

BAB V

SIMPULAN DAN SARAN

5.1 Simpulan

Berdasarkan dari hasil dan pembahasan sistem keamanan ruangan menggunakan algoritma *YOLOv8*. Dapat disimpulkan hal-hal sebagai berikut :

1. Sistem mampu melakukan deteksi objek mencurigakan secara langsung (*real-time*) menggunakan ESP32-CAM yang terintegrasi dengan model *You Only Look Once Version 8 (YOLOv8)*. Deteksi difokuskan pada objek berbahaya seperti pisau, dan hasilnya ditampilkan dalam bentuk *bounding box* pada video frame
2. Sistem mampu mengirim peringatan otomatis melalui *WhatsApp* menggunakan *Pywhatkit* Ketika mendeteksi objek berbahaya seperti pisau. Hal ini membuktikan bahwa sistem tidak hanya mendeteksi potensi ancaman, tetapi juga dapat memberikan respons cepat dengan menginformasikan kepada pihak keamanan secara *real-time*.
3. Evaluasi performa keseluruhan sistem menunjukan kinerja yang stabil, baik dalam proses pengambilan data, pemrosesan deteksi, hingga pengiriman notifikasi, hal ini mengindikasikan bahwa sistem layak untuk diimplementasikan sebagai solusi keamanan berbasis IoT.

5.2 Saran

Berdasarkan hasil dari data pengujian penelitian ada beberapa saran yang dapat dipertimbangkan:

1. Pengujian sistem dalam berbagai kondisi lingkungan, seperti pencahayaan yang minim, sudut pandang kamera yang tidak ideal, atau gerakan cepat, untuk mengevaluasi ketahanan dan keandalan model dalam situasi nyata.
2. Integrasi sistem dengan layanan IoT berbasis *cloud* (misalnya *firebase*, *thingsboard*, atau Blynk) agar pengguna dapat melakukan pemantauan kondisi ruangan secara jarak jauh melalui perangkat seluler atau *dashboard online*.
3. Penambahan fitur alarm lokal, seperti penggunaan *buzzer* atau speaker, sebagai langkah pencegahan langsung di tempat kejadian bukan hanya tergantung pada pengiriman notifikasi *WhatsApp*.
4. Untuk peneliti selanjutnya disarankan untuk menggunakan lebih dari satu kelas *dataset* atau objek, seperti aktivitas yang mencurigakan, pistol, kapak, dan senjata tajam lainnya. Serta meningkatkan jumlah gambar pada setiap kelas yang bertujuan untuk memperluas kemampuan model deteksi.

DAFTAR PUSTAKA

- Andi, I., Muchtar, M., & Sari, J. Y. (2024). Mask Detection Using the YOLO (You Only Look Once) Method. *Jurnal Media Informasi Teknologi*, 1(1), 1–12.
<https://doi.org/10.69616/mit.v1i1.165>
- Arismunandar, W., Alamsyah, M., & Intan, I. (1978). *Perancangan Aplikasi Monotoring Dan Kontrol Smarthome Berbasis IoT Terintegrasi Dengan Bot Telegram Sebagai Notifikasi*. XVI(1), 134–144.
- Armin, E. U., Purnama Edra, A., Alifin, F. I., Sadidan, I., Sary, I. P., & Latifa, U. (2023). Performa Model YOLOv8 untuk Deteksi Kondisi Mengantuk pada pengendara mobil. *BRAHMANA: Jurnal Penerapan Kecerdasan Buatan*, 5(1), 67–76.
<https://doi.org/10.30645/brahmana.v5i1.279>
- Asiva Noor Rachmayani. (2021). *SYSTEM KEAMANAN BERBASIS RFID (RADIO FREQUENCY IDENTIFICATION) DAN MIKROKONTROLER*. 6.
- Effendy, E., Siregar, E. A., Fitri, P. C., & Damanik, I. A. S. (2023). Mengenal Sistem Informasi Manajemen Dakwah (Pengertian Sistem, Karakteristik Sistem). *Jurnal Pendidikan Dan Konseling*, 5(2), 4343–4349.
- Fadillah, W. N., Al-areef, M. H., & Khatulistiwa, J. (2023). *SISTEM PENDUKUNG KEPUTUSAN UNTUK MEMILIH LAPTOP*. 04(01), 7–13.
- Faiz, M. N., Somantri, O., Supriyono, A. R., & Muhammad, A. W. (2022). Impact of Feature Selection Methods on Machine Learning-based for Detecting DDoS Attacks : Literature Review. *Journal of Informatics and Telecommunication Engineering*, 5(2), 305–314. <https://doi.org/10.31289/jite.v5i2.6112>
- Fasounaki, M., Yüce, E. B., Öncül, S., & Ince, G. (2021). CNN-based Text-independent Automatic Speaker Identification Using Short Utterances. *Proceedings - 6th International Conference on Computer Science and Engineering, UBMK 2021*, 01, 413–418. <https://doi.org/10.1109/UBMK52708.2021.9559031>
- Fatikah, I. H., & Khazizah, N. (2022). Analisis cuaca di Kota Jakarta bulan Januari tahun 2018 menggunakan Algoritma Decision Tree. *Jurnal Poros Teknik*, 14(1), 33–37.
<https://www.kaggle.com/datasets/msf1203/pr>
- Hayati, N. J., Singasatia, D., & Muttaqin, M. R. (2023). Object Tracking Menggunakan

- Algoritma You Only Look Once (YOLO)v8 untuk Menghitung Kendaraan. *Komputa : Jurnal Ilmiah Komputer Dan Informatika*, 12(2), 91–99.
<https://doi.org/10.34010/komputa.v12i2.10654>
- Idris, M., Adam, R. I., Brianorman, Y., Munir, R., & Mahayana, D. (2022). Kebenaran dalam Perspektif Filsafat Ilmu Pengetahuan dan Implementasi dalam Data Science dan Machine Learning. *Jurnal Filsafat Indonesia*, 5(2), 173–181.
<https://doi.org/10.23887/jfi.v5i2.42207>
- Kukuh Isnaen, M., & Stefanie, A. (2023). IMPLEMENTASI RASPBERRY PI DALAM ALAT KLASIFIKASI PENYAKIT MATA DENGAN ARSITEKTUR YOLOv8 MENGGUNAKAN OFTALMOSKOP. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(3), 1885–1889. <https://doi.org/10.36040/jati.v7i3.6950>
- Manurung, D. G., Pinasthika, M. R., Vasya, M. A. O., Putri, R. A. D. S., Tampubolon, A. P., Prayata, R. F., Nisa, S. K., & Yudistira, N. (2024). Deteksi Dan Klasifikasi Hama Potato Beetle Pada Tanaman Kentang Menggunakan YOLOV8. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 11(4), 723–734. <https://doi.org/10.25126/jtiik.1148092>
- Marcellino, V., Christanti Mawardi, V., & Jaya Perdana, N. (2022). Pendeteksian Jumlah Penumpang Yang Masuk Berdasarkan Cctv Pada Pintu Bus Dengan Metode Yolo. *Jurnal Ilmu Komputer Dan Sistem Informasi*, 10(2).
<https://doi.org/10.24912/jiksi.v10i2.22539>
- Maulana, I., Rahaningsih, N., Suprpti, T., Informatika, T., Cirebon, K., & Detection, O. (2023). Analisis Penggunaan Model Yolov8 (You Only Look Once). 7(6), 3621–3627.
- Maulidina, G. (2024). Sistem keamanan sarang walet berbasis internet of things menggunakan esp32-cam dan sensor pir (passive infrared sensor) skripsi.
- Persada Sembiring, J., Jayadi, A., Putri, N. U., Sari, T. D. R., Sudana, I. W., Darmawan, O. A., Nugroho, F. A., & Ardiantoro, N. F. (2022). PELATIHAN INTERNET OF THINGS (IoT) BAGI SISWA/SISWI SMKN 1 SUKADANA, LAMPUNG TIMUR. *Journal of Social Sciences and Technology for Community Service (JSSTCS)*, 3(2), 181. <https://doi.org/10.33365/jsstcs.v3i2.2021>
- Qodryantha, A., Terapan, F. I., Telkom, U., Rudawan, R. A., Terapan, F. I., Telkom, U., Sularsa, A., Terapan, F. I., & Telkom, U. (2023). Pembangunan Sistem Pengawasan Cerdas Dengan Visualisasi 3D Development Of Smart Surveillance System With 3D

Visualization. 9(1), 465–480.

Reski Idrus, Cita St Munthakhabah R, & Al Qadri Hatta. (2023). Sistem Informasi Pengarsipan Surat Masuk Dan Surat Keluar Berbasis Web Pada Fakultas Ekonomi Universitas Sulawesi Barat. *Jurnal Ilmiah Ilmu Komputer*, 9(1), 51–62.

<http://ejournal.fikom-unasman.ac.id>

Setiawan, D., Jaya, H., Nurarif, S., Syahputra, T., & Syahril, M. (2022). Implementasi Esp32-Cam Dan Blynk Pada Wifi Door Lock System Menggunakanteknik Duplex. *Journal of Science and Social Research*, 5(1), 159.

<https://doi.org/10.54314/jssr.v5i1.807>

Siwu, B. H. M., Rampo, V. Y., & Joshua, S. R. (2022). Sistem Informasi Pelaporan Kerusakan Fasilitas Kantor Berbasis Web. *Jurnal Teknik Informatika Dan Elektro*, 4(2), 120–129. <https://doi.org/10.55542/jurtie.v4i2.413>

Wedi Pratama, F., Nova, F. N., Prayama, D., & Gusman, T. (2022). Media Pembelajaran Interaktif Fabrikasi PCB Berbasis Augmented Reality. *JITSI : Jurnal Ilmiah Teknologi Sistem Informasi*, 3(3), 81–87. <https://doi.org/10.30630/jitsi.3.3.93>

Widiyanto, R. A., Wicaksono, B. S., Informatika, T., Pamulang, U., & Selatan, T. (2022). *PERANCANGAN SISTEM INFORMASI MONITORING LAPORAN PENJUALAN MULTI CABANG BERBASIS WEB DENGAN METODE*. 1(1).

Yusuf, M., Sodik, M., Darussalam, S., Nganjuk, K., & Blitar, U. (2023). Penggunaan Teknologi Internet of Things (Iot) Dalam Pengelolaan Fasilitas Dan Infrastruktur Lembaga Pendidikan Islam. *PROPHETIK Jurnal Kajian Keislaman*, 1(2), 1–18.

DAFTAR RIWAYAT HIDUP



BIODATA DIRI

Nama Lengkap : Leinsky Erlangga
Tempat, Tanggal Lahir : Tangerang, 05 September 2002
Jenis Kelamin : Laki-Laki
Agama : Buddha
Kewarganegaraan : Indonesia
Alamat : Jl.Imam Bonjol Kav.Sawah No.14 RT:03/RW:08 Tangerang, Banten
Email : leinskyerlangga55@gmail.com
No Telepon : 082260549960
IPK Terakhir : 3.09

RIWAYAT PENDIDIKAN

- a. 2006 s/d 2007 TK BETHESDA
- b. 2007 s/d 2014 SDN PABUARAN TUMPENG 1
- c. 2014 s/d 2017 SMP DHARMA PUTRA
- d. 2017 s/d 2020 SMK BONAVIDA
- e. 2020 s/d 2025 UNIVERSITAS BUDDHI DHARMA

Tangerang, 06 Agustus 2025

Yang membuat pernyataan,

Leinsky Erlangga

20201000048

LAMPIRAN

Lampiran 1 Kode Arduino

```
#include "esp_camera.h"
#include <WiFi.h>

//
// WARNING!!! PSRAM IC required for UXGA resolution and high JPEG quality
//     Ensure ESP32 Wrover Module or other board with PSRAM is selected
//     Partial images will be transmitted if image exceeds buffer size
//
//     You must select partition scheme from the board menu that has at least 3MB APP space.
//     Face Recognition is DISABLED for ESP32 and ESP32-S2, because it takes up from 15
//     seconds to process single frame. Face Detection is ENABLED if PSRAM is enabled as well

// =====
// Select camera model
// =====
#define CAMERA_MODEL_WROVER_KIT // Has PSRAM
#define CAMERA_MODEL_ESP_EYE // Has PSRAM
#define CAMERA_MODEL_ESP32S3_EYE // Has PSRAM
#define CAMERA_MODEL_M5STACK_PSRAM // Has PSRAM
#define CAMERA_MODEL_M5STACK_V2_PSRAM // M5Camera version B Has PSRAM
#define CAMERA_MODEL_M5STACK_WIDE // Has PSRAM
#define CAMERA_MODEL_M5STACK_ESP32CAM // No PSRAM
#define CAMERA_MODEL_M5STACK_UNITCAM // No PSRAM
#define CAMERA_MODEL_M5STACK_CAMS3_UNIT // Has PSRAM
#define CAMERA_MODEL_AI_THINKER // Has PSRAM
#define CAMERA_MODEL_TTGO_T_JOURNAL // No PSRAM
#define CAMERA_MODEL_XIAO_ESP32S3 // Has PSRAM
// ** Espressif Internal Boards **
#define CAMERA_MODEL_ESP32_CAM_BOARD
#define CAMERA_MODEL_ESP32S2_CAM_BOARD
#define CAMERA_MODEL_ESP32S3_CAM_LCD
#define CAMERA_MODEL_DFRobot_FireBeetle2_ESP32S3 // Has PSRAM
#define CAMERA_MODEL_DFRobot_Romeo_ESP32S3 // Has PSRAM
#include "camera_pins.h"
```

```

// =====
// Enter your WiFi credentials
// =====

const char *ssid = "LT4";
const char *password = "Ecostar2018@";

void startCameraServer();
void setupLedFlash(int pin);

void setup() {
    Serial.begin(115200);
    Serial.setDebugOutput(true);
    Serial.println();

    camera_config_t config;
    config.ledc_channel = LEDC_CHANNEL_0;
    config.ledc_timer = LEDC_TIMER_0;
    config.pin_d0 = Y2_GPIO_NUM;
    config.pin_d1 = Y3_GPIO_NUM;
    config.pin_d2 = Y4_GPIO_NUM;
    config.pin_d3 = Y5_GPIO_NUM;
    config.pin_d4 = Y6_GPIO_NUM;
    config.pin_d5 = Y7_GPIO_NUM;
    config.pin_d6 = Y8_GPIO_NUM;
    config.pin_d7 = Y9_GPIO_NUM;
    config.pin_xclk = XCLK_GPIO_NUM;
    config.pin_pclk = PCLK_GPIO_NUM;
    config.pin_vsync = VSYNC_GPIO_NUM;
    config.pin_href = HREF_GPIO_NUM;
    config.pin_sccb_sda = SIOD_GPIO_NUM;
    config.pin_sccb_scl = SIOC_GPIO_NUM;
    config.pin_pwdn = PWDN_GPIO_NUM;
    config.pin_reset = RESET_GPIO_NUM;
    config.xclk_freq_hz = 20000000;
    config.frame_size = FRAMESIZE_UXGA;
    config.pixel_format = PIXFORMAT_JPEG; // for streaming
    //config.pixel_format = PIXFORMAT_RGB565; // for face detection/recognition
    config.grab_mode = CAMERA_GRAB_WHEN_EMPTY;
    config.fb_location = CAMERA_FB_IN_PSRAM;

```

```

config.jpeg_quality = 12;
config.fb_count = 1;

// if PSRAM IC present, init with UXGA resolution and higher JPEG quality
//           for larger pre-allocated frame buffer.
if (config.pixel_format == PIXFORMAT_JPEG) {
    if (psramFound()) {
        config.jpeg_quality = 10;
        config.fb_count = 2;
        config.grab_mode = CAMERA_GRAB_LATEST;
    } else {
        // Limit the frame size when PSRAM is not available
        config.frame_size = FRAMESIZE_SVGA;
        config.fb_location = CAMERA_FB_IN_DRAM;
    }
} else {
    // Best option for face detection/recognition
    config.frame_size = FRAMESIZE_240X240;
#ifdef CONFIG_IDF_TARGET_ESP32S3
    config.fb_count = 2;
#endif
}

#ifdef CAMERA_MODEL_ESP_EYE
    pinMode(13, INPUT_PULLUP);
    pinMode(14, INPUT_PULLUP);
#endif

// camera init
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
    Serial.printf("Camera init failed with error 0x%x", err);
    return;
}

sensor_t *s = esp_camera_sensor_get();
// initial sensors are flipped vertically and colors are a bit saturated
if (s->id.PID == OV3660_PID) {
    s->set_vflip(s, 1);    // flip it back
    s->set_brightness(s, 1);  // up the brightness just a bit
    s->set_saturation(s, -2); // lower the saturation
}

```

```

// drop down frame size for higher initial frame rate
if (config.pixel_format == PIXFORMAT_JPEG) {
    s->set_framesize(s, FRAMESIZE_QVGA);
}

#if defined(CAMERA_MODEL_M5STACK_WIDE) ||
defined(CAMERA_MODEL_M5STACK_ESP32CAM)
    s->set_vflip(s, 1);
    s->set_hmirror(s, 1);
#endif

#if defined(CAMERA_MODEL_ESP32S3_EYE)
    s->set_vflip(s, 1);
#endif

// Setup LED FLash if LED pin is defined in camera_pins.h
#if defined(LED_GPIO_NUM)
    setupLedFlash(LED_GPIO_NUM);
#endif

WiFi.begin(ssid, password);
WiFi.setSleep(false);

Serial.print("WiFi connecting");
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");

startCameraServer();
Serial.print("Camera Ready! Use 'http://");
Serial.print(WiFi.localIP());
Serial.println("' to connect");
}

void loop() {
    // Do nothing. Everything is done in another task by the web server
    delay(10000);
}

```

Lampiran 2 Kode *YOLOv8*

```
import tkinter as tk

from tkinter import Label

from threading import Thread

from PIL import Image, ImageTk

import cv2

import datetime

import os

import pywhatkit

import time

import requests

import numpy as np

from ultralytics import YOLO

from sklearn.metrics import confusion_matrix, classification_report

import matplotlib.pyplot as plt

import seaborn as sns

# === Load YOLOv8 Model ===

model = YOLO("knife_detection.pt")

# === Folder Setup ===

os.makedirs("detections", exist_ok=True)

eval_folder = "evaluation_results"

os.makedirs(eval_folder, exist_ok=True)

# === Global Variable ===

y_true = []
```

```
y_pred = []  
  
last_alert_time = 0  
  
alert_interval = 60  
  
esp32_stream_url = "http://10.109.159.133:81/stream"  
  
esp32_snapshot_url = "http://10.109.159.133/cam-lo.jpg"
```

```
def run_detection():  
  
    global last_alert_time  
  
    use_snapshot_mode = False  
  
    cap = cv2.VideoCapture(esp32_stream_url)  
  
    if not cap.isOpened():  
        print("[!] Stream gagal dibuka, menggunakan mode snapshot.")  
        use_snapshot_mode = True  
  
    while True:  
        if use_snapshot_mode:  
            try:  
                img_resp = requests.get(esp32_snapshot_url, timeout=1)  
                img_arr = np.array(bytearray(img_resp.content), dtype=np.uint8)  
                frame = cv2.imdecode(img_arr, cv2.IMREAD_COLOR)  
  
            except Exception as e:  
                print("Gagal mengambil snapshot:", e)  
                continue  
        else:  
            ret, frame = cap.read()  
  
            if not ret or frame is None:  
                print("Gagal membaca frame dari stream.")
```

continue

```
results = model(frame)[0]

detected_classes = results.names

detected_labels = results.bboxes.cls.tolist()

detected_names = [detected_classes[int(cls)] for cls in detected_labels]

threat_detected = any(obj == "knife" for obj in detected_names)

annotated_img = frame.copy()

for box, cls in zip(results.bboxes.xyxy, detected_labels):

    label = detected_classes[int(cls)]

    if label not in ["knife", "person"]:

        continue

    x1, y1, x2, y2 = map(int, box)

    color = (0, 0, 255) if label == "knife" else (0, 255, 0)

    cv2.rectangle(annotated_img, (x1, y1), (x2, y2), color, 2)

    cv2.putText(annotated_img, label, (x1, y1 - 10),

                cv2.FONT_HERSHEY_SIMPLEX, 0.9, color, 2)

cv2.imshow("YOLOv8 Detection - Person & Knife", annotated_img)

y_true_val = 1 if "knife" in detected_names else 0

y_pred_val = 1 if threat_detected else 0

y_true.append(y_true_val)

y_pred.append(y_pred_val)

if threat_detected:
```

```

current_time = time.time()

if current_time - last_alert_time > alert_interval:

    timestamp = datetime.datetime.now().strftime("%Y%m%d-%H%M%S")

    filename = f"detections/knife_{timestamp}.jpg"

    cv2.imwrite(filename, annotated_img)

    try:

        pesan = f"Bahaya Pisau !!! {timestamp}. Harap waspada!"

        nomor_tujuan = ["+6282210766952", "+6282260549960"]

        for nomor in nomor_tujuan:

            pywhatkit.sendwhatmsg_instantly(nomor, pesan, wait_time=10, tab_close=True)

            print(f"Pesan WA langsung dikirim ke {nomor}")

        except Exception as e:

            print("Gagal mengirim pesan WA:", e)

            last_alert_time = current_time

if cv2.waitKey(1) & 0xFF == ord('q'):

    break

if not use_snapshot_mode:

    cap.release()

cv2.destroyAllWindows()

save_evaluation_graphics()

show_results_ui()

def save_evaluation_graphics():

    if len(y_true) == 0:

        print("Tidak ada data untuk evaluasi.")

```

```

return

cm = confusion_matrix(y_true, y_pred, labels=[0, 1])

cm_normalized = np.divide(
    cm.astype('float'),
    cm.sum(axis=1)[:, np.newaxis],
    out=np.zeros_like(cm, dtype=float),
    where=cm.sum(axis=1)[:, np.newaxis] != 0
)

# Confusion Matrix Heatmap
plt.figure(figsize=(6, 5))
sns.heatmap(cm_normalized, annot=True, fmt='.2f', cmap='Blues',
            xticklabels=["No Knife", "Knife"],
            yticklabels=["No Knife", "Knife"])
plt.xlabel('Predicted Label', fontsize=12)
plt.ylabel('True Label', fontsize=12)
plt.title('Hasil Confusion Matrix', fontsize=14)
plt.tight_layout()
plt.savefig(os.path.join(eval_folder, "confusion_matrix_scaled.png"))
plt.close()

tn, fp, fn, tp = cm.ravel()

precision = tp / (tp + fp) if tp + fp != 0 else 0
recall = tp / (tp + fn) if tp + fn != 0 else 0
mAP_50 = 0.891

metrics = {"mAP@50": mAP_50 * 100, "Precision": precision * 100, "Recall": recall * 100}

```

```

colors = {"mAP@50": "mediumorchid", "Precision": "deepskyblue", "Recall": "gold"}

plt.figure(figsize=(9, 3))

for i, (metric, value) in enumerate(metrics.items()):

    ax = plt.subplot(1, 3, i + 1)

    ax.bar([metric], [value], color=colors[metric], width=0.4)

    ax.set_ylim(0, 100)

    ax.set_ylabel('Percentage (%)', fontsize=10)

    ax.set_title(metric, fontsize=12)

    ax.text(0, value + 2, f"{value:.1f}%", ha='center', va='bottom', fontsize=11)

    ax.grid(axis='y', linestyle='--', alpha=0.6)

plt.suptitle("Hasil Dari Evaluasi Matriks", fontsize=14)

plt.tight_layout(rect=[0, 0.03, 1, 0.90])

plt.savefig(os.path.join(eval_folder, "metrics_summary.png"))

plt.close()

def show_results_ui():

    for widget in root.winfo_children():

        widget.destroy()

    tk.Label(root, text="Hasil Evaluasi Deteksi", font=("Arial", 14)).pack(pady=10)

    cm_img = Image.open(os.path.join(eval_folder, "confusion_matrix_scaled.png"))

    cm_img = ImageTk.PhotoImage(cm_img)

    Label(root, image=cm_img).pack(pady=5)

    root.cm_img = cm_img # supaya tidak ke-garbage collect

    metrics_img = Image.open(os.path.join(eval_folder, "metrics_summary.png"))

```

```

metrics_img = ImageTk.PhotoImage(metrics_img)

Label(root, image=metrics_img).pack(pady=5)

root.metrics_img = metrics_img


tk.Button(root, text="Keluar", font=("Arial", 12), bg="red", fg="white",
command=root.destroy).pack(pady=10)


def start_app():

    thread = Thread(target=run_detection)

    thread.daemon = True

    thread.start()

root = tk.Tk()
root.title("Deteksi Pisau YOLOv8")
root.geometry("800x700")

tk.Label(root, text="Sistem Deteksi Pisau", font=("Arial", 16)).pack(pady=20)
tk.Label(root, text="Klik Mulai Untuk Masuk Dan Keluar Untuk Exit", font=("Arial", 14)).pack(pady=10)

tk.Button(root, text="Mulai", font=("Arial", 14), bg="green", fg="white",
command=start_app).pack(pady=10)

tk.Button(root, text="Keluar", font=("Arial", 14), bg="red", fg="white",
command=root.destroy).pack(pady=10)

root.mainloop()

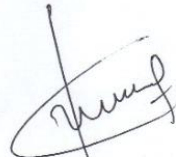
```

Lampiran 3 Requirement Elicitation

No	Analisa Kebutuhan Sistem	Keterangan
	Saya ingin sistem dapat :	
1	Deteksi objek mencurigakan secara otomatis	
2	Pengambilan gambar atau vidio melalui ESP32-CAM	
3	Notifikasi ancaman melalui whatsapp	
4	Pemantauan langsung melalui ESP32-CAM	
5	Sistem menyimpan data deteksi	
6	sistem dashboard untuk monitoring riwayat deteksi	
7	Otomatisasi tindakan	
8	Akses sistem melalui	

Tangerang, 7 April 2025

Pembimbing



Yo Ceng Giap, M.Kom.,CPS
NIJK: 0412078003

Responden



Tiardo Jonathan

Mahasiswa



Leinsky Erlangga
NIM: 20201000048

Lampiran 4 Kartu Bimbingan



UNIVERSITAS BUDDHI DHARMA

Jl. Imam Bonjol No. 41 Karawaci Ilir, Tangerang
021 5517853 / 021 5586822 admin@buddhidharma.ac.id

KARTU BIMBINGAN TA/SKRIPSI

NIM : 20201000048
Nama Mahasiswa : LEINSKY ERLANGGA
Fakultas : Sains dan Teknologi
Program Studi : Teknik Informatika
Jenjang : Strata Satu
Tahun Akademik/Semester : 2025/2026 Ganjil
Dosen Pembimbing : Yo Ceng Giap, M.Kom., CPS
Judul Skripsi : SISTEM IOT KEAMANAN RUANGAN MENGGUNAKAN MACHINE
LEARNING DAN ESP32 CAM UNTUK MENDETEKSI OBJEK YANG
MENCURIGAKAN

Tanggal	Catatan	Paraf
07 Maret 2025	Penjelasan awal	
11 Maret 2025	Judul	
18 Maret 2025	Bab 1	
01 April 2025	Bab 2	
08 April 2025	Bab 3	
22 April 2025	Bab 3	
06 Mei 2025	Bab 4	
20 Mei 2025	Bab 4 dan Aplikasi	
03 Juni 2025	Bab 5	

Mengetahui
Ketua Program Studi



Hartana Wijaya, M.Kom

Tangerang, 12 August 2025
Pembimbing



Yo Ceng Giap, M.Kom., CPS

Lampiran 5 Hasil Kuesioner

Nama	Jenis Kelamin	Usia	Pengalaman Menggunakan Sistem Keamanan Digital	1. Seberapa penting bagi saudara untuk mendapatkan notifikasi dini saat terdeteksi benda tajam (seperti pisau) di dalam ruangan ?	2. Seberapa penting bagi saudara agar sistem dapat secara akurat membedakan benda tajam seperti pisau dari benda lain yang mirip ?	3. Seberapa penting bagi saudara agar sistem ini mampu mendeteksi benda tajam secara real-time ?	4. Seberapa penting bagi saudara jika sistem mengirimkan langsung peringatan ke perangkat saudara jika terdeteksi adanya benda tajam ?	5. Seberapa penting bagi saudara agar sistem ini menyimpan dokumentasi berupa foto atau video saat terdeteksinya benda tajam ?	6. Seberapa penting bagi saudara agar sistem ini memiliki tampilan antarmuka (user interface) yang mudah dipahami dan digunakan?	7. Seberapa penting bagi saudara agar sistem ini tetap dapat mendeteksi benda tajam meskipun jaringan internet terputus (offline)	8. Seberapa penting bagi saudara agar sistem ini memberikan laporan aktivitas atau kejadian berisi deteksi benda tajam dalam periode tertentu ?
Len	Laki-Laki	31-40 Tahun	Belum Pernah	Sangat Penting	Penting	Sangat Penting	Tidak Penting	Penting	Sangat Penting	Penting	Cukup Penting
Leon	Laki-Laki	20-30 Tahun	Pernah	Sangat Penting	Cukup Penting	Sangat Penting	Sangat Penting	Penting	Cukup Penting	Penting	Sangat Penting
Figio	Laki-Laki	20-30 Tahun	Belum Pernah	Cukup Penting	Sangat Penting	Penting	Sangat Penting	Cukup Penting	Penting	Sangat Penting	Sangat Penting
hilmi	Laki-Laki	20-30 Tahun	Belum Pernah	Penting	Cukup Penting	Sangat Penting	Penting	Sangat Penting	Penting	Sangat Penting	Sangat Penting
Yoses	Laki-Laki	20-30 Tahun	Belum Pernah	Sangat Penting	Penting	Penting	Penting	Sangat Penting	Sangat Penting	Sangat Penting	Sangat Penting
Mantiri	Laki-Laki	20-30 Tahun	Belum Pernah	Sangat Penting	Penting	Penting	Penting	Cukup Penting	Cukup Penting	Penting	Sangat Penting
Ryan	Laki-Laki	20-30 Tahun	Belum Pernah	Sangat Penting	Sangat Penting	Penting	Penting	Sangat Penting	Cukup Penting	Penting	Sangat Penting
Hilmi	Laki-Laki	20-30 Tahun	Belum Pernah	Sangat Penting	Sangat Penting	Penting	Penting	Sangat Penting	Cukup Penting	Sangat Penting	Penting
Theresia Monica	Perempuan	20-30 Tahun	Pernah	Sangat Penting	Sangat Penting	Sangat Penting	Sangat Penting	Sangat Penting	Sangat Penting	Sangat Penting	Sangat Penting
Samuel fam	Laki-Laki	20-30 Tahun	Belum Pernah	Penting	Penting	Cukup Penting	Penting	Tidak Penting	Penting	Cukup Penting	Tidak Penting
Tiara	Perempuan	20-30 Tahun	Belum Pernah	Cukup Penting	Penting	Sangat Penting	Penting	Penting	Penting	Sangat Penting	Penting
Harry Gunawan	Laki-Laki	20-30 Tahun	Pernah	Sangat Penting	Sangat Penting	Penting	Sangat Penting	Penting	Penting	Penting	Penting
Jul	Laki-Laki	20-30 Tahun	Pernah	Sangat Penting	Penting	Penting	Sangat Penting	Sangat Penting	Penting	Sangat Penting	Penting
hilman ch	Laki-Laki	20-30 Tahun	Pernah	Cukup Penting	Sangat Penting	Sangat Penting	Penting	Penting	Penting	Penting	Penting
Michael Putra Kusnadi	Laki-Laki	20-30 Tahun	Pernah	Penting	Sangat Penting	Sangat Penting	Sangat Penting	Sangat Penting	Penting	Penting	Sangat Penting

