

IMPLEMENTASI KAMERA NIGHT VISION PADA ROBOT LASER MENGGUNAKAN SMARTPHONE PADA VR BOX

Raensi Taruna Aji¹⁾, Muhammat Maariful Huda²⁾, Eko Kuncoro³⁾

¹⁾Prodi Teknik Telekomunikasi Militer, Poltekad Kodiklatad,
Jl Raya Anggrek Batu-Malang-65324-Telp (0341) 461504

²⁾Jurusan Ilmu Komputer, Universitas Nahdlatul Ulama Blitar

Jl. Masjid No.22, Kauman, Kec. Kepanjenkidul, Kota Blitar (66117)
Komd4416@gmail.com ¹⁾, hudha.maariful@gmail.com²⁾, ekokuncoro68@gmail.com³⁾

IMPLEMENTATION OF NIGHT VISION CAMERA ON LASER ROBOT USING SMARTPHONE ON VR BOX

Abstract: *The military world needs innovation in security systems. Robot are one of the innovations because they are part of the development of technology. However, the efficiency of time and effort, as well as considering safety, requires more innovative development. This study was conducted to answer the question of whether the implementation of night vision camera on laser robot using smartphone on VR Box can accurately detect human objects, and the extent of their detection distance. The experimental method for proving hypotheses through a designed program is the method used in this study. The results of this study concluded that the implementation of night vision cameras can identify human objects in the form of frames that indicate the location where the object is detected in night and normal mode. Identification of human objects in night mode acquires only an accuracy of 81%, while in normal mode the camera is able to show accuracy of 100%. The camera is able to identify human objects with a maximum distance of 9 meters in night mode. Meanwhile, in night mode, the camera is able to identify human objects as far as 11 meters away.*

Abstrak: Dunia militer membutuhkan inovasi dalam sistem keamanan. Robot adalah salah satu inovasi karena bagian dari perkembangan teknologi. Namun, efisiensi waktu dan tenaga, serta mempertimbangkan keamanan, diperlukan pengembangan yang lebih inovatif. Untuk itu, penelitian ini dilakukan untuk menjawab pertanyaan apakah implementasi kamera *night vision* pada robot laser menggunakan *smartphone* pada VR Box dapat mendeteksi objek manusia secara akurat, dan sejauh mana jarak pendeteksiannya. Metode eksperimen untuk membuktikan hipotesis melalui program yang dirancang adalah metode yang digunakan dalam penelitian ini. Hasil penelitian ini menyimpulkan bahwa, Implementasi kamera *night vision* dapat mengidentifikasi objek manusia berupa *frame* yang menunjukkan lokasi di mana objek dideteksi dalam mode malam dan normal. Identifikasi objek manusia dalam mode malam hanya memperoleh akurasi sebesar 81%, sedangkan dalam mode normal kamera mampu menunjukkan akurasi sebesar 100%. Kamera mampu mengidentifikasi objek manusia dengan jarak maksimal 9 meter pada mode malam. Sedangkan, pada mode malam, kamera mampu mengidentifikasi objek manusia sejauh 11 meter.

Kata kunci: Kamera Night Vision, Robot, Smartphone, VR Box, Objek Manusia

PENDAHULUAN

Robot kamera *night vision* dapat digunakan untuk melihat hasil tangkapan kamera yang lebih bagus dalam pelacakan. Dengan begitu, seseorang dapat terbantu untuk mengidentifikasi sesuatu. Seperti yang dalam penelitian al Fairi & Hermanto (2018), dengan robot kamera *night vision*, korban yang mungkin tidak dilihat oleh operator yang menavigasikan robot untuk bergerak di dalam reruntuhan dapat lebih mudah untuk teridentifikasi.

Pun demikian dalam sistem keamanan, kamera *night vision* dapat membantu membaca objek hanya dengan melihat LCD (*Led Crystal Display*). Seperti penelitian yang pernah dilakukan oleh Rachmawati (2021), bahwa keamanan pada sistem yang dirancang dengan *mikro kontroller* berbasis *Raspberry Pi*, dapat membantu membaca objek, dan *liquid crystal display* sebagai penampil objek yang ditangkap kamera.

Berdasarkan penjelasan di atas, maka mengimplementasikan kamera *night vision* pada robot laser juga dapat diterapkan pula di bidang militer sebagai bentuk inovasi yang baru. Sebab dilihat dari *night vision* itu sendiri merupakan kamera yang mampu melihat baik dalam kondisi biologis atau teknologi di lingkungan gelap (Syamsuddin et al., 2021).

Kamera *night vision*, atau teknologi hasil pengembangan AS ini pada dasarnya ditujukan untuk pertahanan

(Gokartha, 2020). Itu sebabnya dalam penelitian yang pernah dilakukan (Majid et al., 2020), dijelaskan bahwa *tracking* kamera *night vision* dapat menggantikan pos dengan berbasis *image processing*. Dengan hasil pengolahan citra mampu kamera mampu mendeteksi keberadaan berupa objek orang berdasarkan bentuk tubuh.

Di samping itu, pada penelitian yang dilakukan oleh Syamsuddin et al. (2021), juga mengemukakan terkait robot dan kamera *night vision* adalah bagian dari inovasi yang dapat diterapkan di bidang militer. Menurut hasil penelitian Syamsuddin, pengolahan citra mampu mendeteksi keberadaan berupa objek orang berdasarkan bentuk tubuh, dan *tracking object* dengan baik dari jarak jauh.

Adapun dalam hal monitoring hasil kamera ini dapat dilakukan melalui *smartphone*, karena berisi sejumlah *chip sirkuit* terintegrasi *IC logam-oksidasemikonduktor* (MOS), yang mendukung komunikasi nirkabel (seperti Bluetooth, Wi-Fi, atau navigasi satelit) (Zega, 2022).

Terlepas dari itu, guna lebih memudahkan hasil tangkapan layar yang ditayangkan pada *smartphone* bisa dihubungkan dengan kacamata VR Box sebab fitur yang dimiliki. Sebab dilihat dari cara kerjanya VR box sendiri merupakan sebuah teknologi yang memungkinkan seseorang untuk melihat suatu pergerakan objek secara nyata dan

membuat pengguna seolah-olah dapat menyentuh objek yang di dalamnya secara fisik (Meidelfi et al., 2018).

Didasarkan pada uraian latar belakang di atas, maka diajukan dua pertanyaan penelitian, yaitu: 1) apakah implementasi kamera *night vision* pada robot laser menggunakan *smartphone* pada *VR Box* dapat mendeteksi objek manusia secara *real time*? 2) Berapa jauh jarak deteksi objek manusia melalui kamera *night vision* pada robot laser menggunakan *smartphone* pada *VR Box*?

METODE PENELITIAN

Metode yang digunakan dalam penelitian ini adalah metode eksperimen yang membuktikan hipotesis melalui program yang di rancang. Kualitas dari rancangan dapat diketahui dengan cara melakukan pengujian terhadap program yang telah dibuat secara terus menerus untuk mendapatkan hasil yang baik dan akurat.

Variabel Penelitian

Pada pengujian ini terdapat 2 variabel penelitian yang telah ditetapkan oleh peneliti, antara lain:

1. Variabel terikat

Variabel terikat adalah yang dipengaruhi oleh variabel bebas, di antaranya:

- a. Akurasi deteksi objek.
- b. Jarak deteksi objek.

2. Variabel Bebas

Variabel bebas adalah nilai yang didapat berdasarkan dari referensi yang

saling terkait yaitu ditentukan atau dapat diubah oleh peneliti. Dalam penyusunan tugas akhir yang berjudul sistem transmisi pada alat, antara lain:

- a. Kamera *Night Vision*
- b. Robot Laser
- c. *Smartphone*
- d. *VR Box*

Pengambilan Data

Data penelitian ini menggunakan data primer dan skunder yang dikumpulkan oleh peneliti melalui teknik pengumpulan data dengan cara eksperimen atau percobaan yang dilakukan kepada alat yang digunakan antara lain, kamera *night vision*, robot laser, *VR Box*, dan *smartphone* untuk mendapatkan objek penelitian yang sempurna. Adapun tahapan pengambilan data dalam penelitian ini adalah sebagai berikut:

Pengambilan Data Primer

Data primer dalam penelitian ini diambil dari hasil uji coba alat yang telah di terapkan sebagai berikut:

1. Pengujian *software* pada kamera *night vision raspberry Pi* agar kamera yang telah di rancang dapat mengidentifikasi objek manusia.
2. Pengujian *software* pada kamera *night vision raspberry Pi* agar mengetahui seberapa jauh jarak maksimal dapat mendeteksi objek.

Pengambilan Data Sekunder

Berdasarkan pada pengambilan data yang dilakukan oleh peniliti untuk melengkapi

data yang akan diterapkan pada tugas akhir ini berupa jurnal tentang sistem yang berkaitan dengan kamera *night vision*, jurnal yang berkaitan dengan *pattern recognition* yang merupakan pengolahan pada pola dan jurnal yang berkaitan dengan *openCV* dengan menerapkan metode *Convolutional Neural Network (CNN)* dan *Local Binary Patterns (LBP)*.

HASIL DAN PEMBAHASAN

Perancangan Sistem

Sebelum melaksanakan rancang bangun alat, penulis membuat gambaran sebagai modal utama, maka penulis diperlukan desain rencana alat sebagai konsep awal untuk pekerjaan langkah selanjutnya. Sehingga penulis mampu membuat dan merancang alat yang sesuai dengan hasil yang optimal.

Konseptual Desain VR Box



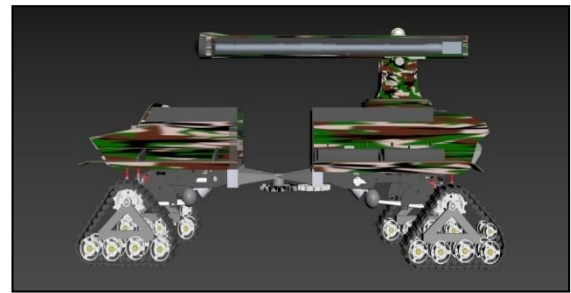
Gambar 1 Desain Smartphone pada VR
Box Terbuka



Gambar 2 Desain Smartphone VR BOX
Tertutup

(Sumber: www.Google.com, 2022)

Konseptual *Camera Night Vision* pada Robot Laser



Gambar 3 Desain Alat Tampak dari
Samping

(Sumber: Diolah oleh peneliti, 2023)



Gambar4 Desain Alat Tampak dari Atas

(Sumber: Diolah oleh peneliti, 2023)



Gambar 5 Kamera Night Vision pada
Bagian Depan Robot Laser

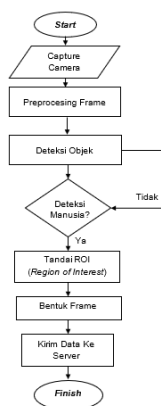
(Sumber: Diolah oleh peneliti, 2023)



Gambar 6 Kamera Night Vision pada Bagian Depan Robot Laser Tampak Samping

(Sumber: Diolah oleh peneliti, 2023)

Flowchart



Gambar 7 Flowchart Alat

Berdasarkan Flowchat di atas dapat dijelaskan sistem kerja alat adalah sebagai berikut:

1. Pengambilan data gambar terhadap sebuah objek berupa data *frame* yang diambil oleh kamera *night vision*.
2. Gambar melalui proses menuju proses untuk mendeteksi objek yang terdeteksi yaitu manusia.

Perancangan Software

Peneliti pada tahapan selanjutnya merancang *software* yang akan diterapkan

pada kamera *night vision* agar dapat mendeteksi objek manusia dengan jarak 11 meter. Proses perancangan dilakukan dengan langkah dan tahapan seperti berikut:

Langkah Perancangan Software

Adapun langkah dalam merancang *software* Menginstal modul *openCV*

```

pi@raspberrypi:~$ sudo pip3 install opencv-python
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Collecting opencv-python
  Downloading https://files.pythonhosted.org/packages/b6/82/0519fdbbcadd0fa8c2568327a8311477315a91b4513738e1d35f0ce715/opencv-python-4.5.4.58.tar.gz (89.8MB)
    63% |#####| 57.4MB 3.9MB/s eta 0:00:09
  
```

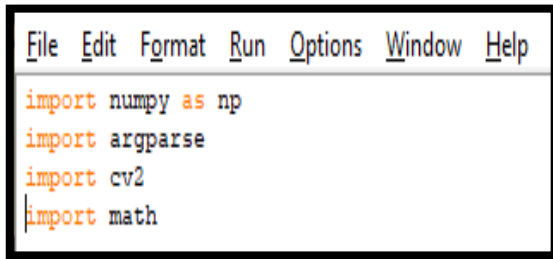
Gambar8 Modul OpenCV

(Sumber: Diolah oleh peneliti, 2023)

Adapun dalam pendeteksian objek manusia maka diperlukan untuk menginstall modul *opencv*. Modul ini berguna sebagai modul untuk melakukan proses citra digital, sehingga script yang akan dibuat dapat melakukan pendeteksian objek, sesuai dengan objek yang dipilih yakni objek manusia. Untuk melakukan instalasi *opencv* maka yang akan dilakukan adalah mengeksekusi perintah “*sudo pip3 install opencv-python*” dan menunggu sampai proses selesai.

Perancangan Program Deteksi Objek Manusia (CNN)

Perancangan CNN meliputi pembacaan library dengan menggunakan statement import. Dimana library yang dibutuhkan untuk CNN ditunjukkan pada gambar berikut:



Gambar 9 Library untuk CNN

(Sumber: Diolah oleh peneliti, 2023)

Library *numpy* digunakan untuk keperluan operasi matrix, sebab CNN pada dasarnya menggunakan operasi *dot matrix* untuk melakukan *back propagation* dan *forward propagation*. *Argparse* digunakan untuk keperluan *debugging* atau keperluan mencari kesalahan program apabila terjadi *error*. *CV2* untuk melakukan operasi image processing, semua operasi yang berhubungan dengan pemrosesan gambar melalui modul *CV2* atau *OpenCV2* yang sebelumnya sudah diinstall ke dalam *raspberry*. Sedangkan *math* digunakan untuk perhitungan matematis dasar.

Adapun objek perancangan

CNN adalah sebagai berikut:

1. Pembacaan Library

```
#pembacaan library yang digunakan, sehingga coding yang dipakai tidak
#tidak terlalu panjang. Hanya memanggil fungsi-fungsi yang disediakan
#oleh library
from picamera.array import PiRGBArray
from picamera import PiCamera
import imutils
from imutils.video import VideoStream
from imutils.video import FPS
import time
from flask import Flask, render_template, Response
import cv2
import numpy as np
import os
import multiprocessing as mp
```

2. Pembacaan Mode Deteksi

```
#pembacaan mode deteksi
f = open('/home/pi/stream/isdetected.txt','r')
isdetect = str(f.read())
f.close()
```

3. Pembacaan Kamera IPCAM

```
#class camera digunakan untuk pembacaan kamera ipcam
class Camera():
    #inisialisasi kamera
    def __init__(self, rtsp_url):
        #load pipe for data transmission to the process
        self.parent_conn, child_conn = mp.Pipe()
        #load process
```

```
self.p = mp.Process(target=self.update,
args=(child_conn, rtsp_url))
#start process
self.p.daemon = True
self.p.start()
#fungsi untuk mengakhiri pembacaan kamera
def end(self):
    #send closure request to process

    self.parent_conn.send(2)
#fungsi untuk membaca frame atau gambar yang ditangkap oleh
kamera
def update(self, conn, rtsp_url):
    #load cam into separate process

    print("Cam Loading...")
    cap = cv2.VideoCapture(rtsp_url, cv2.CAP_FFMPEG)
    cap.set(cv2.CAP_PROP_BUFFERSIZE, 1)
    print("Cam Loaded...")
    run = True

    while run:
        #grab frames from the buffer
        cap.grab()

        #recieve input data
        rec_dat = conn.recv()

        if rec_dat == 1:
            #if frame requested
            ret, frame = cap.read()
            conn.send(frame)

        elif rec_dat == 2:
            #if close requested
            cap.release()
            run = False

    print("Camera Connection Closed")
    conn.close()
#fungsi untuk mengambil frame dengan konfigurasi pengecilan atau
#pembesaran gambar
def get_frame(self, resize=None):
    ##Used to grab frames from the cam connection process

    ##[resize] param : % of size reduction or increase i.e 0.65
    #for 35% reduction or 1.5 for a 50% increase

    #send request
    self.parent_conn.send(1)
    frame = self.parent_conn.recv()

    #reset request
    self.parent_conn.send(0)

    #resize if needed
    if resize == None:
        return frame
    else:
        return self.rescale_frame(frame, resize)
#fungsi untuk mengubah ukuran gambar
def rescale_frame(self, frame, percent=65):
    return cv2.resize(frame, None, fx=percent, fy=percent)

#inisialisasi class objek yang dipakai
tempdir = os.path.abspath('/home/pi/stream/templates')
crosshair = cv2.imread("crosshair.png", cv2.IMREAD_UNCHANGED)
CLASSES = ["background", "aeroplane", "bicycle", "bird", "boat",
"bottle", "bus", "car", "cat", "chair", "cow",
"diningtable",
"dog", "horse", "motorbike", "person", "pottedplant",
"sheep",
"sofa", "train", "tvmonitor"]
COLORS = (0,255,0)

#inisialisasi metode neural network yang dipakai (MobilenetSSD)
net =
cv2.dnn.readNetFromCaffe('/home/pi/stream/MobileNetSSD_deploy.prototx
t', '/home/pi/stream/MobileNetSSD_deploy.caffemodel')

app = Flask(__name__, template_folder=tempdir)
time.sleep(1)

#fungsi yang dipanggil ketika vr headset diakses
def gen_frames():
    global camera
    global isdetect

    #ambil alamat kamera ipcam
    camera = Camera("rtsp://admin:BFGYRP@192.168.8.102/h264_stream")
    time.sleep(1)
    while True:
        t1 = int(time.time()*1000)
        image = camera.get_frame(0.2)
        if not True:
            break
        else:
            #jika metode deteksi

            if '1' in isdetect:

                #jalankan neural network
                (h,w) = image.shape[:2]
                blob = cv2.dnn.blobFromImage(cv2.resize(image, (300,
300)), 0.007843, (300, 300), 127.5)
                net.setInput(blob)
                detections = net.forward()
                #looping atau perulangan program disetiap objek yang terdeteksi
                for i in np.arange(0, detections.shape[2]):
                    confidence = detections[0, 0, i, 2]

#jika objek terdeteksi dengan tingkat keyakinan lebih dari 0.2 atau
20% maka lanjutkan
if confidence > 0.2 and CLASSES[int(detections[0,
0, i, 1])]=='person':
    #bentuk gambar penanda terhadap objek yang berhasil dideteksi
```

```

        idx = int(detections[0, 0, i, 1])
        box = detections[0, 0, i, 3:7] * np.array([w,
        h, w, h])
        box.astype("int")
        (startX, startY, endX, endY) =
        label = "{}: {:.2f}%".format(CLASSES[idx],
        confidence * 100)
        cv2.rectangle(image, (startX, startY), (endX,
        endY),
        COLORS, 2)
        y = startY - 15 if startY - 15 > 15 else
        startY + 15
        t2 = int(time.time()*1000)-t1
        cv2.putText(image, label+' '+str(t2),
        (startX, y),
        cv2.FONT_HERSHEY_SIMPLEX, 0.5, COLORS, 2)
        height, width, channels = image.shape
        #gambar sasaran
        cv2.line(image, (int(width/3),int(height/2)),
        (int(width/3+15),int(height/2)), (0,255,0), 2)
        cv2.line(image, (int(width-width/3),int(height/2)),
        (int(width-width/3-15),int(height/2)), (0,255,0), 2)
        cv2.line(image, (int(width/2),int(height/2+20)),
        (int(width/2),int(height-height/4)), (0,255,0), 2)
        cv2.circle(image, (int(width/2), int(height/2)), 2,
        (0,255,0), 2)
        ret, buffer = cv2.imencode('.jpg', image)
        #rawCapture.truncate(0)
        frame2 = buffer.tobytes()
        #bentuk data hasil
        dengan format gambar jpeg untuk ditampilkan pada web
        yield (b'--frame\r\n'
        b'Content-Type: image/jpeg\r\n\r\n' + frame2 +
        b'\r\n')

#routing web atau jalur web ketika server diakses
#jika jalur yang diminta adalah jalur video_feed maka eksekusi
program pengolahan gambar diatas
@app.route('/video_feed')
def video_feed():
    return Response(gen_frames(), mimetype='multipart/x-mixed-
    replace; boundary=frame')

#jika server diakses maka menuju jalur tampilan pada file index.html
@app.route('/')
def index():
    return render_template('index.html')

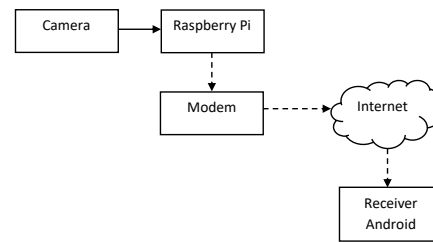
#jalankan server dengan alamat ip yang diberikan oleh modem dengan
port 8080
if __name__ == '__main__':
    app.run(host='0.0.0.0', port=8080)

```

Perencanaan Sistem

Sistem penglihatan pada kamera *night vision* yang dimonitor melalui *Smartphone* sebagai langkah awal *smartphone* akan disetting. Setelah itu, *smartphone* akan melakukan inisialisasi terhadap kamera yang akan digunakan. Apabila sudah siap maka *smartphone* akan melakukan perekaman pada objek. Apabila objek sudah dikenali maka kamera akan menangkap gambar yang kemudian akan diproses dan ditampilkan melalui layar *smartphone*.

Penjelasan Blok Diagram Alat



Gambar 10 Blok Diagram Alat

(Sumber: Diolah oleh peneliti, 2023)

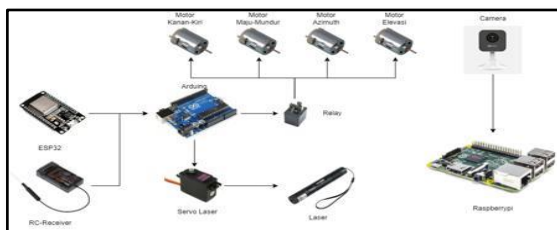
Adapun penjelasan blok diagram alat di atas adalah sebagai berikut:

1. *Ipcam* digunakan sebagai kamera untuk menangkap gambar objek dengan fitur *night vision*, yang aktif secara otomatis akan aktif apabila kamera mendeteksi cahaya yang kurang. Sehingga bagaimanapun keadaan cahaya kamera dapat menangkap gambar beserta pola dari objek yang tertangkap.
2. *Raspberry pi* sebagai pusat dari pengolahan data, di mana fungsi utama *raspberry* adalah sebagai pengolah citra digital yang akan mengolah data dari kamera. Pengolahan untuk pengenalan objek meliputi proses pengambilan gambar, *preprocessing*, *image recognition*, pengenalan pola dengan metode *local binary pattern*, dan pengiriman hasil pengolahan data. Pada proses pengambilan data sistem akan mengambil data *frame* pada alamat ip dari *ipcam*, setelah itu gambar melalui proses *preprocessing* di mana pada proses ini gambar akan dikonversi yang awalnya memiliki

format warna RGB menjadi format warna *grayscale*, setelah itu menuju proses *image recognition* untuk mengetahui keberadaan manusia pada frame yang ditangkap, proses ini mempunyai *output ROI (Region of Interest)* yang berisikan variabel posisi objek yang terdeteksi berupa koordinat x,y beserta panjang atau lebar objek yang dideteksi adalah manusia. Setelah itu proses terakhir adalah pengiriman hasil pendeteksian melalui jaringan *Local Area Network (LAN)* melalui perangkat-perangkat jaringan.

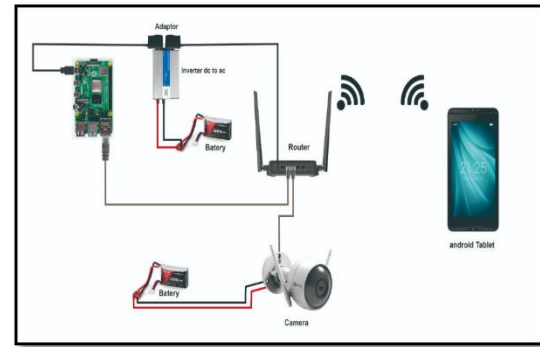
3. *Android* perangkat android digunakan untuk mengetahui hasil dari pengolahan data dari sistem robot untuk pengenalan objek. Selain itu perangkat android juga digunakan sebagai perangkat monitoring untuk memonitor posisi robot pada peta secara *realtime*, sehingga pengguna dapat mengetahui posisi robot tanpa harus dekat dengan robot.

Berdasarkan seluruh penjelasan di atas, perancangan sistem pada penelitian ini dapat digambarkan sebagai berikut:



Gambar 11 Diagram Keseluruhan Sistem Alat

(Sumber: Diolah oleh peneliti, 2023)



Gambar 12 Diagram Sistem pada Software Raspberry Pi

(Sumber: Diolah oleh peneliti, 2023)

Dari Gambar rangkaian *wiring* diagram pada software raspberry Pi dapat dijelaskan bahwa :

- 1) Power Bank *Aukey* sebagai sumber daya untuk mengaktifkan *raspberry Pi*, adaptor dan juga kamera.
- 2) Kamera dihubungkan ke router, guna mengirimkan hasil gambar objek yang selanjutnya di hubungan menuju *raspberry Pi*.
- 3) *Raspberry Pi* memproses data dengan *software* yang telah dirancang kemudian hasil dikirimkan Kembali menuju router.
- 4) *Smartphone* sebagai perangkat *monitoring* untuk menampilkan hasil dari gambar objek yang telah diproses.

Hasil Penelitian

Hasil Pengujian Deteksi

Pada pengujian deteksi objek manusia, peneliti melakukan uji coba untuk mengetahui akurasi dari alat yang dibuat dalam mengidentifikasi objek manusia di kamera dengan dua tampilan yang akan dihubungkan dengan VR box.

Sedangkan untuk objek yang akan dideteksi adalah kotak sasaran tepat berada di seluruh tubuh manusia.

Adapun hasil pengujian dalam penelitian ini dilakukan melalui dua proses pengujian, yaitu mode malam dan normal, sebagaimana gambar 13 dan 14.



Gambar 13 Tangkapan Layar Deteksi Target Mode Normal
(Sumber: Diolah oleh peneliti, 2023)



Gambar 14 Tangkapan Layar Deteksi Target Mode Malam
(Sumber: Diolah oleh peneliti, 2023)

Adapun pengujian yang pertama, yang dilakukan menggunakan mode malam dan mendapatkan nilai akurasi pengujian sebagaimana tabel 1.

Tabel 1 Hasil Pengujian Deteksi Objek Manusia Mode Malam

No	Jarak	Keterangan
1	2 M	Berhasil Terdeteksi
2	3 M	Berhasil Terdeteksi
3	4 M	Berhasil Terdeteksi
4	5 M	Berhasil Terdeteksi

5	6 M	Berhasil Terdeteksi
6	7 M	Berhasil Terdeteksi
7	8 M	Berhasil Terdeteksi
8	9 M	Berhasil Terdeteksi
9	10 M	Tidak Terdeteksi
10	11 M	Tidak Terdeteksi

(Sumber: Diolah oleh peneliti, 2023)

Berdasarkan hasil uji pada tabel di atas, dapat diketahui bahwa, tingkat akurasi alat yang dibuat untuk mendeteksi objek manusia secara *real time* menggunakan mode malam adalah 81% yang diperoleh dengan rumus sebagai berikut:

$$\begin{aligned}
 \text{Accuracy} &= \frac{x}{n} \times 100 \\
 &= \frac{9}{11} \times 100 \\
 &= 0,81 \times 100 \\
 &= 81\%
 \end{aligned}$$

Keterangan:

x = percobaan berhasil

n = jumlah percobaan keseluruhan

Adapun dalam pengujian yang dikedua didapatkan nilai akurasi sebesar 100% sebagaimana tabel 2.

Tabel 2 Hasil Pendeteksian Objek Manusia Mode Normal

No	Jarak	Keterangan
1	2 M	Berhasil Terdeteksi
2	3 M	Berhasil Terdeteksi
3	4 M	Berhasil Terdeteksi
4	5 M	Berhasil Terdeteksi
5	6 M	Berhasil Terdeteksi
6	7 M	Berhasil Terdeteksi
7	8 M	Berhasil Terdeteksi
8	9 M	Berhasil Terdeteksi
9	10 M	Berhasil Terdeteksi
10	11 M	Berhasil Terdeteksi

(Sumber: Diolah oleh peneliti, 2023)

Berdasarkan hasil uji pada tabel di atas, dapat diketahui bahwa, tingkat akurasi alat yang dibuat untuk mendeteksi

NO	PERCOBAAN	JARAK	HASIL	
			Malam	Normal
1.	Percobaan ke-1	2 M	Detek	Detek
2.	Percobaan ke-2	3 M	Detek	Detek
3.	Percobaan ke-3	4 M	Detek	Detek
4.	Percobaan ke-4	5 M	Detek	Detek
5.	Percobaan ke-5	6 M	Detek	Detek
6.	Percobaan ke-6	7 M	Detek	Detek
7.	Percobaan ke-7	8 M	Detek	Detek
8.	Percobaan ke-8	9 M	Detek	Detek
9.	Percobaan ke-9	10 M	Tidak	Detek
10.	Percobaan ke-10	11 M	Tidak	Detek

objek manusia secara *real time* menggunakan mode normal adalah 100%, yang diperoleh dengan rumus sebagai berikut:

$$\begin{aligned}
 \text{Accuracy} &= \frac{x}{n} \times 100 \\
 &= \frac{11}{11} \times 100 \\
 &= 1 \times 100 \\
 &= 100\%
 \end{aligned}$$

Berdasarkan dua pengujian di atas, dapat disimpulkan bahwa, implemementasi kamera *night vision* pada robot laser menggunakan *smartphone* pada VR Box, apabila dilakukan menggunakan mode malam mendapatkan akurasi sebesar 81%, namun jika dilakukan menggunakan mode normal, maka akurasi yang didapatkan adalah sebesar 100%.

Hasil Pengujian Jarak Deteksi Objek

Pada pengujian jarak maksimal deteksi objek peneliti melakukan uji coba dengan menggunakan alat ukur meteran untuk mengetahui seberapa jauh jarak kemampuan dari sistem software pada kamera tersebut didalam melakukan pengidentifikasian objek manusia menggunakan kamera *Night Vision*.

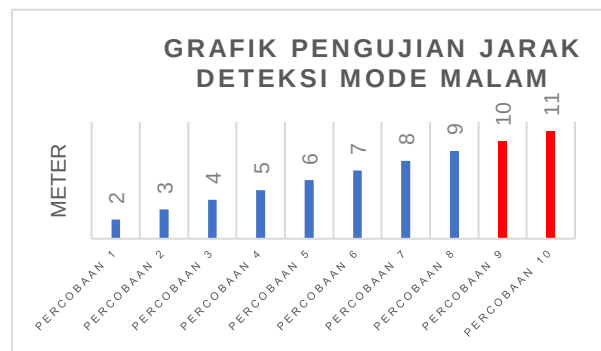
Sebagaimana telah dilakukan dalam pengujian deteksi objek, yaitu

menggunakan dua mode, maka dalam pengujian jarak, dilakukan pula pengujian deteksi jarak dengan dua mode dan mendapatkan hasil seperti pada tabel 3.

Tabel 3 Hasil Pengujian Jarak Deteksi Objek

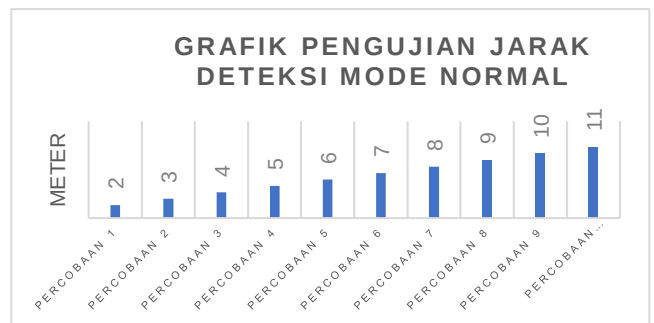
(Sumber: Diolah oleh peneliti, 2023)

Berdasarkan hasil pengujian jarak deteksi objek sebagaimana tabel 3 maka dapat digambarkan dalam bentuk grafik pada gambar 15 dan 16:



Gambar 15 Grafik Pengujian Jarak Deteksi Objek Mode Malam

(Sumber: Diolah oleh peneliti, 2023)



Gambar 16 Grafik Pengujian Jarak Deteksi Objek Mode Normal

(Sumber: Diolah oleh peneliti, 2023)

Pada tabel dan grafik hasil pengujian di atas, dipaparkan bahwa kemampuan *software* pada sistem kamera ini mampu mendeteksi objek manusia dengan jarak maksimal 9 meter dalam mode malam, namun pada mode normal

kamera mampu mendeteksi hingga jarak 11 meter.

KESIMPULAN

Berdasarkan seluruh pengujian yang telah dilakukan, maka pada akhir penulisan Tugas Akhir yang berjudul Implementasi Kamera *Night Vision* Pada Robot Laser Menggunakan *Smartphone* Pada VR Box Secara *Real Time* dapat diambil kesimpulan, bahwa Implementasi kamera *night vision* dapat mengidentifikasi objek manusia berupa *frame* yang menunjukkan lokasi di mana objek dideteksi dalam dua mode, yaitu mode malam dan normal. Identifikasi objek manusia dalam mode malam hanya memperoleh akurasi sebesar 81%, sedangkan dalam mode normal kamera mampu menunjukkan akurasi sebesar 100%.

Kamera mampu mengidentifikasi objek manusia dengan jarak maksimal 9 meter pada mode malam, dari jarak maksimal yang telah ditentukan, yaitu 11 meter. Sedangkan, pada mode malam, kamera mampu mengidentifikasi objek manusia secara maksimal, yaitu 11 meter.

DAFTAR PUSTAKA

- al Fairi, T., & Hermanto. (2018). Al Fairi, T., & Hermanto, L. 2018. "Implementasi Teknologi IOT Pada Robot Kamera NV Untuk Pelacak Korban Gempa di dalam Reruntuhan". . *Jurnal Ilmiah KOMPUTASI*, 17(3), 209–220.
- Gokartha, R. L. (2020). *Pembangunan Aplikasi Dashcam Ride Tracking Untuk Laporan Rekaman Berkendara Supir Pada Kpm Trans Memanfaatkan Google Cloud Vision Pada Smartphone Android* [Skripsi]. Universitas Komputer Indonesia.
- Majid, W. E., Gatut Yulisusianto, & Rahmad Santoso. (2020). Rancang Bangun Sistem Tracking Pada Kamera Night Vision Sebagai Pengganti Pos Dengar Berbasis Image Processing. Malang : Politeknik Angkatan Darat. *Politeknik Angkatan Darat*.
- Meidelfi, D., Mooduto, H. A., & Setiawan, D. (2018). Visualisasi 3D Gedung dengan Konsep Virtual Reality Berbasis Android. *INVOTEK: Jurnal Inovasi Vokasional Dan Teknologi*, 18(1), 59–66. <https://doi.org/10.24036/invotek.v18i1.264>
- Rachmawati, et al. (2021). Perancangan Sistem Keamanan Pintar Kamera Night Vision Auto Color Berbasis Raspberry Pi. *Jurnal Qua Teknika*, 11(2).
- Syamsuddin, A. N., Kuncoro, E., & Chorina, P. (2021). Implementasi Identifikasi dan Tracking Sasaran Secara Otomatis Pada Robot Tempur Cia Versi N2MR3 Dengan Menggunakan Metode Pattern Recognition". . . *Jurnal Telkommil*, 2, 34–41.
- Zega, Y. J. S. (2022). , *Benefit and Sompie, Sherwin R.U.A. 2022. "Alat Monitoring Pemakaian Listrik Menggunakan Arduino Uno"*. . (online). <http://repo.unsrat.ac.id/3957/> [Skripsi]. Universitas Sam Ratulangi.