty dan vnc server untuk membuka sistem operasi raspberry pi dan menginstall python dan opencv.



Gambar 4.1.1 datasheet raspberry pi

Apa itu Pi gpio ? Pi gpio adalah pin untuk tambahan input output LED yang saya gunakan, gambar lingkaran merah, biasa disebut sebagai header GPIO.

#### 1. Antrian 1 (live)

GPIO.setup(2, GPIO.OUT) (hijau)

GPIO.setup(4, GPIO.OUT) (merah)

GND(9).

#### 2. Antrian 2

GPIO.setup(13, GPIO.OUT) (hijau)

GPIO.setup(26, GPIO.OUT) (merah)

GND(14).

#### 3. Antrian 3

GPIO.setup(14, GPIO.OUT) (hijau)

GPIO.setup(18, GPIO.OUT) (merah)

GND (39).

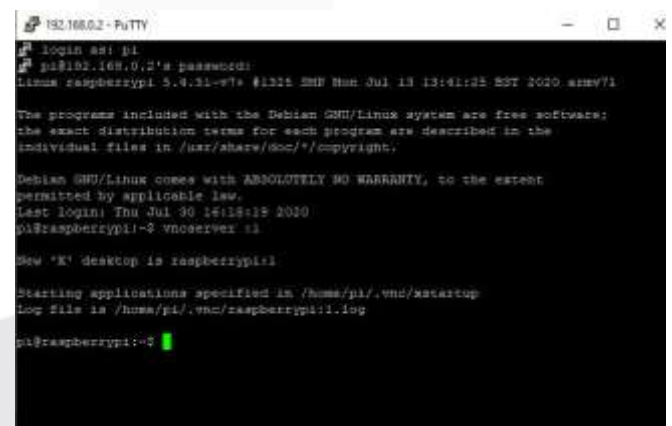
#### 4. Antrian 4

GPIO.setup(21, GPIO.OUT) (hijau)

GPIO.setup(16, GPIO.OUT) (merah)

GND (34).

Setelah jumper sudah terhubung, kemudian melakukan login dan masukkan perintah vnc server :1 agar bisa membuka vncserver yang dapat dilihat pada Gambar 4.1.2.

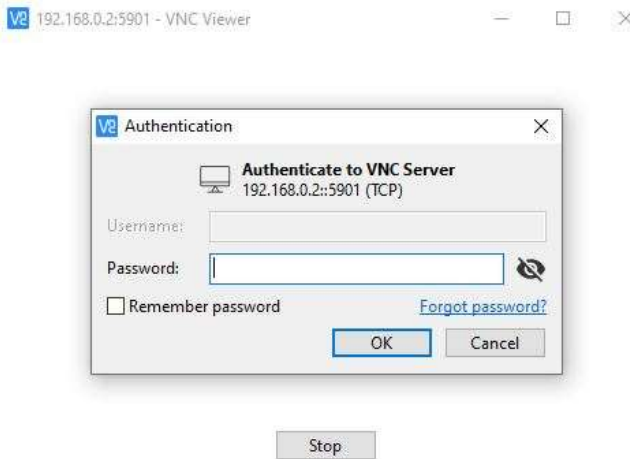


Gambar 4.1.2 login putty

Setelah melakukan perintah vncserver :1, kemudian buka vncserver lakukan login dan dapat dilihat pada Gambar 4.1.3.



Gambar 4.1.3 Buka aplikasi vncserver



Gambar 4.1.4 masukkan password

Kemudian membuka terminal dan download python dan opencv yang dapat dilihat pada Gambar 4.1.5.

```

pi@raspberrypi:~$ sudo pip3 install opencv-python
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Collecting opencv-python
  Downloading https://www.piwheels.org/simple/opencv-python/opencv_python-4.1.1.26-cp37-cp37m-linux_armv7l.whl (10.0MB)
    100% |#####| 10.0MB 29kB/s
Requirement already satisfied: numpy>=1.16.2 in /usr/lib/python3/dist-packages (from opencv-python) (1.16.2)
Installing collected packages: opencv-python
Successfully installed opencv-python-4.1.1.26
pi@raspberrypi:~$ sudo pip3 install opencv-contrib-python
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Collecting opencv-contrib-python
  Downloading https://www.piwheels.org/simple/opencv-contrib-python/opencv_contrib_python-4.1.1.26-cp37-cp37m-linux_armv7l.whl (13.9MB)
    100% |#####| 13.9MB 17kB/s
Requirement already satisfied: numpy>=1.16.2 in /usr/lib/python3/dist-packages (from opencv-contrib-python) (1.16.2)
Installing collected packages: opencv-contrib-python
Successfully installed opencv-contrib-python-4.1.1.26
    
```

Gambar 4.1.5 download python dan opencv

### 4.1 Pengujian

Pengujian pada penelitian ini dilakukan di tempat yang terang penerangan serta menggunakan objek

prototype kendaraan yang diubah banyaknya kendaraan. Ilustrasi objek yang digunakan pada percobaan ini dengan menggunakan mainan mobil. Pengujian terbagi menjadi 2 bagian yaitu, pengujian simpang yang live dan simpang yang menggunakan foto.



### 3.1.2 Pengujian Simpang Live

Tujuan utama pengujian simpang live adalah untuk mengetahui kendaraan yang terdeteksi.

#### 4.2.1.1 Hasil Pengujian Simpang Live

Hasil pengujian simpang live dapat di lihat pada Tabel 4.2.1

Jumlah kendaraan	Status	Gambar
2 kendaraan	Terdeteksi	
4 kendaraan	Terdeteksi	

6 kendaraan an	Terdeteksi	
8 kendaraan an	Terdeteksi	

**Tabel 4.2.1 Hasil Pengujian Sempang Live**

Saat pengujian dilakukan, objek di tambahkan cahaya agar lebih jelas tampilan kendaraannya. Hasil pengujian dapat dilihat pada Tabel 4.2.1.

**4.2.2 Pengujian Sempang Menggunakan Foto**

dikarenakan webcam yang saya gunakan hanya 1 makan sisa sempang yang saya gunakan menggunakan foto.

**4.2.2.1 Antrian 2**



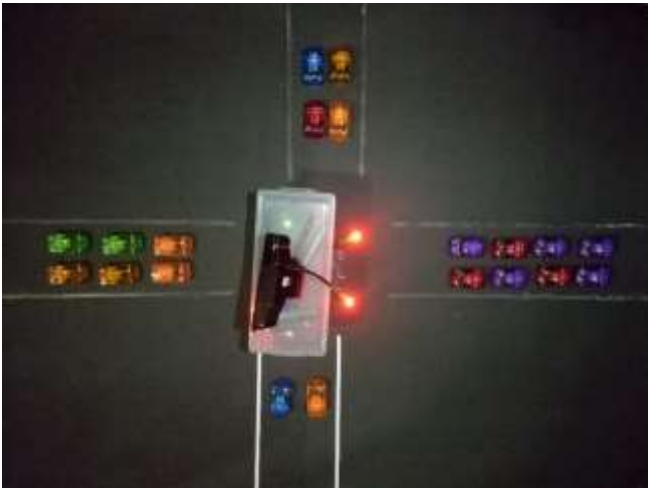
**4.2.2.2 Antrian 3**



**4.2.2.3 Antrian 4**



### 4.2.3 Tampilan Alat dan Program



Gambar 4.2.3.1 Tampilan Alat

#### 4.2.3.1 Pemograman Live

```
# simpangan 1, live
cap = cv2.VideoCapture(0)
ret, imgx = cap.read()
img1, dis1 = distance(imgx, True)
dis.append([1,dis1])
print("Panjang Antrian #1: ", dis1)
cv2.imwrite("antrian1.jpg",img1)
cap.release()
```

#### 4.2.3.2 Pemograman seluruhnya

```
import cv2
import time
import math
from operator import itemgetter
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM)
# LED merah
GPIO.setup(4, GPIO.OUT)
GPIO.setup(18, GPIO.OUT)
GPIO.setup(16, GPIO.OUT)
GPIO.setup(26, GPIO.OUT)
#LED kuning
GPIO.setup(3, GPIO.OUT)
GPIO.setup(15, GPIO.OUT)
GPIO.setup(19, GPIO.OUT)
GPIO.setup(20, GPIO.OUT)
# LED hijau
GPIO.setup(2, GPIO.OUT)
GPIO.setup(13, GPIO.OUT)
GPIO.setup(14, GPIO.OUT)
GPIO.setup(21, GPIO.OUT)

sudut = 30
```

```
while True:

    dis = []

    # simpangan 1, live
    cap = cv2.VideoCapture(0)
    ret, imgx = cap.read()
    img1, dis1 = distance(imgx, True)
    dis.append([1,dis1])
    print("Panjang Antrian #1: ", dis1)
    cv2.imwrite("antrian1.jpg",img1)
    cap.release()

    # simpangan 2
    img2, dis2 = distance("demo3.jpg")
    dis.append([2,dis2])
    print("Panjang Antrian #2: ", dis2)
    cv2.imwrite("antrian2.jpg",img2)

    # simpangan 3
    img3, dis3 = distance("demo.jpg")
    dis.append([3,dis3])
    print("Panjang Antrian #3: ", dis3)
    cv2.imwrite("antrian3.jpg",img3)

    # simpangan 4
    img4, dis4 = distance("demo4.jpg")
    dis.append([4,dis4])
    print("Panjang Antrian #4: ", dis4)
    cv2.imwrite("antrian4.jpg",img4)

    dis = sorted(dis, key=itemgetter(1), reverse=True)
    print(dis)
    cv2.imshow("antrian1",img1)
    cv2.imshow("antrian2",img2)
    cv2.imshow("antrian3",img3)
    cv2.imshow("antrian4",img4)

    durasi = [10,8,6,4] # urutan durasi
    simpangan = 0
    for m in dis:
        if m[0] == 1: # simpangan 1
            GPIO.output(2,GPIO.HIGH) # hijau
            GPIO.output(13,GPIO.LOW)
            GPIO.output(14,GPIO.LOW)
            GPIO.output(21,GPIO.LOW)
            GPIO.output(4,GPIO.LOW) # merah
            GPIO.output(18,GPIO.HIGH)
            GPIO.output(16,GPIO.HIGH)
            GPIO.output(26,GPIO.HIGH)

        elif m[0] == 2: # simpangan 2
            GPIO.output(2,GPIO.LOW) # hijau
            GPIO.output(13,GPIO.HIGH)
            GPIO.output(14,GPIO.LOW)
            GPIO.output(21,GPIO.LOW)
            GPIO.output(4,GPIO.HIGH) # merah
            GPIO.output(18,GPIO.HIGH)
            GPIO.output(16,GPIO.HIGH)
            GPIO.output(26,GPIO.LOW)

        elif m[0] == 3: # simpangan 3
            GPIO.output(2,GPIO.LOW) # hijau
            GPIO.output(13,GPIO.LOW)
```

```

GPIO.output(14, GPIO.HIGH)
GPIO.output(21, GPIO.LOW)
GPIO.output(4, GPIO.HIGH) # merah
GPIO.output(18, GPIO.LOW)
GPIO.output(16, GPIO.HIGH)
GPIO.output(26, GPIO.HIGH)
elif m[0] == 4: # simpangan 4
GPIO.output(2, GPIO.LOW) # hijau
GPIO.output(13, GPIO.LOW)
GPIO.output(14, GPIO.LOW)
GPIO.output(21, GPIO.HIGH)
GPIO.output(4, GPIO.HIGH) # merah
GPIO.output(18, GPIO.HIGH)
GPIO.output(16, GPIO.LOW)
GPIO.output(26, GPIO.HIGH)
if cv2.waitKey(durasi[simpangan]*1000) == :
break
simpangan += 1

```



Gambar 4.2.3.2. sudut

#### 4.2.3.3 Pemrograman port LED

```

import cv2
import time
import math
from operator import itemgetter
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM)
# LED merah
GPIO.setup(4, GPIO.OUT)
GPIO.setup(18, GPIO.OUT)
GPIO.setup(16, GPIO.OUT)
GPIO.setup(26, GPIO.OUT)
#LED kuning
GPIO.setup(3, GPIO.OUT)
GPIO.setup(15, GPIO.OUT)
GPIO.setup(19, GPIO.OUT)
GPIO.setup(20, GPIO.OUT)
# LED hijau
GPIO.setup(2, GPIO.OUT)
GPIO.setup(13, GPIO.OUT)
GPIO.setup(14, GPIO.OUT)
GPIO.setup(21, GPIO.OUT)

sudut = 30

```

Sudut 30 yang dimaksud adalah sudut antara garis merah dan kuning. hasil pengukuran jarak antara vertika, berdasarkan posisi koordinat piksel. Sementara antrian rill adalah yang kuning, jadi faktor sudut harus diperhitungkan. Yang dapat dilihat pada gambar 4.2.3.2.

## BAB 5 KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Sesuai dengan Rumusan Masalah dan Tujuan dari Proyek Akhir ini, maka kesimpulannya adalah :

1. Sensor Webcam dengan menggunakan metode Vehicles Counting dapat menentukan simpang mana yang paling banyak kendaraannya.

Terdapat kesimpulan baru yang didapatkan ketika melakukan pengujian alat yaitu:

1. Fitur dapat lebih banyak menangkap atau menghitung kendaran Ketika webcam yang digunakan lebih luas dan posisi kamera lebih tinggi.

### 5.1 Saran

Dari hasil pengujian terhadap proyek akhir ini dan untuk pengembangan lebih pada sistem proyek akhir ini disarankan:

1. Perancangan sistem yang tambahan lebih baik dengan mengganti kamera PTZ agar dapat

memaksimalkan video atau kualitas gambar yang dihasilkan.

2. Perlu alat untuk mempergerakan kamera dan zoom kamera agar bisa mengatur antrian kendaraan lebih panjang.

#### DAFTAR PUSTAKA

- [1] D. Hansen and Scribd.com, "prinsip kerja lampu lalu lintas.docx," *www.scribd.com*. <https://www.scribd.com/doc/258874283/prinsip-kerja-lampu-lalu-lintas-docx>.
- [2] bagus tris AtmajaNRP and Researchgate.net, "IMPLEMENTASI SISTEM FUZZY UNTUK PENGATURAN LAMPU LALU LINTAS BERDASARKAN KEPADATAN ARUS KENDARAAN," *www.researchgate.net*. [https://www.researchgate.net/publication/280591688\\_IMPLEMENTASI\\_SISTEM\\_FUZZY\\_UNTUK\\_PENGATURAN\\_LAMPU\\_LALU\\_LINTAS\\_BERDASARKAN KEPADATAN\\_ARUS\\_KENDARAAN](https://www.researchgate.net/publication/280591688_IMPLEMENTASI_SISTEM_FUZZY_UNTUK_PENGATURAN_LAMPU_LALU_LINTAS_BERDASARKAN KEPADATAN_ARUS_KENDARAAN).
- [3] Shaleh and Rumahshaleh.com, "pengertian webcam dan sejenisnya," *www.rumahshaleh.com*. <https://rumahshaleh.com/pengertian-webcam-dan-jenisnya/>.
- [4] G. Kaonang and dailysocial.id, "Raspberry Pi 4 Janjikan Performa Sekelas Komputer Desktop Tanpa Korban Fleksibilitasnya," *www.dailysocial.id*. <https://dailysocial.id/post/raspberry-pi-4>.
- [5] dewi putri Suryani and openlibrary.telkomuniversity.ac.id, "PROTOTYPE KONTROL LAMPU RUMAH MENGGUNAKAN RASPBERRY PI BERBASIS WEB," *www.openlibrary.telkomuniversity.ac.id*, 2016, [Online]. Available: <https://openlibrary.telkomuniversity.ac.id/pustaka/122540/prototipe-kontrol-lampu-rumah-menggunakan-raspberry-pi-berbasis-web.html>.
- [6] elektronika-dasar.web.id, "LED (Light Emitting Dioda)," *www.elektronika-dasar.web.id*.
- [7] Mukti911 and Mukti.top, "python again apa itu python part1," *www.mukti.top*. <https://www.mukti.top/2018/08/python-again-apa-itu-python-part1.html?m=1>.
- [8] hanugra aulia Sidharta and Binus.ac.id, "introduction to openCV,"

