Information and Technology Vol. 5, No. 2, 2025 , Pages 1-11

Analysis of Face Detection with Head Accessories Using Haar Cascade in Image Processing

Heru Susanto*, Yuza Reswan, Nuri David Maria Veronika, Rozali Toyib

Universitas Muhammadiyah Bengkulu

DOI: https://doi.org/10.53697/jkomitek.v5i2.3146
*Correspondence: Heru Susanto
E-mail: herustudy11@gmail.com

Received: 22-10-2025
Accepted: 22-11-2025
Published: 22-12-2025

Abstract: This research is entitled Analysis of Face Detection With Accessories On The Head Using Haar Cascade In Image Processing. The problem faced is how accurate the haar cascade algorithm is in detecting faces with accessories on the head, whether the haar cascade algorithm can still detect faces even though there are accessories that are felt around the face, therefore a study was conducted on face detection with accessories on the head using the haar cascade algorithm in several facial conditions such as facial angles and in several lighting conditions in the detection process. This study aims to analyze the accuracy of the haar cascade algorithm in detecting faces with accessories on the head, whether haar cascade can still detect faces even though there are accessories around the face that block, and 20 facial data sets were taken with a real-time laptop camera. The facial data obtained was taken in several places such as campus areas, houses and boarding houses and with various lighting conditions. From 20 face data taken in real time, researchers obtained 2 face data that were not detected due to poor lighting when taking face data and obtained 18 face data that were successfully detected even though using accessories around the face and it is certain that haar cascade has very good accuracy in detecting faces even though they were blocked by accessories. The testing methods applied are precision, recall and accuracy for calculating the results obtained. The results of the study obtained a precision of 100%, recall 90% and accuracy of 90%.

Keywords: Face Detection, Haar Cascade Algorithm, Image Processing, Accuracy

Introduction

The Haar Cascade Classifier algorithm is used for the process of detecting faces or objects in the form of digital images, this algorithm displays a mathematical function in the form of a box by displaying the RGB value at each pixel, after that Viola-Jones developed this algorithm, where each box is processed and produces several values in the form of dark and light areas, and these values will be used as the basis for image processing so that it is known as the Haar-Like Feature (Mantara et al, 2022) . The Haar Cascade algorithm is a popular approach to detect objects in images, which can be applied in various fields such as face recognition, object recognition, and others. In this thesis, it is possible to discuss how the Haar Cascade technique is used to identify faces in images, as well as the analysis of the performance and detection accuracy achieved by the method. The main objective is to analyze the effectiveness and reliability of this face detection method in the context of image processing.

In today's era where technology is increasingly integrating itself into everyday life, image processing has become one of the fields that has received great attention. One important aspect of image processing is face detection, which has various important

applications in various functions such as security, facial recognition, and human-machine interaction. Face detection is also often known as facial identification, is a computer vision-based AI technology used to find and recognize human faces in images, digital videos and even using a device's real-time camera.

In computer science there is a science of artificial intelligence systems where this system applies intelligence like humans. With artificial intelligence, computers will be trained with a lot of data that aims to recognize an object. By detecting facial objects, security systems such as object verification can be created. Face detection methods have experienced significant developments in recent years. There are many algorithms that can be used so that computers can be trained to recognize an object, one of which is the haar cascade classifier algorithm. This *haar cascade classifier algorithm* is a method developed by viola and jones in 2001 (Nugroho, 2024) .

Methodology

This study aims to conduct an in-depth analysis of the use of Haar Cascade in face detection with head accessories in image processing. Researchers will explore the basic principles of the Haar Cascade algorithm and apply and test its performance in various accessory scenarios. Researchers will analyze the strengths and weaknesses of this approach, as well as explore possible improvements for the future. Through this study, it is hoped that a better understanding will be obtained on how Haar Cascade can be optimized for better face detection with accessories.

Problem Analysis and Program Design

Research Based on the title of the research , the Malsallalh taken, this research focuses on the main Malsallalh, namely analyzing the accuracy and accuracy of face detection with accessories on the head using the *Haar cascade algorithm.*

Therefore, research that has been conducted shows that using the *Haar Cascade method* and *the open cv Python library* has a high level of accuracy and accuracy in detecting faces with several accessories around the face.

Identification of problems

Based on the problems that arise, this research focuses on a problem, namely analyzing the results of the accuracy and accuracy of face detection using accessories around the face on human facial objects using *the Haar Cascade algorithm* and *the Open CV library.*

Python, can the Haar Cascade algorithm still detect faces with accessories around the face accurately and precisely.

Program Design

At this stage it will be designed and tested using a real-time laptop camera to capture the face that will be detected later, the facial data that will be taken is 20 facial data sets in real time and the data will be taken in several places later and will be tested directly using the haar cascade algorithm and with the open cv library and the python programming language which will be tested completely on *Visual Studio Code (VS Code) software* and after

this testing stage the data that has been obtained will be collected for further processing and analysis.

Facial data testing

At the beginning of this data collection, the researcher began to take or collect data in the form of faces with several accessories around the face totaling 20 different facial objects, with different facial positions and conditions of observation that will be adjusted to the location of the facial data collection.

Results and Discussion

After the previous testing process, from 20 facial data sets that have been tested in real time using a laptop camera and *Visual Studio Code (VS Code) software,* the results obtained were that almost all of the 20 data were successfully detected well even though the facial data objects used several kinds of accessories such as hats, glasses, headbands, hoodies, helmets, skullcaps and from different facial angles, as well as in several places such as campus areas, homes and boarding houses with varying lighting quality so as to obtain the expected facial detection results with accessories using *the haar cascade algorithm* and *the python open cv library*.

Discussion

The process of implementing research to detect faces with accessories around the face with *haar cascade* and *the open cv python library* and using *software* that has been installed to run the program with the python programming language, namely *Visual Studio Code (VS Code)* and using a computer/laptop device with *HP elitebook 840 G1 laptop specifications* and *an Intel Core i5 processor, 8GB RAM* and Windows 10.

The steps in this research are as follows:

The process of running a program on Visual Studio Code software

The first step taken by the researcher was to run the program using *Visual Studio Code software, Visual Studio Code* is *a* very lightweight, yet powerful source code editor that runs from the desktop. It comes with built-in support for *JavaScript,* TypeScript, and *Node.js* and has a diverse array of extensions available for other languages, including C++, C #, *Python, and PHP.* (Erika & Saputro, 2021) . to run the program using the python language and also with *the open Cv library* to connect the haar cascade algorithm so that later it can detect faces with the haar cascade.

Figure 1. Initial display image of VS code software

Before running the program on *Visual Studio Code Software*

Place the prepared Python Open Cv library file into the folder that will be used to run the face detection program with accessories on the face , so that later the face detection program with Haar Cascade can be connected to the prepared Python program of course.

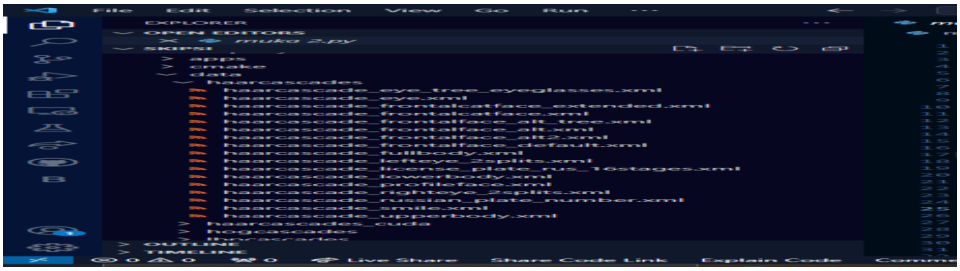


Figure 2. Image of the Open CV library display

Image processing process in python in visual studio code.

In the process of calling a face with Haar Cascade, using a command that is connected to the Open CV Python library and from the Haar Cascade data set that already exists in the Open CV library, it can be seen as below.

```
face_cascade=cv2.CascadeClassifier('opencv4.x/data/haarcascades/haarcasca

de_frontalface_default.xml')
```

After that, from the first step that has been carried out, the program will open the camera that will be used for detection.

```
cap = cv2.VideoCapture(0)
```

Next, the program will read the frame from the camera that has been connected, as in the program below.

```
ret, frame = cap.read()
```

And after that the frame will be converted to Grayscale and the face will be dedetected with the Haar Cascade command that has been applied.

```
gray=cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
```

⇩

```
faces = face_cascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5)
```

And then the program will display a box image and will circle the face that will be detected later, this box image functions as the center of focus on the detected face as well as the point of face detection with accessories on the face , such as the program below.

```
for (x, y, w, h) in faces: cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0), 2)
```

Then the face frame will be displayed in the box image that was displayed before as per the command below.

```
cv2.imshow('Wajah', frame)
```

Above are the program steps in detecting faces with accessories on the face using haar cascade and the python open cv library so as to obtain 20 facial data sets from real-time detection results and screenshots from the device directly as shown in the image below.

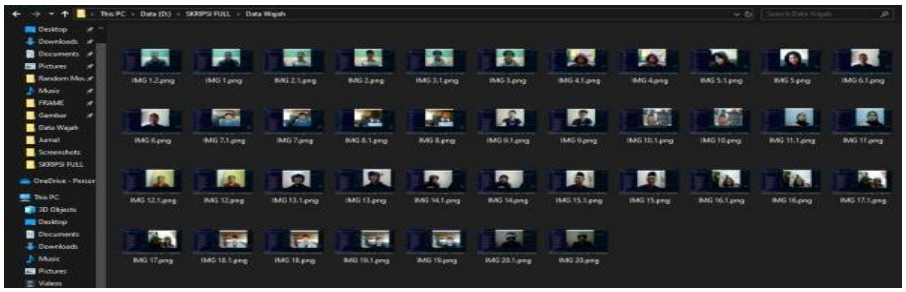


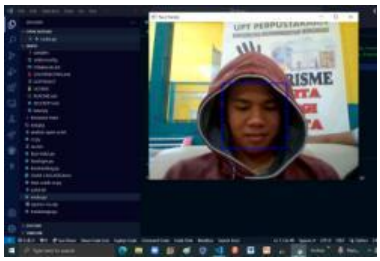
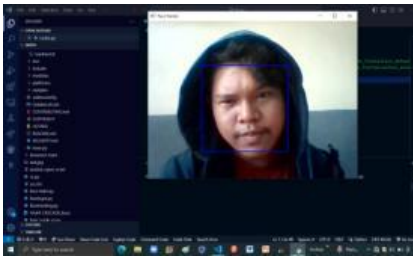
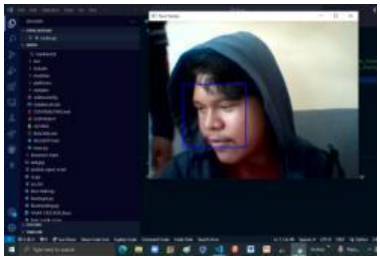
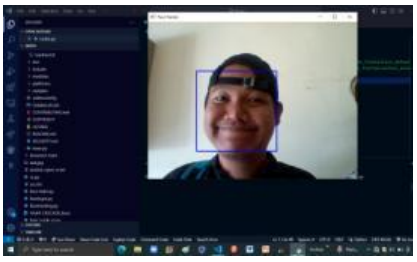
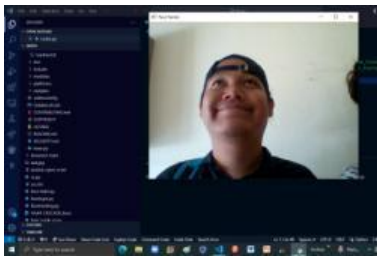
Figure 3. Figure 20 facial datasets obtained

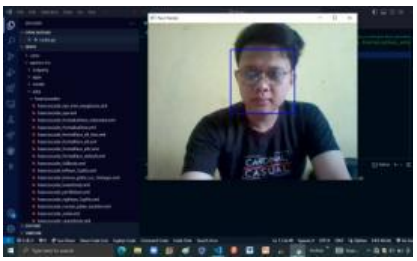
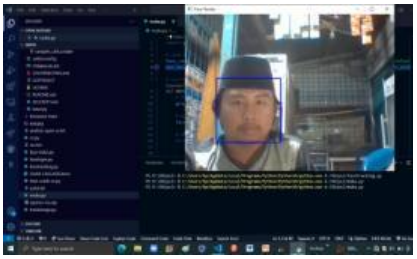
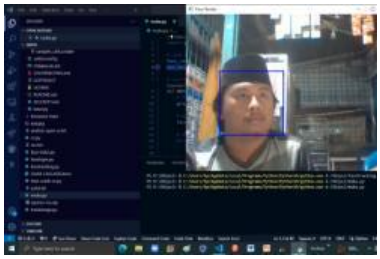
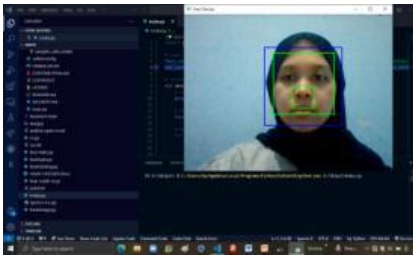
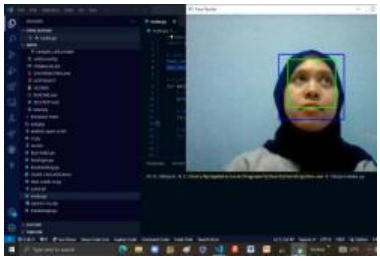
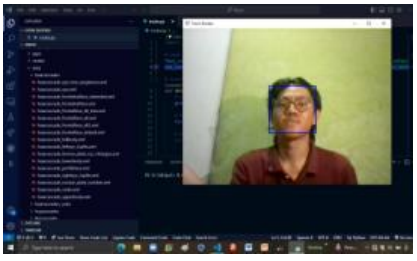
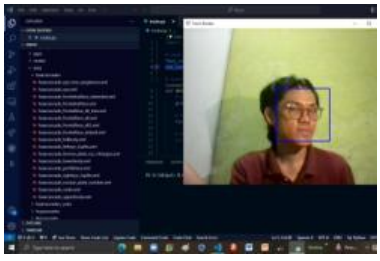
Face detection results with accessories around the face

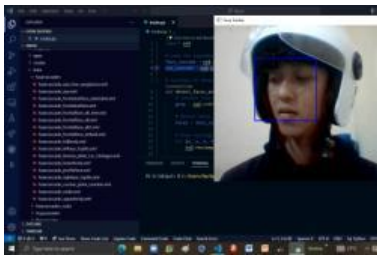
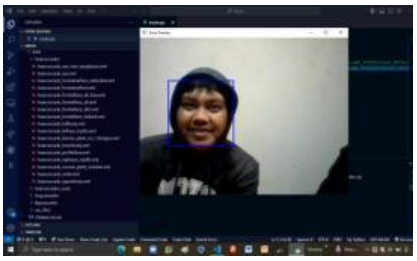
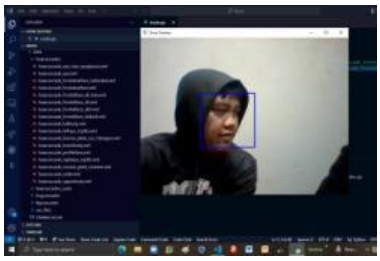
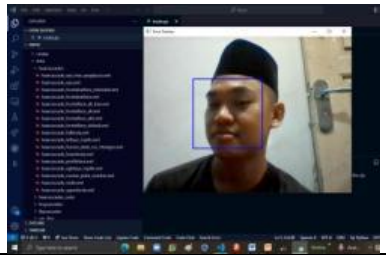
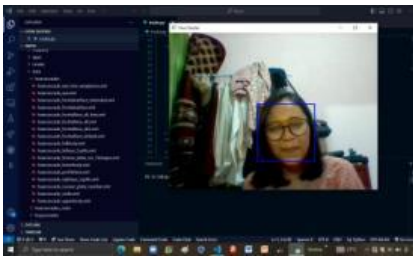
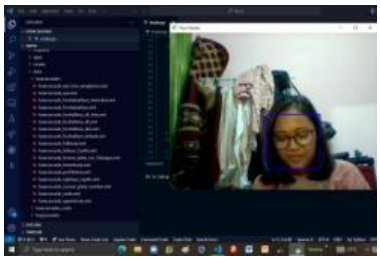
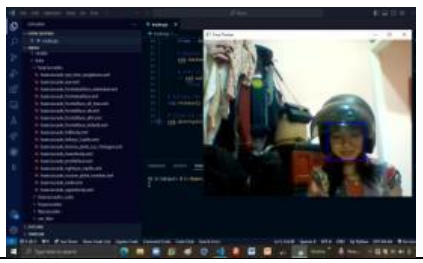
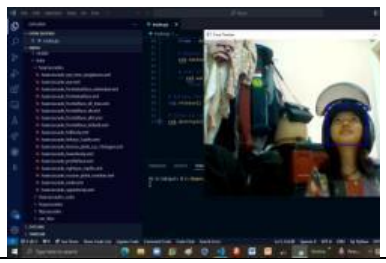
The following table shows the results of face detection with accessories that were obtained after the previous program process:

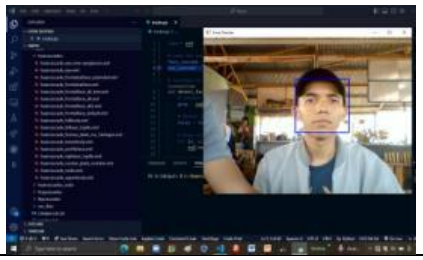
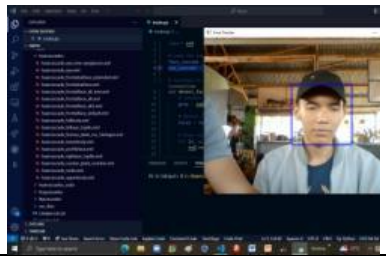
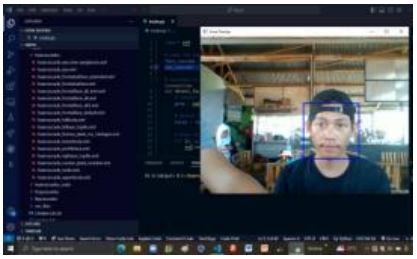
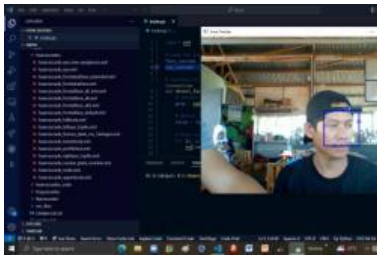
Table 1. Results of face detection with accessories

Test data	Face Detection Result 1	Face detection results 2	Information
1.Png			True Positive
2.Png			True Positive

Test data	Face Detection Result 1	Face detection results 2	Information
3.Png			True Positive
4.Png			True Positive
5.Png			True Positive
6.Png			True Positive
7.Png			True Positive

Test data	Face Detection Result 1	Face detection results 2	Information
8.Png			False Negative
9.Png			True Positive
10.Png			True Positive
11.Png			True Positive
12.Png			True Positive

Test data	Face Detection Result 1	Face detection results 2	Information
13.Png			True Positive
14.Png			True Positive
15.Png			True Positive
16.Png			True Positive
17.Png			True Positive

Test data	Face Detection Result 1	Face detection results 2	Information
18.Png			True Positive
19.Png			True Positive
20.Png			False Negative

In the table above, two images are shown from two different angles of view of the results of testing 20 facial data sets with accessories. The results obtained from the tests carried out on the Visual Studio Code software so that they get satisfactory facial detection results, namely getting almost 20 facial data sets with accessories that can be detected well and there are 2 data sets that are not detected due to the lighting being a little dark so that it can interfere with the system in detecting faces and backlight factors or against the light so that the system has difficulty in detecting faces with accessories. The meaning of the description is "True Positive" meaning the results are detected (as expected) and False Negative meaning the results are not detected. The results of facial detection that cannot be found in the system are in data set 8 and data set 20 which are caused by factors of insufficient lighting and backlight (against the light) so that the system has difficulty in detecting the face

Conclusion

From the results of the facial data set analysis that has been done, it can be concluded that from the research that has been mentioned, the author concludes that the analysis of facial detection with accessories on the face uses haar cascade in image processing. This haar cascade algorithm is still able to detect faces well, testing was carried out on visual studio code software and with an HP elitebook 840 G1 laptop device and an Intel Core i5 processor with 8GB RAM and a Windows 10 operating system and with a real-time camera device

with a *precision percentage* of 100%, *a recall* of 90% and *an accuracy* of 90%. At the beginning of this research, researchers have conducted testing with 20 different facial datasets with accessories and in different places such as homes, campus areas and boarding houses. The facial dataset was taken during the day, namely in the campus area and at night in the home and boarding house area, with a distance of several cm from the real-time camera of the computer/laptop device. And from 20 facial data sets with the tested accessories, 18 out of 20 data successfully detected faces even with accessories around the face and there were 2 facial datasets with accessories that were not detected due to insufficient lighting when taking facial data and the presence of backlight or devices against the light so that the system gets excessive reflection of light so that the system has difficulty in detecting faces with accessories and there are also factors from the perspective of taking facial data sets that have been done.

References

Andrekha, MZ, & Huda, Y. (2021). *Mangosteen Color Detection Using Image Processing with Opencv Python P - ISSN : 2302-3295* . *9* (4).

Christopher, A., & Mulyana, TMS (2022). Classification of Angiosperm Plants Using the K-Nearest Neighbor Algorithm Based on Leaf Shape. *JIPI (Scientific Journal of Informatics Research and Learning)* , *7* (4), 1233–1243. https://doi.org/10.29100/jipi.v7i4.3211

Erika, Y., & Saputro, H. (2021). Mahakarya Information System Journal (JSIM). *Mahakarya Information System Journal (JSIM)* , *4* (2), 10–18.

Fahmi Chairulloh Widia Sumantri, & Sutisna. (2022). Digital Image Processing of Motor Vehicle License Plates Using K-NN. *Journal of Informatics, Technology, and Science* , *4* (2), 101–104. https://doi.org/10.51401/jinteks.v4i2.1999

Gupta, S. (2024). Best Image Processing for Higher Face Detection Rate Using Haar Cascades. *Eai Springer Innovations in Communication and Computing*, 113-128, ISSN 2522-8595, https://doi.org/10.1007/978-3-031-64495-5_9

Hartika, B., & Ahmad, D. (2021). *Face Recognition Using the Haar Cascade Classifier Algorithm and Convolutional Neural Network* . *6* (3), 12–19.

Harshini, B. (2024). Face Detection and Face Recognition using Haar Cascade Algorithm. *5th International Conference on Recent Trends in Computer Science and Technology Icrtcst 2024 Proceedings*, 425-428, https://doi.org/10.1109/ICRTCST61793.2024.10578546

Intan, D., Saputra, S., Anjar, WS, Nuzul, KA, Pamungkas, RA, Studi, P., Informatika, T., & Langsung, V. (nd). *Face tracking and detection using live video on webcam* . *10* (1), 50–59.

Mantara, D., Sudoro, W., Kurniasari, A., & Rosid, J. (2022). *Face recognition using the Haar Cascade and Facenet methods* . *3* (1), 30–34.

Mithun, M. (2024). College Library Face Detection and Management System with Anti-spoofing Mechanism. *2024 Asian Conference on Intelligent Technologies Acoit 2024*, https://doi.org/10.1109/ACOIT62457.2024.10941336

Nugroho, FT (2024). *Facial Image Detection Using the Haar Cascade Classifier Algorithm* . *1* , 15–26.

Opencv-python, M., & Dwiparaswati, W. (2022). *51 UG JOURNAL VOL.16 Issue 02 February 2022 . 16* , 51–59.

Pauzi, CA, Yahya, AB, Wildan, FT, & Hidayatulloh, A. (2024). *Application of Haar Cascade Classifier in Facial Shape Pattern Recognition Using OpenCV . 2* (1), 7–15.

Politeknik, J., & Riau, C. (2021). *Application of Haar Cascade Classifier in Detecting Faces and Grayscale Image Transformation Using OpenCV . 7* (1), 100–109.

Qorik, G., Pratamasunu, O., Ilmandira, O., Farisi, R., Jannah, M., Informatika, P., Teknik, F., Nurul, U., Probolinggo, KP, & Mukaromah, IF (2020). *ON VIDEO USING HAAR CASCADE AND METHOD* .

Sahfitri, I., Simanjuntak, M., & Nurhayati. (2021). Application of the Viola Jones Method in a Face Detection System. *National Informatics Seminar (SENATIKA)* , 473–480.

Sandag, GA (2020). App Store App Rating Prediction Using Random Forest Algorithm. *CogITo Smart Journal* , *6* (2), 167–178. https://doi.org/10.31154/cogito.v6i2.270.167-178

Shukla, D. (2024). Human Face Detection and Emotion Recognition Using OpenCV through AI. *Iemecon 2024 12th International Conference on Internet of Everything Microwave Embedded Communication and Networks*, https://doi.org/10.1109/IEMECON62401.2024.10845980

Utami, F., & Mujib, MA (2021). *Implementation of the Haar Cascade Algorithm in Face Recognition Applications . 03* (01), 33–38.