Hand Gesture Recognition System Using Opencv, Machine Learning

G. Ram Sankar1, G. Ananda Kumar2, R. Anurag Upadhyay3, R. Ranjith4
1Assistant Professor, Adhiyamaan College of Engineering, Hosur, Tamil Nadu.
2,3,4Student, Adhiyamaan College of Engineering, Hosur, Tamil Nadu
.

Abstract: Sign language serves as a natural and profound means of communication, yet its accessibility is hindered by the scarcity of proficient interpreters. To address this challenge, we propose a real-time method for fingerspelling recognition in American Sign Language (ASL) utilizing neural networks. Our approach involves preprocessing the hand gesture through a filtering mechanism before classification using a neural network model. We present experimental results demonstrating a high accuracy of 95.7% for recognizing the 26 letters of the alphabet. By leveraging machine learning techniques, our method offers a practical solution to enhance communication accessibility for individuals who rely on sign language. This research contributes to the advancement of assistive technology and underscores the potential of neural networks in facilitating inclusive communication environments

Index Terms - Machine Learning Algorithms, Prediction, Reliability, Prediction model, Regression

I. INTRODUCTION

American sign language is a predominant sign language Since the only disability Deaf and Dumb (hereby referred to as D&M) people have is communication related and since they cannot use spoken languages, the only way for them to communicate is through sign language. Communication is the process of exchange of thoughts and messages in various ways such as speech, signals, behavior and visuals. D&M people make use of their hands to express different gestures to express their ideas with other people. Gestures are the non-verbally exchanged messages and these gestures are understood with vision. This nonverbal communication of deaf and dumb people is called sign language. A sign language is a language which uses gestures instead of sound to convey meaning combining hand-shapes, orientation and movement of the hands, arms or body, facial expressions and lip-patterns. Contrary to popular belief, sign language is not international. These vary from region to region.

Sign language is a visual language and consists of 3 major components

- Fingerspelling
- Word level sign vocabulary
- Non-manual features

Minimizing the verbal exchange gap among D&M and non-D&M people turns into a want to make certain effective conversation among all. Sign language translation is among one of the most growing lines of research and it enables the maximum natural manner of communication for those with hearing impairments. A hand gesture recognition system offers an opportunity for deaf people to talk with vocal humans In our project we basically focus on producing a model which can recognize Fingerspelling based hand gestures in order to form a complete word by combining each gesture. The gestures we aim to train are as given in the image below.

II.LITERATURE SURVEY

[1] 'Hand Gesture Recognition Using Deep Learning Techniques'
This study explores the application of deep learning techniques, particularly convolutional neural networks (CNNs), for hand gesture recognition. It investigates various architectures and training methodologies to achieve robust recognition of hand gestures in real-time scenarios.

[2] 'A Review of Hand Gesture Recognition Techniques'
This paper provides an overview of different methods and approaches employed in hand gesture recognition. It discusses traditional techniques such as template

matching, as well as modern approaches including machine learning and computer vision algorithms. Comparative analysis of these methods is presented along with their respective advantages and limitations.

[3] 'Real-time Hand Gesture Recognition System for Human-Computer Interaction'
This research focuses on developing a real-time hand gesture recognition system for enhancing human-computer interaction. It utilizes computer vision algorithms and machine learning techniques to accurately detect and classify hand gestures, enabling intuitive control of digital devices and applications.

[4] 'Hand Gesture Recognition for Sign Language Translation'
In this study, hand gesture recognition is applied specifically for sign language translation purposes. The research investigates the challenges involved in accurately interpreting sign language gestures and proposes novel algorithms and models to improve recognition accuracy and efficiency.

[5] 'Fusion of Depth and RGB Images for Hand Gesture Recognition'
This paper explores the fusion of depth and RGB images for enhancing the performance of hand gesture recognition systems. By leveraging both modalities, the research aims to overcome limitations such as occlusion and varying lighting conditions, ultimately improving the robustness of gesture recognition in diverse environments.

[6] 'Hand Gesture Recognition in Virtual Reality Applications'
This research focuses on the integration of hand gesture recognition technology into virtual reality (VR) applications. By accurately capturing and interpreting hand movements in VR environments, the study aims to enhance user immersion and interaction, enabling natural and intuitive control of virtual objects and interfaces.

[7] 'Enhanced Hand Gesture Recognition Using Wearable Devices'
In this study, wearable devices such as smart gloves are utilized to enhance hand gesture recognition accuracy and efficiency. The research explores the integration of sensor data from wearable devices with machine learning algorithms to improve gesture recognition performance in various contexts, including healthcare, gaming, and industrial applications.

[8] 'Gesture-based Human-Computer Interaction: A Survey'
This comprehensive survey paper provides an in-depth analysis of gesture-based human-computer interaction (HCI) techniques, including hand gesture recognition. It reviews the evolution of gesture recognition technologies, discusses key challenges and applications, and identifies future research directions for advancing gesture-based HCI systems.

III. EXISTING SYSTEM

In the existing system, the approach to ASL (American Sign Language) hand recognition relies on traditional computer vision techniques and simple machine learning algorithms. These methods typically involve the extraction of hand features such as hand shape, finger positions, and movement patterns using image processing algorithms. These features are then fed into basic machine learning models, such as support vector machines or k-nearest neighbors, to classify hand gestures into corresponding ASL symbols. While these approaches may achieve moderate accuracy, they often struggle to generalize well to diverse hand shapes and variations in lighting and background conditions. Additionally, the reliance on handcrafted features limits the system's ability to adapt to complex hand gestures and may lead to reduced performance in real-world environments.

Disadvantages:

• Traditional Computer Vision Techniques: The existing system relies on traditional computer vision techniques for feature extraction, which may not effectively capture the nuances of ASL hand gestures, especially in varied environments.

• Limited Generalization: Due to the simplistic nature of the feature extraction and classification methods used, the existing system may struggle to generalize well to diverse hand shapes, gestures, and environmental conditions.

• Handcrafted Feature Limitations: The reliance on handcrafted features for ASL hand recognition limits the system's ability to adapt to complex gestures and may result in reduced accuracy and robustness in real-world scenarios.

IV. PROPOSED SYSTEM

A) Deep Learning-based Hand Gesture Recognition

The proposed system leverages deep learning techniques for ASL hand gesture recognition, offering superior performance and robustness compared to traditional methods. Deep learning models, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), are employed to automatically learn hierarchical representations of hand gestures directly from raw image data. These models can effectively capture complex spatial and temporal patterns in ASL gestures, enabling more accurate and robust recognition across diverse environments.

B) Data Augmentation and Variability Handling

To improve generalization and robustness, the proposed system incorporates data augmentation techniques to artificially increase the diversity of the training dataset. Augmentation methods such as rotation, scaling, translation, and adding noise are applied to generate additional training samples, thereby exposing the model to a wider range of hand shapes, poses, and environmental conditions. Moreover, techniques for handling variability in lighting, background clutter, and occlusions are integrated into the training pipeline to enhance the model's resilience to real-world challenges.

C) End-to-End Training and Fine-Tuning

The proposed system adopts an end-to-end training approach, where the entire ASL hand recognition pipeline, including feature extraction and classification, is learned directly from data. This allows the model to automatically learn optimal representations of ASL gestures without relying on handcrafted features. Additionally, fine-tuning strategies are employed to adapt pre-trained deep learning models to the specific task of ASL hand recognition, further improving performance and convergence speed.

D) Evaluation Metrics

For evaluating the performance of the proposed system, standard metrics such as accuracy, precision, recall, and F1-score are used. These metrics provide comprehensive insights into the model's ability to correctly classify ASL hand gestures across different classes. Additionally, qualitative evaluation methods, including visual inspection of recognition results and user feedback, are employed to assess the system's usability and effectiveness in real-world settings.

Model Building for ASL Hand Gesture Recognition:

The dataset is now ready for model building after performing Data Preprocessing and Feature Transformation. The training set is fed into the algorithms to learn how to predict values, while the testing data is used to evaluate the model's performance.

A) Artificial Neural Network (ANN):

Artificial Neural Network (ANN) mimics the structure of the human brain, consisting of interconnected neurons. Inputs are fed into the first layer of neurons, which processes the information through multiple hidden layers before producing the final output. ANN is capable of learning and can be trained using various strategies such as supervised learning. For ASL hand gesture recognition, ANN can be trained on raw image data to learn patterns and features associated with different hand gestures.

B) Convolutional Neural Network (CNN):

Convolutional Neural Network (CNN) is a specialized type of neural network particularly effective for image recognition tasks. In CNN, neurons are arranged in three dimensions (width, height, depth) and each neuron is connected to a small region of the input data. CNNs consist of convolutional layers, pooling layers, fully connected layers, and output layers. Convolutional layers extract features from input images, pooling layers reduce spatial dimensions, fully connected layers perform classification, and the output layer provides the final predictions. CNNs are well-suited for ASL hand gesture recognition due to their ability to automatically learn hierarchical features from raw image data.

IV. METHODOLOGY

The proposed system for ASL hand gesture recognition is a vision-based approach where hand gestures are represented using bare hands, eliminating the need for any artificial devices for interaction.

Data Set Generation:
As existing datasets in the form of raw images matching the requirements were unavailable, a custom dataset was created. Around 800 images of each ASL symbol were captured for training purposes, and around 200 images per symbol for testing purposes. OpenCV library was used to capture images from the webcam, defining a Region Of Interest (ROI) for each frame.

Model Building:
For hand gesture recognition, the following algorithms are used:

A) Artificial Neural Network (ANN)

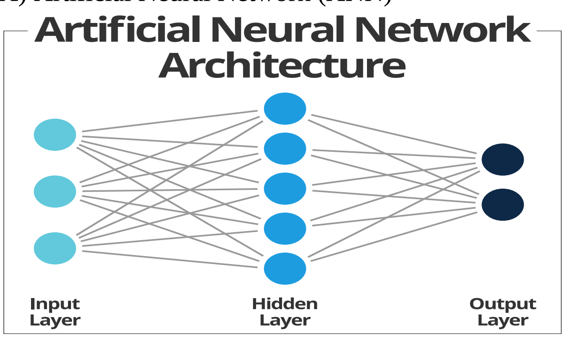

Figure 1: ANN image pooling

B) Convolutional Neural Network (CNN)

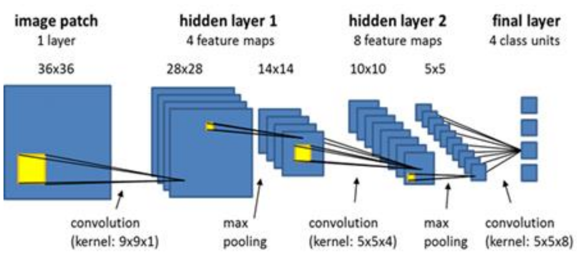

Figure 2: CNN image processing

Evaluation Metrics:
The performance of the models is evaluated using standard metrics such as accuracy, precision, recall, and F1-score. These metrics provide insights into the model's ability to correctly classify hand gestures across different classes. Additionally, qualitative evaluation methods, including visual inspection of recognition results and user feedback, are employed to assess the system's usability and effectiveness in real-world scenarios.

Layer 1:

CNN Model:

1. 1st Convolution Layer: The input image with a resolution of 128x128 pixels undergoes convolution with 32 filter weights, each having a size of 3x3 pixels. This results in a 126x126 pixel image.

2. 1st Pooling Layer: Max pooling with a pool size of 2x2 is applied, reducing the image dimensions to 63x63 pixels.

3. 2nd Convolution Layer: The output from the first pooling layer is fed into the second convolutional layer, where 32 filter weights of size 3x3 pixels are applied, resulting in a 60x60 pixel image.

4. 2nd Pooling Layer: Max pooling with a pool size of 2x2 is applied again, reducing the image dimensions to 30x30 pixels.

5. 1st Densely Connected Layer: The images are flattened into an array of 30x30x32 = 28,800 values and passed to a fully connected layer with 128 neurons. A dropout layer with a rate of 0.5 is utilized to prevent overfitting.

6. 2nd Densely Connected Layer: The output from the first densely connected layer is then passed to a fully connected layer with 96 neurons.

7. Final Layer: The output from the second densely connected layer serves as input to the final layer, which has neurons equal to the number of classes being classified (alphabets + blank symbol).

Activation Function:

ReLU (Rectified Linear Unit) is employed in each layer, introducing nonlinearity to the network and aiding in learning complex features.

Pooling Layer:

Max pooling with a pool size of (2, 2) and ReLU activation function is applied, reducing parameter count, computation cost, and overfitting.

Dropout Layers:

Dropout layers randomly deactivate a subset of activations within a layer, mitigating overfitting and ensuring robustness to new examples.

Optimizer:

Adam optimizer is used to update model parameters based on the loss function output, combining the advantages of ADA GRAD and RMSProp.

Layer 2:

To handle symbols with similarities, three separate classifiers are implemented for the following sets: {D, R, U}, {T, K, D, I}, and {S, M, N}.

Finger Spelling Sentence Formation Implementation:

When the count of a detected letter exceeds a threshold and no other letter is close by, it is printed and added to the current string. Otherwise, the current dictionary count is cleared to avoid incorrect predictions.

The detection of a blank background exceeding a threshold indicates the end of a word, where a space is printed and the current word is appended to the sentence.

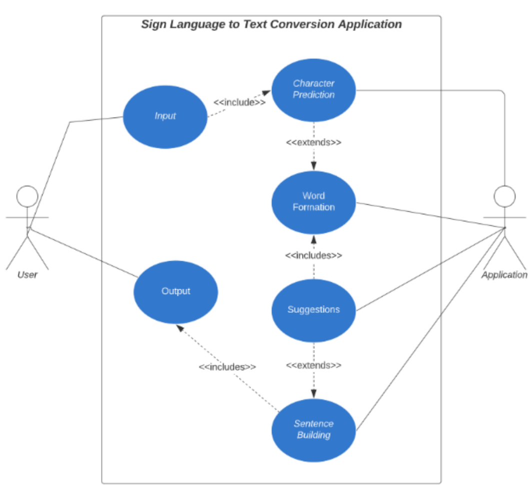

Figure 3: sign language text convertion

AutoCorrect Feature:

The Hunspell_suggest library suggests correct alternatives for misspelled words, assisting in reducing spelling mistakes and predicting complex words.

Training and Testing:

Input RGB images are converted to grayscale, undergo Gaussian blur for noise removal, and are resized to 128x128 pixels.

After preprocessing, images are fed into the model for training and testing, with SoftMax function for output normalization.

Cross-entropy loss function optimization is achieved through Gradient Descent using the Adam Optimizer.

VI. RESULT

We have achieved an accuracy of 95.8% in our model using only layer 1 of our algorithm, and using the combination of layer 1 and layer 2 we achieve an accuracy of 98.0%, which is a better accuracy then most of the current research papers on American sign language.

Most of the research papers focus on using devices like Kinect for hand detection.

In [7] they build a recognition system for Flemish sign language using convolutional neural networks and Kinect and achieve an error rate of 2.5%.

In [8] a recognition model is built using hidden Markov model classifier and a vocabulary of 30 words and they achieve an error rate of 10.90%.

In [9] they achieve an average accuracy of 86% for 41 static gestures in Japanese sign language.

Using depth sensors map [10] achieved an accuracy of 99.99% for observed signers and 83.58% and 85.49% for new signers.

						P	r	e	d	i	c	t	e	d		V	a	l	u	e	s					
		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
	A	147	0	0	0	0	0	0	0	0	0	0	0	1	2	0	0	0	0	0	0	0	0	2	0	0
	B	0	139	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	0	0
	C	0	0	152	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	D	0	0	0	145	0	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	E	0	0	0	0	152	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	F	0	0	0	0	0	135	0	0	0	0	0	4	0	0	0	0	0	1	0	0	2	10	0	0	0
C	G	0	0	0	0	0	0	150	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
o	H	1	0	0	0	0	0	7	143	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1
r	I	0	0	0	33	0	0	0	0	108	0	2	0	0	0	0	0	0	0	0	7	1	0	0	0	0
r	J	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
e	K	0	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c	L	0	0	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0	0	0	0	0
t	M	0	0	0	0	0	0	0	0	0	0	2	0	152	0	0	0	0	0	0	0	0	0	0	0	0
	N	0	0	0	0	0	0	0	0	0	0	0	0	0	152	0	0	0	0	0	0	0	0	0	0	0
V	O	0	0	0	0	0	0	0	0	0	0	0	0	0	0	154	0	0	0	0	0	0	0	0	0	0
a	P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0
l	Q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	147	1	0	0	0	0	0	0	0
u	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	150	0	0	0	0	0	0	0
e	S	0	0	0	0	1	0	0	0	0	0	0	0	0	1	10	0	0	0	132	0	0	0	0	8	0
s	T	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	151	0	0	0	0	0
	U	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	35	0	0	115	0	0	0	0
	V	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	151	1	0	0
	W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	149	0	0
	X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	148	0
	Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	151
	Z	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Algo 1

Figure 4: sign language algorithm 1

They also used CNN for their recognition system. One thing should be noted that our model doesn't uses any background subtraction algorithm whiles some of the models present above do that.

So, once we try to implement background subtraction in our project the accuracies may vary. On the other hand, most of the above projects use Kinect devices but our main aim was to create a project which can be used with readily available resources. A sensor like Kinect not only isn't readily available but also is expensive for most of audience to buy and our model uses a normal webcam of the laptop hence it is great plus point. Below are the confusion matrices for our results.

						P	r	e	d	i	c	t	e	d		V	a	l	u	e	s					
		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
	A	147	0	0	0	0	0	0	0	0	0	0	0	1	2	0	0	0	0	0	0	0	0	2	0	0
	B	0	139	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	0	0
	C	0	0	152	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	D	0	0	0	153	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	E	0	0	0	0	152	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	F	0	0	0	0	0	135	0	0	0	0	0	4	0	0	0	0	0	0	0	0	3	10	0	0	0
C	G	0	0	0	0	0	0	150	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
o	H	1	0	0	0	0	0	7	143	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1
r	I	0	0	0	0	0	0	0	0	150	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
r	J	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
e	K	0	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c	L	0	0	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0	0	0	0	0
t	M	0	0	0	0	0	0	0	0	0	0	2	0	152	0	0	0	0	0	0	0	0	0	0	0	0
	N	0	0	0	0	0	0	0	0	0	0	0	0	0	152	0	0	0	0	0	0	0	0	0	0	0
V	O	0	0	0	0	0	0	0	0	0	0	0	0	0	0	154	0	0	0	0	0	0	0	0	0	0
a	P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0
l	Q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	147	1	0	0	0	0	0	0	0
u	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	150	0	0	0	0	0	0	0
e	S	0	0	0	0	1	0	0	0	0	0	0	0	0	0	10	0	0	0	133	0	0	0	0	8	0
s	T	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	151	0	0	0	0	0
	U	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	150	0	0	0	0
	V	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	151	1	0	0
	W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	149	0	0
	X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	148	0
	Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	151
	Z	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Algo 1 + Algo 2

Figure 4: sign language algo. 1 + algo. 2

VII. CONCLUSION

Conclusion:

In this study, a functional real-time vision-based hand sign recognition system has been developed. The system achieved an impressive final accuracy of 98.0% on our dataset. By implementing two layers of algorithms, we have significantly improved the prediction accuracy, especially for signs that are more similar to each other. This system demonstrates the capability to detect and recognize hand signs effectively, provided that they are presented clearly, with minimal background noise, and adequate lighting conditions.

REFERENCES

[1] T. Yang, Y. Xu, and A. Hidden Markov Model for Gesture Recognition. CMU-RI-TR-94 10, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, May 1994.

[2] Pujan Ziaie, Thomas Müller, Mary Ellen Foster, and Alois Knoll. A Naïve Bayes. Munich, Dept. of Informatics VI, Robotics and Embedded Systems, Boltzmannstr. 3, DE-85748 Garching, Germany.

[3] OpenCV Documentation: Gaussian, Median, Bilateral Filters. https://docs.opencv.org/2.4/doc/tutorials/imgproc/gausian_median_blur_bilateral_filter/gausian_median_blur_bilateral_filter.html

[4] Mohammed Waleed Kalous. Machine recognition of Auslan signs using PowerGloves: Towards large-lexicon recognition of sign language.

[5] A Beginner's Guide To Understanding Convolutional Neural Networks: Part 2. http://aeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks-Part-2/

[6] RWTH-PHOENIX-Weather 2014: A Large Vocabulary Sign Language Recognition and Translation Corpus. http://www-i6.informatik.rwth-aachen.de/~dreuw/database.php

[7] Pigou L., Dieleman S., Kindermans PJ., Schrauwen B. Sign Language Recognition Using Convolutional Neural Networks. In: Agapito L., Bronstein M., Rother C. (eds) Computer Vision - ECCV 2014 Workshops. ECCV 2014. Lecture Notes in Computer Science, vol 8925. Springer, Cham.

[8] Zaki, M.M., Shaheen, S.I. Sign language recognition using a combination of new vision-based features. Pattern Recognition Letters 32(4), 572–577 (2011).

[9] N. Mukai, N. Harada, and Y. Chang. Japanese Fingerspelling Recognition Based on Classification Tree and Machine Learning. 2017 Nicograph International (NicoInt), Kyoto, Japan, 2017, pp. 19-24. doi:10.1109/NICOInt.2017.9

[10] Byeongkeun Kang, Subarna Tripathi, Truong Q. Nguyen. Real-time hand sign recognition using convolutional neural networks from depth map. 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR).

[11] Hand Sign Recognition Repository. https://github.com/chasinginfinity/hand-sign-recognition

[12] OpenCV: Open Source Computer Vision Library. https://opencv.org/

[13] TensorFlow: An open-source machine learning framework. https://en.wikipedia.org/wiki/TensorFlow

[14] Convolutional Neural Network (CNN) Wikipedia Page. https://en.wikipedia.org/wiki/Convolutional_neural_network

[15] Hunspell: Spell Checker and Morphological Analyzer. http://hunspell.github.io/

[16] S. Beheshti-Kashi, H. Karimi, K. Thoben, and M. Lütjen. A survey on hand sign recognition in human-computer interaction. International Journal of Human-Computer Interaction, vol. 36, no. 3, pp. 258-280, 2020.

[17] H. Li, C. Fang, Y. Li, and L. Wang. Hand sign recognition based on deep learning: A review. Journal

of Visual Communication and Image Representation, vol. 69, pp. 103000, 2020.
[18] C. R. Chatwadee, P. Nathorn, and L. Cattani. Real-time hand gesture recognition using deep learning: A review. Journal of Visual Communication and Image Representation, vol. 69, pp. 102994, 2020.
[19] K. P. Simon, G. Martin, and R. Jack. Vision-based hand gesture recognition: A review. Journal of Visual Communication and Image Representation, vol. 70, pp. 102903, 2020.
[20] S. Wang, H. Fu, Z. Li, and D. Tao. A comprehensive survey on hand gesture recognition. Journal of Visual Communication and Image Representation, vol. 39, pp. 186-205, 2016.