Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 216 (2023) 13–20

Procedia Computer Science

www.elsevier.com/locate/procedia

7th International Conference on Computer Science and Computational Intelligence 2022

Sign language recognition system for communicating to people with disabilities

Yulius Obi[a], Kent Samuel Claudio[a], Vetri Marvel Budiman[a], Said Achmad[a,*], Aditya Kurniawan[a,*]

[a]*Computer Science Department, School of Computer Science, Bina Nusantara University, Jakarta, 11480, Indonesia*

Abstract

Sign language is one of the most reliable ways of communicating with special needs people, as it can be done anywhere. However, most people do not understand sign language. Therefore, we have devised an idea to make a desktop application that can recognize sign language and convert it to text in real time. This research uses American Sign Language (ASL) datasets and the Convolutional Neural Networks (CNN) classification system. In the classification, the hand image is first passed through a filter and after the filter is applied, the hand is passed through a classifier which predicts the class of the hand gestures. This research focuses on the accuracy of the recognition. Our Application resulted in 96,3% accuracy for the 26 letters of the alphabet.

Keywords: Computer Vision; Convolutional Neural Networks; American Sign Language (ASL); Sign Language Recognition

1. Introduction

AI or Artificial Intelligence is one of the fields of computer science that studies human intelligence to make artificial intelligence capable of solving problems. Computer vision is a subcategory of Artificial Intelligence (AI). The goal of computer vision is to extract useful information from images. However, it is challenging to implement. Computer vision has been used to manufacture robots and photo scans and is also used in the automotive, medical, mathematical,

*Corresponding author.
E-mail address: said.achmad@binus.edu; adkurniawan@binus.edu

1877-0509
10.1016/j.procs.2022.12.106

and industrial fields [1, 2].

Deaf people have problems communicating with normal people in their daily lives. One reason for that is that not many people understand American Sign Language (ASL) [3, 4]. As such, this research aims to recognize hand gestures or ASL, which the system will change into text that can be read in real-time, making communication with people with special needs easier. Hand gesture recognition is also in Human-Computer Interaction (HCI) because it interacts with the user directly. Human-Computer Interaction (HCI) is the study, planning, or design of interaction between users and computers. One of the functional interactions for a hand gesture recognition system is displaying text composed of alphabets read by the system [5, 6].

In this research, we will utilize Computer Vision and Pattern Recognition technology to create a desktop application that can detect hand movements in real-time using a webcam/live camera. Afterward, we will use American Sign Language (ASL) datasets and the Convolutional Neural Networks (CNN) classification system. This research focuseson the accuracy of recognizing letters of the alphabet and provides the results in a text in real-time.

2. Literature Review

Human-computer interaction (HCI) is generally done using a mouse, keyboard, remote control, or touch screen. However, interpersonal communication is done more naturally through voice and physical movement, which is generally considered more flexible and efficient [5]. According to Zhi-Hua Chen et al. [7], new types of HCI are needed due to the rapid development of software and hardware. In particular, speech recognition and gesture recognition have received significant attention in the field of HCI. Artificial Intelligence (AI) technology is also needed to perform gesture recognition, precisely computer vision. In computer vision technology, several things can be researched, one of which is real-time motion-based recognition. There are a variety of different ways that can be used to create recognition systems. The aim of research in this field is usually to increase the accuracy of the recognition performed and to perform gesture recognition, such as hand movements, sign language, and body movements.

In general, recognition technology can recognize many things, such as patterns, faces, body movements, or hand movements, for different purposes. In 2013, there was a study on gesture recognition in real-time, which already had a success rate of more than 68% for each gesture. This study uses an optical flow feature and is combined with a face detector [8]. However, Tarek Frikha and Abir Presentche were more interested in making hand gesture recognition [3]. To support research on hand gesture recognition, a study in 2018 introduced a dataset and a benchmark called EgoGesture [1]. EgoGesture uses Hierarchical Hidden Markov Model and Classification methods, as well as the Cambridge Hand Gesture dataset, and can produce a dataset that is very useful for research. One application of recognition technology is Automotive Human-Machine technology. In its field, hand gesture recognition controls applications on mobile tablets. This research uses ToF sensors, PCA-based preprocessing, and Convolutional Neural Network, which gives satisfactory results for the riders [9].

Although many applications are made with recognition technology, the focus of this research is on the field of communication, especially sign language. Communication is the most important thing for every human being to be able to share their thoughts and ideas. Communication can be said to be successful if the communication partner receives and understands the message [10]. However, for people who have hearing difficulties or cannot speak, communication will be difficult, so other forms of communication are needed [11], such as by writing or using body gestures. However, written communication is less practical because people with hearing problems are generally less proficient in writing spoken language. Moreover, this type of communication is impersonal and slow in face-to-face conversation. For example, agile communication with the doctor is often required when there is an accident, and written communication is not always possible [12]. Of these various forms of communication, sign language is the most effective tool of communicating [13]. Although sign language is the most effective, communication is sometimes difficult because only some understand sign language [14, 15]. Sign Language has different forms and hand gestures in each country, such as American Sign Language (ASL), which is used in a study by Shruti Chavan, Xinrui Yu, and Jafar Saniie [16] and Indonesian Sign Language (ISL) that is used in a study conducted in [4]. Therefore, we need a tool that can recognize and convert sign language into understandable text with a high level of accuracy as well as easy-to-use [17].

To be able to recognize and change the sign language into understandable text, the device must be able to re- ceive a photo or video input. The photo or video input is then processed before finally being classified. This photo orvideo processing can be done in several ways or steps, such as background blurring or removal [18, 2], edge detection[17, 2, 19], point cloud processing [17], skin-color detection [20, 21]. converting Region of Interest (ROI) to grayscaleimage

and blurring ROI with Gaussian blur, Contour-Extraction [20, 19]. Data augmentation such as re-scaling, zoom- ing, shearing, rotation, width and height-shifting also into the dataset [15, 22], HSV colorspace, Global Threshold, Adaptive Gaussian Threshold [11], KAZE feature detection, KMeans algorithm [13], as well as converting BRG to RGB and converting RGB to thresholding [10]. After doing the image processing using these various methods, the

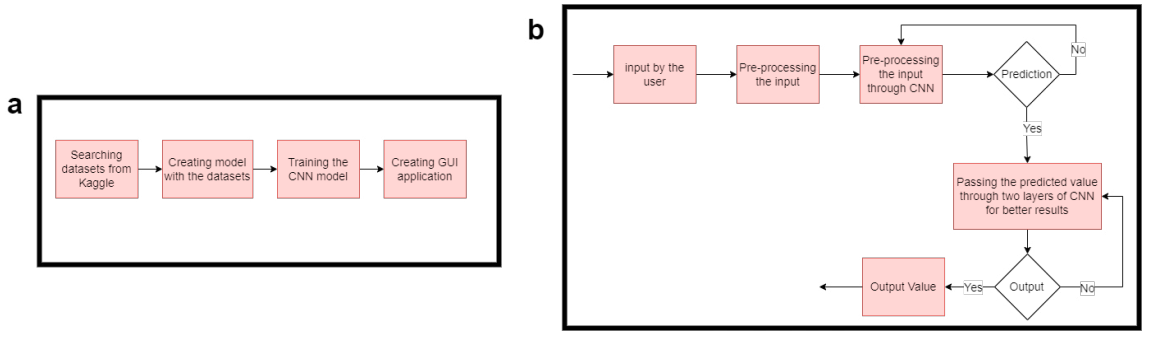

Fig. 1. (a) Research Process; (b) Gesture classification process.

results of the image processing are then classified to get the results. The classification can also be done in various ways, such as with Histogram matching algorithm [17], Nearest Neighbor [13, 21], SVM, NBC [13], Hidden Markov model [21], Extreme Learning Method (ELM) [6] and the most frequently used method is the Convolution neural network or CNN [23, 24, 10, 11, 15, 2].

Therefore, we are looking for references for suitable methods and algorithms. In 2019, there was a research that discussed the best out of these three methods, the methods being Wavelet Transforms method, Empirical Mode Decomposition method and Convolutional Neural Networks method which gave the result that the CNN method was more accurate but had relatively high memory usage [25]. Furthermore, in 2019 there was also research on Indonesian Language Recognition (ISL) using the CNN method and the YOLOv3 architecture, which also uses the ISL dataset. The research gave an accuracy of 100% for images, and 73% for videos [26]. Then in the following year, the Machine Learning method using OpenCV and TensorFlow gave positive results and was supported with a different dataset from other research [27].

The conclusion that we can convey is that to facilitate communication with people with hearing difficulties or who cannot speak, Human-computer interaction (HCI) media is needed. So, in our current times, a sign language recognition system is needed to make communication between deaf people and normal people easier. Research on sign language recognition has been carried out by researchers using different methods. The topic is very interesting, and the results positively impact society. Therefore, this research will focus on making a sign language recognition system that uses the American Sign Languages (ASL) dataset with the Convolutional Neural Network (CNN) method, which is easy to understand and has a high level of accuracy.

3. Methodology

For this research, first, we search the required datasets from Kaggle [28]. Then, we will create the model with Kaggle's dataset. After that, we will train the CNN model before finally creating the GUI for our desktop application. We have included our flowchart for our research process, as seen in Figure 1(a).

The steps required to run the language recognition model are generally divided into several sections. First, the application will enter input from the user's live camera, then the input will be read, and the photo results will be displayed as characters which will finally be assigned to a word.

3.1. Datasets

The dataset that we will use is a dataset that we took from the Kaggle website entitled "ASL Hand Sign Dataset (Grayscaled Thresholded)" [28], which contains 24 classes that have been applied a gaussian blur filter. However, because the source code requires 27 classes, we use the 3 dataset classes taken from Nikhil Gupta, namely the datasets

for classes J, Z, and 0 (blank), to fill the gaps in the datasets in Kaggle. Compared to the datasets from the code we used [29], our datasets have a higher number of images. The images that we use are 30526 images which are divided into 27 classes for training data and 8958 images which are divided into 27 classes for testing data, while the datasets from the source code that we use only have 12845 images which are divided into 27 classes for training data and 4268 images which are divided into 27 classes for testing data.

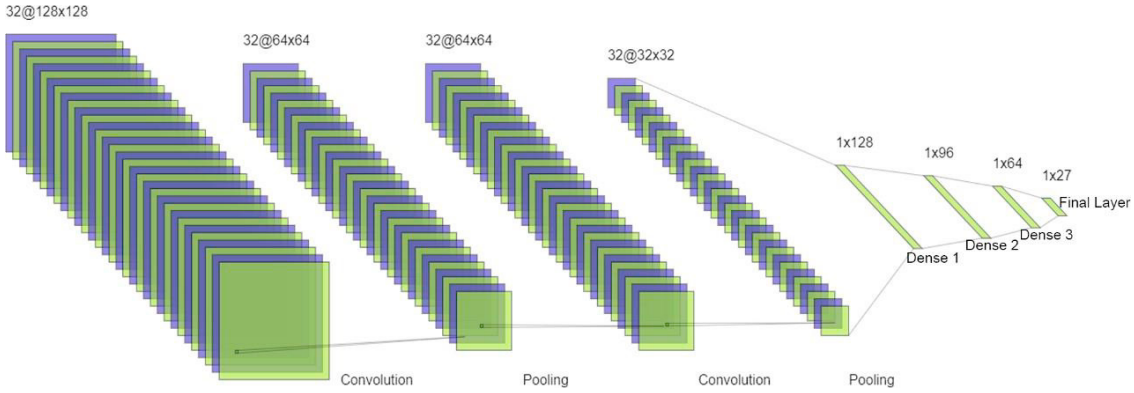

Fig. 2. Model Architecture

3.2. Implementing Algorithm

This research uses a dataset from Kaggle [28] because it has a larger database than Nikhil Gupta's. Using Kaggle's dataset, we can analyze the differences from Nikhil Gupta's research caused by having a different dataset. After that, we used that dataset for training and testing the model created by Nikhil Gupta. This model uses the Convolutional Neural Network (CNN) method for Gesture Classification, which consists of 2 layers. At Layer 1, this model performs image processing to predict and detect the number of frames of user input. Layer 1 consists of the CNN model in which there are 7 layers (1st Convolution Layer, 1st Pooling Layer, 2nd Convolution Layer, 2nd Pooling Layer, 1st Densely Connected Layer, 2nd Densely Connected Layer, and Final Layer) which can be seen in the Figure 2, Activation Function, Pooling Layer, Dropout Layer, and Optimizer. In Layer 2, there are two layers of algorithms that check and predict symbols or letters that look similar to each other so that they can detect and display an accurate letter. In addition, our application uses the Hunspell library to implement the autocorrect feature, where the user can click on one of the application's three suggestions to form a word. We have included our flowchart for our gesture classification, which can be seen in Figure 1(b). During training phases, the parameters that may have affected the model's accuracy are the amount of dataset that's being used for training and testing the model and the number of iterations that are used in the testing of the model.

After the model was modified and trained using the new datasets, we used the Sign Language To Text Converter application created by Nikhil Gupta, which can recognize and convert sign languages to letters/texts in real-time. The application used python programming language and the following libraries: TensorFlow, NumPy, OpenCV, OS-sys, operator, string, Tkinter, Hunspell, Keras, Enchant, and Pillow. The application process is as such: The hand image captured by the camera is directly converted to grayscale and then processed using gaussian blur and adaptive threshold. Then, the image will be processed by the model that has been made to make predictions and display the output character. Next, the predicted result or the letter must remain the same for a few seconds to be combined intoa word. We also modified the GUI of the application by adding an image containing a list of ASL symbols to make it easier for the user. The application's modified GUI (Graphical User Interface) can be seen in Fig 3.

3.3. Evaluate

Performance and accuracy measurements will be carried out by looking at the results of the train data and test data

using epochs 20 times. The iteration produces accuracy values and loss values for train data and test data to find the values that indicate the success and loss values. In conducting trials and evaluations of the application that has been made, we use a confusion matrix to calculate the model's performance and accuracy. The data used for the confusion matrix is a dataset created by Nikhil Gupta in his research. Specific parameters or variables may affect application performance when executing the application as such:

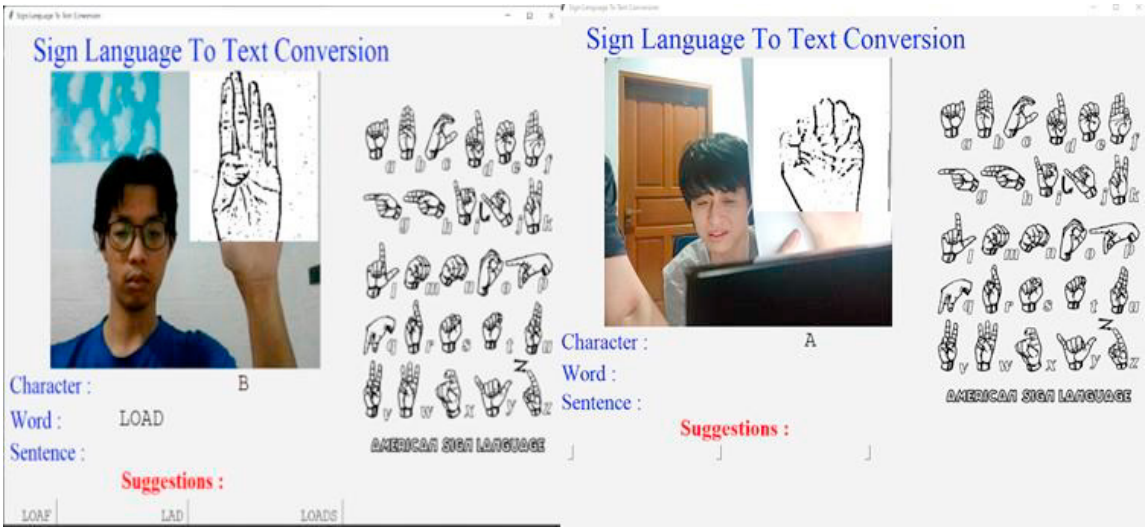

Fig. 3. Application GUI

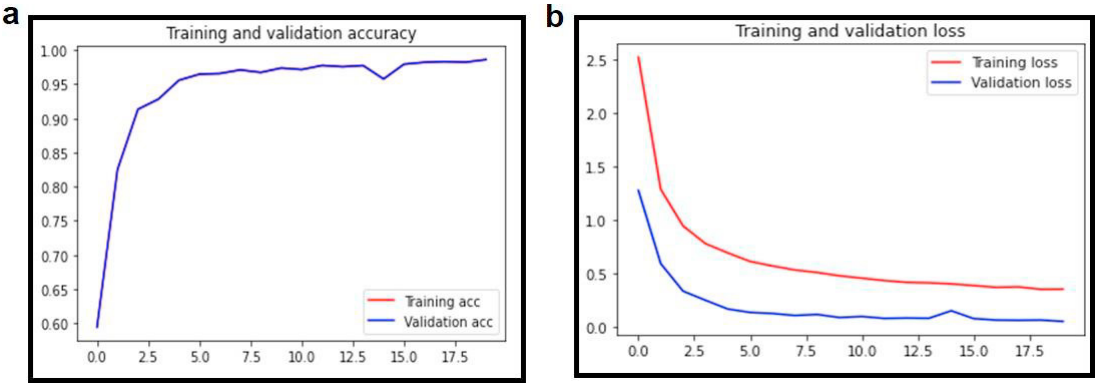

Fig. 4. (a) Train Val Accuracy; (b) Train Val Loss.

- Device 1's specification used for testing the application has an i5-9300H processor, 1650Ti VGA, 8GB RAM,1TB HDD and 256GB SSD, Windows 11 Operating System, and HD WEBCAM LAPTOP
- Device 2's specification used for testing the application has an i7-4790H Processor, NVidia GeForce GTX 970VGA, 16GB RAM, 1TB HDD and 128GB SSD, Windows 10 Operating System, and a POCOPHONE F1 smartphone camera
- Background complexity in testing the application.

4. Result and Discussion

Using a different training and testing dataset and number of epochs from the research by Nikhil Gupta, we produceda model with training and validation accuracy of 89.1% and 98.6%, respectively, as well as training and training

validation errors of 35.5% and 5.3% respectively. We have visualized the accuracy, and the loss from our training and validation model of our research algorithm in the form of graphs can be seen in Figure 4.

The results of our model compared with the one by Nikhil Gupta are very different. Namely, our model has a greater loss value in training and validation, having 35.5% loss value compared to Nikhil Gupta's 3.7% in training and 5.3%loss value compared to Nikhil Gupta's 0.18% in validation. Our model also has a lower accuracy value in training and validation than Nikhil Gupta's model, having 89.1% accuracy compared to Nikhil Gupta's 99% in training and 98.6% accuracy compared to Nikhil Gupta's 99.9% in validation. This difference in results could be influenced by thenumber of epochs performed for different models and the difference in the amount of data in the dataset. In testing andevaluating the model, we use a confusion matrix created by testing using a dataset created by Nikhil Gupta's researchbecause it has the same number of classes, the same type of data, and a smaller amount of data. The accuracy resultingfrom the confusion matrix is 96,3%. We have provided an image of the confusion matrix for our results in fig 5.

After the model has been implemented, our application recognizes and converts sign languages to letters/texts in real-time. Some variables that may have affected the application's ability to recognize the sign language are lighting and background correctly. So when running the application, we need good lighting and a plain background that does not have objects on the box that reads hand gestures. In testing the application, to test the "blank" gesture (a box to read hand gestures must be empty with nothing in it), our application requires an empty background condition. For the application to be able to read and convert the rest of the gesture to a letter/text, our hand gesture must stay in the box, and the letter shown in the "Character" row must be the same for approximately 5 seconds until that letter appeared in the "Word" row.

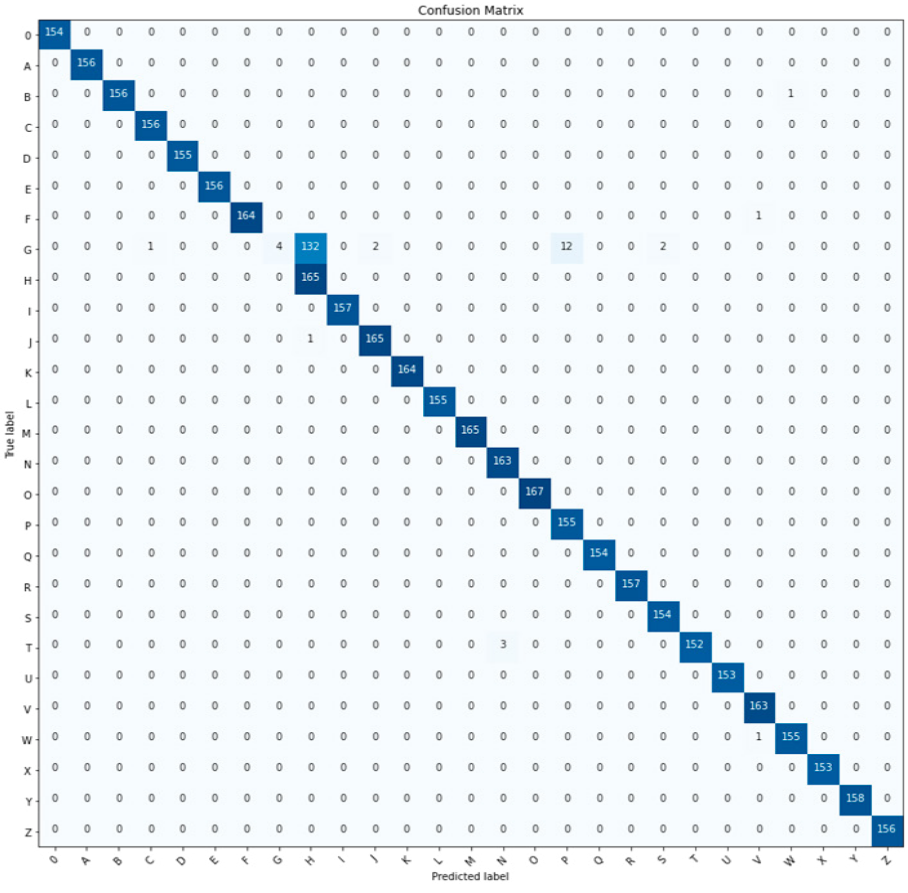

Fig. 5. Confusion matrix

From the result, the author analyzes why there are errors in the recognition process: In the confusion matrix, thereis a problem where the letter G is detected as the letter H. The cause of this problem is when we were testing our model using the dataset used by Nikhil Gupta, the gesture for the letter G in Nikhil Gupta's dataset was different from our training dataset's gesture for the letter G. Furthermore, the testing dataset's gesture for the letter G is very similar to the training dataset's gesture for the letter G, which causes the letter G to be recognized as the letter H while testing. This problem is very prominent when recognizing gestures in real time where similar gestures are present, as it may cause the application to misinterpret some gestures. Other than that, misinterpretation errors can also be caused by a low-resolution camera, insufficient lighting, or the background being too complex.

5. Conclusion

In this study, the steps we took were finding a dataset from the Kaggle website, creating a new model using the two layers Convolutional Neural Network (CNN) method using the dataset we got, training our CNN model, and creating a GUI for our application. As a result, our application can read hand gestures in real-time and combine them into a word with a final accuracy rate of 96,3%. However, forming letters into words requires the letters to remain the same for a few seconds. Therefore, for future research, we suggest implementing background removal methods, searching for a way to speed up the process of forming letters into a word so the wait time will be shortened, and multiplying the layers in the CNN model so the accuracy of the model will be greater. One might also reconsider using methods other than CNN, as it might lead to better results.

References

[1] Zhang Y, Cao C, Cheng J, Lu H. EgoGesture: A new dataset and benchmark for egocentric hand gesture recognition. IEEE Transactions on Multimedia. 2018;20(5):1038-50.

[2] Juneja S, Juneja A, Dhiman G, Jain S, Dhankhar A, Kautish S. computer Vision-Enabled character recognition of hand Gestures for patients with hearing and speaking disability. Mobile Information Systems. 2021;2021.

[3] Zeineb A, Chaala C, Frikha T, Hadriche A. Hand gesture recognition system. International Journal of Computer Science and Information Security (IJCSIS). 2016;14(6).

[4] ZULKARNAIN IYW. INDONESIAN SIGN LANGUAGE CONVERTER INTO TEXT AND VOICE AS SOCIAL INTERACTION TOOL FOR INCLUSION STUDENT IN VOCATIONAL HIGH SCHOOLS.

[5] Zhang Q, Xiao S, Yu Z, Zheng H, Wang P. Hand gesture recognition algorithm combining hand-type adaptive algorithm and effective-area ratio for efficient edge computing. Journal of Electronic Imaging. 2021;30(6):063026.

[6] Miao A, Liu F. Application of human motion recognition technology in extreme learning machine. International Journal of Advanced Robotic Systems. 2021;18(1):1729881420983219.

[7] Chen Zh, Kim JT, Liang J, Zhang J, Yuan YB. Real-time hand gesture recognition using finger segmentation. The Scientific World Journal. 2014;2014.

[8] Bayazit M, Couture-Beil A, Mori G. Real-time Motion-based Gesture Recognition Using the GPU. In: MVA. Citeseer; 2009. p. 9-12.

[9] Zengeler N, Kopinski T, Handmann U. Hand gesture recognition in automotive human–machine interaction using depth cameras. Sensors. 2018;19(1):59.

[10] Chava Sri Varshini MCGSSKS Gurram Hruday. Sign Language Recognition. INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH TECHNOLOGY (IJERT). 2020;09(05).

[11] Perdana IPI, Putra IKGD, Dharmaadi IPA. Classification of Sign Language Numbers Using the CNN Method. JITTER: Jurnal Ilmiah Teknologi dan Komputer. 2021;2(3):485-93.

[12] Pigou L, Dieleman S, Kindermans PJ, Schrauwen B. Sign language recognition using convolutional neural networks. In: European conferenceon computer vision. Springer; 2014. p. 572-8.

[13] Nivedita S, Ramyapriya Y, Tanmaya H, et al. Sign Language Recognition System using Machine Learning. 2022.

[14] Kadhim RA, Khamees M. A real-time american sign language recognition system using convolutional neural network for real datasets. Tem Journal. 2020;9(3):937.

[15] Al-Obodi AH, Al-Hanine AM, Al-Harbi KN, Al-Dawas MS, Al-Shargabi AA. A Saudi Sign Language Recognition System based on Convolutional Neural Networks. Department of Information Technology, College of Computer, Qassim University, Buraydah, Saudi Arabia.

[16] Chavan S, Yu X, Saniie J. Convolutional Neural Network Hand Gesture Recognition for American Sign Language. In: 2021 IEEE International Conference on Electro Information Technology (EIT). IEEE; 2021. p. 188-92.

[17] Rajan* GS, Nagarajan R, Sahoo AK, Sethupathi MG. Interpretation and Translation of American Sign Language for Hearing Impaired Individuals using Image Processing. International Journal of Recent Technology and Engineering (IJRTE). 2019 Nov;8(4):415–420. Available from: http://dx.doi.org/10.35940/ijrte.D6966.118419.

[18] Zhang D, Lee DJ, Chang YP. A New Profile Shape Matching Stereovision Algorithm for Real-time Human Pose and Hand Gesture Recognition. International Journal of Advanced Robotic Systems. 2014;11(2):16.

[19] AlSaedi AKH, AlAsadi AHH. A new hand gestures recognition system. Indonesian Journal of Electrical Engineering and Computer Science. 2020;18(1):49-55.

[20] Huang H, Chong Y, Nie C, Pan S. Hand gesture recognition with skin detection and deep learning method. In: Journal of Physics: Conference Series. vol. 1213. IOP Publishing; 2019. p. 022001.

[21] Kapuscinski T, Oszust M, Wysocki M, Warchol D. Recognition of hand gestures observed by depth cameras. International Journal of Advanced Robotic Systems. 2015;12(4):36.

[22] Islam MZ, Hossain MS, ul Islam R, Andersson K. Static hand gesture recognition using convolutional neural network with data augmentation. In: 2019 Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR). IEEE; 2019. p. 324-9.

[23] Vazquez Lopez I. Hand Gesture Recognition for Sign Language Transcription. 2017.

[24] Cooper H, Holt B, Bowden R. Sign language recognition. In: Visual analysis of humans. Springer; 2011. p. 539-62.

[25] Alnaim N, Abbod M. Mini gesture detection using neural networks algorithms. In: Eleventh International Conference on Machine Vision (ICMV 2018). vol. 11041. SPIE; 2019. p. 547-55.

[26] Daniels S, Suciati N, Fathichah C. Indonesian sign language recognition using yolo method. In: IOP Conference Series: Materials Science and Engineering. vol. 1077. IOP Publishing; 2021. p. 012029.

[27] Abhishek B, Krishi K, Meghana M, Daaniyaal M, Anupama H. Hand gesture recognition using machine learning algorithms. Computer Science and Information Technologies. 2020;1(3):116-20.

[28] Akdeniz F. ASL Handsign Dataset (Grayscaled Thresholded); 2022. Available from: https://www.kaggle.com/datasets/furkanakdeniz/asl-handsign-dataset-grayscaled-thresholded?resource=download-directory&select=asl-dataset.

[29] Gupta N. Sign Language To Text Conversion; 2021. Available from: https://github.com/emnikhil/Sign-Language-To-Text-Conversion.