2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N) | 978-1-6654-7436-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICAC3N56670.2022.10074007

A Smart System for Sign Language Recognition using Machine Learning Models

Dr. B. Rajalingam, Associate Professor-CSE, St. Martins Engineering College, Secunderabad, rajalingam35@gmail.com

Dr. R. Santhosh kumar, Associate Professor - AIDS, St. Martins Engineering College, Secunderabad, santhoshkumar.aucse@gmail.com

Dr. P. Deepan, Assistant Professor-AIML, St. Martins Engineering College, Secunderabad, deepanp87@gmail.com

Dr. P. Santosh Kumar Patra, Professor - CSE, St. Martins Engineering College, Secunderabad, principal@smec.ac.in

Dr. S.Bavankumar, Assistant Professor-CSE, St. Martins Engineering College, Secunderabad, sbavankumar55gmail.com

Abstract - **When people exchange information, thoughts, or sentiments via the medium of communication, it is called communication. Both parties must be conversant in the same language in order for communication to occur. On the other hand, those who are deaf or dumb must use various communication methods. If you can't hear, you have deafness; if you can't speak, you have dumbness. As a group, they use sign language to communicate, but most people don't see it as a valuable skill. To communicate with someone who is deaf or mute is difficult since not everyone is familiar with or understands sign language. You may develop a model using machine learning to get around this roadblock. It is possible to teach a model to recognize different gestures used in sign language and to translate those gestures into English. Many people will get the ability to converse with the deaf and dumb as a result of this. By using a camera to capture Sign Language data, we may then use machine learning techniques like Convolutional Neural Networks to the datasets. Some other well-known models have been compared to a new strategy that has been presented for the same problem. Principal Component Analysis (PCA), Histogram of Gradients (HOG), and Local Binary Patterns (LBP) are some of the pre-processing approaches employed. ORB, Canny edge detection, and the state of word approach are all used to create the new model. This pre-processed data is then sent through a variety of classifiers in order to provide useful findings (including Random Forests and Support Vector Machines, as well as Nave Bayes and Logistic Regression). The new models are substantially more accurate than the previous ones. The method achieves good accuracy even with a little dataset.**

Keywords: Classification, Support vector Machine, Principal Component Analysis, ORB, CNN

1 INTRODUCTION

The ability to express oneself via human conversation is crucial. We communicate in a variety of ways, including speaking, body language, gestures, reading, visual aids and writing. The minority of persons who are deaf or hard of hearing nevertheless face a relationship gap. An interpreter are used to communicate with them. In an emergency, however, these methods would be impractical due to their time and expense. Sign Language depends mostly on physical communication to convey meaning. In order to convey the speaker's thoughts, it is necessary to use a variety of hand shapes, orientations, and movements at the same time.

There are two distinct styles of sign language: character-by-character spelling with finger gestures. Because it enables you to convey names, addresses, and other things that don't have a word-level meaning in sign language, finger spelling is an essential ability in the language. Despite this, many people do not utilize finger spelling since it is difficult to learn and put into practise effectively. As a result, there is no universal sign language, and only a small number of people can communicate effectively using it.

Finger spelling classification in sign language may be used to solve this puzzle. Several machine learning techniques are used in this work to record and compare accuracies, and the results are presented here.

1.1 Classification Algorithms

Support Vector Machine (SVM)

In the support vector machine (SVM), each data item in an n-dimensional space is represented by the coordinate value of one of its features (n is the number of features). Determining a hyper-plane that most effectively separates the classes is how the classification is carried out.

ISBN: 978-1-6654-7436-8/22/$31.00 ©2022 IEEE

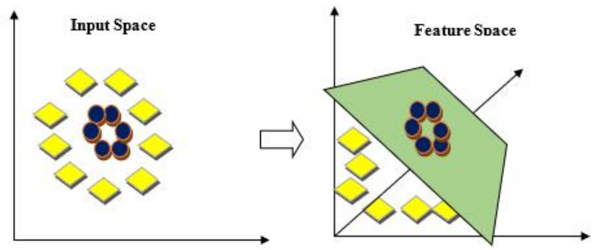

Fig. 1. Baseline SVM Classifier

K Nearest Neighbors (KNN)

According to KNN classification, an item is allocated to its closest k-neighbors based on their majority vote, with the object given to the class that is the most prevalent among its nearest neighbours. Class membership is the algorithm's final result.

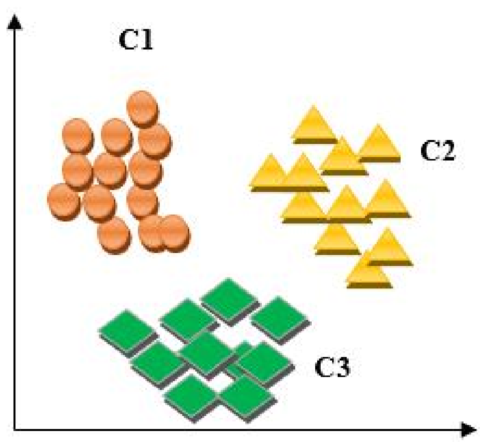

Fig. 2. Baseline KNN

CNN

A CNN model's four primary operations are:

- Convolution layer
- Non-linearity (Relu) layer
- Pooling layer
- Fully-connected layer

Convolution: Extracting features from a picture using convolution. It keeps the spatial link between pixels intact by learning picture properties from tiny input squares. Relu is the most common follow-up.

Relu: In the feature map, it is a zero-replacement process for all negative pixel values. To make a convolution network more interesting, it adds non-linearity to the equations.

Pooling: Reducing the dimension of each feature map while retaining crucial data is achieved by pooling (also known as downsampling).

Fully-connected layer: Perceptron with softmax function in output layer is multi-layered. Features extracted from previous layers may be used to classify images into different classes depending on the training data.

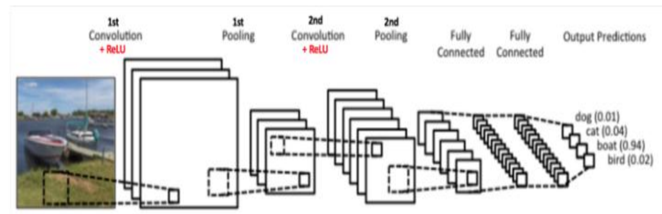

Fig. 3. Baseline CNN model

1.2 Research Paper Organization

2 LITERATURE REVIEW

[3] Sign languages are described as an ordered set of hand movements with specific meanings used by the hearing impaired to communicate in daily life. Because they are visual languages, they rely on body language, facial expressions, and hand gestures to convey meaning. Everywhere in the globe, there are more than 300 distinct sign languages. There are many distinct sign languages, but only a small fraction of the public knows any of them, making it difficult for persons with disabilities to communicate freely with others. SLR makes it possible to communicate in sign language without having any prior knowledge of the language. As a result, it can identify gestures and translate them into a language that is widely spoken, such as English.

A lot of work has already been done on SLR, but there is still a lot of room for improvement. Electronic systems can make judgments based on data, thanks to machine learning algorithms. Training and testing datasets are required for classification algorithms. [6] The training set gives the classifier with experience, whereas the testing set is used to evaluate the model. Many researchers have devised effective data collecting and categorization techniques [3][7]. Direct measurement techniques and vision-based procedures [3] may be divided into two categories depending on the data gathering method. Motion data gloves, motion capturing systems, or sensors are used for direct measurements. As a result of the motion data, it is possible to construct strong SLR algorithms that can monitor fingers, hands, and other body parts. Visual techniques to SLR depend on RGB pictures to extract spatial and temporal information that may be used to discriminate objects. Methods that use vision to monitor and extract hand regions before classifying them as movements [3] are common. In order to recognise hands, semantic segmentation and skin colour detection are used [8][9], since the colour of the skin is usually easy to discern. In order to distinguish just the moving elements in a scene, newer hand detection systems additionally employ face recognition and subtraction as well as backdrop subtraction to distinguish only the hands from other body parts like faces and arms [10][11]. Filtering methods, including as Kalman and particle filters

ISBN: 978-1-6654-7436-8/22/$31.00 ©2022 IEEE

[10][12], were used to provide precise and robust hands tracking, particularly in the presence of obstacles.

Both direct measurement and vision-based approaches need the usage of a variety of instruments in order to collect data. [13] In an SLR system, the camera is the principal input device. Microsoft Kinect, for example, delivers a colour video stream and a depth video stream both at the same time for input. For background segmentation, the depth data is useful. Additionally, accelerometers and sensory gloves are used in the collection of data. The San Francisco-based technology firm "Leap Motion," now known as "Ultraleap," created the touchless Leap Motion Controller (LMC) [14][15] for data collection. It has a frame rate of roughly 200 fps and is capable of detecting and following objects that resemble fingers, including hands and fingers. As it is difficult to get a sign language training dataset, the majority of researchers use their signer to capture their training data. There have been a variety of processing techniques used in the development of the SLR system. SLR often employs the HMM [12], a hidden Markov model. HMMs such as Light HMM and Tied-Mixture Density HMM (MSHMM) have been employed in a variety of ways, but the most common one is MSHMM. Neural network processing models have also been employed. [19] Nave Bayes Classifier (NBC) and Multilayer Perceptron (MLP) [14, unsupervised neural network] [20][21][22][23].

An SOM, SOFM, SRN, SVM, and 3D convolutional residual network [28] are some of the methods that may be used to create these maps. The wavelet-based technique [29] and Eigen Value Euclidean Distance [30] have also been employed by researchers. Variations in processing and application techniques have produced a wide range of accuracy outcomes to far. 83.6 percent, 86.7 percent, 97.5 percent, 97.7 percent and 100% were the results of the Light HMM, the MSHMM, the SVMs, Eigen Values and Wavelet Families respectively. [2][31][22][30]. However, despite the fact that many models have produced high accuracy results, the accuracy does not just rely on the processing model employed, but on numerous aspects such as dataset size and quality, data gathering techniques, devices utilised, and so forth.

Isolated SLR and continuous SLR are the two kinds of SLR systems. The system is taught to detect a single gesture in isolated SLR. Whether it's a letter, a number, or some other motion, each picture is labelled to symbolise something specific. Continuous SLR is distinct from the categorization of individual gestures. It's possible to identify and interpret whole words in continuous SLR, rather than just one gesture. [33][34]. Even with all the SLR research, there are still numerous gaps that need to be filled. To begin, these are a few of the concerns and obstacles that need to be addressed. Each word must be labelled meticulously in isolated SLR techniques. A non-trivial pre-processing phase, temporal segmentation unavoidably propagates faults into later stages, and sentence synthesis are used as building blocks in continuous SLR approaches.

Data collecting devices are expensive, hence a low-cost technique is required for SLR systems to be commercialised. Using a webcam instead of a higher-end camera may save money, but the picture may be distorted, making it less useful. In addition, there are several drawbacks to sensor data collecting, such as noise, poor human handling, and poor ground connection. This is a problem with vision-based methods since the hand and finger overlap. There are no large datasets available.

Many people have the impression that sign languages are universal, however this is not true. Sign language is based on spoken language, not universal.

3 PROPOSED METHODOLOGY

3.1 Datasets for Sign Language

For this project, we have used two datasets: ASL dataset and ISL dataset

American Sign Language (ASL) dataset: It is based on the ASL dataset by B. Kang et al. There are 31,000 photos in all, with 1000 photographs assigned to each of the 31 categories. For a total of five individuals, these motions have been documented. Numbers 0-9 and alphabets A-Z are included in the motions, however 'J' and 'Z,' which require hand movements and cannot be recorded in a picture, are not. (0/o), (V/2) and (W/6) are three examples of hand motions that are very similar. Context or meaning is used to categorise these items.

Indian Sign Language (ISL) Dataset: For ISL, there was no predefined dataset to work with. As a result, we turn to data compiled by IISc M.E. student Mukesh Kumar Makwana. Each of the 35 hand movements was represented by 1,250 of the 43,750 depth photos in the dataset. A total of 5 people were involved in the recording of these. With the exception of "2," which is identical to the letter 'v,' the gestures encompass all alphabets (A-Z) and numbers (0-9) The photos are in grayscale and have a 320x240 resolution.

Pre-processing: Segmentation is defined as the RGB picture is distorted into a grayscale image with a single grayscale channel. Using Canny Edge Detection, the grayscale image is turned into an image with just the most prominent edges remaining. Detecting edges in photos using the Canny edge detection method is both effective and widely utilized. It is an excellent and commonly used approach. Multi-stage techniques are used to identify sharp discontinuities in the incoming data stream.. This reduces background noise, allowing for more effective use of other methods.

3.2 Feature Extraction Techniques

To reduce the number of dimensions, feature extraction techniques are utilised to choose just the most relevant data from the original features. To decrease a large amount of data into something that is easier to analyse, it is possible to pare down the

ISBN: 978-1-6654-7436-8/22/$31.00 ©2022 IEEE

algorithm's input to a more manageable set of features. Classification algorithms are used in combination with feature extraction techniques such as PCA, LBP, and HoG. As a result, the model consumes less memory and performs better.

Pricipal Component Analysis (PCA)

Data is reduced in dimensionality by projecting it to a lower dimension, which is called PCA. One of the most significant features is one that has the greatest variation or spread, since this correlates to the greatest entropy and hence encodes the greatest amount of information. It is so decided to retain the most variable dimension while reducing other dimensions.

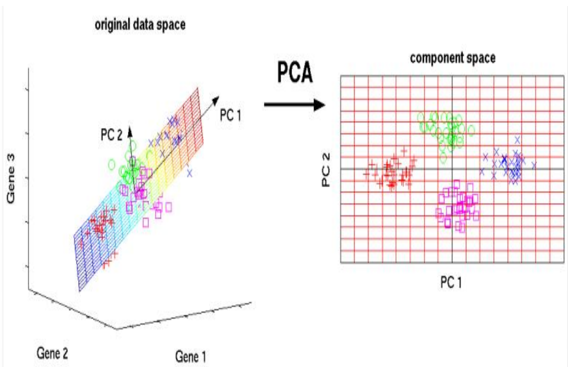

Fig. 4. Feature Extraction

HoG (Histogram of Gradients)

One technique to simplify an image is to utilise a feature descriptor, which is a representation or portion of an image that does just that. There are nine bins in the gradient histogram: 0, 20, 40, 60... 160. Hog creates the gradient histogram for the picture pixels using these nine angles: 0, 20, 40, 60... 160. In order to build a histogram for each each pixel, the pictures are separated into cells (typically 8x8) and the gradient magnitude and gradient angle are determined. After a block of cells' histograms have been normalised, the final feature vector for the picture as a whole may be produced.

Local Binary Pattern

The texture operator known as "Local Binary Patterns" effectively identifies pixels in an image by establishing a threshold and then estimating the nearby pixels of the target pixels [22]. As a binary string, the result is regarded as such. LBP is often used to analyze gray scale pictures, but it can also be used to analyze images such as RGB pixels may be used to generate three binary digits that can be described as a single feature with minimal adjustments.

Segmentation
Canny Edge Detection

↓

Feature Extraction
HOG, LBP, ORB
(Oriented FAST and Rotated BRIEF)

↓

PCA
for Dimensionality Reduction

↓

Classification
Random Forest, SVM, KNN, Naïve Bayes, CNN

Fig. 5. Flowchart of ORB Approach

The dataset include 26 alphabets, the nothing character (a specific image used to identify a lack of signs as default predictions), space characters, and deletes characters. Fig. 5 illustrates the proposed method's four essential steps: segmentation, feature extraction, building of a visual vocabulary histogram, and classification. In this proposed solution, a pre-processing technique was used to extract feature descriptors from a photograph.

FAST keypoint detector approach is used to compute key points and patch orientation, which is computed by calculating the direction of the vector from the patch's situated corner point to the patch's intensity-weighted centroid, which is then used to calculate the key points and orientation. ORB uses a multi-scale image pyramid since orientation is not included in the FAST attributes. Each pyramid level has a downsampled version of the original picture at the bottom, which contains all of the important places.

To calculate descriptors, a rotation of the BRIEF method is utilised since the BRIEF technique performs poorly when rotated. Use of a Gaussian kernel in the BRIEF approach smoothes the image, which reduces noise sensitivity and improves descriptor stability. The coordinates of the feature pixels are stored in a matrix, which is subsequently rotated (steered) depending on the patch's orientation. The figure 7 shows the ORB feature detection using identify patches in the image. In order to identify each patch, the algorithm generates a 32-dimensional vector. This generates a 32-dimensional feature vector for each photo in a supplied collection of sign pictures corresponding to a single sign class.

ISBN: 978-1-6654-7436-8/22/$31.00 ©2022 IEEE

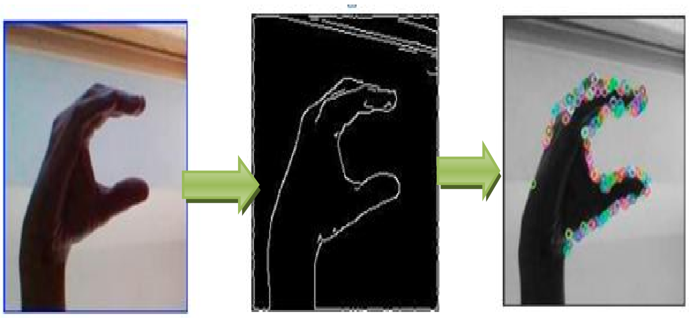

Fig. 6. Input image Fig. 7. Edge Detected Image Fig. 8. ORD Feature extraction

Histogram of Visual Vocabulary Generation

The many kinds of sign images can each be characterized by a distinct group of key descriptors that hold common qualities, as was discussed earlier. Feature models from this collection are used to build a training photo bag, which is subsequently utilized to produce training pictures. In order to get K groups with equivalent descriptors, K-means clustering is used. Nearby clusters are allocated to each patch in this image. A histogram of codewords is generated for each picture after the descriptors have been mapped to their appropriate clusters. It is possible to generate a language of code words by bagging together a number of feature histograms as shown in Fig 8. This method assumes that the number of code words created for each picture is 150.

In order to train the feature vector of 150 code words that was previously produced for each image, a number of different classification models, such as K Nearest Neighbor (KNN), Support Vector Machine (SVM), Random Forest, Naive Bayes and CNN are utilised. These classifiers are used for this research experiment. It is determined whether or not a model is accurate by continually putting it through its paces using the testing set.

5. RESULT AND DISCUSSION

A number of different classification models using ORB feature extraction in addition to other methods of feature extraction were considered in order to locate the decision-making process that works best in conjunction with our method. The goal was to find the optimal combination of these two factors. ORB's feature vector integrates the visual qualities of a variety of hand movements in a condensed yet meaningful fashion, which allows it to be far more accurate than previous systems. When compared to earlier methods, this is a significant advancement. When compared to ORB's cheap training costs for machine learning algorithms in sequence and memory, higher feature representations such as PCA and HOG are computationally inefficient. ORB has an additional benefit over LBP and PCA in that it is based on overall picture visual characteristics rather than the study of image textural characteristics or the examination of statistical data. This gives it an advantage over both of these other methods.

ORB's feature vector is significantly smaller than those created by PCA and HOG, which enables it to deliver superior average results when interconnected with canny edge detection algorithm and a bag of words. Instead of utilizing PCA and HOG, ORB's feature vector is significantly smaller than those created by PCA and HOG (which are both much greater). In the charts and tables that follow, we compare the outcomes obtained using our approach with those obtained using various well-known methods as shown in Figure 9.

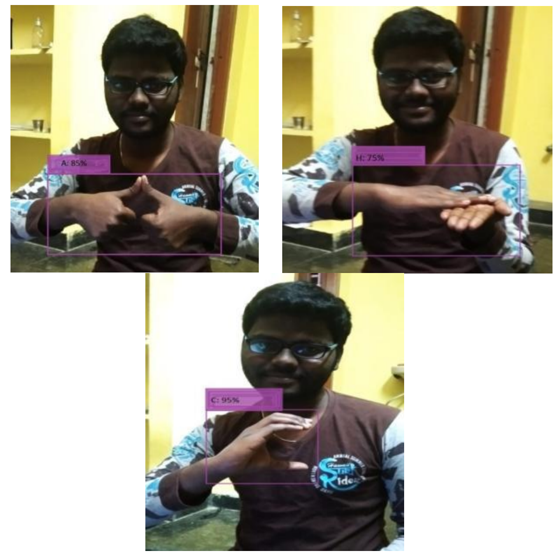

Fig. 9. Tested Output

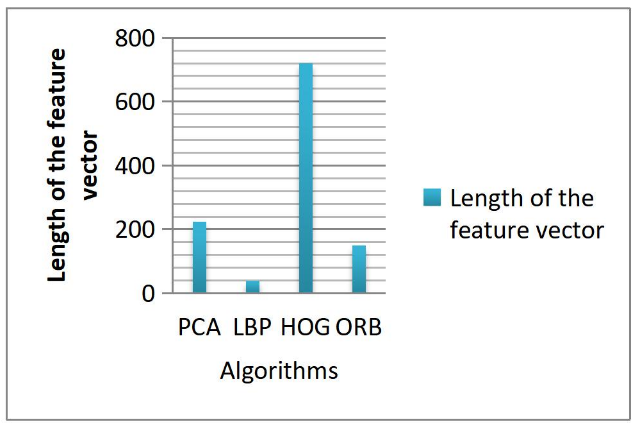

Fig. 10. Feature Vector Graph

ISBN: 978-1-6654-7436-8/22/$31.00 ©2022 IEEE

Table 1. Error Rate Comparison of Methodologies

Feature Extraction Technique	Feature Vector Length	Error Rate on Support Vector Machine	Error Rate on Random Forest	Error Rate on K-Nearest Neighbours	Error Rate on Gaussian Naive Bayes	Error Rate on Multilayer Perceptron	Error Rate on Logistic Regression
Oriented FAST and Rotated BRIEF	150	14.75	7.31	4.04	27.77	3.04	15.41
Histogram of Gradients	720	12.6	8.0	5.89	52.87	20.09	22.61
Local Binary Pattern	8	65.2	53.25	37.02	70.59	38.77	65.87
Principal Component Analysis	224	7.13	1.7	04.19	59.92	1.69	27.34

Table 2. Comparative Analysis with other Models

Dataset	Number of Test Images	Classifier Used	Feature Extraction Technique	Accuracy
ASL[1]	5	SVM	HOG	80
ASL + Digits [18]	100	SVM	YCbCr-HOG	89.54
Mobile-ASL [25]	800	SVM	SIFT	92.25
ASL (Proposed Approach)	17400	KNN	ORB	95.81
ASL (Proposed Approach)	17400	MLP	ORB	96.96

6. CONCLUSION

On the same dataset, the suggested approach of ORB feature extraction was tested against a range of different pre-processing approaches, such as Histogram of Gradients, Local Binary Pattern, and PCA, as detailed in the article. These techniques were used to analyse the input data. These methods have been tested using a variety of well-known classifiers, including KNN, SVM, Random Forest, Nave Bayes, Logistic Regression, and Multi-Layer Perceptron, and they have not produced any notable errors during the process. When it comes to the Nave Bayes, Logistic Regression, and KNN classifiers, the proposed methodology outperforms all other pre-processing techniques. On the other hand, principal component analysis outperforms all other pre-processing approaches when it comes to the MLP, Random Forest, and SVM classifiers. despite the fact that the method offers a very high level of accuracy in the identification of motions. Even though it is only being tested on still gesture images for the time being, it has the potential to be developed in the near future to recognise dynamic motion in movies in real time. Currently, the testing is only being done on static gesture images. There is potential for the system to be modified such that it can instantaneously recognise RGBD photos taken by Kinetic Sensors. Combining deep learning approaches such as modified convolutional neural networks, optimising using quantum computing, and applying evolutionary algorithms for feature selection after feature extraction are some ways to make the article better. In conjunction with graph theory, the use of actual hand models equipped with sensors may be implemented in an effort to achieve a higher level of precision in the model.

REFERENCE

[1]. Kapur, R.: The Types of Communication. MIJ. 6, (2020).

[2]. Suharjito, Anderson, R., Wiryana, F., Ariesta, M.C., Kusuma, G.P.: Sign Language Recognition Application Systems for Deaf-Mute People: A Review Based on Input-Process Output. Procedia Comput. Sci. 116, 441–448 (2017). https://doi.org/10.1016/J.PROCS.2017.10.028.

[3]. Konstantinidis, D., Dimitropoulos, K., Daras, P.: Sign language recognition based on hand and body skeletal data. 3DTV-Conference. 2018-June, (2018). https://doi.org/10.1109/3DTV.2018.8478467.

[4]. Dutta, K.K., Bellary, S.A.S.: Machine Learning Techniques for Indian Sign Language Recognition. Int. Conf. Curr. Trends Comput. Electr. Electron. Commun. CTCEEC 2017. 333–336 (2018). https://doi.org/10.1109/CTCEEC.2017.8454988.

[5]. Bragg, D., Koller, O., Bellard, M., Berke, L., Boudreault, P., Braffort, A., Caselli, N., Huenerfauth, M., Kacorri, H., Verhoef, T., Vogler, C., Morris, M.R.: Sign Language Recognition, Generation, and Translation: An Interdisciplinary Perspective. 21st Int. ACM SIGACCESS Conf. Comput. Access. (2019). https://doi.org/10.1145/3308561.

[6]. Rosero-Montalvo, P.D., Godoy-Trujillo, P., Flores-Bosmediano, E., Carrascal-Garcia, J., Otero-Potosi, S., Benitez-Pereira, H., Peluffo-Ordonez, D.H.: Sign Language Recognition Based on Intelligent Glove Using Machine Learning Techniques. 2018 IEEE 3rd Ecuador Tech. Chapters Meet. ETCM 2018. (2018). https://doi.org/10.1109/ETCM.2018.8580268.

[7]. Zheng, L., Liang, B., Jiang, A.: Recent Advances of Deep Learning for Sign Language Recognition. DICTA 2017 - 2017 Int. Conf. Digit. Image Comput. Tech. Appl. 2017-Decem, 1–7 (2017). https://doi.org/10.1109/DICTA.2017.8227483.

[8]. Rautaray, S.S.: A Real Time Hand Tracking System for Interactive Applications. Int. J. Comput. Appl. 18, 975–8887 (2011).

[9]. Zhang, Z., Huang, F.: Hand tracking algorithm based on super-pixels feature. Proc. - 2013 Int. Conf. Inf. Sci. Cloud Comput. Companion, ISCC-C 2013. 629–634 (2014). https://doi.org/10.1109/ISCC-C.2013.77.

[10]. Lim, K.M., Tan, A.W.C., Tan, S.C.: A feature covariance matrix with serial particle filter for isolated sign language recognition. Expert Syst. Appl. 54, 208–218 (2016). https://doi.org/10.1016/J.ESWA.2016.01.047.

[11]. Kapur, R.: The Types of Communication. MIJ. 6, (2020).

[12]. Suharjito, Anderson, R., Wiryana, F., Ariesta, M.C., Kusuma, G.P.: Sign Language Recognition Application Systems for Deaf-Mute People: A Review Based on InputProcess-Output. Procedia Comput. Sci. 116, 441–448 (2017). https://doi.org/10.1016/J.PROCS.2017.10.028.

[13]. Konstantinidis, D., Dimitropoulos, K., Daras, P.: Sign language recognition based on hand and body skeletal data. 3DTV-Conference. 2018-June, (2018). https://doi.org/10.1109/3DTV.2018.8478467.

[14]. Dutta, K.K., Bellary, S.A.S.: Machine Learning Techniques for Indian Sign Language Recognition. Int. Conf. Curr. Trends Comput. Electr. Electron. Commun. CTCEEC 2017. 333–336 (2018). https://doi.org/10.1109/CTCEEC.2017.8454988.

[15]. Bragg, D., Koller, O., Bellard, M., Berke, L., Boudreault, P., Braffort, A., Caselli, N., Huenerfauth, M., Kacorri, H., Verhoef, T., Vogler, C., Morris, M.R.: Sign Language Recognition, Generation, and Translation: An Interdisciplinary Perspective. 21st Int. ACM SIGACCESS Conf. Comput. Access. (2019). https://doi.org/10.1145/3308561.

[16]. Lim, K.M., Tan, A.W.C., Tan, S.C.: Block-based histogram of optical flow for isolated sign language recognition. J. Vis. Commun. Image Represent. 40, 538–545 (2016). https://doi.org/10.1016/J.JVCIR.2016.07.020.

[17]. Gaus, Y.F.A., Wong, F.: Hidden Markov Model - Based gesture recognition with overlapping hand-head/hand-hand estimated using Kalman Filter. Proc. - 3rd Int. Conf. Intell. Syst. Model. Simulation, ISMS 2012. 262–267 (2012). https://doi.org/10.1109/ISMS.2012.67.

[18]. Nikam, A.S., Ambekar, A.G.: Sign language recognition using image based hand gesture recognition techniques. Proc. 2016 Online Int. Conf. Green Eng. Technol. IC-GET 2016. (2017). https://doi.org/10.1109/GET.2016.7916786.

ISBN: 978-1-6654-7436-8/22/$31.00 ©2022 IEEE

[19]. Mohandes, M., Aliyu, S., Deriche, M.: Arabic sign language recognition using the leap motion controller. IEEE Int. Symp. Ind. Electron. 960–965 (2014). https://doi.org/10.1109/ISIE.2014.6864742.
[20]. Enikeev, D.G., Mustafina, S.A.: Sign language recognition through Leap Motion controller and input prediction algorithm. J. Phys. Conf. Ser. 1715, 012008 (2021). https://doi.org/10.1088/1742-6596/1715/1/012008.
[21]. Cheok, M.J., Omar, Z., Jaward, M.H.: A review of hand gesture and sign language recognition techniques. Int. J. Mach. Learn. Cybern. 2017 101. 10, 131–153 (2017). https://doi.org/10.1007/S13042-017-0705-5.
[22]. Wadhawan, A., Kumar, P.: Sign Language Recognition Systems: A Decade Systematic Literature Review. Arch. Comput. Methods Eng. 2019 283. 28, 785–813 (2019). https://doi.org/10.1007/S11831-019-09384-2.
[23]. Camgöz, N.C., Koller, O., Hadfield, S., Bowden, R.: Sign language transformers: Joint end-to-end sign language recognition and translation. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 10020–10030 (2020). https://doi.org/10. 1109/CVPR 42600.2020.01004.
[24]. Cui, R., Liu, H., Zhang, C.: A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training. IEEE Trans. Multimed. 21, 1880–1891 (2019). https://doi.org/10.1109/TMM.2018.2889563
[25]. Hore, S., Chatterjee, S., Santhi, V., Dey, N., Ashour, A.S., Balas, V.E., Shi, F.: Indian Sign Language Recognition Using Optimized Neural Networks. Adv. Intell. Syst. Comput. 455, 553–563 (2017). https://doi.org/10.1007/978-3-319-38771-0_54.
[26]. Kumar, P., Roy, P.P., Dogra, D.P.: Independent Bayesian classifier combination based sign language recognition using facial expression. Inf. Sci. (Ny). 428, 30–48 (2018). https://doi.org/10.1016/J.INS.2017.10.046.
[27]. Sharma, A., Sharma, N., Saxena, Y., Singh, A., Sadhya, D.: Benchmarking deep neural network approaches for Indian Sign Language recognition. Neural Comput. Appl. 2020 3312. 33, 6685–6696 (2020). https://doi.org/10.1007/S00521-020-05448-8.
[28]. Kishore, P.V.V., Prasad, M. V.D., Prasad, C.R., Rahul, R.: 4-Camera model for sign language recognition using elliptical fourier descriptors and ANN. Int. Conf. Signal Process. Commun. Eng. Syst. - Proc. SPACES 2015, Assoc. with IEEE. 34–38 (2015). https://doi.org/10.1109/SPACES.2015.7058288.
[29]. Tewari, D., Srivastava, S.K.: A Visual Recognition of Static Hand Gestures in Indian Sign Language based on Kohonen Self-Organizing Map Algorithm. Int. J. Eng. Adv. Technol. 165 (2012).
[30]. Gao, W., Fang, G., Zhao, D., Chen, Y.: A Chinese sign language recognition system based on SOFM/SRN/HMM. Pattern Recognit. 37, 2389–2402 (2004). https://doi.org/10.1016/J.PATCOG.2004.04.008.
[31]. Quocthang, P., Dung, N.D., Thuy, N.T.: A comparison of SimpSVM and RVM for sign language recognition. ACM Int. Conf. Proceeding Ser. 98–104 (2017). https://doi.org/10.1145/3036290.3036322.
[32]. Pu, J., Zhou, W., Li, H.: Iterative alignment network for continuous sign language recognition. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 2019-June, 4160–4169 (2019). https://doi.org/10.1109/CVPR.2019.00429. 29. Kalsh, E.A., Garewal, N.S.: Sign Language Recognition System. Int. J. Comput. Eng. Res. 6.
[33]. Singha, J., Das, K.: Indian Sign Language Recognition Using Eigen Value Weighted Euclidean Distance Based Classification Technique. IJACSA) Int. J. Adv. Comput. Sci. Appl. 4, (2013).

ISBN: 978-1-6654-7436-8/22/$31.00 ©2022 IEEE