Computer Engineering and Applications Vol. 13, No. 3, October 2024

The Eye and Nose Identification Chip Controller-Based on Robot Vision Using Weightless Neural Network Method

Ahmad Zarkasi[1*], Huda Ubaya[2], Kemahyanto Exaudi[3], Megi Fitriyanto[4]

[1,2,3,4]Department of Computer Engineering,Faculty of Computer Science, Universitas Sriwijaya
ahmadzarkasi@unsri.ac.id

ABSTRACT

Increasingly advanced image analysis in computer vision, allowing computers to interpret, identify, and analyze pictures with accuracy comparable to humans. The availability of data sources in decimal, hexadecimal, or binary forms enables researchers to take the initiative in applying their study findings. Decimal formats are typically used on traditional computers like desktops and minicomputers, whereas hexadecimal and binary formats were utilized on single-chip controllers. Weightless Neural Network is a method that can be implemented in a single chip controller. The aim of this research is to develop a facial recognition system, for eye and mouth identification, that works in a single chip controller or also called a microcontroller. The suggested method is a Weightless Neural Network with Immediate Scan approach for processing and identifying eye and nose patterns. The data will be handled in many memory locations that are specifically designed to handle massive volumes of data. The data is made up of primary face data sheets and face input data. The data sets utilized are (x,y) pixels, and frame sizes range from 90x90 pixels to 110x110 pixels. Each face shot will be processed by selecting the region of the eyes and nose and saving it as an image file. The eye and nose will identify the face frame. Next, the photos will be converted to binary format. A magazine matrix will be used to transmit binary data from a minicomputer to a microcontroller via serial connection. Based on a known pattern, the resultant similarity accuracy is 83,08% for the eye and 84,09% for the sternum. In contrast, the similarity percentage for an eye ranges from 70% to 85% for an undefined pattern.

Keywords: Single-chip controllers, Weightless Neural Network, the Immediate Scan approach, facial recognition, eye and mouth identification.

1. INTRODUCTION

Faces may communicate and transmit information through a variety of emotions and facial expressions [1][2]. It is essential for social engagement in society. Human faces have a vital part in the identification of individuals. Every person has distinct face traits in terms of form, color, and gesture. The human brain has an incredible capacity to recognize a person by the face, even when the individual has physical changes, such as age, hairdo changes, or the use of spectacles [3][4]. A typical way for recognizing persons is to gaze, which involves taking photos of items to be recognized and storing face data in brain memory [5][6]. After capturing the image, the following step is to look for the properties of the previously viewed item. This approach was eventually adopted by a computer, which became known as computer vision [7][8].

Computer Vision enables computers to interpret and recognize pictures with accuracy comparable to human capabilities [9][10]. Some researchers have used this capacity to build a number of solutions for computer vision. Among these are methods for recognizing vehicle patterns, underwater objects, and face patterns [11][12][13] Pattern recognition algorithms have been applied in the field of robotics, particularly on vision robots used in industrial, home, and military applications. Robot vision technology has been programmed to identify a variety of face patterns [14][15][16]. Facial pattern recognition on a vision robot can be implemented using either computer technology or a single chip controller [17][18]. Researchers must consider various factors while designing a single chip controller, including cost-effectiveness, ease of repair, hardware availability, and pattern recognition algorithms that are compatible with the hardware system [19][20][21].

The Local Binary Pattern Histogram (LBPH) [22][23][24] and Weightless Neural Network approaches (WNN) [25][26][27]. But based on the results of the surveillance, both methods are a more realistic Weightless Neural Network approach to be implemented in managing facial pattern data. Weightless Neural Network has advantages in terms of speed in processing and data similarity. The data of an information will be grouped in several memory locations, then the layout will be studied, and compared to the output pattern of the system to obtain the accuracy of the data pattern resemblance. Based on prior explanations, the beginning of this study proposes a Weightless Neural Network approach to face pattern recognition, which will be confirmed using a vision robot. In this study, facial pattern data only includes binary data for the eyes and nose. The aim of the study was to develop facial recognition systems using the Weightless Neural Network method using an immediate scan approach, which focuses on processing eye and mouth data. Hopefully this research could contribute to the development of facial pattern recognition methods on embedded system engineering platforms.

2. DESIGN AND METHOD

This part seeks to improve the research by introducing organized and dynamic stages of work. This part will also cover the overall design of hardware and software, as well as the design of research methodologies. A vision robot with a mobile design is comprised of a control system, an actuator, and a camera sensor. The software is an algorithm for identifying patterns and motions in the robot's vision.

2.1 SYSTEM DESIGN

The design of the system is the first stage in determining the research framework. The first step is to design a system-wide schematic block that separates the facial detection process from the facial, mouth, and nose identification patterns [28][29][30]. In general, the method begins with the construction of a dataset, which is an image of the researcher's face that will be utilized in the facial recognition process. The next stage is to extract features from the face, specifically the eyes and nose, and convert the picture to a binary array. Once the binary data is accessible, an Arduino library for facial recognition is built using the file header and source file. Figure 1 depicts a system schematic block.

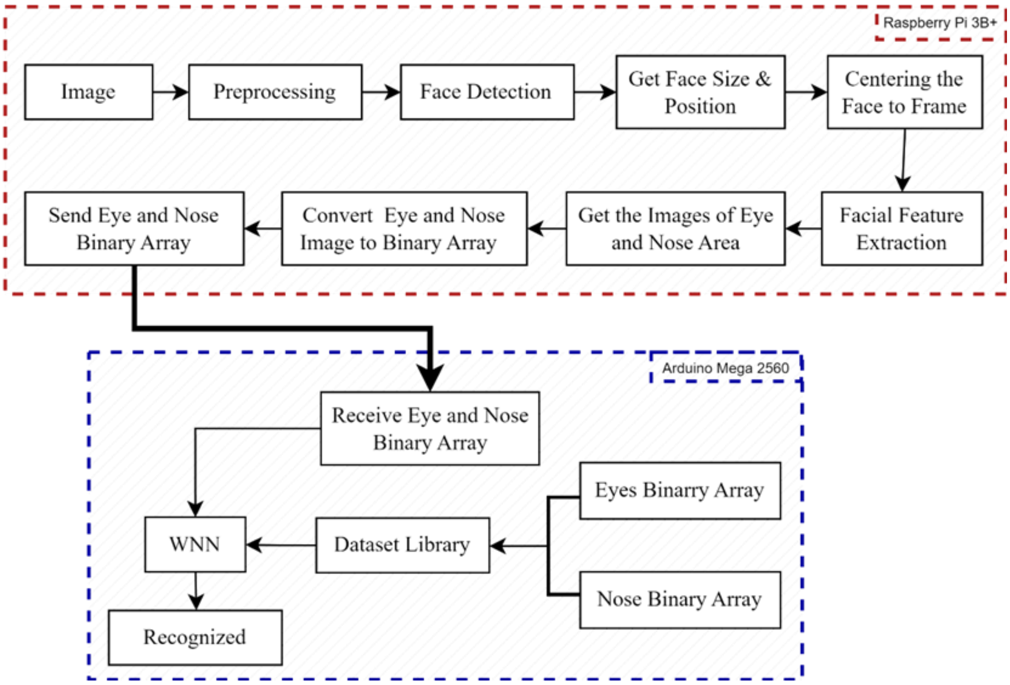


FIGURE 1. System schematic block.

When the camera records a picture in real time, the author's detection algorithm runs a preprocessing step that transforms an RGB image to grayscale. Prior to completing her feature search for facial detection, the image will be transformed to an integral image. When a face is identified, the size and location of the face frame are computed, and the DC motor moves to change the face position of the frame. The face frames will be back in search of her features to detect both eyes and nose. After getting the eye and nose area, both images will be converted to binary format. Binary data will be sent from the Raspberry Pi to the Arduino Mega via serial for continued identification process. Once binary eye and nose data is obtained, it will be employed in facial recognition with the Weighless Neural Network method. The supplied binary data will be compared to datasets prepared and saved in the preceding process's library. The process of comparing input patterns to datasets on the comparator yields the amount of similarity utilized to assess whether or not faces are identified.

2.2 HARDWARE DESIGN

The next stage is to design the hardware and determine the system requirements. The hardware is made up of various components that work together to function properly. This hardware is an essential component of the vision robot. The components include a Raspberry Pi 3B+ with an ARM A3 CPU, 1 GB of RAM, and a Pi Camera V2.1. Then there is one AVR microcontroller with the Arduino board series Mega256, which has 256 KB ROM capacity. or the drive system consists of 1 free wheel, 2 drive wheels with 2 DC motors equipped with 2 drivers for each motor and 1 Lipo Battery. All of those components are assembled and placed on the vision robot. Figure 2, showing the wiring diagram of the component and the design of the vision robot.

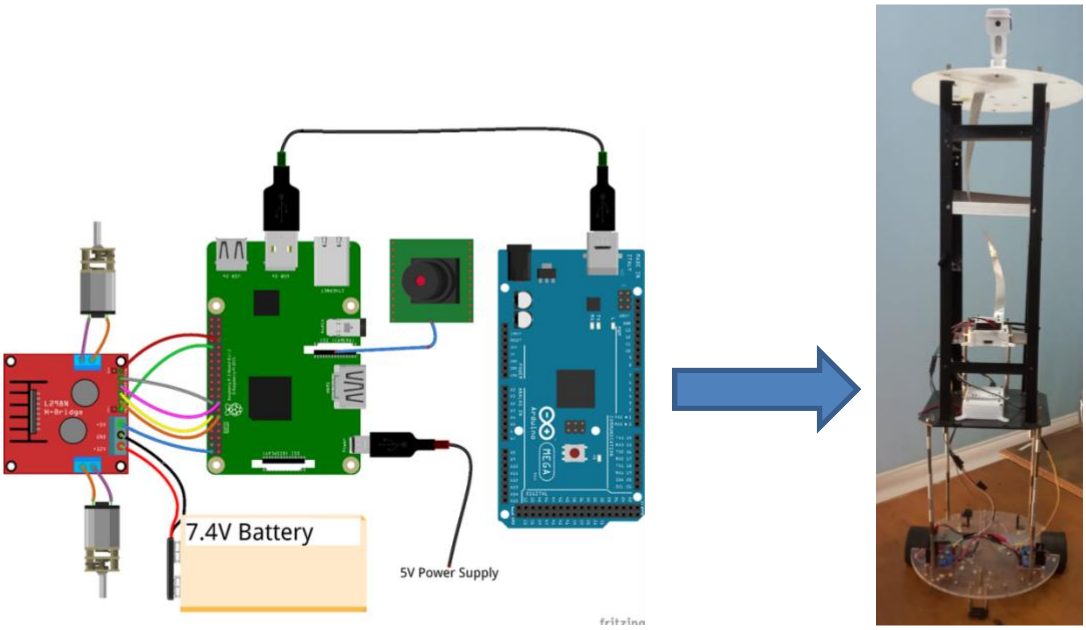


FIGURE 2. Wiring of robot vision system

In general, the system works by having robots wander randomly in a certain location while active cameras collect environmental data. When a facial picture is taken and identified by the camera, the vision robot moves to a specific distance. As seen in the preceding block diagram, the picture location within the frame will act as a reference and input for the hardware to move. The Raspberry Pi 3B+, a Mini PC, will be utilized to process images (binary recognition and conversion) and control robotic actuators. Two DC motors that have been coupled will be utilized to operate the Mobile Robot. The Lipo battery will serve as the driver's power supply.

2.3 WWN ALGORITHM DESIGN

The software design is divided into two phases: facial detection and facial pattern recognition. Face detection is conducted by utilizing an ARM processor on a Raspberry module and the Python programming language. Facial recognition is performed utilizing Arduino's single chip AVR controller and C++ programming languages. The Viola-Jones algorithm is used to detect faces, and the Weightless Neural Network method is used with the Immediate Scan approach to recognize faces. The Immediate Scan approach method is a fundamental tool for data verification on the compressor. This method works by directly comparing input pattern data to known patterns on a bit level. The data identification and verification procedure operates in the following order of conditions: if the input data contains image resolution data of the same size as the reference pattern, the data will be immediately compared to the data on the referencing pattern. Figure 3 depicts an Immediate Scan Approach Scheme.

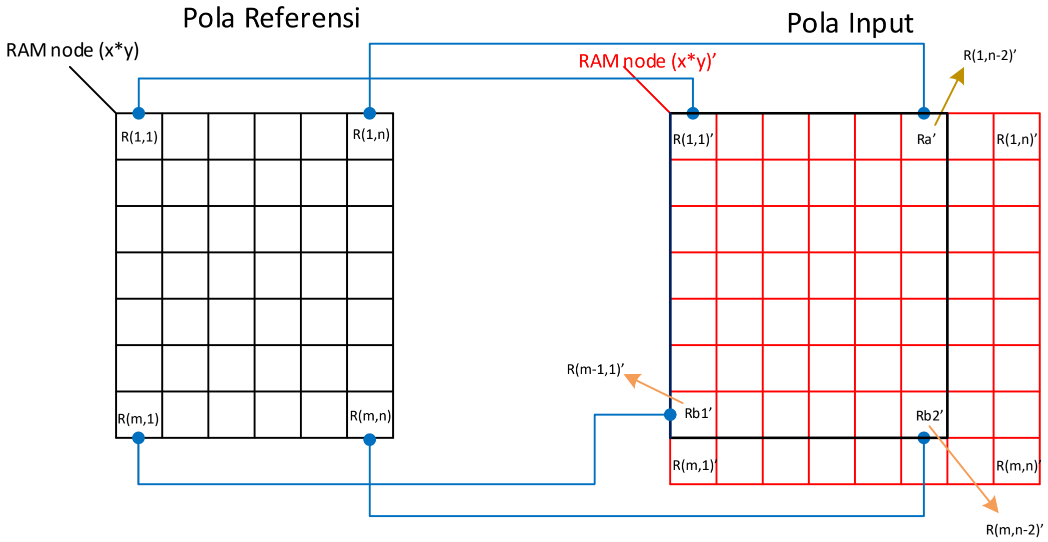


FIGURE 3. Immediate Scan Approach Scheme.

Figure 3 shows that the reference pattern contains the node's RAM resolution data (x*y), whereas the input patterns contain the RAM node's resolution data (x * y). The data verification method is tailored to the specific area where the data RAM node began and concluded. The check begins from the RAM point of the first row R(1.1) and proceeds to the RAM spot R(1,n), ending with the RAM final row point R(m,1) and R(m, n). The next step is to create a face detection algortima flow diagram.

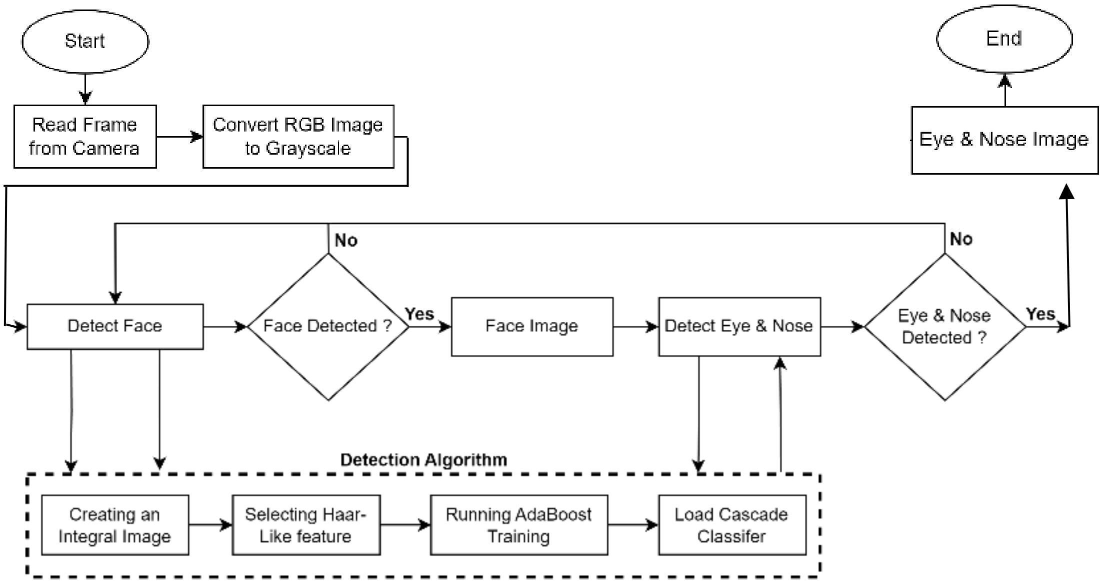


FIGURE 4. Face detection process.

In the face detection process, when you find a face in an image taken from a camera frame, the system will calculate the position of the face using coordinates. (x,y). This information will be processed directly in the microcomputer, which then tries to adjust the center point of the face area to the center of the camera frame. Figure 4 is a face detection process. After that, proceed with feature extraction. In the process of extracting facial features, the images obtained are eye and nose data. This data will go through the next process of facial recognition. The method used in facial pattern recognition is the Weightless Neural Network. (WNN). The WNN architectural design is shown in Figure 5.

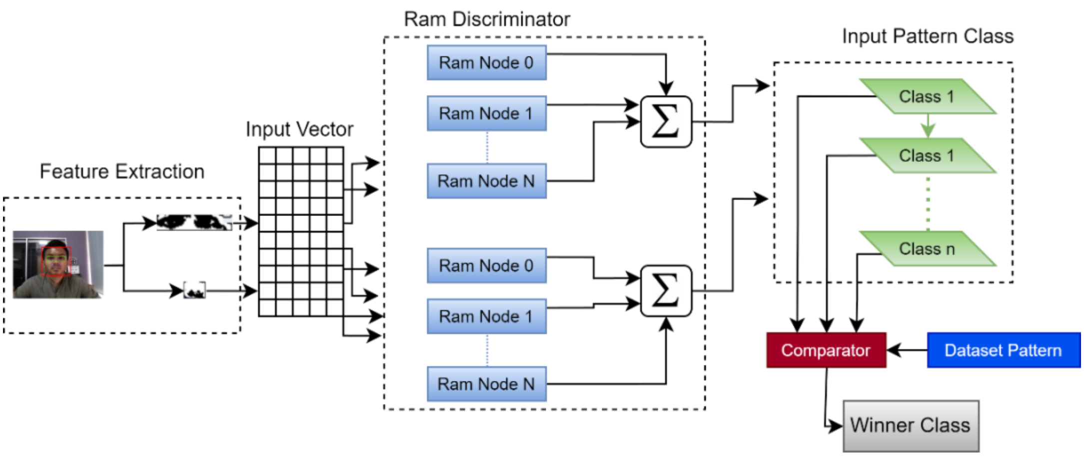


FIGURE 5. WNN architectural design

Figure 5 shows the feature extraction pattern adjusted in a binary format to encompass the location of the eye and nose. The data from this part is utilized as a vector input. The vector input data is then combined into numerous RAM nodes, as is standard procedure for pattern data placement. RAM nodes are then organized into many RAM discriminators for the purpose of learning data on each of their nodes. The RAM discriminator is also taught to recognize input patterns and compare them to reference patterns in the address comparator. The class with the greatest Euclidean distance to the reference face pattern is chosen as the winner.

3. RESULTS AND DISCUSSION

3.1 FACE TRACKING MOBILE ROBOT

The test aims to determine the performance of the hardware, in this case the mobile robot in its response to facial movements. The frame that already contains facial detection results will be divided into several tracking areas. The direction of motion of the DC motor first depends on the face position (x) and then follows the width of the face (w). As seen in Figure 6, the frame width of 320x240 is divided into several different pixel sized areas.

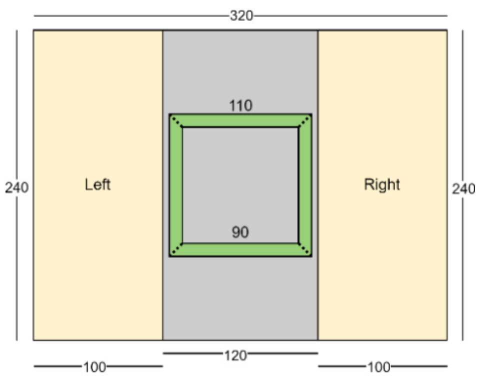
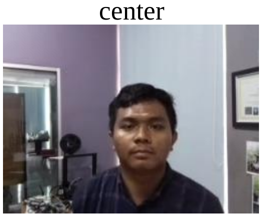


FIGURE 6. Face tracking area

The left and right area frames have widths of 100x240, the outer face frame is 110x110 and the inside frame is 90x90. The grey area indicates that the robot is already in the right position so the robot only needs to make adjustments on the width of the face. Figure 6 shows how the robot's behavior will be strongly influenced by the location of the face. If the face is in one of the pixel area's places, the robot moves in accordance with its movement regulations. For example, if the face is in the left area, the motor will travel right until the face is in a brace position, and if the face is on the right side, the robot will move left until the face is in the middle region. This is done to ensure that the image acquired by the camera is in the best possible position and that the facial pattern can be clearly seen. Thus, the acquired face pattern data is likely to be more accurate. Table 1 result of face tracking.

TABLE 1.
Face Tracking Result

Face Area	**Robot Responce**
center 	Size=106, Position=120 Stop Size=109, Position=120 Stop Size=94, Position=130 Stop Size=97, Position=128
rght 	Size=102, Position=200 Right Size=96, Position=205 Right Size=100, Position=202 Right Size=102, Position=203
Left 	Size=86, Position=48 Left Size=83, Position=49 Left Size=86, Position=49 Left Size=84, Position=50

In Table 1, the center position size resolution achieved indicates the smallest size of 94 pixels, with the face positioned in the pixel area of 130, and the maximum size of 106 pixels, with the face positioned in the pixel area of 120. For faces on the right side, the smallest pixel resolution is 96 and the biggest is 102, with pixel locations of 205 and 200. The least pixel resolution for faces on the left side is 83 with a pixel area of 49, while the greatest size is 86 with the same pixel area of 49.

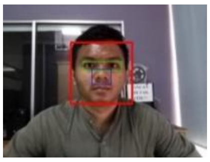
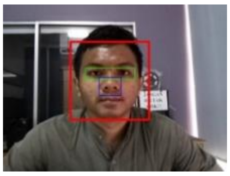
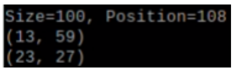
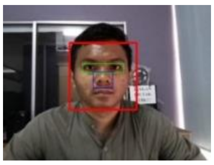
3.2 FACE RECOGNITION WNN ALGORITHM

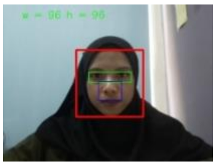
This test is conducted with the aim of evaluating the extent of the performance of the face detection system applied in this research. In this study, the face detection system is designed to detect front-facing faces with the condition that the width of the face frame is within the range of 90 to 110 pixels. The testing process carried out plays a crucial role in measuring the extent to which the system can meet the established resolution limits. Based on the results of the tests, the facial images taken can all be detected by the Viola-Jones algorithm. Face recognition testing was conducted using two types of image inputs. The first test was performed with pre-prepared face image files, while the second test utilized real-time face image input from a Pi Camera. This recognition process will return 3 values: the highest eye similarity, the highest nose similarity, and a decision on whether the face is recognized or not. Table 2 is the face recognition testing result.

.

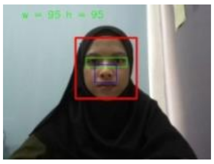
TABLE 2.

Face Recognition Result

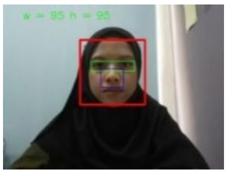
Face Image (Offline)	**Face Recognition**	**Face Image (realtime)**	**Face Recognition Realtime**
	MaxEye:93.99%; MaxNose:94.00%; Recognized; Time Required: 579ms	 	Dataset1: Eye = 82.66%; Nose = 79.87% Dataset2: Eye = 84.09%; Nose = 84.54% Dataset3: Eye = 79.01%; Nose = 77.78% Dataset4: Eye = 77.71%; Nose = 79.07% Dataset5: Eye = 82.79%; Nose = 83.41% Dataset6: Eye = 80.96%; Nose = 80.03% Dataset7: Eye = 78.23%; Nose = 83.25% Dataset8: Eye = 83.96%; Nose = 85.83% Dataset9: Eye = 76.53%; Nose = 78.42% Dataset10: Eye = 80.83%; Nose = 79.87% MaxEye:84.09%; MaxNose:85.83% Not Recognized; Time Required:509ms
	MaxEye:87.00%; MaxNose:95.68%; Recognized; Time Required: 568ms		
	axEye:78.92%; MaxNose:86.61% Not Recognized; Time Required:600ms	 	Dataset1: Eye = 62.60%; Nose = 74.81% Dataset2: Eye = 65.78%; Nose = 79.76% Dataset3: Eye = 65.92%; Nose = 77.49% Dataset4: Eye = 66.31%; Nose = 77.08% Dataset5: Eye = 64.19%; Nose = 77.60% Dataset6: Eye = 66.98%; Nose = 78.54% Dataset7: Eye = 68.30%; Nose = 81.55% Dataset8: Eye = 63.13%; Nose = 81.99% Dataset9: Eye = 62.07%; Nose = 76.88% Dataset10: Eye = 67.77%; Nose = 79.87% MaxEye:68.30%; MaxNose:81.99% Not Recognized; Time Required:579ms
	MaxEye:67.26%; MaxNose:86.16% Not Recognized; Time Required:700ms		

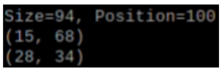


MaxEye:82.62%; MaxNose:71.28% Not Recognized; Time Required:589ms



.79%; MaxNose:62.95% Not Recognized; Time Required:631ms





Dataset1: Eye = 72.14%; Nose = 78.29%
Dataset2: Eye = 74.71%; Nose = 84.37%
Dataset3: Eye = 77.45%; Nose = 79.20%
Dataset4: Eye = 78.11%; Nose = 80.21%
Dataset5: Eye = 76.23%; Nose = 82.27%
Dataset6: Eye = 76.00%; Nose = 82.75%
Dataset7: Eye = 72.83%; Nose = 86.76%
Dataset8: Eye = 76.27%; Nose = 85.09%
Dataset9: Eye = 75.12%; Nose = 79.24%
Dataset10: Eye = 71.71%; Nose = 83.47%
MaxEye:78.11%; MaxNose:86.76%
Not Recognized; Time Required:713ms

Out of the ten faces evaluated, nine were identifiable, but one could not be recognized since the greatest eye similarity score was just 81.29%. Then, among the ten faces of the researcher's buddy 1 that were evaluated, the average nose similarity score was 86%, while the eye similarity score was only approximately 65%, resulting in the system returning the judgment of unidentifiable. Furthermore, of the ten faces belonging to the researcher's buddy 2, the similarity scores for both eyes and nose were less than 80%.

In real-time testing, the photos generated may contain noise owing to the robot's movement during the tracking process, therefore the similarity threshold must be somewhat reduced. A face is considered identifiable when the similarities between the eyes and nose exceed 85%. Out of the ten faces studied by the researcher, eight were recognizable, but two could not be recognized since the maximum similarity of the eyes was only 83.08% and 84.09%, respectively. Furthermore, the algorithm determined that the ten faces of the researcher's pals were unidentifiable.because it only generates a similarity level for the eyes of approximately 70%, while the similarity level for the nose is around 85%.

3.3 DATA COMMUNICATION ON DEVICES

In this example, the Raspberry Pi and Arduino are linked by USB to allow data transfer and control of the Arduino from the Raspberry Pi. As illustrated in Figure 7, after the USB is connected, the Arduino Mega will be identified as a serial port with the designation "/dev/ttyACM0 (Arduino Mega or Mega 2560)" in the Arduino IDE loaded on the Raspberry Pi.

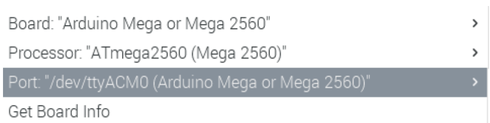


FIGURE 7. Serial port configuration

In this test, the Raspberry Pi will send a String message to the Arduino Mega, which will then read the message and respond by sending a reply if it gets it successfully. As illustrated in Figure 8, the Raspberry Pi will display the Arduino's reply message, confirming that serial connection is working properly. The Arduino

Mega can receive data well. In this test, the Raspberry Pi sent "10011000" ten times, and the Arduino Mega was able to receive it properly, displaying it on the Serial Monitor.

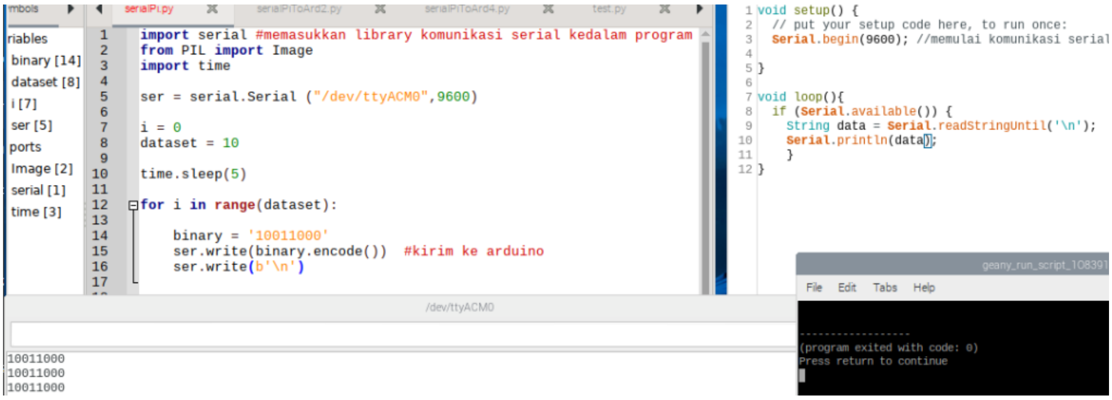


FIGURE 8. Data communication between Raspberry Pi and Arduino Mega

4. CONCLUSION

Overall, system testing has gone smoothly for both hardware and software. The mobile robot algorithm generates suitable answers depending on the test results. The robot may first move the facial position to the center of the frame by moving left or right, and then alter the breadth of the face by moving forward or backward. The face identification method detects faces with pixel widths (x,y) ranging from 90 to 110 pixels and is capable of recognizing facial characteristics accurately. The Weightless Neural Networks algorithm-based face recognition system, when tested with the researcher's facial picture samples, yielded similarity scores for the eyes and nose that above 90%. The average similarity score for real-time system testing was 85% due to the shift away from facial tracking technology. The setup utilized in the study operated admirably, with USB serial connectivity between the Raspberry Pi and Arduino Mega capable of transmitting full binary data without data loss.

ACKNOWLEDGEMENTS

The researc Upublication of this article was funded by DIPA of Public Service Agency of Universitas Sriwijaya 2023. Nomor SP DIPA-023.17.2.67751512023, On November 30, 2022. In accordance with the Rector's Decree Number: 0188/UN9.3.1/SK/2023, On April lB, 2023," was supported by Universitas Sriwijaya. The first author would like to thank the Faculty of Computer Science, Faculty of Engineering, Intelligence System Research Group, and COMNETS Research Group, as well as Universitas Sriwijaya's Embedded Systems, Control System, and Robotic Laboratory, for their assistance with this research.

REFERENCES

[1] F. Marcolin, E. Vezzetti, and M. G. Monaci, "Face perception foundations for pattern recognition algorithms," *Neurocomputing*, vol. 443, pp. 302–319, 2021, doi: 10.1016/j.neucom.2021.02.074.

[2] M. O. Oloyede, G. P. Hancke, and H. C. Myburgh, "A review on face recognition systems: recent approaches and challenges," *Multimed. Tools Appl.*, vol. 79, no. 37–38, pp. 27891–27922, 2020, doi: 10.1007/s11042-020-09261-2.

[3] G. M. Zafaruddin and H. S. Fadewar, *Face recognition using eigenfaces*, vol. 810. Springer Singapore, 2018.

[4] Y. Kortli, M. Jridi, A. Al Falou, and M. Atri, "Face recognition systems: A survey," *Sensors (Switzerland)*, vol. 20, no. 2, 2020, doi: 10.3390/s20020342.

[5] M. Taskiran, N. Kahraman, and C. E. Erdem, "Face recognition: Past, present and future (a review)," *Digit. Signal Process. A Rev. J.*, vol. 106, p. 102809, 2020, doi: 10.1016/j.dsp.2020.102809.

[6] R. Rothe, R. Timofte, and L. Van Gool, "Deep Expectation of Real and Apparent Age from a Single Image Without Facial Landmarks," *Int. J. Comput. Vis.*, vol. 126, no. 2–4, pp. 144–157, 2018, doi: 10.1007/s11263-016-0940-3.

[7] G. Bradski, A. Kaehler, and V. Pisarevsky, "Learning-Based Computer Vision with Intel&apos," *Intel Technol. J.*, vol. 9, no. 2, 2005.

[8] V. E.Dahiphale and S. R, "Real-Time Computer Vision System for Continuous Face Detection and Tracking," *Int. J. Comput. Appl.*, vol. 122, no. 18, pp. 1–5, Jul. 2015, doi: 10.5120/21797-5100.

[9] X. Feng, Y. Jiang, X. Yang, M. Du, and X. Li, "Computer vision algorithms and hardware implementations: A survey," *Integration*, vol. 69, no. April, pp. 309–320, 2019, doi: 10.1016/j.vlsi.2019.07.005.

[10] A. John Rajan, K. Jayakrishna, T. Vignesh, J. Chandradass, and T. T. M. Kannan, "Development of computer vision for inspection of bolt using convolutional neural network," *Mater. Today Proc.*, vol. 45, no. xxxx, pp. 6931–6935, 2020, doi: 10.1016/j.matpr.2021.01.372.

[11] S. Wan and S. Goudos, "Faster R-CNN for multi-class fruit detection using a robotic vision system," *Comput. Networks*, vol. 168, p. 107036, 2020, doi: 10.1016/j.comnet.2019.107036.

[12] N. O'Mahony *et al.*, *Deep Learning vs. Traditional Computer Vision BT - Advances in Computer Vision*. Springer International Publishing, 2020.

[13] S. Zhang, L. Wen, H. Shi, Z. Lei, S. Lyu, and S. Z. Li, "Single-Shot Scale-Aware Network for Real-Time Face Detection," *Int. J. Comput. Vis.*, vol. 127, no. 6–7, pp. 537–559, Jun. 2019, doi: 10.1007/s11263-019-01159-3.

[14] S. Nurmaini, A. Zarkasi, D. Stiawan, B. Yudho Suprapto, S. Desy Siswanti, and H. Ubaya, "Robot movement controller based on dynamic facial pattern recognition," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 22, no. 2, p. 733, 2021, doi: 10.11591/ijeecs.v22.i2.pp733-743.

[15] A. Zarkasi, S. Nurmaini, D. Stiawan, and B. Y. Suprapto, "Weightless Neural Networks Face Recognition Learning Process for Binary Facial Pattern," *Indones. J. Electr. Eng. Informatics*, vol. 10, no. 4, pp. 955–969, 2022, doi: 10.52549/ijeei.v10i4.3957.

[16] A. Zarkasi, H. Ubaya, K. Exaudi, A. Almuqsit, and O. Arsalan, "Robot Vision

Pattern Recognition of the Eye and Nose Using the Local Binary Pattern Histogram Method," vol. 12, no. 3, pp. 147–158, 2023.

[17] G. Cerutti, R. Prasad, and E. Farella, "Convolutional Neural Network on Embedded Platform for People Presence Detection in Low Resolution Thermal Images," *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, vol. 2019-May, pp. 7610–7614, 2019, doi: 10.1109/ICASSP.2019.8682998.

[18] X. Lv, M. Su, and Z. Wang, "Application of Face Recognition Method Under Deep Learning Algorithm in Embedded Systems," *Microprocess. Microsyst.*, p. 104034, 2021, doi: 10.1016/j.micpro.2021.104034.

[19] M. Wang and W. Deng, "Deep face recognition: A survey," *Neurocomputing*, vol. 429, pp. 215–244, 2021, doi: 10.1016/j.neucom.2020.10.081.

[20] R. E. Fairley, "Automobile Embedded Real-Time Systems," in *Systems Engineering of Software-Enabled Systems*, Wiley, 2019, pp. 377–389.

[21] H. Jang, K. Han, S. Lee, J. J. Lee, and W. Lee, "MMNoC: Embedding Memory Management Units into Network-on-Chip for Lightweight Embedded Systems," *IEEE Access*, vol. 7, pp. 80011–80019, 2019, doi: 10.1109/ACCESS.2019.2923219.

[22] Q. Zhao, "Research on the application of local binary patterns based on color distance in image classification," *Multimed. Tools Appl.*, vol. 80, no. 18, pp. 27279–27298, 2021, doi: 10.1007/s11042-021-10996-9.

[23] A. Anand, V. Jha, and L. Sharma, "An improved local binary patterns histograms technique for face recognition for real time applications," *Int. J. Recent Technol. Eng.*, vol. 8, no. 2 Special Issue 7, pp. 524–529, 2019, doi: 10.35940/ijrte.B1098.0782S719.

[24] K. C. Paul and S. Aslan, "An Improved Real-Time Face Recognition System at Low Resolution Based on Local Binary Pattern Histogram Algorithm and CLAHE," *Opt. Photonics J.*, vol. 11, no. 04, pp. 63–78, 2021, doi: 10.4236/opj.2021.114005.

[25] L. Santiago *et al.*, "Weightless Neural Networks as Memory Segmented Bloom Filters," *Neurocomputing*, vol. 416, pp. 292–304, 2020, doi: 10.1016/j.neucom.2020.01.115.

[26] M. De Gregorio and M. Giordano, "An experimental evaluation of weightless neural networks for multi-class classification," *Appl. Soft Comput. J.*, vol. 72, pp. 338–354, 2018, doi: 10.1016/j.asoc.2018.07.052.

[27] R. Barbosa, D. O. Cardoso, D. Carvalho, and F. M. G. França, "Weightless neuro-symbolic GPS trajectory classification," *Neurocomputing*, vol. 298, pp. 100–108, 2018, doi: 10.1016/j.neucom.2017.11.075.

[28] K. B. Schauder, W. J. Park, Y. Tsank, M. P. Eckstein, D. Tadin, and L. Bennetto, "Initial eye gaze to faces and its functional consequence on face identification abilities in autism spectrum disorder," *J. Neurodev. Disord.*, vol. 11, no. 1, pp. 1–20, 2019, doi: 10.1186/s11689-019-9303-z.

[29] W. Ali, W. Tian, S. U. Din, D. Iradukunda, and A. A. Khan, *Classical and modern face recognition approaches: a complete review*, vol. 80, no. 3. Multimedia Tools and Applications, 2021.

[30] J. H. Shah, M. Sharif, M. Yasmin, and S. L. Fernandes, "Facial expressions classification and false label reduction using LDA and threefold SVM," *Pattern Recognit. Lett.*, vol. 139, pp. 166–173, 2020, doi: 10.1016/j.patrec.2017.06.021.