DAFTAR PUSTAKA

A Gentle Introduction to Semi Supervised Learning | by Gayatri Sharma | Medium. (n.d.). Retrieved 28 December 2023, from https://medium.com/@gayatri_sharma/a-gentle-introduction-to-semi-supervised-learning-7afa5539beea

Abdel-Hamid, S. El, Saleem, A. S. Y., Youssef, M. I., Mohammed, H. H., & Abdelaty, A. I. (2020). Influence of housing systems on duck behavior and welfare. *Journal of Advanced Veterinary and Animal Research*, *7*(3), 407–413. https://doi.org/10.5455/javar.2020.g435

Abdul Muiz Khalimi. (n.d.). *Perhitungan Confusion Matrix Multi-Class Clasification 3x3*. Retrieved 30 June 2023, from https://www.pengalaman-edukasi.com/2020/11/menghitung-confusion-matrix-3-kelas.html

adminweb. (2022, October 26). *Kenali Apa Itu Internet of Things, Cara Kerja & Manfaatnya*. Bluepowertechnology. https://ofis.bluepowertechnology.com/blog-detail/kenali-apa-itu-internet-of-things-cara-kerja-manfaatnya/

Ahmad Roihan, Po Abas Sunarya, & Ageng Setiani Rafika. (2019). Pemanfaatan Machine Learning dalam Berbagai Bidang. *IJCIT (Indonesian Journal on Computer and Information Technology)*, *5*(1), 75–82.

Angga Yana, Iwan Setiawan, & Dani Garnida. (2016). Behaviour Exploration Of Muscovyduck (Cairina Moschata) On Incubating Ducks Eggin Wet And Dry Condition. *FakultasPeternakan, Universitas Padjadjaran*.

Buntoro, G. A. (2017). Analisis Sentimen Calon Gubernur DKI Jakarta 2017 Di Twitter. *INTEGER: Journal of Information Technology*, *2*(1). https://doi.org/10.31284/J.INTEGER.2017.V2I1.95

Deepanshu Tyagi. (2019, September 1). *Introduction to ORB (Oriented FAST and Rotated BRIEF) | by Deepanshu Tyagi | Medium*. 1 Januari 2019. https://medium.com/@deepanshut041/introduction-to-orb-oriented-fast-and-rotated-brief-4220e8ec40cf

Dini Widianigrum, Tuti Widyastuti, Asep Anang, & Iwan Setiawan. (2021). Mating Behavior Of Ciayumajakuning Muscovy Duck In The Caged Maintenance System. *Jurnal Ilmu Pertanian Dan Peternakan*, *09*, 121–130.

https://doi.org/https://doi.org/10.31949/Agrivet/V9i2.1722

doxygen. (2019, April 21). *Feature Matching*. Man.Hubwiz.Com. http://man.hubwiz.com/docset/OpenCV.docset/Contents/Resources/Documents/dc/dc3/tutorial_py_matcher.html

espressif. (2021, May 23). *ESP8266 Wi-Fi MCU I Espressif Systems*. Espressif. https://www.espressif.com/en/products/socs/esp8266

hewanpedia. (2021, April 17). *Definisi Unggas - Hewanpedia*. Hewanpedia.Com. https://hewanpedia.com/definisi-unggas/

Khoharja, G., & Nathania Purbowo, A. (n.d.). *Aplikasi Deteksi Nilai Uang pada Mata Uang Indonesia dengan Metode Feature Matching*.

Kusban, M. (2011). *Morphological Image Processing*. www.mathworks.com

Maisarah, M., Ningsih, S., Maisari, S., Atifah, Y., Biologi, J., Matematika, F., Ilmu, D., & Alam, P. (n.d.). *Karakteristik dan Tingkah Laku Ayam Kampung (Gallus gallus domesticus) Saat Memasuki Musim Kawin (Characteristics and Behavior of Local Chickens (Gallus gallus domesticus) When Entering the Mating Season)*.

Marhamah. (2020). *IMPLEMENTASI ALGORITMA SUPPORT VECTOR MACHINE (SVM) DALAM MEMPREDIKSI KELULUSAN JALUR MASUK PERGURUAN TINGGI BANDA ACEH (STUDI KASUS MAHASISWA BARU TAHUN AJARAN 2019)*. UNIVERSITAS ISLAM NEGERI AR-RANIRY.

Marpaung, F., Aulia, F., Suryani SKom, N., & Cyra Nabila SKom, R. (n.d.). *COMPUTER VISION DAN PENGOLAHAN CITRA DIGITAL*. www.pustakaaksara.co.id

Nurul Huda. (2022, October 18). *Edge Computing: Pengertian, Kelebihan, dan Kekurangannya*. Dewaweb. https://www.dewaweb.com/blog/edge-computing/

Penjelasan Sederhana tentang Apa Itu SVM? | by Samsudiney | Medium. (n.d.). Retrieved 25 June 2023, from https://medium.com/@samsudiney/penjelasan-sederhana-tentang-apa-itu-svm-149fec72bd02

Pratomo, A. H., Kaswidjanti, W., & Mu'arifah, S. (2020). *IMPLEMENTASI ALGORITMA REGION OF INTEREST (ROI) UNTUK MENINGKATKAN PERFORMA ALGORITMA DETEKSI DAN KLASIFIKASI KENDARAAN*

IMPLEMENTATION OF REGION OF INTEREST (ROI) ALGORITHM TO IMPROVE CAR DETECTION AND CLASSIFICATION ALGORITHM. *7*(1), 155–162. https://doi.org/10.25126/jtiik.202071718

Prof Ali Agus. (2018, September 1). *Ancaman Impor Daging Itik*. Media Agrebisnis Peternakan. http://troboslivestock.com/detail-berita/2018/09/01/22/10677/prof-ali-agus--ancaman-impor-daging-itik

Rajawali. (2021, April 8). *IMOU Cue 2 IPC-C22EP (2MP Wifi CCTV Camera) 1080p*. Tokopedia. https://www.tokopedia.com/rajawalicompmksr/imou-cue-2-ipc-c22ep-2mp-wifi-cctv-camera-1080p?extParam=ivf%3Dfalse%26src%3Dsearch

Ramadhani, M. (2018). *Klasifikasi Ikan Menggunakan ORB-PCA dan KNN*. INSTITUT TEKNOLOGI SEPULUH NOPEMBER.

Ramadhani, M., Darlis, D., & Murti, H. (2018). KLASIFIKASI IKAN MENGGUNAKAN ORIENTED FAST AND ROTATED BRIEF (ORB)DAN K-NEAREST NEIGHBOR (KNN). *Jurnal Ilmiah Teknologi Informasi*, *16*(2), 115–124.

Reza Fadhlurrohman, Delvia Fitri Suarman, Muhammad Zainal Umar, & Yusni Atifah. (2021). Pengaruh Faktor Lingkungan Terhadap Reproduksi Ayam Ras Petelur. *Prosiding SEMNAS BIO 2021*.

Rui Santos, & Sara Santo. (2019, April 23). *ESP32 with BME280 Sensor using Arduino IDE (Pressure, Temperature, Humidity)*. Randomnerdtutorials. https://randomnerdtutorials.com/esp32-bme280-arduino-ide-pressure-temperature-humidity/

Saptadi, A. H., & Kiswanto, A. (2020). Rancang Bangun Web Server Penampil Data Cuaca Berbasis Arduino Menggunakan Sensor BME280 dan BH1750FVI dengan Tiga Mode Tampilan Data. *Jurnal Teknik Elektro Dan Komputasi (ELKOM)*, *2*(2), 112–121. https://doi.org/10.32528/elkom.v2i2.3516

Shpakov, O. N., & Bogomolov, G. V. (1981). PENGOLAHAN CITRA DIGITAL UNTUK MENDETEKSI OBYEK MENGGUNAKAN PENGOLAHAN WARNA MODEL NORMALISASI RGB. *Studies in Environmental Science*, *17*(C), 329–332. https://doi.org/10.1016/S0166-1116(08)71924-1

Siti Badriyah. (2018, August 17). *Pengertian Konvensional dan Contoh Penerapannya Dalam Masyarakat*. Gramedia Blog. https://www.gramedia.com/literasi/memahami-istilah-konvensional/

Vijaysinh Lendave. (2021, August 25). *Using Background Subtraction Methods in Image Processing*. Analyticsindiamag. https://analyticsindiamag.com/using-background-subtraction-methods-in-image-processing/

Wang, J., Wang, N., Li, L., & Ren, Z. (2020). Real-time behavior detection and judgment of egg breeders based on YOLO v3. *Neural Computing and Applications*, *32*(10), 5471–5481. https://doi.org/10.1007/s00521-019-04645-4

Yasin K. (2022, April 24). *Pengertian MySQL, Fungsi, dan Cara Kerjanya (Lengkap)*. Niagahoster. https://www.niagahoster.co.id/blog/mysql-adalah/

Ye, P., Li, M., Liao, W., Ge, K., Jin, S., Zhang, C., Chen, X., & Geng, Z. (2019). Hypothalamic transcriptome analysis reveals the neuroendocrine mechanisms in controlling broodiness of Muscovy duck (Cairina moschata). *PLoS ONE*, *14*(5). https://doi.org/10.1371/journal.pone.0207050

Yuliani, Tonny Suhendra, & Hollanda Arief Kusuma. (2022). ANALISIS PERBANDINGAN AKURASI PADA SENSOR TEKANAN BMP280 DAN BME280. *Student Online Jurnal*, *3*(1).

Zhuang, X., & Zhang, T. (2019). Detection of sick broilers by digital image processing and deep learning. *Biosystems Engineering*, *179*, 106–116. https://doi.org/10.1016/j.biosystemseng.2019.01.003

LAMPIRAN

1. Kode Program Penenalan Pola Gambar.

```
#Import Necessary Libraries
import cv2
import numpy as np
import datetime
import matplotlib.pyplot as plt
import os
import pickle
import time
import os
import pickle
from tkinter import Tk
from tkinter.filedialog import askopenfilename
import matplotlib.pyplot as plt
from skimage.io import imread
from skimage.transform import resize

#Fungsi Waktu
t = datetime.datetime.now()

#Melakukan Pembacaan Video
cap = cv2.VideoCapture('data-video/data-video (10).avi')

#Ekstraksi Frame
output_folder1 = 'fgmask-background-substraction'
if not os.path.exists(output_folder1):
    os.makedirs(output_folder1)

#Perulangan Mulai dari Nol
frame_count = 0

#Ektraksi fitur ORB Template
image_template = cv2.imread('template-matching/mengeram.png', 0)

#Penetapan Waktu untuk Database
current_time = datetime.datetime.now()
kernel = None

#Deteksi Background Substraction
backgroundObject = cv2.createBackgroundSubtractorMOG2(detectShadows=True)

#Deteksi fitur keypoint dengan ORB
def ORB_detector(new_image, image_template):
    image1 = cv2.cvtColor(new_image, cv2.COLOR_BGR2GRAY)
    orb_detector = cv2.ORB_create(500, 1.2)
    kp1, des1 = orb_detector.detectAndCompute( image1, None)
    kp2, des2 = orb_detector.detectAndCompute( image_template, None)
    bf = cv2.BFMatcher(cv2.NORM_HAMMING, crossCheck=True)
    matches = bf.match(des1, des2)
    matches = sorted(matches, key=lambda val: val.distance)
    return len(matches)
```

```
#Perulangan video
if (cap.isOpened() == False):
   print("Error opening video stream or file")

while (cap.isOpened()):
   ret, frame = cap.read()

#ROI 1
x1, y1 = 19, 80
x4, y4 = 99, 190
cv2.rectangle(frame, (x1, y1), (x4, y4), (0, 255, 0), 2)
roi_1 = frame[y1:y4, x1:x4]

#ROI 2
   x1, y1 = 102, 39
   x4, y4 = 600, 425
   cv2.rectangle(frame, (x1, y1), (x4, y4), (0, 255, 0), 2)
   roi_2 = frame[y1:y4, x1:x4]

#Pengenalan POLA dengan ORB
matches = ORB_detector(roi_1, image_template)
   output_string = "Mengeram: " + str(matches) + "%"
   if matches > 25:
      # 19 25
      cv2.putText(frame, output_string, (20, 70), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 0,
255), 2)
      # print("Mengeram", current_time)
      print_text = "Mengeram "
   else:
      cv2.putText(frame, "", (20, 70), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 0, 255), 2)
      # print("", current_time, "d")

#Pengenalan POLA Background Substraction dan SVM
fgmask = backgroundObject.apply(roi_2)
   _, fgmask = cv2.threshold(fgmask, 250, 255, cv2.THRESH_BINARY)
   fgmask = cv2.erode(fgmask, kernel, iterations=1)
   fgmask = cv2.dilate(fgmask, kernel, iterations=10)
   contours, _ = cv2.findContours(fgmask, cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE)

   for cnt in contours:
      if cv2.contourArea(cnt) > 2000:
         x, y, w, h = cv2.boundingRect(cnt)

         Categories = ['Mating', 'Not-Mating']
         # print(os.path.abspath(os.getcwd()))
         model = pickle.load(open('model-220.p', 'rb'))

         # Adjust the width and height to 100x100
         w = 100
         h = 100
         # Calculate the new x and y coordinates to keep the center of the bounding box the
same
         x = x + int((w - 100) / 2)
         y = y + int((h - 100) / 2)

         roi_box = roi_2[y:y+h, x:x+w]
         img_resize = resize(roi_box, (150, 150, 3))
```

```
        l = [img_resize.flatten()]
        probability = model.predict_proba(l)

        cv2.rectangle(roi_2, (x, y), (x + w, y + h), (0, 0,255 ), 2)
        cv2.putText(roi_2, Categories[model.predict(l)[0]], (x, y - 10),
cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 0, 255), 2, cv2.LINE_AA)
        # print("Detected and Croping", current_time)
        # print(Categories[model.predict(l)[0]], current_time)
        perilaku = Categories[model.predict(l)[0]]
        print_text=perilaku;
        # if perilaku == 'Mating':
        #     print_text = 'Mating'

    # Extract the foreground from the frame using the segmented mask.
    foregroundPart = cv2.bitwise_and(roi_2, roi_2, mask=fgmask)
    # Stack the original frame, extracted foreground, and annotated frame.
    stacked = np.hstack((roi_2, foregroundPart, roi_2))

    #Menampilkan Hasil
    if ret == True:
        cv2.imshow('FRAME ROI-1 DAN ROI-2', frame)
        cv2.imshow('MASKING', fgmask)
```

2. Kode Program Training SVM

```
#Impor Library
import pandas as pd
from sklearn import svm
from sklearn.model_selection import GridSearchCV
import os
import cv2
import matplotlib.pyplot as plt
from skimage.transform import resize
from skimage.io import imread
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report,accuracy_score,confusion_matrix
import pickle
from PIL import Image

#Pengenalan Data
mypath= 'training/'
file_name = []
tag = []
full_path = []
for path, subdirs, files in os.walk(mypath):
    for name in files:
        full_path.append(os.path.join(path, name))
        tag.append(path.split('/')[-1])
        file_name.append(name)

# memasukan variabel yang sudah dikumpulkan pada looping di atas menjadi sebuah
dataframe agar rapih
df = pd.DataFrame({"path":full_path,'file_name':file_name,"tag":tag})
df.groupby(['tag']).size()
```

```
#Pengkategorian Data
Categories=['mating','not-mating']
flat_data_arr=[]
target_arr=[]
datadir='training/'
for i in Categories:
    print(f'loading... category : {i}')
    path=os.path.join(datadir,i)
    for img in os.listdir(path):
        img_array=imread(os.path.join(path,img))
        img_resized=resize(img_array,(150,150,3))
        flat_data_arr.append(img_resized.flatten())
        target_arr.append(Categories.index(i))
        print(f'loaded category:{i} successfully')
        flat_data=np.array(flat_data_arr)
        target=np.array(target_arr)
        df=pd.DataFrame(flat_data)
        df['Target']=target
```

```
#Split Data
x=df.iloc[:,:-1]
y=df.iloc[:,-1]
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=77,stratify=
y)
print('Splitted Successfully')
```

```
#Pelatihan Model
param_grid={'C':[0.1,1,10,100],'gamma':[0.0001,0.001,0.1,1],'kernel':['rbf','poly']}
svc=svm.SVC(probability=True)
print("Training model dimulai, Mohon menunggu sampai proses Traing model selesai")
model=GridSearchCV(svc,param_grid)
model.fit(x_train,y_train)
print('Model Telah berhasil di Training')
model.best_params_
```

3. Kode Program Pembacaan Sensor

```
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
#include <WiFiClientSecureBearSSL.h>
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>

// Replace with your network credentials
const char* ssid     = "NSAndi";  //nama ssid wifi, hanya support wifi "2,5Gh"
const char* password = "123456788"; //pasword wifi

// REPLACE with your Domain name and URL path or IP address with path
const char* serverName = "https://192.168.42.63/sensordata/post-esp-data.php";
//const char* serverName = "http://192.168.42.63/post-esp-data.php";

// Keep this API Key value to be compatible with the PHP code provided in the project page.
// If you change the apiKeyValue value, the PHP file /post-esp-data.php also needs to have the
same key
String apiKeyValue = "tPmAT5Ab3j7F9";
```

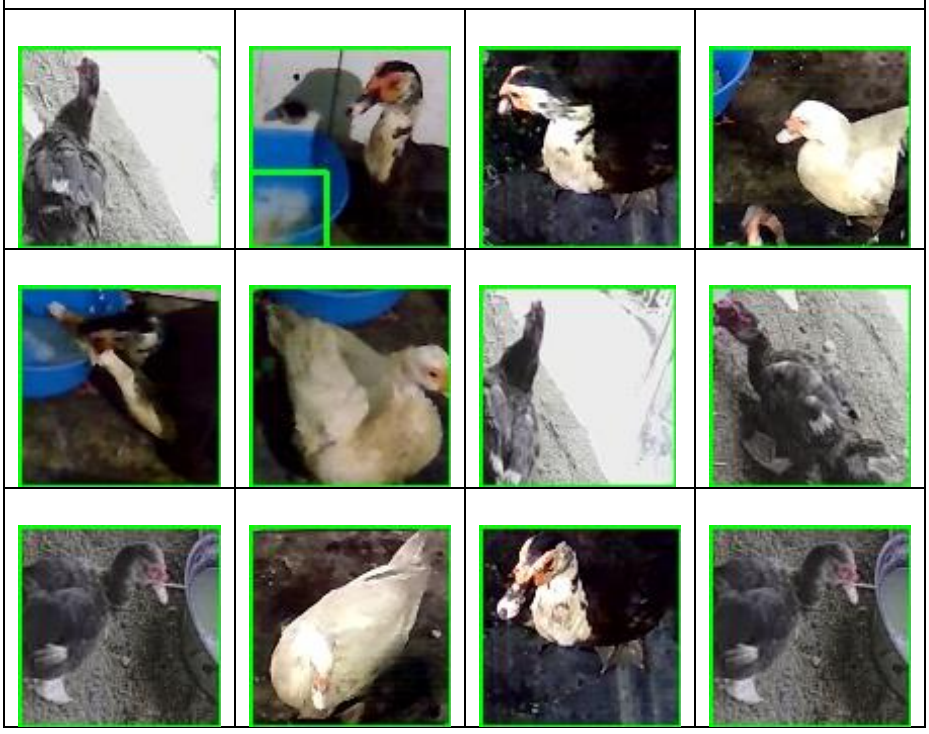
```
String sensorName = "BME280";
String sensorLocation = "Office";

/*#include <SPI.h>
#define BME_SCK 18
#define BME_MISO 19
#define BME_MOSI 23
#define BME_CS 5*/

#define SEALEVELPRESSURE_HPA (1013.25)

Adafruit_BME280 bme;  // I2C
//Adafruit_BME280 bme(BME_CS);  // hardware SPI
//Adafruit_BME280 bme(BME_CS, BME_MOSI, BME_MISO, BME_SCK);  // software SPI

void setup() {
  Serial.begin(115200);

  WiFi.begin(ssid, password);
  Serial.println("Connecting");
  while(WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.print("Connected to WiFi network with IP Address: ");
  Serial.println(WiFi.localIP());

  // (you can also pass in a Wire library object like &Wire2)
  bool status = bme.begin(0x76);
  if (!status) {
    Serial.println("Could not find a valid BME280 sensor, check wiring or change I2C
address!");
    while (1);
  }
}

void loop() {
  //Check WiFi connection status
  if(WiFi.status()== WL_CONNECTED){

    std::unique_ptr<BearSSL::WiFiClientSecure>client(new BearSSL::WiFiClientSecure);

    // Ignore SSL certificate validation
    client->setInsecure();

    //create an HTTPClient instance
    HTTPClient https;

    // Your Domain name with URL path or IP address with path
    https.begin(*client, serverName);

    // Specify content-type header
    https.addHeader("Content-Type", "application/x-www-form-urlencoded");

    // Prepare your HTTP POST request data
    String httpRequestData = "api_key=" + apiKeyValue + "&sensor=" + sensorName
```

```
                + "&location=" + sensorLocation + "&value1=" +
String(bme.readTemperature())
                + "&value2=" + String(bme.readHumidity()) + "&value3=" +
String(bme.readPressure()/100.0F) + "";
   Serial.print("httpRequestData: ");
   Serial.println(httpRequestData);

   // You can comment the httpRequestData variable above
   // then, use the httpRequestData variable below (for testing purposes without the BME280
sensor)
   //String                    httpRequestData                    =
"api_key=tPmAT5Ab3j7F9&sensor=BME280&location=Office&value1=24.75&value2=49.
54&value3=1005.14";

   // Send HTTP POST request
   int httpResponseCode = https.POST(httpRequestData);

   // If you need an HTTP request with a content type: text/plain
   //http.addHeader("Content-Type", "text/plain");
   //int httpResponseCode = https.POST("Hello, World!");

   // If you need an HTTP request with a content type: application/json, use the following:
   //http.addHeader("Content-Type", "application/json");
   //int                    httpResponseCode                    =
https.POST("{\"value1\":\"19\",\"value2\":\"67\",\"value3\":\"78\"}");

   if (httpResponseCode>0) {
    Serial.print("HTTP Response code: ");
    Serial.println(httpResponseCode);
   }
   else {
    Serial.print("Error code: ");
    Serial.println(httpResponseCode);
   }
   // Free resources
   https.end();
  }
  else {
   Serial.println("WiFi Disconnected");
  }
  //Send an HTTP POST request every 30 seconds
  delay(30000);
}
```

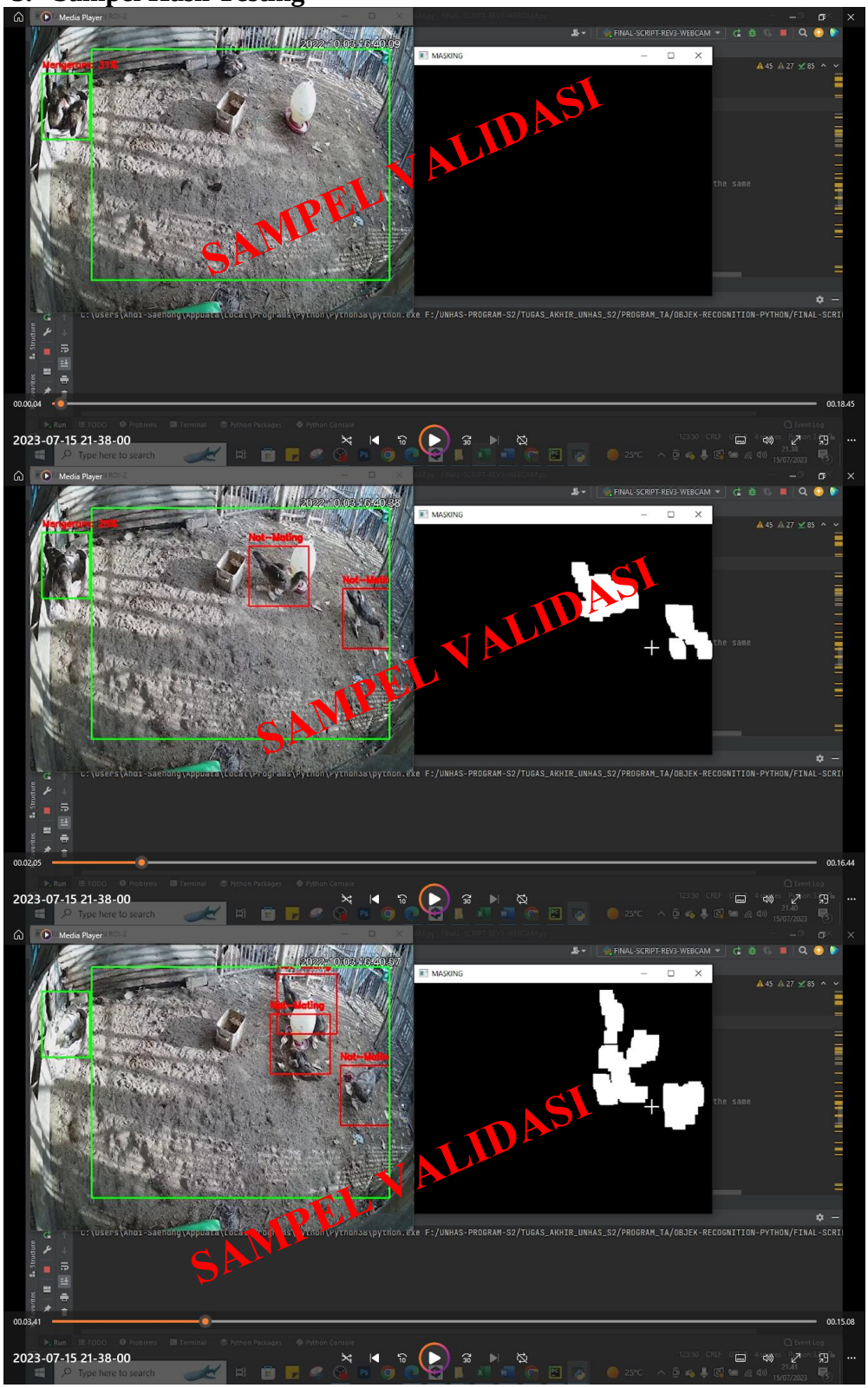
4. Sampel Training

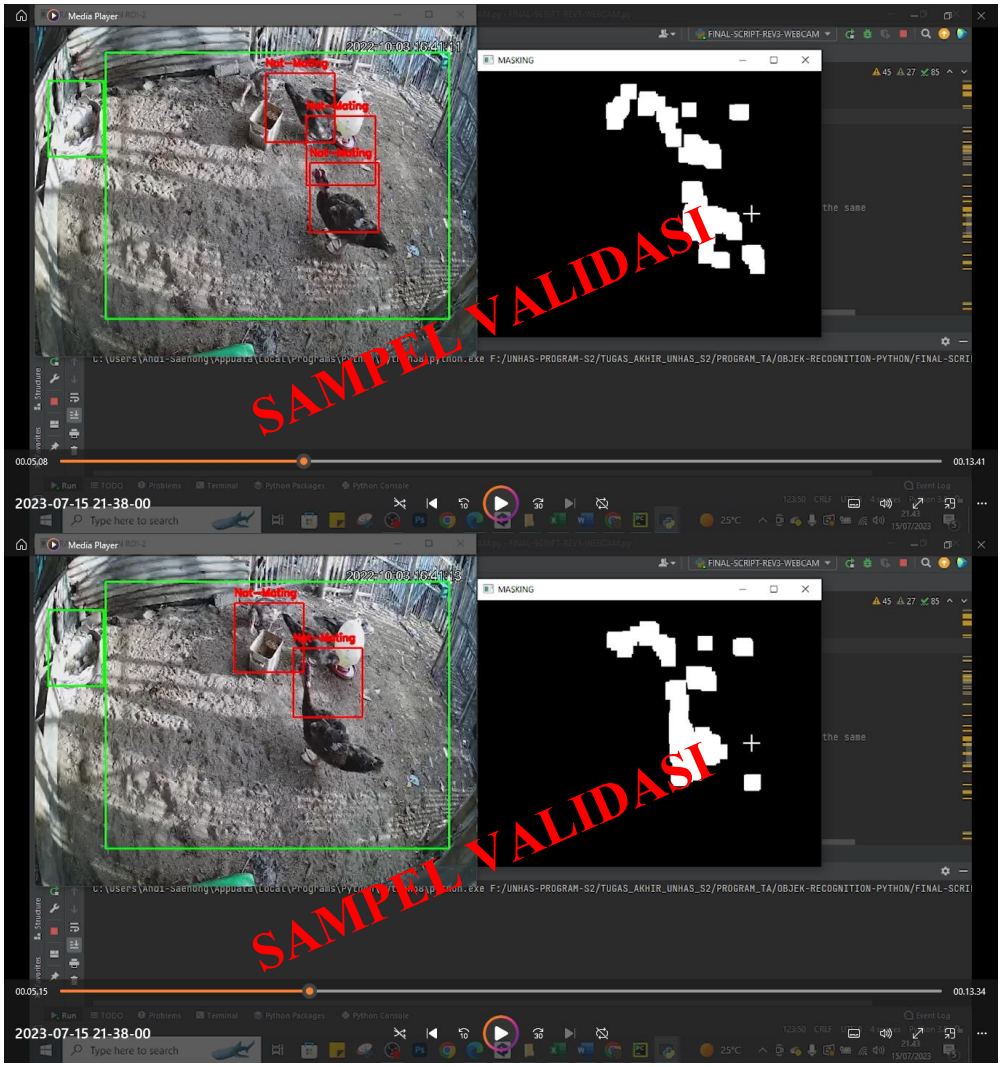
Sampel Perilaku Kawin unggas

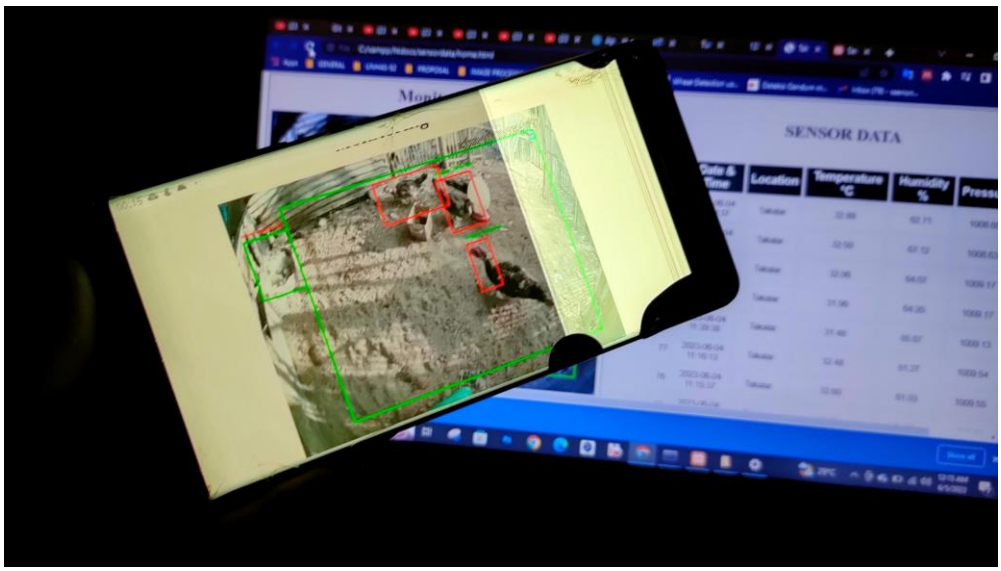
Sampel Perilaku Tidak Kawin



5. Sampel Hasil Testing

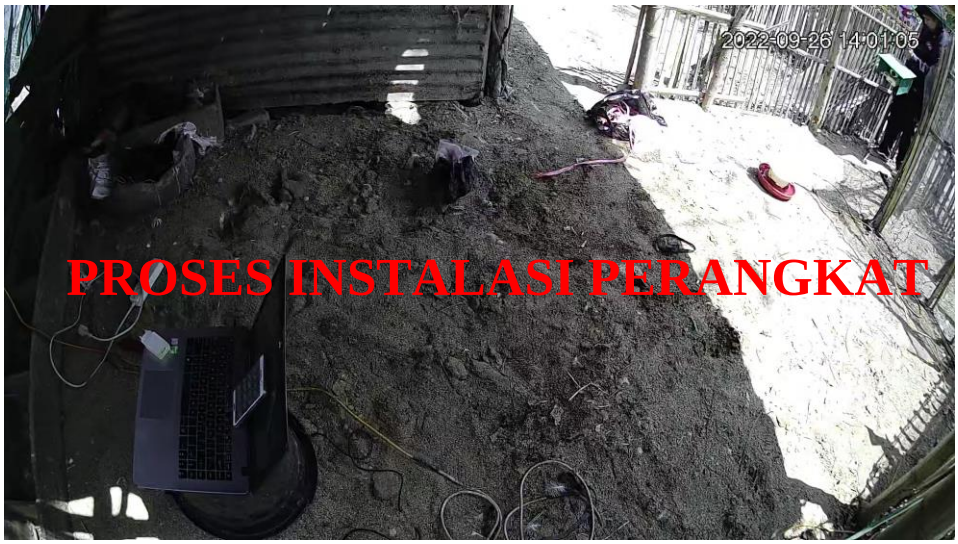






PENGUJIAN MENGGUNAKAN JARINGAN INTERNET


Media Player
PENGUJIAN SECARA REAL-TIME
VID20230708135246
Type here to search


2022-09-26 14:01:05
PROSES INSTALASI PERANGKAT


Monitoring
SENSOR DATA
PENGUJIAN KINERJA DATABASE