Intelligent Wheelchair Control System Based on Finger Pose Recognition

Iswahyudi
Dept. of Electrical Engineering
University of Jember
Jember, Indonesia
email: yudiokyes@gmail.com

Khairul Anam
Dept. of Electrical Engineering
University of Jember
Jember, Indonesia
email: Khairul@unej.ac.id

Azmi Saleh
Dept. of Electrical Engineering
University of Jember
Jember, Indonesia
email: azmi2009@gmail.com

***Abstract*— In the old day, wheelchairs are moved manually by using hand or with the assistance of someone else. Users of this wheelchair get tired quickly if they have to walk long distances. The electric wheelchair emerged as a form of innovation and development for the manual wheelchair. This paper presented the control system of the electric wheelchair based on finger poses using the Convolutional Neural Network (CNN). The camera is used to take pictures of five-finger poses. Images are selected only in certain sections using Region of Interest (ROI). The five-finger poses represent the movement of the electric wheelchair to stop, right, left, forward, and backward. The experimental results indicated that the accuracy of the finger pose detection is about 93.6%. Therefore, the control system using CNN can be a potential solution for an electric wheelchair.**

Keywords*— *Object Recognition, Region of Interest (ROI), Convolutional Neural Network (CNN)

I. INTRODUCTION

Persons with disabilities are people who experience physical, intellectual, mental, and or sensory limitations for an extended time. When they interact with the environment, they meet some obstacles and difficulties [1]. Data from the Inter-Census Population Survey or SUPAS (2015) [2] shows that as many as 21.84 million or about 8.56 percent of Indonesia's population, are those with disabilities. We can know that people with foot disabilities, whether caused by accidents or due to congenital disabilities, have difficulty walking. To walk around, they need a wheelchair. A wheelchair is generally moved manually by using hand strength or with the assistance of another person. One of the problems when using a wheelchair is that the user will get tired quickly especially for long distance travelling. To solve this problem, the manual wheelchair was tranformed into electric wheelchairs by adding an electric motor to drive the wheelchair. The development of wheelchairs continues not only in terms of the propulsion system, but the navigation control system for wheelchair movements has also been developed. Currently, electric wheelchairs have also experienced developments in the navigation or movement control system. Research on the movement control system for electric wheelchairs has also been developed to provide convenience for the wearer. There are many ways to control the movement of a wheelchair, such as using a joystick, voice signal, and a camera. In this study, we equip the electric wheelchair with a camera using the Convolutional Neural Network (CNN) algorithm in recognizing the pose finger. Image data in the form of five types of finger poses representing the classified control of movement of an electric wheelchair to stop, right, left, forward and backward.

Several previous studies has published research on hand gesture detection such as [3]. In this study, the classification process for finger pose recognition was carried out using Zernike Moment (ZMs) for feature extraction and three classifications, i.e. k-Nearest Neighbour (KNN), Artificial Neural Network (ANN), and Support Vector Machine (SVM) with accuracy of 77,5%, 82,5%, and 91%, respectively. Another research [4] focused on finger pose recognition to control robot movements. They extracted fature using ZMs and classified those features using ANN. The results of this study are expected that the proposed system provides an alternative way to control the robot. Another research employed Deep learning for hand gesture recognition[5],, using Convolutional Neural Networks (CNNs) and Stacked Denoising Autoencoders (SDAEs) with the accuracy of 91.33 and 92.83 %, respectively. Furhermore [6] focused on wheelchair movement control based on the identification of the pupil movement of the eyeball. Extraction feature using Face and Eye Detection and Canny Edge Detection whereas classification techniques using Eye Pupil Center Classification with the control system Raspberry Pi. The results of the wheelchair movement control study experienced some time delay, dark light places affected wheelchair performance and it was difficult to track pupils in dark light.

Based on some previous research, researcher chose CNN[7][8][9][10]. The reason for choosing CNN is because it is effective in solving many problems related to object recognition. The purpose of this paper to employ CNN to identify the pose finger as the control source of electrical wheelchair.

II. THE PROPOSED METHOD

A. Object Recognition

Object recognition is a technique contained in computer vision to identify objects in an image or video. The purpose of object recognition is to teach computers how humans can recognize an object in an image or video.

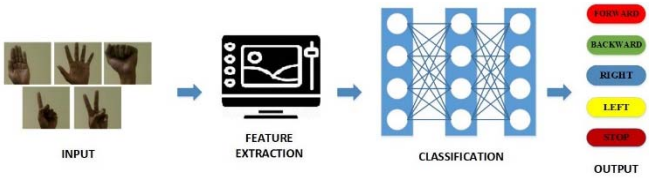


Fig. 1. Ilustration of Hand Pose Recognition

Fig. 1 explains the image recognition techniques that can identify and classify five types of finger poses that represent the movement of an electric wheelchair. In this study, researchers proposed image recognition techniques using the method deep learning with algorithm convolutional neural network (CNN).

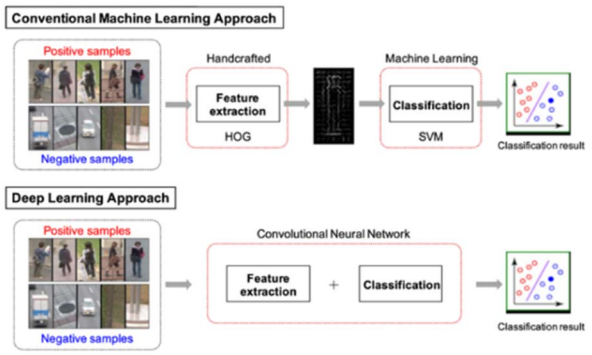


Fig. 2. ML and DL Image Recognition[11]

Fig. 2 is a picture of the steps that are generally taken in designing an object recognition system or object recognition in machine learning (ML) and deep learning (DL) algorithms. The fundamental difference between ML and DL is in DL the feature extraction and classification process become an inseparable part, but in ML the feature extraction and classification processes are carried out separately.

B. *Region of Interest (ROI)*

In digital image processing, sometimes we only need an image processor only in certain parts or areas. The desired area is called the Region of Interest (ROI)[12]. The purpose of using ROI in this study is to optimize the performance of the system in detecting finger poses in real time. Without ROI, image processing is performed on all image pixels without exception

Fig. 3. The Results of Cropping And Finger Detection in ROI

Figure 3 is the result of the cropping process to get the image area that is needed using the cropping technique. Certain areas of the image are delimited by a box with a width of 230 pixels and a height of 300 pixels. The bounding box in the frame has top, right, bottom, left = 100, 370, 400, 600.

C. *Convolutional Neural Network (CNN)*

Convolutional Neural Network (CNN) is a development of a Multilayer Perceptron (MLP) which is designed to process two-dimensional data in the form of images

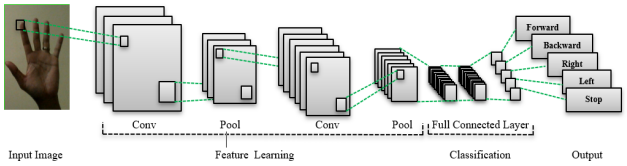


Fig. 4. Architecture of Convolutional Neural Network

CNN's ability is claimed to be the best model for solving problems object recognition and object detection. CNN consists of 2 main parts, namely the feature learning and classification features. Feature learning consisting of Convolutional layer, Pooling, and ReLU. Classification feature contains Fully Connected Layer (FCL).

D. *Convolutional Layer*

The convolution process makes use of what is known as a filter. Like images, filters have a certain height, width, and thickness. Filters are defined by specific or random values. The convolutional layer is the most important component of CNN. Convolution is a mathematical term where application of a function to the output of another function repeatedly. The mathematical operation of the convolution can be written as follows:

$$s(t) = (x * t)(t) = \sum_{t=1}^{\infty} x(\alpha) * w(t - \alpha) \quad (1)$$

From equation (1) it can be explained the function s(t) provides a single output in the form of a feature map. The first argument is input which is *x* and the second argument is *w* as the kernel or filter. When viewed as a two-dimensional image input, it can be said that t is a pixel.
To calculate the number of activated neurons in an output using a hyperparameter using the formula:

$$(W - F + 2P)/(S + 1) \quad (2)$$

From equation (2) it can be explain, it can be calculated the spatial size of the output volume where the hyperparameters used are the volume size (W), filter (F), applied Stride (S) and the amount of zero padding used (P). Stride is the value used to shift the filter through the image input and Zero Padding is the value to get zeros around the image border.

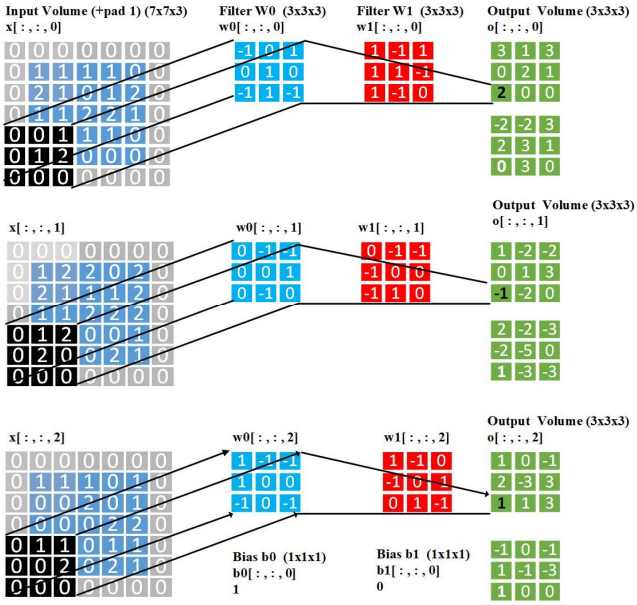


Fig. 5. Ilustration Convolution Process with 2 Filter

In Fig. 5 can be explained at each image position a number is generated which is dot product between parts of the image with filter used. Process shifting (convolve) filter in every possible position filter on the resulting image a activation map.

E. *Activation Function*

The activation function is at the stage after carrying out convolution and before carrying out the pooling layer. In the results of the convolution, an activation function is carried out.

The function of activation to determine whether neuron must be active or not based weighted sum from the input. In general there are 2 types of activation functions, namely Linier and Non Linier Activation Function. The activation function commonly used on CNN is the activation function ReLU (Rectified Linier Unit) and tanh( ).

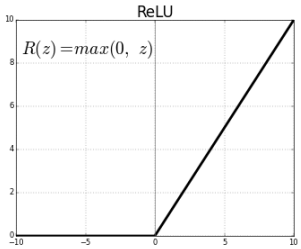


Fig. 6. Activation function of ReLU

F. Pooling Layer

Pooling is a reduction in the size of the matrix, usually after the convolution process, consisting of a filter of a certain size and stride which will alternately shift the entire feature map area.

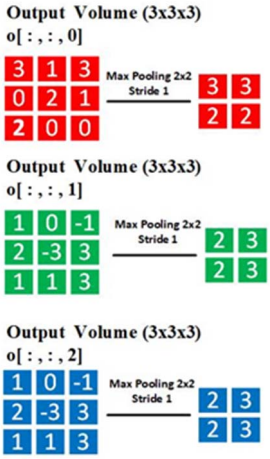


Fig. 7. Example of max pooling

. Based on the picture above shows the process of max-pooling. The output of the pooling process is a matrix with smaller dimensions compared to the initial image. The pooling layer above will operate at each slice of the input volume depth in turn. When viewed from the image above, the max-pooling operation uses a 2x2 filter size. The input to the process is 3x3 in size, from each of the 3 numbers in the operation input the maximum value is taken then proceed to make a new output size to be 2x2

G. Fully Connected Layer

Fully connected layer (FCL) is a layer where all the activation neurons from the previous layer are connected all the neuron on the next layer as is done in an ordinary artificial neural network. Any output from the previous convolutional layer is converted first into one-dimensional data before it can be linked to all neuron in the layer Fully Connected. The purpose of the FCL is to process data so that it can be classified. In the FCL layer all neuron are connected as a whole

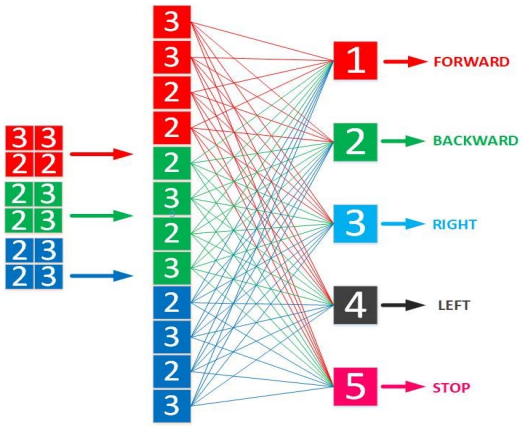


Fig. 8. Fully connected layer

H. Creating Model Dataset

The first thing to prepare is that we have to create and collect data samples. In this study the authors used five types of finger poses to identify and classify. Finger pose data samples were taken from selected subjects. The finger pose data will be divided into two, namely 80% as training data and 20% as test data. Data samples were taken using a webcam camera. 500 pictures were taken of each finger pose. The number of collected image datasets is 2500. Image resolution value is 230x300 with plain white background and varied backgrounds.

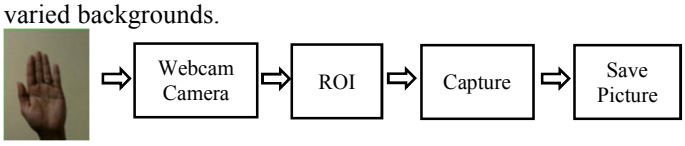
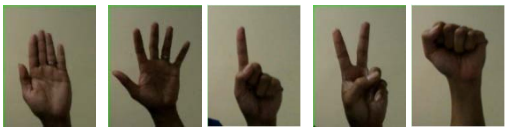


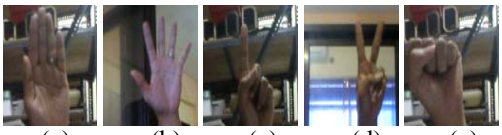
Fig. 9. Flow of the data sampling process



(a) (b) (c) (d) (e)

Fig. 10. Five finger pose white background

(a)Forward (b) Backward (c) Right (d) Left (e) Stop



(a) (b) (c) (d) (e)

Fig. 11. Five finger pose variated backgrounds

(a)Forward (b) Backward (c) Right (d) Left (e) Stop

I. Creating Model Dataset

The The data analysis method used in this research is Convolutional Neural Network (CNN) which aims to classify the five finger poses.

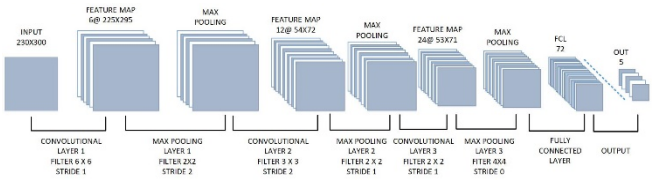


Fig. 12. Eight layers of convolutional neural network

As shown in figure 12 there are three layers convolutional in the CNN model with 8 layers. Researchers can explain as follows:

- The first convolution An image input with a resolution of 230x300x3 is entered at the first convolution with the filter size 6x6 and depth 6 and stride 2 produce output feature map amounting to 225x295x3. Feature map The first convolution is then maxpooling with a 2x2 stride 2 filter resulting in a feature map of 112x147x3 input.
- The second convolution, the result of the first convolution 112x147x3 is entered into the second convolution with a filter size of 3x3 and depth 12 and stride 2 produce output feature map amounting to 12@55x73x3. Feature map the result of the second convolution is then at maxpooling with filter 2x2 stride 1 produce a feature map of 12 @ 54x72x3.
- The third convolution, the result of the second convolution 54x72x3, is inserted into the second convolution with a filter size of 3x3 and a depth of 24 with stride 2 resulting in a feature map output of 24 @ 53x71x3. The second convoluted feature map is then maxpooling with a 4x4 stride 1 filter to produce a feature map of 24 @ 50x68x3.
- At the final stage, namely proses fully connected layer, which is to combine all the results of the convolution into one line, then a probability process is carried out for each row to be carried out the classification process into 5 classification outputs, namely forward, backward, right, left, and stop. Process learning using 100 epoch

TABLE I. NETWORKS PARAMETERS OF CNN

Layer	Jumlah node
Input Layer	230 x 300
Convolutional Layer 1	Filter: 6x6 Depth: 6 Stride: 2
Max Pooling Layer 1	Filter: 2x2 Stride 2
Convolutional layer 2	Filter: 3x3 Depth:12 Stride: 2
Max Pooling Layer 2	Filter: 2x2 Stride 1
Convolutional layer 3	Filter: 3x3 Depth:24 Stride: 1
Max Pooling Layer 3	Filter: 4x4 Stride 1
Full Connected Layer	72
Output layer	5

III. RESULTS AND DISCUSSION

A. *Training Data*

After a dataset of five types of finger poses is created, the dataset will be trained to train the data based on the CNN model algorithm that has been created. The number of training datasets is 2500 image data from five types of finger poses. The training process is carried out by entering the dataset into the model for each model created using the python program on Jupiter notebooks. The number of epochs used in this training process is 100.

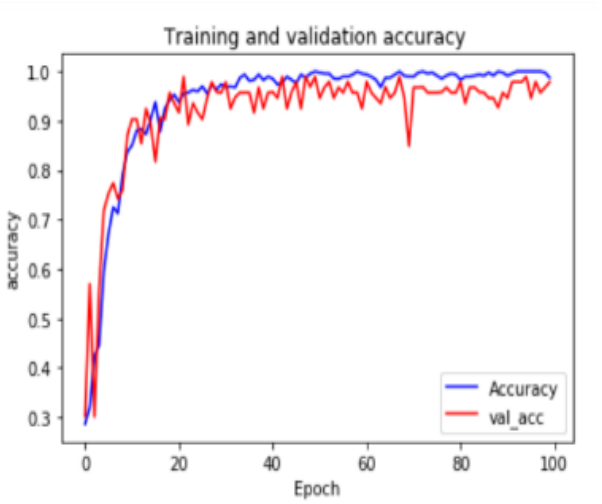


Fig. 13. Training and validation of accuracy

Figure 13 shows the results of training data with the CNN model. From these results it can be explained that at epoch 1 to 10 the accuracy value is between 28.31% to 83.09%. Whereas for epoch 11 to 100 the accuracy value shows an increase between 84.92% to 98.75%. From these results it can be seen that the average level of accuracy at the time of training data by adding an accuracy of 100 epochs, then divided by the number of epochs used, in order to obtain an average accuracy rate of 93.75%.

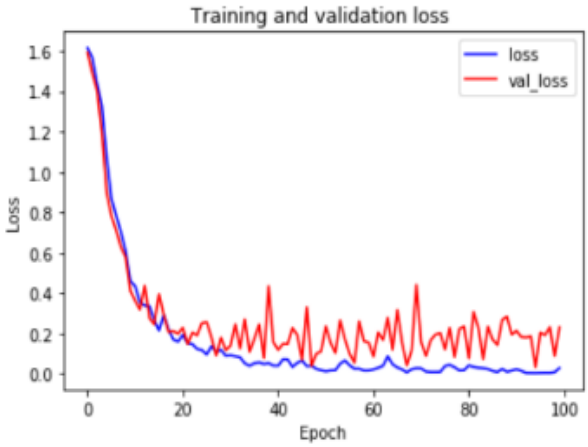


Fig. 14. Training dan validation Loss

In Figure 14, it can be seen that the resulting loss value at epoch 1 to 4 ranges from 16%-10%. Meanwhile, from epoch 5 to 100 the value of the loss is getting lower, namely between 0.8% to 0.03%. These results indicate that the smaller the resulting loss, the greater the value of the level of accuracy. This loss value is influenced by the amount of data used in the training process.

B. *Testing Data*

The next stage after carrying out the learning process is to carry out the testing process of the training model that has been created by testing the video in real time and randomly with a resolution of 240 x 215. The test was carried out by saving the captured video in the form of a frame at a rate of 5 frames per second (fps) for five seconds. Each finger pose produces 25 image frame, so that a total of 125 image frames were tested.

TABLE II. REAL TIME DATA TESTING RESULT

Types of Pose Finger	Total	Prediction		Accuracy (%)
		True	*False*	
Forward	25	25	0	100
Back off	25	25	0	100
Right	25	22	3	88
Left	25	22	3	88
Stop	25	23	2	92

Table 2 iss the result of the data testing process. Data testing is carried out based on a previously created dataset. The total in testing is 125 for each different type of finger pose. From the resultas, it is known that the correct detected forward finger pose is 25, 25 backward finger pose, 22 right hand finger pose, 22 left hand finger pose, and finger pose. correctly detected stop hand is 23. Based on these results it is known that the average accuracy rate in online testing is 93.6%.

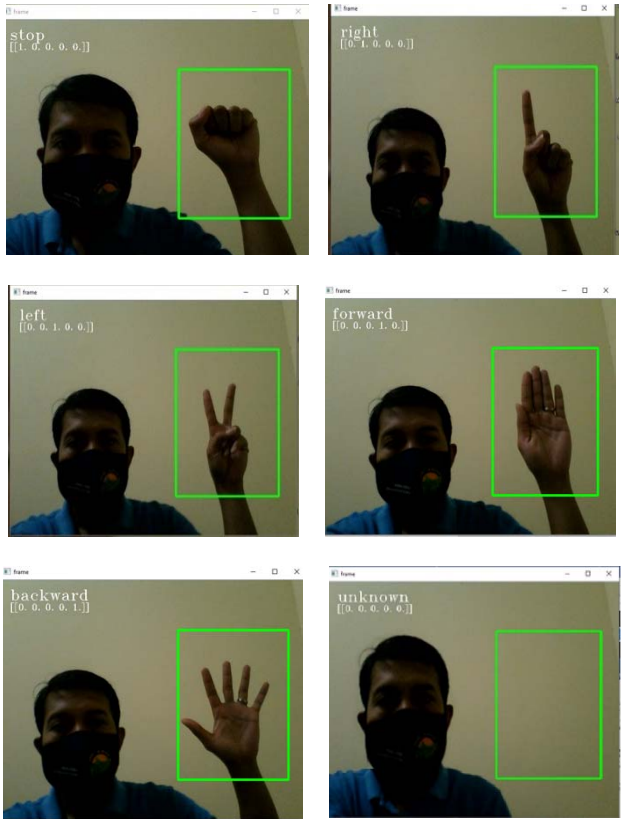


Fig. 15. The results of testing the predictions of 5 finger poses in real time

IV. CONCLUSION

This paper present a wheelchair control system based on pose finger recognition using CNN. There were five pose finger tested. The detection results was utilized to control the direction of the electrical wheelchair. The experimental results shows that the system achieved the average accuracy of 93.6%. These results are promising but it should be improved for real-time application.

ACKNOWLEDGMENT

I would like to thank the lecturers and friends, especially the electrical engineering master program at Jember University who participated in writing this article so that it was completed on time.

REFERENCES

[1] A. Cristea, "Undang Undang RI NOMOR 8 TAHUN 2016," *PENYANDANG Disabil.*, p. 102, 2016.

[2] "Penduduk Indonesia hasil SUPAS 2015." 2015.

[3] A. H. Kulkarni and S. A. Urabinahatti, "Performance comparison of three different classifiers for hci using hand gestures," *Int. J. Mod. Eng. Res.*, vol. 2, no. 4, pp. 2857–2861, 2012.

[4] T. Nadu, "Robot Navigation Control Using Hand Gestures," *Int. J. Mod. Trends Eng. Res. Predict. Thyroid Dis. USING DATAMINING*, pp. 320–327, 2015.

[5] O. K. Oyedotun and A. Khashman, "Deep learning in vision-based static hand gesture recognition," *Neural Comput. Appl.*, vol. 28, no. 12, pp. 3941–3951, 2017.

[6] D. Sahu, "Camera Based Eye Controlled Electronic Wheelchair System Using Raspberry Pi," vol. 0869, no. 3, pp. 83–87, 2016.

[7] Matthew D. Zeiler and Rob Fergus, "Visualizing and understanding convolutional networks," *Eur. Conf. Comput. Vis.*, vol. 12, pp. 818–833, 2014.

[8] B. Hu and J. Wang, "Deep Learning Based Hand Gesture Recognition and UAV Flight Controls," *Int. J. Autom. Comput.*, vol. 17, no. 1, pp. 17–29, 2020.

[9] P. S. Neethu, R. Suguna, and D. Sathish, "An efficient method for human hand gesture detection and recognition using deep learning convolutional neural networks," *Soft Comput.*, vol. 5, 2020.

[10] W. Cheng, Y. Sun, G. Li, G. Jiang, and H. Liu, "Jointly network: a network based on CNN and RBM for gesture recognition," *Neural Comput. Appl.*, vol. 31, pp. 309–323, 2019.

[11] H. Fujiyoshi, T. Hirakawa, and T. Yamashita, "Deep learning-based image recognition for autonomous driving," *IATSS Res.*, vol. 43, no. 4, pp. 244–252, 2019.

[12] S. M. Abbas and S. N. Singh, "Region-based Object Detection and Classification using Faster R-CNN," *Int. Conf. "Computational Intell. Commun. Technol. CICT 2018*, no. Cict, pp. 1–6, 2018.