Journal of Advanced Research in Computer Graphics and Multimedia Technology
Volume 5, Issue 1 - 2023, Pg. No. 14-19
Peer Reviewed Journal

Research Article

Research Report on the Automatic Recognition System for Sign Language

Akshat Rattan[1], Naman Jain[2]

[1,2]Student 6th Semester, DAV Institute of Engineering and Technology, Kabir Nagar, Jalandhar, India.

I N F O

Corresponding Author:
Akshat Rattan, DAV Institute of Engineering and Technology, Kabir Nagar, Jalandhar, India.
E-mail Id:
akshatrattan85@gmail.com
Orcid Id:
https://orcid.org/0009-0004-7307-6417
How to cite this article:
Rattan A, Jain N. Research report on the Automatic Recognition System for Sign Language. *J Adv Res Comp Graph Multim Tech.* 2023; 5(1): 14-19.

Date of Submission: 2023-03-24
Date of Acceptance: 2023-05-10

A B S T R A C T

This article discusses a potential concept for a recognition system for dynamic sign language. The final user will be able to learn and interpret sign language thanks to this technological advancement. The use of machine learning has become more prevalent in the field of Optical Character Recognition (OCR), which is capable of recognizing printed as well as handwritten characters. Using the concepts of supervised learning, we have constructed a broad array of classification, prediction, identification systems. Although earlier algorithms can detect sign language with a level of accuracy comparable to ours, our technique also makes use of the identification of live video streams. As a consequence of this, it provides a higher level of engagement than the systems that are already in place. Sign language is one method that may be used while attempting to communicate with deaf individuals. It is necessary to acquire sign language in order to communicate with them. The majority of learning takes place in social settings with peers. There aren't too many available learning materials when it comes to sign language. As a direct consequence of this, being educated in sign language is a very difficult process. Fingerspelling is the initial stage in learning sign language, it is also used when there is no sign that corresponds to the word being communicated or when the signer is uninformed of the sign. The vast majority of the sign language learning systems that are now on the market depend on more expensive peripheral sensors. We expect to make headway in this field by amassing a dataset and using a variety of feature extraction strategies in order to obtain data that is relevant to the study. After that, this information is inputted into a number of supervised learning algorithms. Sign language is an essential tool for bridging the communication gap between those who are deaf or hard of hearing and others whose hearing is normal. The variety of the nearly 7000 current sign languages, as well as changes in motion position, hand shape, body part location, make automatic sign language recognition (ASLR) a challenging system. Researchers are researching better ways to build ASLR systems in order to find intelligent strategies to tackle such complexity, they have shown significant success in this area. This study takes a look at the research that has been conducted on intelligent systems for sign language recognition over the course of the last two decades and publishes its findings.

Keywords: Sign Language, Optical Character Recognition, Algorithms, Sensors, ASLR Systems

Introduction

Communication is essential to our existence because it enables us to express ourselves in meaningful ways. We are able to communicate with one another via the use of voice, body language, reading, writing, or even the use of visual aids; nonetheless, speaking is one of the most common techniques. Fortuitously, a barrier to communication still exists for the subgroup of people who suffer from speech and hearing impairments. Interpreters and other visual aids are used in order to communicate with these individuals. However, these approaches are not suitable for use in urgent situations since they are both time-consuming and costly. To do this, the speaker will concurrently combine different hand shapes, orientations, movements of the hands, arms, or torso to communicate their views.[1]

Sign Language is comprised of two components: finger-spelling, which includes writing down words with one's fingers, word-level association, which requires using hand gestures to express the meaning of individual words. The ability to communicate names, addresses, other terms that do not hold meaning in the word-level association is made possible by the use of a technique called finger-spelling, which is an essential component of sign language. [2] In spite of this, finger-spelling is not used very often since it is difficult to understand and apply. A further disadvantage of using sign language as a replacement for verbal communication is that there is no universal sign language, even fewer people are acquainted with it. This problem may be handled by using a method that categorizes finger spelling in sign language. This paper makes use of a number of different machine-learning algorithms and compares and contrasts the accuracy of those techniques.

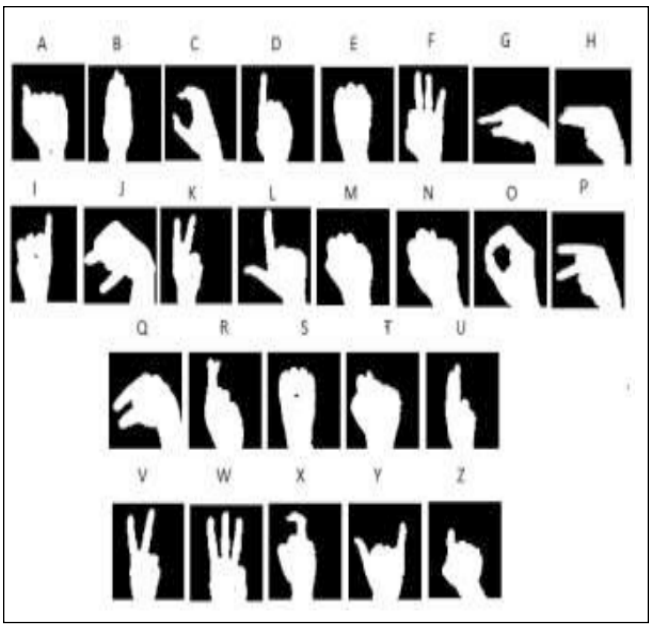

Figure 1.Indian Sign Language[3]

Algorithms Used

During the supervised learning process, classification machine learning algorithms such as SVM and k-NN are used. Supervised learning requires the dataset to be categorized before it is fed into the algorithm for training purposes. SVM, k-NN, CNN are some of the classification methods that were used in this research project. Feature extraction techniques are used to construct a subset of the original features for the purpose of dimensionality reduction. This ensures that the algorithm only gets data that is relevant to its purpose. When the input to the algorithm grows too huge to manage or looks to be redundant (like a pattern of pictures that is created by pixels repeatedly), it may be streamlined into a collection of characteristics that is easier to work with. In order to do this, classification strategies together with feature extraction methods such as PCA, LBP, HoG are used. As a direct consequence of this, a lower amount of memory is required, the performance of the model is improved. The following algorithms are what are being used:

Support Vector Machine Abbreviated as SVM

Each data point in the SVM is shown in an n-dimensional space, where n is the number of features. The value of each feature is the value of a given coordinate. It has been revealed that the classification may be carried out with the use of a hyperplane that effectively divides the classes.

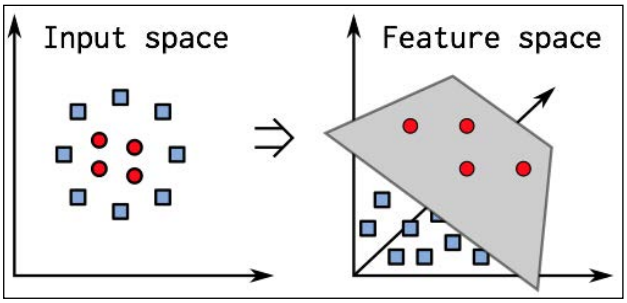

Figure 2.Indian Sign Language[3]

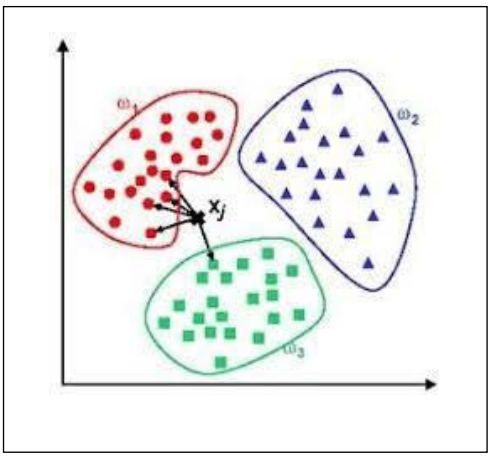

Figure 3.K-NN algorithm[1]

k-NN (k-Nearest Neighbors)

In k-NN classification, an item is assigned to the class that is the most common among its k-nearest neighbors.[4] In this context, k is a positive number that is most often rather small. The class membership is produced as an output by the algorithm.

CNN

When processing data that has a topology similar to a grid, such as images that may be represented as a two-dimensional array of pixels, a kind of neural network called a convolutional neural network (CNN) is used. Convolution, nonlinearity (Relu), pooling, classification (Fully-connected layer) are the four fundamental processes that make up a CNN model. Convolution: The process of convolution allows for the extraction of features from the input of a picture. The spatial connection that exists between pixels may be maintained while learning picture properties via the use of tiny squares of input data. In most cases, Relu is performed following it.

An element-wise method known as Relu replaces any pixels in the feature map with a value of zero once all negative pixel values are removed. The aim of this technique is to provide a convolutional network with nonlinearity.

Pooling: Down sampling, often known as pooling, is a popular name for a specific kind of down sampling that reduces the dimensionality of each feature map while maintaining essential information.

Fully-connected layer: The SoftMax function is used in the perception's output layer, which is part of a multi-layer representation. It attempts to classify the input picture into a variety of categories by using the training data and characteristics from preceding layers. A CNN model is produced by combining these layers into one. The topmost layer has complete connections.

networks. The learned information is stored as "weights" and may be exported to and imported from other models. It is feasible to utilize the pre-trained model as a feature extractor if one stack fully connected layers on top of the previous layers in the model. The model is then trained using the initial dataset once the previously-saved weights have been loaded.

The acronym PCA stands for "Principal Component Analysis."

PCA is used to decrease the dimensionality of the data before it is projected to a lower dimension. Because it contains the highest amount of entropy and, as a result, the biggest amount of information, the character with the greatest amount of variation or dispersion is the most important. This leads in maintaining the dimension that has the largest variation while decreasing the variance of the other dimensions.

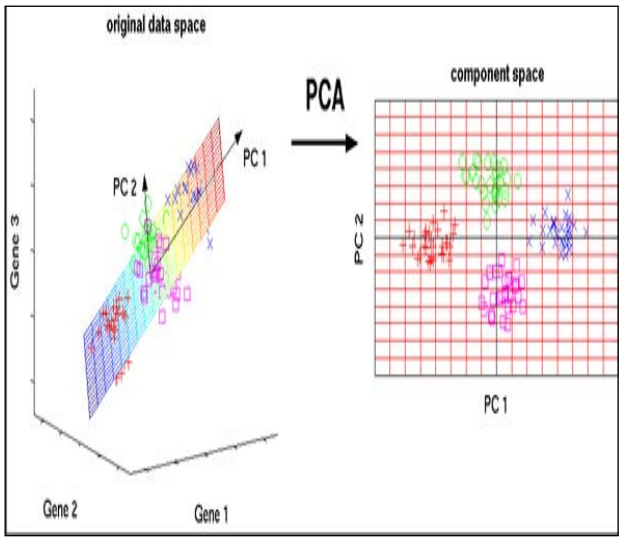

Figure 5. PCA Feature Engineering[1]

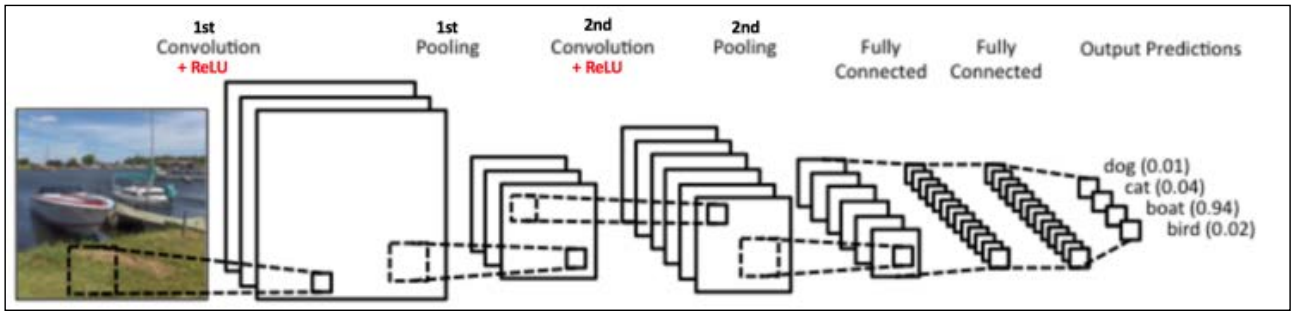

Figure 4: Fully Connected Layer 5

Using the transfer learning concept, the model is first pre-trained on a dataset that is different from the one it was originally developed on. By carrying out these steps, the model is provided with the opportunity to acquire knowledge that may be communicated with other neural

LBPLBP (Local Binary Patterns)

LBP computes a local representation of texture by evaluating each pixel in relation to its surrounding or neighboring pixels. The output is recorded as an array as an LBP 2D array, which is later converted to decimal.

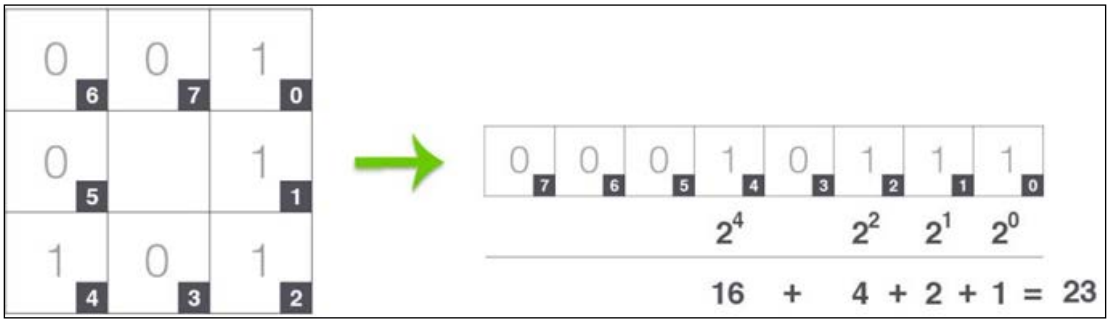

Figure 6.Conversion of the pixel into LBP representation[1]

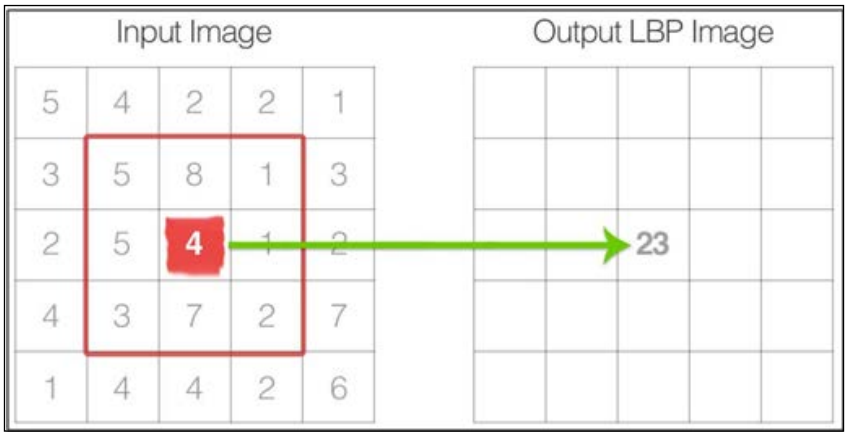

Figure 7.Local Binary Pattern[1]

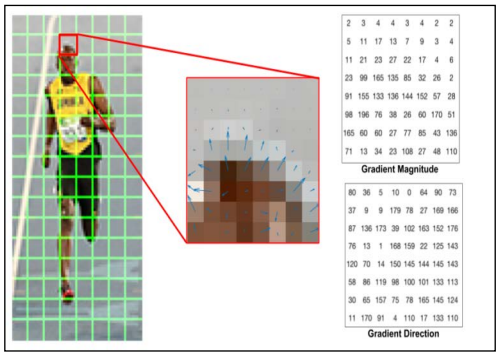

Figure 8.Calculation of Gradient Magnitude and Gradient Direction[1]

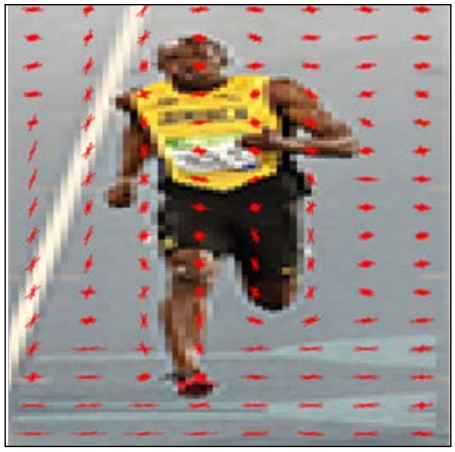

Figure 9.HOG Visualization[1]

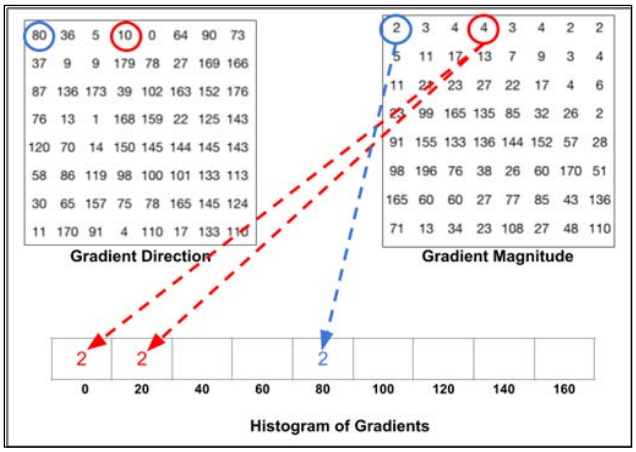

Figure 10.Creating histogram from Gradient of magnitude and direction[1]

HoG (Histogram of Gradients)

A representation of an image or an image patch is referred to as a feature descriptor. This representation simplifies the original picture by removing any extraneous information. The Hog feature descriptor generates a vector for the image pixels consisting of nine bins, or integers, each of which corresponds to one of the following angles: 0 degrees, 20 degrees, 40 degrees, 60 degrees, or 160 degrees. After the photographs have been segmented into cells (often 8 by 8), a histogram is generated for each cell by applying the calculations for the gradient magnitude and gradient angle. The final feature vector for the whole picture is produced

once the histogram of a block of cells has been normalized. When determining what kinds of features can be identified quickly and accurately, it is necessary to take into account the characteristics of the backdrop, the presence of other objects (known as occlusion), the lighting. An technique that finds the histogram of an oriented gradient is built in order to facilitate the detection of ISL.[6]

Experiments on ASL

SVM+PCA

The support vector machine (SVM) classifier is built with the help of the sklearn package's built-in support vector machine (SVM) module. PCA is used for feature extraction, the PCA module is included in the sklearn. decomposition package is used to implement it. A graph of "number of components vs. variance" is produced so that a graph of "number of components vs. variance" may be used to discover the ideal number of components to which we can decrease the original feature set without sacrificing the important qualities. When looking at the graph, 53 components are thought to be the ideal number since the related variance is so close to being at its highest point. Up to the age of 53, the variance of each component steadily reduces and eventually approaches a constant state.

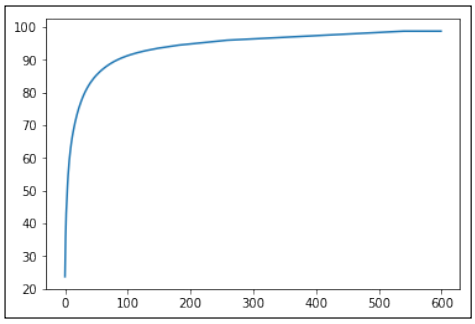

Figure 11.Y-axis: Variance, X-axis: No. of Components[1]

With PCA, we were able to reduce the algorithm's complexity and training time by going from 65536 to 53 components.

SVM+HoG :

The highest accuracy results to yet came from combining SVM with HoG. HoG was implemented using the Scikit-Image library's HoG module.

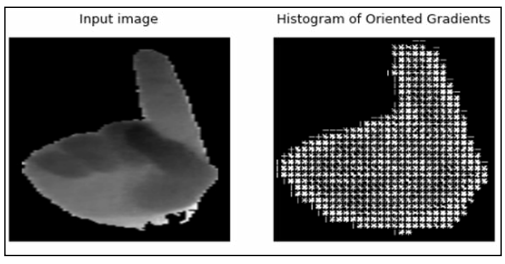

Figure12.Histogram of Gradients[1]

Table 1

Pixels per cell	Cells per block	Average accuracy
8x8	1x1	77.64
8x8	2X2	65.33
16X16	1X1	72.05

The outcomes of varying the parameters pixels per cell and cells per block are as follows:

The parameter was picked since it showed the highest degree of accuracy (8x8, 1x1).

Accuracies Recorded

The following accuracies were noted after the ASL dataset implementation of the datasets that exhibited promising results. The following table shows the maximum accuracies recorded for each algorithm:

Table 2

Images/ Class	Testing Datasets	Classifier	Parameters	Max. accuracy
200	8	SVM + HOG	Pixels per cell: (8,8) Cells per block: (1,1)	81.88 %
200	8	SVM + PCA	No. of compo-nents = 53	78.67 %

The table below shows the average accuracies recorded for each algorithm:

Table 3

Images/ class	Classifier	Parameters	Avg. accuracy
200	SVM + HOG	Pixels per cell: (9.9) Cells per block: (4,4)	74.32%
200	SVM + PCA	No. of components = 53	78.9 %

Conclusion

In conclusion, we find that SVM+HoG and convolutional neural networks are useful tools for the recognition of sign language when used in conjunction with classification methods. To demonstrate an increase in accuracy, pre-training must be carried out using a more extensive dataset. employing SVM+HoG, we were able to get a performance that was 71.88% better than the accuracy that was published

in earlier works of literature for the ISL dataset. This was accomplished by employing 4 subjects for training and a different subject for testing.

Future Work

A user-dependent model with pre-training: The model is able to work well even when trained with a small dataset by first pre-training it on a bigger dataset (such as the 14,000-class ILSRVC) and then fine-tuning it using the ISL dataset. This allows the model to perform effectively even when trained with a small dataset. When utilizing user-dependent models, the user will provide the model a set of training images so that it can get familiar with the user. This allows the model to learn more about the user. For a certain kind of user, the model will perform admirably when used in this manner.

References

1. https://edu.authorcafe.com/academies/6813/sign-language-recognition
2. https://github.com/Goutam1511/Sign-Language-Recognition-using-Scikit-Learn-and-CNN/blob/master/README.md
3. https://www.irjet.net/archives/V7/i3/IRJET-V7I3418.pdf
4. https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm
5. https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/
6. https://ijcsit.com/docs/Volume%205/vol5issue03/ijcsit20140503220.pdf
7. https://scikit-learn.org/stable/modules/svm.html#svm-classification
8. https://scikit-image.org/docs/stable/auto_examples/features_detection/plot_hog.html
9. Sako, H. and Smith, A. (1996)Real-time facial expression recognition based on features' position and dimension. in Proceedings of the International Conference on Pattern Recognition, ICPR'96.
10. Sagawa H, et al., Pattern recognition and synthesis for a sign language translation system. *Journal of Visual Languages and Computing* 1996. 7: 109-127.
11. Sagawa H, Takeuchi M, Ohki M. Description and recognition methods for sign language based on gesture components. in Proceedings of IUI 97. Orlando, Florida: ACM 1997.
12. Sagawa H, Takeuchi M, Ohki M. Sign language recognition based on components of gestures - integeration of symbols and patters. in RWC '97 1997.
13. Liddell SK. American Sign Language Syntax. Approaches to Semiotics, The Hague: Mouton. 1980; 194.
14. Stokoe WC. Sign Language Structure: An Outline of the Visual Communication Systems of the American Deaf. University of Buffalo 1960.
15. Sweeney GJ, Downton AC. Towards appearance-based multi-channel gesture recognition, in Progress in Gestural Interaction: Proceedings of Gesture Workshop '96, P.A. Harling and A.D.N. Edwards, Editor. 1996, Springer: London. p. 7-16.
16. Kyle JG, Woll B. Sign Language The Study of Deaf People and their Language. Cambridge: Cambridge University Press. 1988; 318.
17. Gazdar G, Mellish C. Natural Language Processing in Prolog: An Introduction to Computational Linguistics. Wokingham, England: Addison-Wesley 1989; 504.