

# Towards Controlling Mobile Robot Using Upper Human Body Gesture Based on Convolutional Neural Network

Muhammad Fuad <sup>1\*</sup>, Faikul Umam <sup>2</sup>, Sri Wahyuni <sup>3</sup>, Nuniek Fahriani <sup>4</sup>, Ilham Nurwahyudi <sup>5</sup>, Mochammad Ilham Darwaman <sup>6</sup>  
Fahmi Maulana <sup>7</sup>

<sup>1, 2, 3, 5, 6, 7</sup> Department of Electrical Engineering, Universitas Trunojoyo Madura, Bangkalan, Indonesia

<sup>4</sup> Universitas Muhammadiyah Surabaya, Surabaya, Indonesia

Email: <sup>1</sup> fuad@trunojoyo.ac.id, <sup>2</sup> faikul@trunojoyo.ac.id, <sup>3</sup> s.wahyuni@trunojoyo.ac.id, <sup>4</sup> nuniekfahriani@ft.um-surabaya.ac.id, <sup>5</sup> 180491100025@student.trunojoyo.ac.id, <sup>6</sup> 190491100057@student.trunojoyo.ac.id,

<sup>7</sup> 190491100010@student.trunojoyo.ac.id

\*Corresponding Author

**Abstract**—Human-Robot Interaction (HRI) has challenges in investigation of a nonverbal and natural interaction. This study contributes to developing a gesture recognition system capable of recognizing the entire human upper body for HRI, which has never been done in previous research. Preprocessing is applied to improve image quality, reduce noise and highlight important features of each image, including color segmentation, thresholding and resizing. The hue, saturation, value (HSV) color segmentation is executed by utilizing blue color backdrop and additional lighting to deal with illumination issue. Then thresholding is performed to get a black and white image to distinguish between background and foreground. The resizing is completed to adjust the image to match the size expected by the model. The preprocessed data image is used as input for gesture recognition based on Convolutional Neural Network (CNN). This study recorded five gestures from five research subjects in difference gender and body posture with total of 450 images which divided into 380 and 70 images for training and testing respectively. Experiments that performed in an indoor environment showed that CNN achieved 92% of accuracy in the gesture recognition. It has lower level of accuracy compare to AlexNet model but with faster training computation time of 9 seconds. This result was obtained by testing the system over various distances. The optimal distance for a camera setting from user to interact with mobile robot by using gesture was 2.5 m. For future research, the proposed method will be improved and implemented for mobile robot motion control.

**Keywords**—Gesture; Upper Human Body; Convolutional Neural Network; Mobile Robot Control.

## I. INTRODUCTION

Human-Robot Interaction (HRI) is a field of research on how to develop systems that allow robots to interact with humans in human environments. HRI has the challenge of developing nonverbal interactions with a natural approach that must be carried out in real-time. Several previous studies had been carried out for HRI investigations [1]. One of problems required to solve for the mobile robot is how to control position and orientation of its wheels [2]. Control technology allowed human to manage the robot movement by using joystick [3]–[14] and android-based smart device [15]–[32]. Another approach to control the robot motion is by utilizing computer vision technology [33]–[37].

Computer vision technology has been emerged to be solution for developing motion control of a mobile robot based on perception of feedback from camera. Computer vision that used in robotics are known as robot vision have been explored in some previous researches to solve position and orientation control of mobile robot. Kinect sensor was investigated in [33] to control the motion of omnidirectional three wheeled vacuum cleaning mobile robot by interpreting distance and heading angle based on RGB-D images. Other study on motion control of an omnidirectional mobile robot was utilized the landmarks of the environment [34]. Linear and angular velocity of mobile robot was controlled by proposing distance estimation and landmark recognition based on RGB and depth images from Kinect sensor. Work on mobile robot orientation estimation by utilizing progressive probabilistic Hough transform with law of cosines, quadrant principle, and voting mechanism was explored in [35] to control its turning movement. In other side, controlling translation motion of mobile robot was investigated in [36] by exploiting RGB and depth image matching. RGB-D images from depth camera were placed in the side of mobile robot.

Robotic vision in special forms with a focus on gesture recognition has developed into a state of the art of HRI researches in [38]–[120]. Application of gesture recognition involves virtual reality [121], augmented reality [122], bio medics [123], and robotics [37]. Robot control based on gesture recognition have been investigated in [37], [124]–[126]. The movement of robot arm manipulator SCORBOT-ER 9 Pro was controlled in [37] by detecting human body landmarks. The joints of human arm were proposed as landmarks to enable ability of robot arm to imitate the user action. Geometry algorithm in the form of law of cosines was exploited to provide the angle of each robot's joint. Gesture was recognized by detecting the landmarks represented by a configuration of joints in human arm.

In recent years, Convolutional Neural Networks (CNN) have experienced rapid development and made a positive impression on the field of image classification [127] and robot vision [128]. In the work [129], CNN was used for image recognition of the US postal handwritten digit dataset.



Researchers carried out a series of processes to improve the performance of the Faster R-CNN algorithm to recognize hand posture from NUS dataset [130]. In that work, gesture was constructed by using one hand with complex background with human noise. For controlling mobile robot motion based on human intention, it is desirable to investigate a natural way of interaction. This research has objective to investigate a gesture recognition system to identify the entire human upper body, not just one hand posture as in previous research, in order to move mobile robots by using natural interaction signal.

Based on challenge to improve the simplicity of human-robot interaction, this research aims to utilize nonverbal and natural interaction. Three contributions of this study are developing of upper body gesture dataset consist of 450 images recorded from five research subjects with different gender and body posture, preprocessing using hue, saturation, value (HSV) color segmentation and the architecture of CNN for gesture recognition.

## II. METHOD

This study manages its research methodology according to the flow such as shown in Fig. 1. It begins with the first contribution in this study in that the developing of gesture dataset consist of images of upper body of human.

### A. Development of Gesture Dataset

There are 450 images which have proportion 80% and 20% for training and testing respectively. This gesture dataset is recorded from upper body gestures of five person with different gender and body posture that classified into 5 motions. The recording images process begins with taking a picture of body gestures by using a webcam and adjusting in environment such as described in Fig. 2. Then, preprocessing based on HSV color segmentation is applied to images of the upper body human gesture dataset.

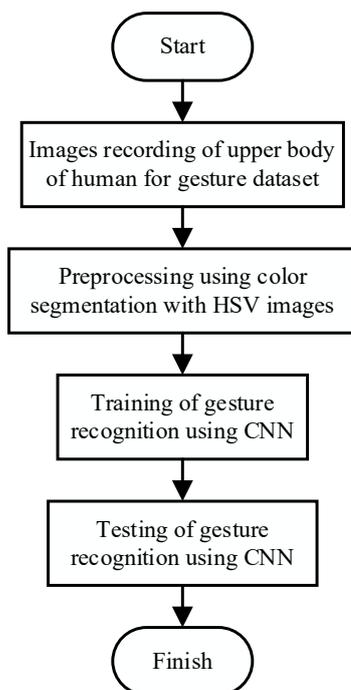


Fig. 1. Research methodology of CNN-based gesture recognition



Fig. 2. Environment setting in image recording process

### B. Preprocessing

The preprocessing sequential processes aim to improve image quality by reducing noise and highlighting the important features of image. This study proposes a series of processes consisting of color segmentation of background image, thresholding, and reducing the size of the image to match the input data model such as depicted in Fig. 3. The RGB image recorded by webcam is converted to HSV. The blue color backdrop is utilized to facilitate color segmentation on HSV image. The HSV trackbar is used for searching of background color. The brightness level of the light at the time of testing is determined by the range of minimum and maximum values for hue, saturation, and value. The image change from RGB to binary in preprocessing phase is illustrated in Fig. 4.

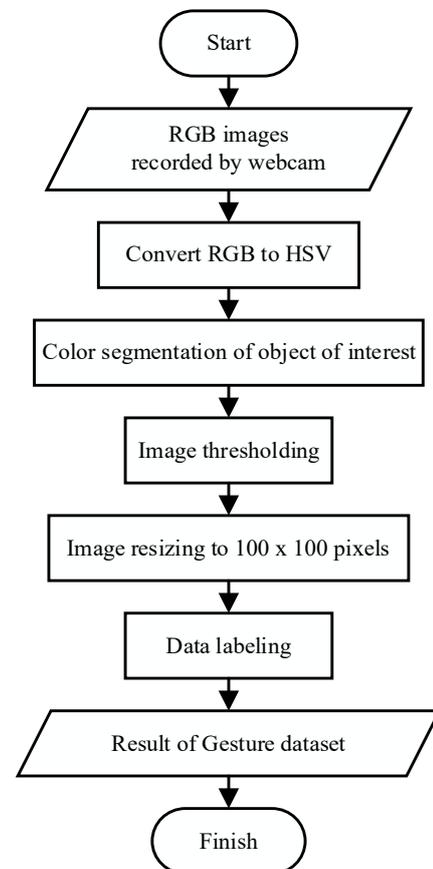


Fig. 3. Sequential process in preprocessing step

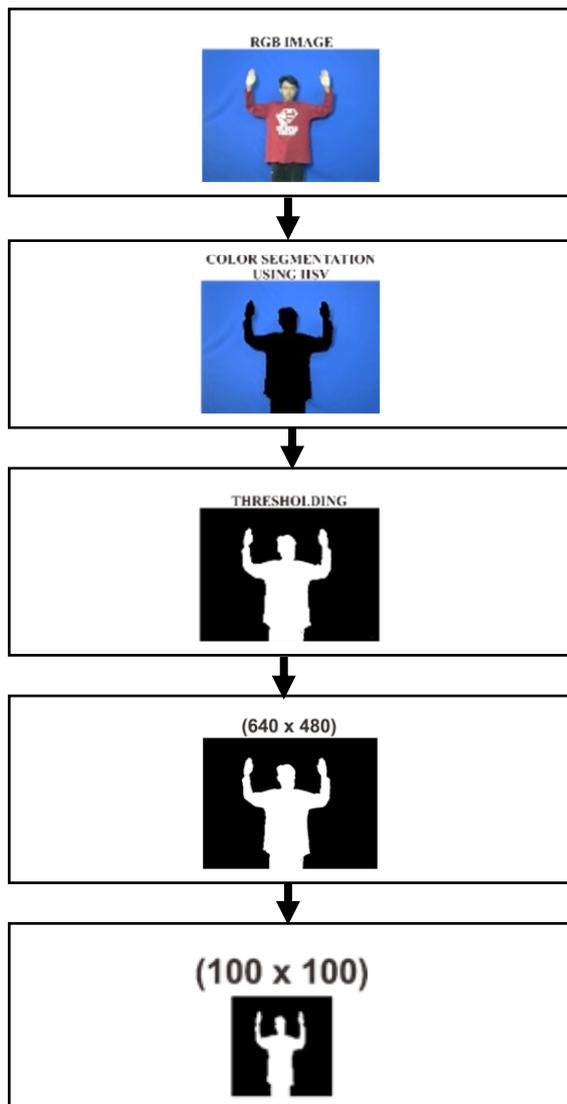


Fig. 4. Image change in preprocessing step

Labelling is carried out after the image data has been obtained. For supporting the model to learn and identify patterns or relationships between input and output features appropriately, the correct labels must be provided. To be more precise, data labelling facilitates the model to learn the rules and correlations existing in the data. Data labelling is used to save preprocessed image into each class directory. Python code for data labelling implementation is described in Fig. 5.

```

if interrupt & 0xFF == ord('1') :
    cv2.imwrite(
        dirs + "0/" + "Maju" + str(i)
        + ".jpg", frame)
    print("data gesture Maju ke-",i)
    i+=1
if interrupt & 0xFF == ord("2") :
    cv2.imwrite(dirs + "1/" +
        "Mundur" + str(j) +
        ".jpg", frame)
    print("data gesture Mundur ke-
        ",j)
    j+=1

```

Fig. 5. Code for image data labelling

### C. The Architecture of CNN

At this stage, the design of a CNN system for body gesture classification is discussed. CNN is a type of Deep Learning architecture that is effective in processing image data with high pattern recognition capabilities. The CNN system flowchart will be explained in detail in Fig. 6. Based on Fig. 6, the CNN process begins by reading the training data. Then the webcam captures image data from the operator's gestures and a series of preprocessing is carried out to adjust the data type to the CNN model created. The feature extraction process is carried out according to the training model that has been created previously.

In creating a CNN network architecture, there are several parameters that need to be considered. The following are several parameters of the CNN network architecture:

#### 1. Filters

Filters in CNN are also known as convolution filters. Convolution filters are filters (multidimensional data) used in convolution layers to help extract certain features from input data. The features detected are edges, curves, shapes, etc. that can be learned by CNN. There are various types of filters such as Gaussian Blur, Prewitt Filter and many more.

#### 2. Padding

Padding is the process of adding zeros to the input matrix symmetrically. If observed, the output size will be smaller than the input for each filter. So to keep the output dimensions the same as the input, you need to use padding.

#### 3. Strides

Stride shows how many steps are moved at each step in the convolution. By default, the stride used is 1. The filter will shift 1 pixel horizontally and then vertically. The smaller the stride, the more detailed the information obtained from an input, but it requires more computation compared to a large stride.

A simple CNN is formed from a series of layers, each layer converting one activation volume into another volume through a different function. The following are several layers in the CNN network architecture:

#### 1. Input Layers

The input layer in CNN is image data represented by a three-dimensional matrix, namely height, width and number of image channels. The 3-dimensional matrix values contained in an image will then be calculated in a convolutional process.

#### 2. Convolutional Layers

The convolutional layer is the first layer that changes an input with the value in the filter. At this stage a convolution process occurs based on the output of the previous layer. Part of the image will be connected to a filter based on the number of kernels used, then shifted with a predetermined stride and performs the same operation again. The process repeats until the filter successfully completes all image data. The output from the convolutional layer will be the input for the next layer. In producing an output there are influencing parameters such as stride and zero padding.

### 3. Activation Function

One of the important parameters in determining the output, accuracy and efficiency of the training model is determined by the activation function. The activation function can determine whether to activate a neuron with reference to the input to the network being important or not in the prediction process using simple mathematical operations. There are several examples of activation functions such as Sigmoid, Tanh, ReLU, Leaky ReLU, Parametrized ReLU, Exponential Linear Unit, Swish, and SoftMax. The commonly used activation function is ReLU. Using ReLU can speed up the training process by thresholding the zero value of the pixel values in the input image. The task of the ReLU activation function is to change pixels that are negative or less than zero in the image to 0.

### 4. Pooling Layer

Pooling is a process to reduce the input size spatially or reduce the number of parameters with down sampling operations. The dimensions of each feature map carried out by the pooling process will be reduced but still retain important information in each data. Pooling layers can also be used to speed up the performance of the entire convolutional layer. The most common form of pooling layer is to use a  $2 \times 2$  filter with steps of 2. There are two methods for pooling layers, namely max pooling and average pooling.

### 5. Flattening Layers

After the feature extraction stage is complete, proceed to the final stage in CNN, namely the classification stage. The first classification stage is flattening, which is the process of creating a 1D vector that is used to store the previous layer's output data. Input to the fully connected process requires individual features like other classifiers. So it is necessary to change the output part of the CNN which was originally in the form of a 3D matrix into a 1D vector so that it is used in the fully-connected process.

### 6. Fully Connected Layer

Fully connected layer is a basic structure in a neural network that can connect neurons in one layer to all neurons in other layers. Fully connected layers involve weights, biases and neurons. These layers are used to classify images between different categories by training. This layer is usually implemented at the end of the network. This layer uses a softmax activation function in the output layer which aims for classification. Softmax activation function to solve multi-class neural network learning and image classification problems with a set of pixels as input. The output from softmax can represent the distribution of a class. Softmax

calculates all the probabilities of labels with values between 0 and 1, if they are all added up they will have a value of 1.

### 7. Outputs

The output is the final layer of the CNN network carrying information that has been learned through hidden layers and providing a final value as a result.

The CNN model used is 2 convolution layers with ReLU activation function, 2 pooling layers (MaxPooling), 1 hidden layer (256), output layer with softmax activation function. After the output layer, the probability of each class will be

known. The class that has the largest probability value is considered a detected gesture. The process to find out which class has the largest probability value is by sorting the probability values between classes through a program.

The architecture of the CNN model created by the researcher is shown in Fig. 7. The CNN architecture created consists of 2 repetitions of the convolution operation.

The first convolution operation uses filter 32 with a kernel of  $3 \times 3$  and stride 1. Then the Rectified Linear Unit (ReLU) function is used to change the output from the operation which was originally negative to 0. Then perform image data reduction or a down sampling process with pooling. The pooling layer here is used to reduce the spatial size of the data by taking the maximum value from the  $2 \times 2$  kernel. This process is repeated to obtain more optimal feature extraction. Optimizing the feature extraction process on a CNN is done by designing and setting the CNN architectural parameters so that the most relevant features can be extracted from the image. The goal of this optimization is to improve the performance and efficiency of the model in image classification tasks according to what researchers want.

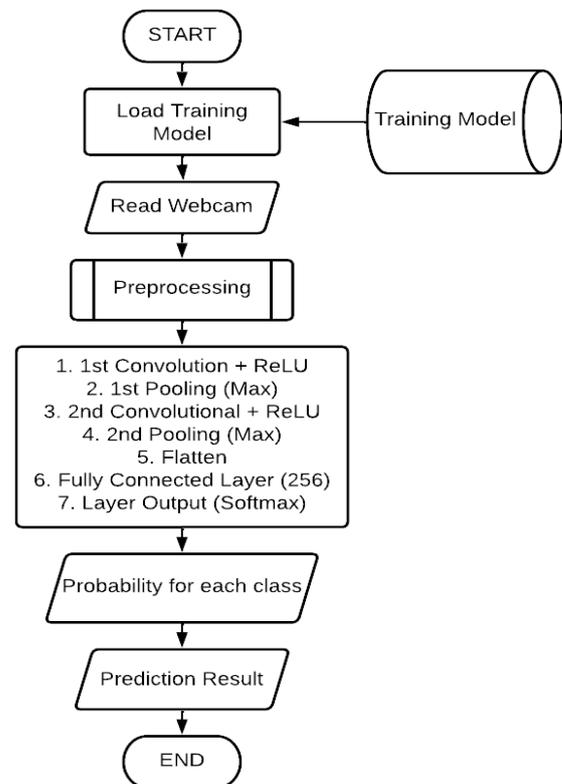


Fig. 6. Flowchart of gesture recognition

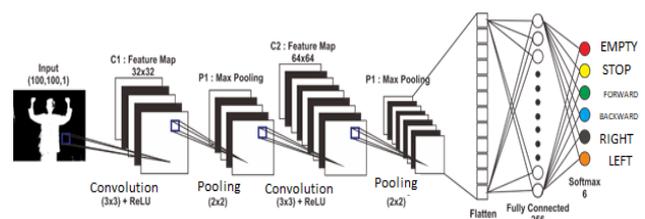


Fig. 7. Architecture of CNN

CNN architecture in Fig. 6 made simple with the aim of reducing the computational process when training the model.

The feature extraction process in CNN generally occurs repeatedly by applying many filters. In this research, the features of the image data have first been extracted through data preprocessing. There is no need for repeated feature extraction because the special characteristics of each image data have been extracted.

Before the gesture recognition process can be carried out, training of the detection system is required, the results of this training will become a CNN model. Training was carried out on 80% of gesture dataset that recorded a total of 450 images from five participants each demonstrating five gestures. Computer with AMD Ryzen 5 3500 U Quad Core 64 bit processor, Radeon Vega 8 GPU, 8 GB RAM, Realtek RTL8723DE 802.11b/g/n for networking and Bluetooth® 4.2 combo was used to complete the training process. Researchers compared the training time process with the AlexNet architecture model and found that the model created was 9 seconds faster than the AlexNet architecture. The researcher's architecture took 40.22 seconds, while the AlexNet model took 49.89 seconds when training the model.

#### D. Training Stage

Gesture is a movement from various poses. In this research, 5 gestures (forward, backward, stop, turn right, turn left) were chosen as case studies for gesture recognition. Before carrying out gesture recognition using the CNN method, it is necessary to carry out a training process as shown in Fig. 8.

It is known that the CNN model designed requires an input image measuring  $100 \times 100$  pixels with a black and white image type. Then there is a feature extraction process such as convolution with two layers and pooling with two layers. The type of pooling used is max pooling, which takes the largest value based on the specified number of kernels. Visualization of several steps in CNN is described in Table I. The following are the feature extraction stages.

##### 1. First Convolution

The image from the preprocessing results will be subjected to a convolution process. The value that is obtained from this process will be used as input for the ReLU. To find out the height and width of the output of the convolution process, it can be determined using the following equation:

$$Output = \frac{W - F + 2P}{S} + 1 \quad (1)$$

Where, *Output* is size of the feature map in the output, *W* is the size of the input feature map, *P* is the padding size, *F* is the filter size, *S* is stride size.

In the first convolution process, input measuring  $100 \times 100$  is then processed using a filter measuring  $3 \times 3$ , stride of 1 and adding padding of 1. This results in feature maps measuring  $100 \times 100$ .

##### 2. ReLU

ReLU is a function that is used to change negative values to 0 of the output of the first convolutional layer.

TABLE I. STEPS IN CNN

Label	Input Image (100x100)	Convolution 1 (100x100)	ReLU (100x100)	Max Pooling 1 (50x50)	Convolution 2 (50x50)	ReLU (100x100)	Max Pooling 2 (25x25)
Stop Gesture							
Move Forward Gesture							
Move Backward Gesture							
Turn Right Gesture							
Turn Left Gesture							

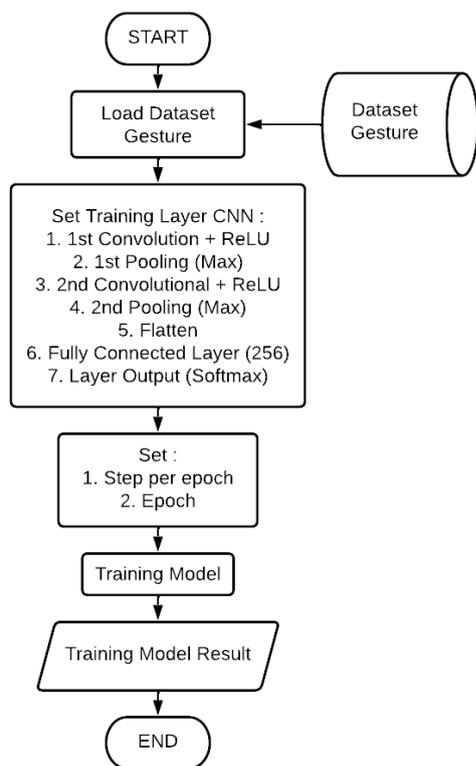


Fig. 8. Training phase of CNN

### 3. First Pooling

The pooling process is a process of reducing the size of the matrix to become smaller. There are two types of pooling, namely max and average pooling. Max pooling uses the largest value of the selected matrix, while average pooling uses the average value of the selected matrix. The pooling process used is average pooling with a filter size of  $2 \times 2$  and stride 2. In the first pooling process, input measuring  $100 \times 100$  is then processed using a filter measuring  $2 \times 2$ , stride of 2 and without adding padding. It produces feature maps measuring  $50 \times 50$ .

### 4. Second Convolution

The image from the first pooling will be subjected to a second convolution process. Then a value is obtained, this value will be used as input for the ReLU process. In the convolution process, the two inputs are  $50 \times 50$  in size and then processed using a  $3 \times 3$  filter, a stride of 1 and additional padding of 1. This results in feature maps measuring  $50 \times 50$ .

### 5. ReLU

ReLU is a function that is used to change negative values to 0 of the output of the second convolution.

### 6. Second Pooling

The pooling process is a process of reducing the size of the matrix to become smaller. The pooling process used is average pooling with a filter size of  $2 \times 2$  and stride 2. In the pooling process, the two inputs measuring  $50 \times 50$  are then processed using a filter measuring  $2 \times 2$ , stride of 2 and without adding padding. It produces feature maps measuring  $25 \times 25$ .

### 7. Flattening

The flattening process is the process of changing the resulting matrix from the feature extraction process into a

single vector which will later become the input layer for the CNN process.

### 8. Fully Connected Layer

Fully Connected Layer (FCL), also known as Dense Layer, is a type of layer that is generally used in CNN architectures to extract further features from features that have been extracted by previous convolution layers and pooling layers. FCL differs from convolution layers in that it has convolution operations based on kernel matrices. The FCL consists of neurons that are fully connected to all neurons in the previous layer. Based on the previous results, the values obtained are carried out further calculations and activation functions such as ReLU can be applied.

$$N_{[k]} = W_{[k]} \times I + B_{[k]} \quad (2)$$

$$a_{[k]} = f(N_{[k]}) = \text{maks}(0, N_{[k]}) \quad (3)$$

where  $N_{[k]}$  summing junction in layer  $k$ ;  $W_{[k]}$  value of the weights in layer  $k$ ;  $I$  is the input;  $a_{[k]}$  is value of neurons in layer  $k$ ;  $B_{[k]}$  value of biases in layer  $k$ .

### 9. Softmax

Softmax is an activation function that serves to distribute the probabilities of existing classes. In There are five neurons that represent the class to be predicted, namely stop, forward, backward, turn right, and turn left gesture respectively.

After completing the training process, a training results file is created with the extension .json and .h5. These files in machine learning are generally used for storing and managing CNN training results. These two formats have different purposes, json files are used to store the structure, model configuration, and other information related to the architecture of the CNN model being created. Meanwhile, the h5 file is used to store the weights and biases in each layer produced during the model training process. This file is used in the real time gesture detection by using a webcam in testing stage such as described in the following section.

### E. Testing Stage

At testing stage, the gesture detection process using the CNN method begins by reading the training results files in json and h5 files to be compared with the result of gesture recognition process of human gesture frame data via a webcam. A prediction process is performed based on the results of training and webcam. The index with the highest prediction probability results is selected as a successfully detected gesture. The results of the gesture detection as depicted in Fig. 9, will be used as control commands for the mobile robot.



Fig. 9. Result of Model Testing

### III. RESULTS AND DISCUSSION

Experiments were carried out to determine the level of success of the proposed method in recognizing human upper body gestures towards controlling the movement of mobile robot. Gesture was demonstrated by human operator in front of webcam at a distance of 2.0 m as illustrated in Fig. 10. The list of names of gestures for testing is shown in Table II. It presents the overall test result of the gesture-based mobile robot movement control system, researchers carried out the test 10 times on 5 participants which demonstrate 5 gesture of test data samples. From tests that were carried out 250 times, there were 227 trials where the gesture was successfully recognized, while 23 trials where the gesture failed to be identified. The percentage accuracy of system testing in identifying gestures was 90.8%. Fig. 11 shows the result of tests that have been carried out by involving variance in gender and size of posture. Small man, big man, small woman, and big woman are four variances used in the experiments.

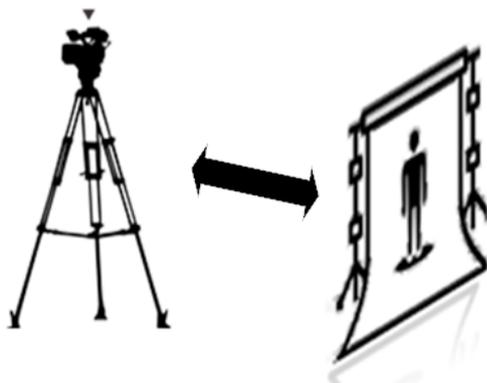


Fig. 10. Experiment's scenario for investigating the impact of distance

TABLE II. RESULT OF UPPER HUMAN BODY GESTURE RECOGNITION

State	Move Forward	Move Backward	Stop	Turn Right	Turn Left
Succeed	46	45	50	45	44
Failed	4	5	0	5	6
Total	50	50	50	50	50
Total number of experiments					250
Total number of failed					23
Total number of succeed					227

This research extends to some experiments to examine the impact of difference distance between human users and cameras on the reliability of the proposed method. There are seven test scenarios by setting the distance between 1.0 m and 4.0 m. Table III summarize the accuracy of gesture recognition from various distances. The first scenario reported 4.4% of accuracy where the distance between the webcam and the user was 1.0 m. In the second scenario where the distance was 1.5 m, resulting in an average success of the system in recognizing gestures of 9.0%.

The third test scenario was resulting in an average success of the system in recognizing gestures of 90% where the distance was 2.0 m. The best accuracy was achieved in the fourth test scenario that resulting in an average success of the system in recognizing gestures of 92% where the distance between the webcam and the user was 2.5 m.

TABLE III. TESTING THE DISTANCE OF THE GESTURE DETECTION SYSTEM

Scenario	Accuracy (%)					Ave rage	Std Dev
	Move For ward	Move Back ward	Stop	Turn Right	Turn Left		
Scenario 1 (1.0 m)	0	6	10	6	0	4.4	3.87
Scenario 2 (1.5 m)	5	10	10	10	10	9.0	2.00
Scenario 3 (2.0 m)	80	90	100	90	90	90	6.32
Scenario 4 (2.5 m)	92	90	100	90	88	92	4.19
Scenario 5 (3.0 m)	80	80	100	90	90	88	7.48
Scenario 6 (3.5 m)	60	70	100	80	80	78	13.26
Scenario 7 (4.0 m)	10	0	0	0	0	2.0	4.00

The average accuracy of the proposed system in recognizing gestures decreased to 88% in the fifth scenario where the distance between the webcam and the user was 3.0 m. The accuracy has a lower score of 78% in the sixth scenario where the user-webcam distance was 3.5 m. The lowest average score of 2.0% resulted in the seventh scenario where the distance between the webcam and the user was 4.0 m. Of the various distances that have been tested, the optimal distance between the webcam and the user to be used to control the movement of a mobile robot is 2.5 m.

Gesture for Stop is relatively easy to identify, while other gestures such as Move Forward, Move Backward, Turn Right, and Turn Left are relatively difficult to recognize. This condition occurs because of the emergence of false positives due to noise in the image. By removing noise through preprocessing by using Gaussian filter or such other filter, it is hoped that the false positive rate can be reduced so that accuracy increases. Several CNN architectures for image denoising will also be investigated in future research. Adaptive HSV color segmentation which will be explored in the next research could also be the solution to overcome the influence of lighting and clothing color in gesture recognition.

These experiments include the scenario at the nearest distance of 1.0 m and at the furthest distance from the capture device of 4.0 m. The data distribution in scenario 4 with a distance of 2.5 m, the best accuracy in these experiments 92% has the smallest value compared, namely 4.19, compared to the data distribution in other scenarios with accuracy above 70%. This study results lower computation time of 9 seconds with 92% of accuracy. Although it has lower computation time, but it has lower accuracy than AlexNet model that scores 100%.

Based on the experimental results, it can be seen that there are still several limitations in this proposed approach in the form of limited sensor range, dependence on backdrops, HSV settings in the color segmentation section are still manual, and accuracy is still low. In the next research, some of these limitations will be corrected and the results will be implemented in the control of mobile robots in human work environments. Viewed from a qualitative perspective based on user experience, in future research it is necessary to examine several additional gestures, especially for more complex mobile robot movements.

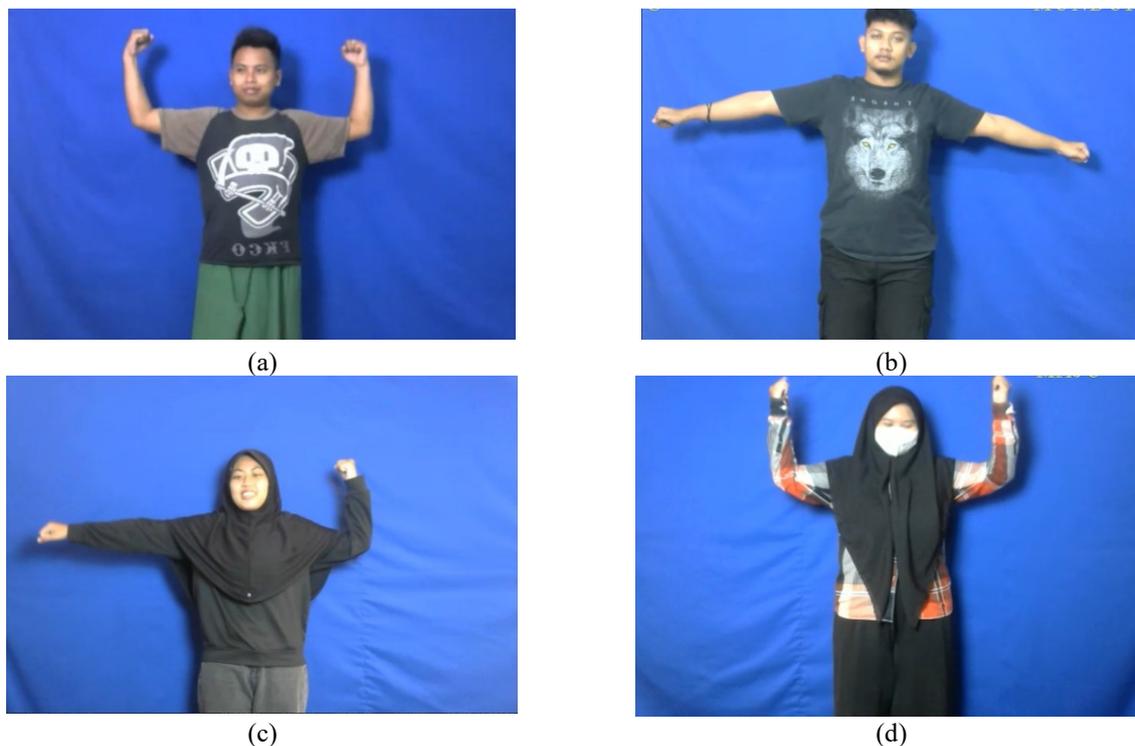


Fig. 11. Results of several experiments with gender and the size of posture variance: (a) man with small posture, (b) man with big posture, (c) woman with small posture, (d) woman with big posture

#### IV. CONCLUSION

The CNN architecture created by the author succeeded in detecting gestures supported by our developed gesture dataset and HSV-based color segmentation resulted the accuracy on average by 92%. The testing process was carried out optimally at a distance of 2.5 m in front of the webcam sensor. In the detection process, lighting settings and the color of the clothes worn have a big influence on the system. It will be difficult to carry out color segmentation so that the feature map of each gesture image is not formed perfectly. A gesture image that is not formed will make it difficult for CNN to detect because of the noise that exists even though the CNN already has a filter at the convulsive stage. Tests were carried out at different distances to determine the robustness of the system from various distances as explained in the experiment stage above. Based on the comparison results, the research model has a faster model training computing time of 9 seconds with 92% of accuracy, but has a lower level of accuracy than the AlexNet model that has accuracy with score 100%.

For future research, several improvements must be developed to overcome limited sensor range, independence system of the cluttered background, adaptive color segmentation, enhance the accuracy of system, decrease the computational time, and system implementation to real mobile robot in the human environment, gaming, and assistive technologies.

#### ACKNOWLEDGMENT

Author would like to acknowledge to the Institute for Research and Community Service (Lembaga Penelitian dan

Pengabdian kepada Masyarakat, LPPM) University of Trunojoyo Madura for financial support with contract No. 5892/UN46.4.1/PT.01.03/2023 and also give the highest appreciation to the Department of Mechatronics Engineering Faculty of Engineering University of Trunojoyo Madura for providing laboratory and technical support in order to make this research to be realized.

#### REFERENCES

- [1] M. A. Goodrich and A. C. Schultz, *Human-robot interaction: a survey*. in Foundations and trends in human-computer interaction, no. 1,3. Boston, Mass.: Now, 2007.
- [2] Barteck, Christoph, Balpaeme, Tony, Eyssel, Frederike, Kanda, Takayuki, Keijsers, Meyer, and Sabanovic, Selma, "Human-Robot Interaction: An Introduction," *Camb. Univ. Press*, 2020, doi: 10.1017/9781108676649.
- [3] E. S. Neo, K. Yokoi, S. Kajita, and K. Tanie, "Whole-Body Motion Generation Integrating Operator's Intention and Robot's Autonomy in Controlling Humanoid Robots," *IEEE Trans. Robot.*, vol. 23, no. 4, pp. 763–775, 2007, doi: 10.1109/TRO.2007.903818.
- [4] S. K. Cho, H. Z. Jin, J. M. Lee, and B. Yao, "Teleoperation of a Mobile Robot Using a Force-Reflection Joystick With Sensing Mechanism of Rotating Magnetic Field," *IEEEASME Trans. Mechatron.*, vol. 15, no. 1, pp. 17–26, 2010, doi: 10.1109/TMECH.2009.2013848.
- [5] S. K. Agrawal, X. Chen, C. Ragonese, and J. C. Galloway, "Training Toddlers Seated on Mobile Robots to Steer Using Force-Feedback Joystick," *IEEE Trans. Haptics*, vol. 5, no. 4, pp. 376–383, 2012, doi: 10.1109/TOH.2011.67.
- [6] Y. Kim, T. Oyabu, G. Obinata, and K. Hase, "Operability of Joystick-Type Steering Device Considering Human Arm Impedance Characteristics," *IEEE Trans. Syst. Man Cybern. - Part Syst. Hum.*, vol. 42, no. 2, pp. 295–306, 2012, doi: 10.1109/TSMCA.2011.2162501.
- [7] D. Scaradozzi, L. Sorbi, S. Zingaretti, M. Biagiola, and E. Omerdic, "Development and integration of a novel IP66 Force Feedback Joystick for offshore operations," in *22nd Mediterranean Conference*

- on *Control and Automation*, pp. 664–669, 2014, doi: 10.1109/MED.2014.6961449.
- [8] H. Zhang and S. K. Agrawal, “An Active Neck Brace Controlled by a Joystick to Assist Head Motion,” *IEEE Robot. Autom. Lett.*, vol. 3, no. 1, pp. 37–43, 2018, doi: 10.1109/LRA.2017.2728858.
- [9] K. A. Radhika, B. L. Raksha, B. R. Sujatha, U. Pruthviraj, and K. V. Gangadharan, “IoT Based Joystick Controlled Pibot Using Socket Communication,” in *2018 IEEE Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER)*, Mangalore (Mangaluru), pp. 121–125, 2018, doi: 10.1109/DISCOVER.2018.8674130.
- [10] M. Mohammadi, H. Knoche, M. Gaiheide, B. Bentsen, and L. N. S. Andreasen Struijk, “A high-resolution tongue-based joystick to enable robot control for individuals with severe disabilities,” in *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*, pp. 1043–1048, 2019, doi: 10.1109/ICORR.2019.8779434.
- [11] R. Rahman, M. S. Rahman, and J. R. Bhuiyan, “Joystick controlled industrial robotic system with robotic arm,” in *2019 IEEE International Conference on Robotics, Automation, Artificial-intelligence and Internet-of-Things (RAAICON)*, pp. 31–34, 2019, doi: 10.1109/RAAICON48939.2019.18.
- [12] Z. Wang *et al.*, “Joystick Car Drive System and its Application to Self-driving Microbus,” in *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*, pp. 632–637, 2020, doi: 10.1109/IECON43393.2020.9254364.
- [13] H. Zhang, B.-C. Chang, Y.-J. Rue, and S. K. Agrawal, “Using the Motion of the Head-Neck as a Joystick for Orientation Control,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 2, pp. 236–243, 2019, doi: 10.1109/TNSRE.2019.2894517.
- [14] M. Mohammadi, H. Knoche, and L. N. S. A. Struijk, “Continuous Tongue Robot Mapping for Paralyzed Individuals Improves the Functional Performance of Tongue-Based Robotic Assistance,” *IEEE Trans. Biomed. Eng.*, vol. 68, no. 8, pp. 2552–2562, 2021, doi: 10.1109/TBME.2021.3055250.
- [15] J. C. Yepes, J. J. Yepes, J. R. Martinez, and V. Z. Perez, “Implementation of an Android based teleoperation application for controlling a KUKA-KR6 robot by using sensor fusion,” in *2013 Pan American Health Care Exchanges (PAHCE)*, pp. 1–5, 2013, doi: 10.1109/PAHCE.2013.6568286.
- [16] P. Rouanet, P.-Y. Oudeyer, F. Danieau, and D. Filliat, “The Impact of Human–Robot Interfaces on the Learning of Visual Objects,” *IEEE Trans. Robot.*, vol. 29, no. 2, pp. 525–541, 2013, doi: 10.1109/TRO.2012.2228134.
- [17] J. Nadvornik and P. Smutny, “Remote control robot using Android mobile device,” in *Proceedings of the 2014 15th International Carpathian Control Conference (ICCC)*, pp. 373–378, 2014, doi: 10.1109/CarpathianCC.2014.6843630.
- [18] G. Mohanarajah, V. Usenko, M. Singh, R. D’Andrea, and M. Waibel, “Cloud-Based Collaborative 3D Mapping in Real-Time With Low-Cost Robots,” *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 2, pp. 423–431, 2015, doi: 10.1109/TASE.2015.2408456.
- [19] S. P. Donohoe, S. A. Velinsky, and T. A. Lasky, “Mechatronic Implementation of a Force Optimal Underconstrained Planar Cable Robot,” *IEEEASME Trans. Mechatron.*, vol. 21, no. 1, pp. 69–78, 2016, doi: 10.1109/TMECH.2015.2431192.
- [20] G. Loianno, C. Brunner, G. McGrath, and V. Kumar, “Estimation, Control, and Planning for Aggressive Flight With a Small Quadrotor With a Single Camera and IMU,” *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 404–411, 2017, doi: 10.1109/LRA.2016.2633290.
- [21] K.-T. Song, S.-Y. Jiang, and M.-H. Lin, “Interactive Teleoperation of a Mobile Manipulator Using a Shared-Control Approach,” *IEEE Trans. Hum.-Mach. Syst.*, vol. 46, no. 6, pp. 834–845, 2016, doi: 10.1109/THMS.2016.2586760.
- [22] Y. Wang *et al.*, “Monitoring Aquatic Debris Using Smartphone-Based Robots,” *IEEE Trans. Mob. Comput.*, vol. 15, no. 6, pp. 1412–1426, 2016, doi: 10.1109/TMC.2015.2460240.
- [23] B. Gokcen, F. Baygul, F. Cakmak, E. Uslu, M. F. Amasyali, and S. Yavuz, “Android application for simultaneously control of multiple land robots which have different drive strategy,” in *2017 International Conference on Computer Science and Engineering (UBMK)*, pp. 724–728, 2017, doi: 10.1109/UBMK.2017.8093513.
- [24] K.-R. Kim, S.-H. Jeong, W.-Y. Kim, Y. Jeon, K.-S. Kim, and J.-H. Hong, “Design of small mobile robot remotely controlled by an android operating system via bluetooth and NFC communication,” in *2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pp. 913–915, 2017, doi: 10.1109/URAI.2017.7992864.
- [25] G. Zheng, S. Bi, H. Min, K. Yang, and Y. Zhang, “Design and Implementation of Chinese Speech Robot Control System Based on Android Embedded Platform,” in *2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, pp. 498–503, 2017, doi: 10.1109/CYBER.2017.8446468.
- [26] J. Lim, G. S. Tewolde, J. Kwon, and S. Choi, “Design and Implementation of a Network Robotic Framework Using a Smartphone-Based Platform,” *IEEE Access*, vol. 7, pp. 59357–59368, 2019, doi: 10.1109/ACCESS.2019.2915344.
- [27] M. Alwateer, S. W. Loke, and N. Fernando, “Enabling Drone Services: Drone Crowdsourcing and Drone Scripting,” *IEEE Access*, vol. 7, pp. 110035–110049, 2019, doi: 10.1109/ACCESS.2019.2933234.
- [28] R. K. Megalingam *et al.*, “Amaran: An Unmanned Robotic Coconut Tree Climber and Harvester,” *IEEEASME Trans. Mechatron.*, pp. 1–1, 2020, doi: 10.1109/TMECH.2020.3014293.
- [29] L. Wu, R. Alqasemi, and R. Dubey, “Development of Smartphone-Based Human-Robot Interfaces for Individuals With Disabilities,” *IEEE Robot. Autom. Lett.*, vol. 5, no. 4, pp. 5835–5841, 2020, doi: 10.1109/LRA.2020.3010453.
- [30] W. Dudek and T. Winiarski, “Scheduling of a Robot’s Tasks With the TaskER Framework,” *IEEE Access*, vol. 8, pp. 161449–161471, 2020, doi: 10.1109/ACCESS.2020.3020265.
- [31] D. Kiryanov, R. Lavrenov, R. Safin, M. Svinin, and E. Magid, “Mobile application for controlling multiple robots,” in *2021 IEEE 16th Conference on Industrial Electronics and Applications (ICIEA)*, pp. 1913–1917, 2021, doi: 10.1109/ICIEA51954.2021.9516091.
- [32] Y.-B. Lin, H. Luo, C.-C. Liao, and Y.-F. Huang, “PuppetTalk: Conversation Between Glove Puppetry and Internet of Things,” *IEEE Access*, vol. 9, pp. 6786–6797, 2021, doi: 10.1109/ACCESS.2020.3048697.
- [33] M. Fuad and D. Purwanto, “Wall-Following Using a Kinect Sensor for Corridor Coverage Navigation,” *Journal of Theoretical and Applied Information Technology*, vol. 70, no. 1, pp. 106–111, 2014.
- [34] M. Fuad, “RGB-D Image Based Landmark Identification for Corridor Coverage Navigation,” *Journal of Theoretical and Applied Information Technology*, vol. 79, no. 3, pp. 437–443, 2015.
- [35] M. Fuad, T. Agustinah, D. Purwanto, T. A. Sardjono, and R. Dikaiono, “Robot Orientation Estimation Based on Single-Frame of Fish-eye Image,” *J. Phys. Conf. Ser.*, vol. 1569, no. 2, p. 022092, 2020, doi: 10.1088/1742-6596/1569/2/022092.
- [36] M. Fuad, “Translational Motion Estimation Using Kinect,” *J. Phys. Conf. Ser.*, vol. 1569, p. 032067, 2020, doi: 10.1088/1742-6596/1569/3/032067.
- [37] M. Fuad, “Skeleton based gesture to control manipulator,” in *2015 International Conference on Advanced Mechatronics, Intelligent Manufacturing, and Industrial Automation (ICAMIMIA)*, pp. 96–101, 2015, doi: 10.1109/ICAMIMIA.2015.7508010.
- [38] Y. Liao, P. Xiong, W. Min, W. Min, and J. Lu, “Dynamic Sign Language Recognition Based on Video Sequence With BLSTM-3D Residual Networks,” *IEEE Access*, vol. 7, pp. 38044–38054, 2019, doi: 10.1109/ACCESS.2019.2904749.
- [39] W. Fang, Y. Ding, F. Zhang, and J. Sheng, “Gesture Recognition Based on CNN and DCGAN for Calculation and Text Output,” *IEEE Access*, vol. 7, pp. 28230–28237, 2019, doi: 10.1109/ACCESS.2019.2901930.
- [40] A. D. Berenguer, M. C. Oveneke, H.-U.-R. Khalid, M. Alioscha-Perez, A. Bourdoux, and H. Sahli, “GestureVLAD: Combining Unsupervised Features Representation and Spatio-Temporal Aggregation for Doppler-Radar Gesture Recognition,” *IEEE Access*, vol. 7, pp. 137122–137135, 2019, doi: 10.1109/ACCESS.2019.2942305.
- [41] Z. Wang *et al.*, “Hand Gesture Recognition Based on Active Ultrasonic Sensing of Smartphone: A Survey,” *IEEE Access*, vol. 7, pp. 111897–111922, 2019, doi: 10.1109/ACCESS.2019.2933987.

- [42] K. Cheng, N. Ye, R. Malekian, and R. Wang, "In-Air Gesture Interaction: Real Time Hand Posture Recognition Using Passive RFID Tags," *IEEE Access*, vol. 7, pp. 94460–94472, 2019, doi: 10.1109/ACCESS.2019.2928318.
- [43] C. Liu, Y. Li, D. Ao, and H. Tian, "Spectrum-Based Hand Gesture Recognition Using Millimeter-Wave Radar Parameter Measurements," *IEEE Access*, vol. 7, pp. 79147–79158, 2019, doi: 10.1109/ACCESS.2019.2923122.
- [44] Y. Wang, S. Wang, M. Zhou, Q. Jiang, and Z. Tian, "TS-I3D Based Hand Gesture Recognition Method With Radar Sensor," *IEEE Access*, vol. 7, pp. 22902–22913, 2019, doi: 10.1109/ACCESS.2019.2897060.
- [45] T. Zhang, T. Song, D. Chen, T. Zhang, and J. Zhuang, "WiGrus: A Wifi-Based Gesture Recognition System Using Software-Defined Radio," *IEEE Access*, vol. 7, pp. 131102–131113, 2019, doi: 10.1109/ACCESS.2019.2940386.
- [46] Y. Wang, A. Ren, M. Zhou, W. Wang, and X. Yang, "A Novel Detection and Recognition Method for Continuous Hand Gesture Using FMCW Radar," *IEEE Access*, vol. 8, pp. 167264–167275, 2020, doi: 10.1109/ACCESS.2020.3023187.
- [47] M. Lee and J. Bae, "Deep Learning Based Real-Time Recognition of Dynamic Finger Gestures Using a Data Glove," *IEEE Access*, vol. 8, pp. 219923–219933, 2020, doi: 10.1109/ACCESS.2020.3039401.
- [48] H. Heydarian, P. V. Rouast, M. T. P. Adam, T. Burrows, C. E. Collins, and M. E. Rollo, "Deep Learning for Intake Gesture Detection From Wrist-Worn Inertial Sensors: The Effects of Data Preprocessing, Sensor Modalities, and Sensor Positions," *IEEE Access*, vol. 8, pp. 164936–164949, 2020, doi: 10.1109/ACCESS.2020.3022042.
- [49] M. Al-Hammadi *et al.*, "Deep Learning-Based Approach for Sign Language Gesture Recognition With Efficient Hand Gesture Representation," *IEEE Access*, vol. 8, pp. 192527–192542, 2020, doi: 10.1109/ACCESS.2020.3032140.
- [50] S. Aly and W. Aly, "DeepArSLR: A Novel Signer-Independent Deep Learning Framework for Isolated Arabic Sign Language Gestures Recognition," *IEEE Access*, vol. 8, pp. 83199–83212, 2020, doi: 10.1109/ACCESS.2020.2990699.
- [51] Y. Guo, S. Yang, N. Li, and X. Jiang, "Device-Free Localization Scheme With Time-Varying Gestures Using Block Compressive Sensing," *IEEE Access*, vol. 8, pp. 88951–88960, 2020, doi: 10.1109/ACCESS.2020.2993576.
- [52] Y. Che and Y. Qi, "Embedding Gesture Prior to Joint Shape Optimization Based Real-Time 3D Hand Tracking," *IEEE Access*, vol. 8, pp. 34204–34214, 2020, doi: 10.1109/ACCESS.2020.2974551.
- [53] J. Y. Oh, J.-H. Park, and J.-M. Park, "FingerTouch: Touch Interaction Using a Fingernail-Mounted Sensor on a Head-Mounted Display for Augmented Reality," *IEEE Access*, vol. 8, pp. 101192–101208, 2020, doi: 10.1109/ACCESS.2020.2997972.
- [54] Z. Ni and Q. Li, "Fusion Learning Model for Mobile Face Safe Detection and Facial Gesture Analysis," *IEEE Access*, vol. 8, pp. 61043–61050, 2020, doi: 10.1109/ACCESS.2019.2948714.
- [55] S. M. Aslam and S. Samreen, "Gesture Recognition Algorithm for Visually Blind Touch Interaction Optimization Using Crow Search Method," *IEEE Access*, vol. 8, pp. 127560–127568, 2020, doi: 10.1109/ACCESS.2020.3006443.
- [56] G. Luo, P. Yang, M. Chen, and P. Li, "HCI on the Table: Robust Gesture Recognition Using Acoustic Sensing in Your Hand," *IEEE Access*, vol. 8, pp. 31481–31498, 2020, doi: 10.1109/ACCESS.2020.2973305.
- [57] E. A. Ibrahim, M. Geilen, M. Li, and J. P. De Gyvez, "Multi-Angle Fusion for Low-Cost Near-Field Ultrasonic in-Air Gesture Recognition," *IEEE Access*, vol. 8, pp. 191204–191218, 2020, doi: 10.1109/ACCESS.2020.3031677.
- [58] P. V. Rouast, H. Heydarian, M. T. P. Adam, and M. E. Rollo, "OREBA: A Dataset for Objectively Recognizing Eating Behavior and Associated Intake," *IEEE Access*, vol. 8, pp. 181955–181963, 2020, doi: 10.1109/ACCESS.2020.3026965.
- [59] F. Ma, F. Song, Y. Liu, and J. Niu, "Quantitative Analysis on the Interaction Fatigue of Natural Gestures," *IEEE Access*, vol. 8, pp. 190797–190811, 2020, doi: 10.1109/ACCESS.2020.3031967.
- [60] R. Ji, "Research on Basketball Shooting Action Based on Image Feature Extraction and Machine Learning," *IEEE Access*, vol. 8, pp. 138743–138751, 2020, doi: 10.1109/ACCESS.2020.3012456.
- [61] C. Wu *et al.*, "sEMG Measurement Position and Feature Optimization Strategy for Gesture Recognition Based on ANOVA and Neural Networks," *IEEE Access*, vol. 8, pp. 56290–56299, 2020, doi: 10.1109/ACCESS.2020.2982405.
- [62] H. Bi, J. Zhang, and Y. Chen, "SmartGe: Identifying Pen-Holding Gesture With Smartwatch," *IEEE Access*, vol. 8, pp. 28820–28830, 2020, doi: 10.1109/ACCESS.2020.2967770.
- [63] F. S. Khan, M. N. H. Mohd, D. M. Soomro, S. Bagchi, and M. D. Khan, "3D Hand Gestures Segmentation and Optimized Classification Using Deep Learning," *IEEE Access*, vol. 9, pp. 131614–131624, 2021, doi: 10.1109/ACCESS.2021.3114871.
- [64] A. Macintosh, N. Vignais, E. Desailly, E. Biddiss, and V. Vigneron, "A Classification and Calibration Procedure for Gesture Specific Home-Based Therapy Exercise in Young People With Cerebral Palsy," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 144–155, 2021, doi: 10.1109/TNSRE.2020.3038370.
- [65] U. Farooq, M. S. Mohd Rahim, N. S. Khan, S. Rasheed, and A. Abid, "A Crowdsourcing-Based Framework for the Development and Validation of Machine Readable Parallel Corpus for Sign Languages," *IEEE Access*, vol. 9, pp. 91788–91806, 2021, doi: 10.1109/ACCESS.2021.3091433.
- [66] K.-C. Lin and R.-J. Wai, "A Feasible Fall Evaluation System via Artificial Intelligence Gesture Detection of Gait and Balance for Sub-Healthy Community- Dwelling Older Adults in Taiwan," *IEEE Access*, vol. 9, pp. 146404–146413, 2021, doi: 10.1109/ACCESS.2021.3123297.
- [67] N. Mohamed, M. B. Mustafa, and N. Jomhari, "A Review of the Hand Gesture Recognition System: Current Progress and Future Directions," *IEEE Access*, vol. 9, pp. 157422–157436, 2021, doi: 10.1109/ACCESS.2021.3129650.
- [68] M. A. Bencherif *et al.*, "Arabic Sign Language Recognition System Using 2D Hands and Body Skeleton Data," *IEEE Access*, vol. 9, pp. 59612–59627, 2021, doi: 10.1109/ACCESS.2021.3069714.
- [69] S. A. Raurale, J. McAllister, and J. M. D. Rincon, "EMG Biometric Systems Based on Different Wrist-Hand Movements," *IEEE Access*, vol. 9, pp. 12256–12266, 2021, doi: 10.1109/ACCESS.2021.3050704.
- [70] E. Rahimian, S. Zabihi, A. Asif, D. Farina, S. F. Atashzar, and A. Mohammadi, "FS-HGR: Few-Shot Learning for Hand Gesture Recognition via Electromyography," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1004–1015, 2021, doi: 10.1109/TNSRE.2021.3077413.
- [71] P. Nallabolu, L. Zhang, H. Hong, and C. Li, "Human Presence Sensing and Gesture Recognition for Smart Home Applications With Moving and Stationary Clutter Suppression Using a 60-GHz Digital Beamforming FMCW Radar," *IEEE Access*, vol. 9, pp. 72857–72866, 2021, doi: 10.1109/ACCESS.2021.3080655.
- [72] D. F. Q. Melo, B. M. C. Silva, N. Pombo, and L. Xu, "Internet of Things Assisted Monitoring Using Ultrasound-Based Gesture Recognition Contactless System," *IEEE Access*, vol. 9, pp. 90185–90194, 2021, doi: 10.1109/ACCESS.2021.3089940.
- [73] M. Altmann, P. Ott, N. C. Stache, and C. Waldschmidt, "Multi-Modal Cross Learning for an FMCW Radar Assisted by Thermal and RGB Cameras to Monitor Gestures and Cooking Processes," *IEEE Access*, vol. 9, pp. 22295–22303, 2021, doi: 10.1109/ACCESS.2021.3056878.
- [74] A. R. Elshenaway and S. K. Guirguis, "On-Air Hand-Drawn Doodles for IoT Devices Authentication During COVID-19," *IEEE Access*, vol. 9, pp. 161723–161744, 2021, doi: 10.1109/ACCESS.2021.3131551.
- [75] X. Jiang *et al.*, "Open Access Dataset, Toolbox and Benchmark Processing Results of High-Density Surface Electromyogram Recordings," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1035–1046, 2021, doi: 10.1109/TNSRE.2021.3082551.
- [76] M. Chmurski, M. Zubert, K. Bierzynski, and A. Santra, "Analysis of Edge-Optimized Deep Learning Classifiers for Radar-Based Gesture Recognition," *IEEE Access*, vol. 9, pp. 74406–74421, 2021, doi: 10.1109/ACCESS.2021.3081353.

- [77] A. Pradhan, J. He, and N. Jiang, "Performance Optimization of Surface Electromyography Based Biometric Sensing System for Both Verification and Identification," *IEEE Sens. J.*, vol. 21, no. 19, pp. 21718–21729, 2021, doi: 10.1109/JSEN.2021.3079428.
- [78] N. Pan, "Research on Music Wireless Control Based on Motion Tracking Sensor and Internet of Things," *IEEE Access*, vol. 9, pp. 48803–48810, 2021, doi: 10.1109/ACCESS.2021.3064565.
- [79] P. V. Rouast and M. T. P. Adam, "Single-Stage Intake Gesture Detection Using CTC Loss and Extended Prefix Beam Search," *IEEE J. Biomed. Health Inform.*, vol. 25, no. 7, pp. 2733–2743, 2021, doi: 10.1109/JBHI.2020.3046613.
- [80] Z. A. Kahar, P. S. Sulaiman, F. Khalid, and A. Azman, "Skeleton Joints Moment (SJM): A Hand Gesture Dimensionality Reduction for Central Nervous System Interaction," *IEEE Access*, vol. 9, pp. 146640–146652, 2021, doi: 10.1109/ACCESS.2021.3123570.
- [81] M.-C. Su, J.-H. Chen, A. M. Arifai, S.-Y. Tsai, and H.-H. Wei, "Smart Living: An Interactive Control System for Household Appliances," *IEEE Access*, vol. 9, pp. 14897–14904, 2021, doi: 10.1109/ACCESS.2021.3051253.
- [82] Z. Chen, J. Yang, and H. Xie, "Surface-Electromyography-Based Gesture Recognition Using a Multistream Fusion Strategy," *IEEE Access*, vol. 9, pp. 50583–50592, 2021, doi: 10.1109/ACCESS.2021.3059499.
- [83] H. Zhou, C. Tawk, and G. Alici, "A 3D Printed Soft Robotic Hand With Embedded Soft Sensors for Direct Transition Between Hand Gestures and Improved Grasping Quality and Diversity," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 550–558, 2022, doi: 10.1109/TNSRE.2022.3156116.
- [84] R. Hu, X. Chen, H. Zhang, X. Zhang, and X. Chen, "A Novel Myoelectric Control Scheme Supporting Synchronous Gesture Recognition and Muscle Force Estimation," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 1127–1137, 2022, doi: 10.1109/TNSRE.2022.3166764.
- [85] Q. Zengyu *et al.*, "A Simultaneous Gesture Classification and Force Estimation Strategy Based on Wearable A-Mode Ultrasound and Cascade Model," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 2301–2311, 2022, doi: 10.1109/TNSRE.2022.3196926.
- [86] X. Song, S. S. Van De Ven, L. Liu, F. J. Wouda, H. Wang, and P. B. Shull, "Activities of Daily Living-Based Rehabilitation System for Arm and Hand Motor Function Retraining After Stroke," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 621–631, 2022, doi: 10.1109/TNSRE.2022.3156387.
- [87] S. B. Abdullahi and K. Chamnongthai, "American Sign Language Words Recognition Using Spatio-Temporal Prosodic and Angle Features: A Sequential Learning Approach," *IEEE Access*, vol. 10, pp. 15911–15923, 2022, doi: 10.1109/ACCESS.2022.3148132.
- [88] A. Ali *et al.*, "End-to-End Dynamic Gesture Recognition Using MmWave Radar," *IEEE Access*, vol. 10, pp. 88692–88706, 2022, doi: 10.1109/ACCESS.2022.3199411.
- [89] J. Li, S. Ray, V. Rajanna, and T. Hammond, "Evaluating the Performance of Machine Learning Algorithms in Gaze Gesture Recognition Systems," *IEEE Access*, vol. 10, pp. 1020–1035, 2022, doi: 10.1109/ACCESS.2021.3136153.
- [90] J.-W. Choi, C.-W. Park, and J.-H. Kim, "FMCW Radar-Based Real-Time Hand Gesture Recognition System Capable of Out-of-Distribution Detection," *IEEE Access*, vol. 10, pp. 87425–87434, 2022, doi: 10.1109/ACCESS.2022.3200757.
- [91] T. Wang, Y. Zhao, and Q. Wang, "Hand Gesture Recognition With Flexible Capacitive Wristband Using Triplet Network in Inter-Day Applications," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 2876–2885, 2022, doi: 10.1109/TNSRE.2022.3212705.
- [92] Q. M. Areeb, Maryam, M. Nadeem, R. Alroobaea, and F. Anwer, "Helping Hearing-Impaired in Emergency Situations: A Deep Learning-Based Approach," *IEEE Access*, vol. 10, pp. 8502–8517, 2022, doi: 10.1109/ACCESS.2022.3142918.
- [93] C. Canuto, E. O. Freire, L. Molina, E. A. N. Carvalho, and S. N. Givigi, "Intuitiveness Level: Frustration-Based Methodology for Human–Robot Interaction Gesture Elicitation," *IEEE Access*, vol. 10, pp. 17145–17154, 2022, doi: 10.1109/ACCESS.2022.3146838.
- [94] J. E. Lara, L. K. Cheng, O. Rohrl, and N. Paskaranandavivel, "Muscle-Specific High-Density Electromyography Arrays for Hand Gesture Classification," *IEEE Trans. Biomed. Eng.*, vol. 69, no. 5, pp. 1758–1766, 2022, doi: 10.1109/TBME.2021.3131297.
- [95] J. Li, J. Meng, H. Gong, and Z. Fan, "Research on Continuous Dynamic Gesture Recognition of Chinese Sign Language Based on Multi-Mode Fusion," *IEEE Access*, vol. 10, pp. 106946–106957, 2022, doi: 10.1109/ACCESS.2022.3212064.
- [96] J. Xu, H. Wang, J. Zhang, and L. Cai, "Robust Hand Gesture Recognition Based on RGB-D Data for Natural Human–Computer Interaction," *IEEE Access*, vol. 10, pp. 54549–54562, 2022, doi: 10.1109/ACCESS.2022.3176717.
- [97] B. Zhu, D. Zhang, Y. Chu, Y. Gu, and X. Zhao, "SeNic: An Open Source Dataset for sEMG-Based Gesture Recognition in Non-Ideal Conditions," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 1252–1260, 2022, doi: 10.1109/TNSRE.2022.3173708.
- [98] D. R. Kothadiya, C. M. Bhatt, T. Saba, A. Rehman, and S. A. Bahaj, "SIGNFORMER: DeepVision Transformer for Sign Language Recognition," *IEEE Access*, vol. 11, pp. 4730–4739, 2023, doi: 10.1109/ACCESS.2022.3231130.
- [99] B. Fang *et al.*, "Simultaneous sEMG Recognition of Gestures and Force Levels for Interaction With Prosthetic Hand," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 2426–2436, 2022, doi: 10.1109/TNSRE.2022.3199809.
- [100] Z. Pan, X. Qian, and H. Li, "Speaker Extraction With Co-Speech Gestures Cue," *IEEE Signal Process. Lett.*, vol. 29, pp. 1467–1471, 2022, doi: 10.1109/LSP.2022.3175130.
- [101] Z. Lu, S. Cai, B. Chen, Z. Liu, L. Guo, and L. Yao, "Wearable Real-Time Gesture Recognition Scheme Based on A-Mode Ultrasound," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 2623–2629, 2022, doi: 10.1109/TNSRE.2022.3205026.
- [102] F. Sufyan, S. Sagar, Z. Ashraf, S. Nayel, M. S. Chishti, and A. Banerjee, "A Novel and Lightweight Real-Time Continuous Motion Gesture Recognition Algorithm for Smartphones," *IEEE Access*, vol. 11, pp. 42725–42737, 2023, doi: 10.1109/ACCESS.2023.3255402.
- [103] S. Khetavath *et al.*, "An Intelligent Heuristic Manta-Ray Foraging Optimization and Adaptive Extreme Learning Machine for Hand Gesture Image Recognition," *Big Data Min. Anal.*, vol. 6, no. 3, pp. 321–335, 2023, doi: 10.26599/BDMA.2022.9020036.
- [104] B. Hampiholi, C. Jarvers, W. Mader, and H. Neumann, "Convolutional Transformer Fusion Blocks for Multi-Modal Gesture Recognition," *IEEE Access*, vol. 11, pp. 34094–34103, 2023, doi: 10.1109/ACCESS.2023.3263812.
- [105] Q. Li, Z. Luo, R. Qi, and J. Zheng, "DeepTPA-Net: A Deep Triple Attention Network for sEMG-Based Hand Gesture Recognition," *IEEE Access*, vol. 11, pp. 96797–96807, 2023, doi: 10.1109/ACCESS.2023.3312219.
- [106] S. Wan, L. Yang, K. Ding, and D. Qiu, "Dynamic Gesture Recognition Based on Three-Stream Coordinate Attention Network and Knowledge Distillation," *IEEE Access*, vol. 11, pp. 50547–50559, 2023, doi: 10.1109/ACCESS.2023.3278100.
- [107] A. S. M. Miah, Md. A. M. Hasan, and J. Shin, "Dynamic Hand Gesture Recognition Using Multi-Branch Attention Based Graph and General Deep Learning Model," *IEEE Access*, vol. 11, pp. 4703–4716, 2023, doi: 10.1109/ACCESS.2023.3235368.
- [108] Y. Ghasemi, H. Jeong, K.-B. Park, S. H. Choi, and J. Y. Lee, "Evaluating User Interactions in Wearable Extended Reality: Modeling, Online Remote Survey, and In-Lab Experimental Methods," *IEEE Access*, vol. 11, pp. 77856–77872, 2023, doi: 10.1109/ACCESS.2023.3298598.
- [109] Y. Pang, Y. Gong, and X. Hao, "Hand Gesture Recognition Based on Point Cloud Sequences and Inverse Kinematics Model," *IEEE Access*, vol. 11, pp. 44082–44091, 2023, doi: 10.1109/ACCESS.2023.3272746.
- [110] H. Ansar *et al.*, "Hand Gesture Recognition for Characters Understanding Using Convex Hull Landmarks and Geometric Features," *IEEE Access*, vol. 11, pp. 82065–82078, 2023, doi: 10.1109/ACCESS.2023.3300712.
- [111] H.-Q. Nguyen *et al.*, "Hand Gesture Recognition From Wrist-Worn Camera for Human–Machine Interaction," *IEEE Access*, vol. 11, pp. 53262–53274, 2023, doi: 10.1109/ACCESS.2023.3279845.
- [112] W. E. Villegas-Ch, J. García-Ortiz, and S. Sánchez-Viteri, "Identification of Emotions From Facial Gestures in a Teaching Environment With the Use of Machine Learning Techniques," *IEEE*

- Access*, vol. 11, pp. 38010–38022, 2023, doi: 10.1109/ACCESS.2023.3267007.
- [113] S. B. Abdullahi and K. Chamnongthai, “IDF-Sign: Addressing Inconsistent Depth Features for Dynamic Sign Word Recognition,” *IEEE Access*, vol. 11, pp. 88511–88526, 2023, doi: 10.1109/ACCESS.2023.3305255.
- [114] K. Wang, Y. Chen, Y. Zhang, X. Yang, and C. Hu, “Iterative Self-Training Based Domain Adaptation for Cross-User sEMG Gesture Recognition,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 2974–2987, 2023, doi: 10.1109/TNSRE.2023.3293334.
- [115] Y. Liu, Q. Qiao, S. Shi, X. Wang, M. Yang, and X. Gao, “Keyframe Extraction and Process Recognition Method for Assembly Operation Based on Density Clustering,” *IEEE Access*, vol. 11, pp. 13564–13573, 2023, doi: 10.1109/ACCESS.2023.3243083.
- [116] A. Raza, A. M. Qadri, I. Akhtar, N. A. Samee, and M. Alabdulhafith, “LogRF: An Approach to Human Pose Estimation Using Skeleton Landmarks for Physiotherapy Fitness Exercise Correction,” *IEEE Access*, vol. 11, pp. 107930–107939, 2023, doi: 10.1109/ACCESS.2023.3320144.
- [117] J. Ma, X. Ren, H. Li, W. Li, V. Y. Tsviatkou, and A. A. Boriskevich, “Noise-Against Skeleton Extraction Framework and Application on Hand Gesture Recognition,” *IEEE Access*, vol. 11, pp. 9547–9559, 2023, doi: 10.1109/ACCESS.2023.3240313.
- [118] A. Nimbekar, Y. V. S. Dinesh, A. Gautam, V. Hunsigida, A. R. Nali, and A. Acharyya, “Reconfigurable VLSI Design Architecture for Deep Learning Established Forelimb and Hindlimb Gesture Recognition for Rehabilitation Application,” *IEEE Access*, vol. 11, pp. 70061–70070, 2023, doi: 10.1109/ACCESS.2023.3293422.
- [119] Y. Lin, R. Palaniappan, P. De Wilde, and L. Li, “Robust Long-Term Hand Grasp Recognition With Raw Electromyographic Signals Using Multidimensional Uncertainty-Aware Models,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 962–971, 2023, doi: 10.1109/TNSRE.2023.3236982.
- [120] E. F. Asl, S. Ebadollahi, R. Vahidnia, and A. Jalali, “Statistical Database of Human Motion Recognition Using Wearable IoT—A Review,” *IEEE Sens. J.*, vol. 23, no. 14, pp. 15253–15304, 2023, doi: 10.1109/JSEN.2023.3282171.
- [121] S. Zhang and S. Zhang, “A Novel Human-3DTV Interaction System Based on Free Hand Gestures and a Touch-Based Virtual Interface,” *IEEE Access*, vol. 7, pp. 165961–165973, 2019, doi: 10.1109/ACCESS.2019.2953798.
- [122] E. S. Goh, M. S. Sunar, and A. W. Ismail, “3D Object Manipulation Techniques in Handheld Mobile Augmented Reality Interface: A Review,” *IEEE Access*, vol. 7, pp. 40581–40601, 2019, doi: 10.1109/ACCESS.2019.2906394.
- [123] S. Jacob, V. G. Menon, F. Al-Turjman, V. P. G., and L. Mostarda, “Artificial Muscle Intelligence System With Deep Learning for Post-Stroke Assistance and Rehabilitation,” *IEEE Access*, vol. 7, pp. 133463–133473, 2019, doi: 10.1109/ACCESS.2019.2941491.
- [124] H. Brock, J. Ponce Chulani, L. Merino, D. Szapiro, and R. Gomez, “Developing a Lightweight Rock-Paper-Scissors Framework for Human-Robot Collaborative Gaming,” *IEEE Access*, vol. 8, pp. 202958–202968, 2020, doi: 10.1109/ACCESS.2020.3033550.
- [125] A. Ciscal, J. Perez-Turiel, J.-C. Fraile, D. Sierra, and E. De La Fuente, “RobHand: A Hand Exoskeleton With Real-Time EMG-Driven Embedded Control. Quantifying Hand Gesture Recognition Delays for Bilateral Rehabilitation,” *IEEE Access*, vol. 9, pp. 137809–137823, 2021, doi: 10.1109/ACCESS.2021.3118281.
- [126] J. A. Mahmud, B. C. Das, J. Shin, K. Md. Hasib, R. Sadik, and M. F. Mridha, “3D Gesture Recognition and Adaptation for Human–Robot Interaction,” *IEEE Access*, vol. 10, pp. 116485–116513, 2022, doi: 10.1109/ACCESS.2022.3218679.
- [127] A. Rachmad, M. Fuad, and E. M. S. Rochman, “Convolutional Neural Network-Based Classification Model of Corn Leaf Disease,” *Math. Model. Eng. Probl.*, vol. 10, no. 2, pp. 530–536, 2023, doi: 10.18280/mmep.100220.
- [128] X. Guan, J. Huang, and T. Tang, “Robot vision application on embedded vision implementation with digital signal processor,” *International Journal of Advanced Robotic Systems*, vol. 17, no. 1, 2020, doi:10.1177/1729881419900437.
- [129] B. Zhang, C. Quan and F. Ren, “Study on CNN in the recognition of emotion in audio and images,” 2016 IEEE/ACIS 15th *International Conference on Computer and Information Science (ICIS)*, pp. 1–5, 2016, doi: 10.1109/ICIS.2016.7550778.
- [130] J. Chang, J. Xiao, J. Chai, and Z. Zhou, “An Improved Faster R-CNN Algorithm for Gesture Recognition in Human-Robot Interaction,” in *2019 Chinese Automation Congress (CAC)*, pp. 5761–5764, 2019, doi: 10.1109/CAC48633.2019.8997339.