Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 173 (2020) 181–190

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on *Smart Sustainable Intelligent Computing and Applications* under ICITETM2020

# Hand Gesture Recognition using Image Processing and Feature Extraction Techniques

Ashish Sharma[a], Anmol Mittal[a], Savitoj Singh[a], Vasudev Awatramani[a]

[a]Maharaja Agrasen Institute of Technology, Guru Gobind Singh Indraprastha University, New Delhi-110085, Delhi, India

**Abstract**

Image identification is becoming a crucial step in most of the modern world problem-solving systems. Approaches for image detection, analysis and classification are available in glut, but the difference between such approaches is still arcane. It essential that proper distinctions between such techniques should be interpreted and they should be analyzed. Standard American Sign Language (ASL) images of a person's hand photographed under several different environmental conditions are taken as the dataset. The main aim is to recognize and classify such hand gestures to their correct meaning with the maximum accuracy possible. A novel approach for the same has been proposed and some other widely popular models have compared with it. The different preprocessing techniques used are Histogram of Gradients, Principal Component Analysis, Local Binary Patterns. The novel model is made using canny edge detection, ORB and bag of word technique. The preprocessed data is passed through several classifiers (Random Forests, Support Vector Machines, Naïve Bayes, Logistic Regression, K-Nearest Neighbours, Multilayer Perceptron) to draw effective results. The accuracy of the new models has been found significantly higher than the existing model.



*Keywords:* Hand Gesture Recognition; Machine Learning; Image Processing; American Sign Language; Multi-Class Classification

## 1. Introduction

Understanding sign language is an arduous task and it is a skill that has to be learned with practice. But with this paper, we aim to provide several schemes of identifying and understanding such letters without learning the sign language. We focus primarily on the development of new procedures to understand sign language, and to find differences between the approaches and best method of recognition of the sign language. There are several difficulties in developing a better method for sign recognition such as, in real life the images captured are so


10.1016/j.procs.2020.06.022

excessively noisy that high level of pre-processing is required, the datasets available online are generally so noiseless, that working on them leads to the development of models trained only to handle images with less or nearly no noise, hence being impractical for real-life application. Thus, it is imperative to create a model that can handle noisy images and also be able to produce positive results.

American sign language is a widely used language for physically impaired (as shown in Fig 1). The main model is constructed to recognize sign gesture images of the hand, which utilizes Oriented FAST and Rotated BRIEF (ORB) as a feature detector, having efficacy and performance better than widely used feature detectors such as SIFT and SURF, etc. The model utilizes a combination of several other techniques and classifiers.

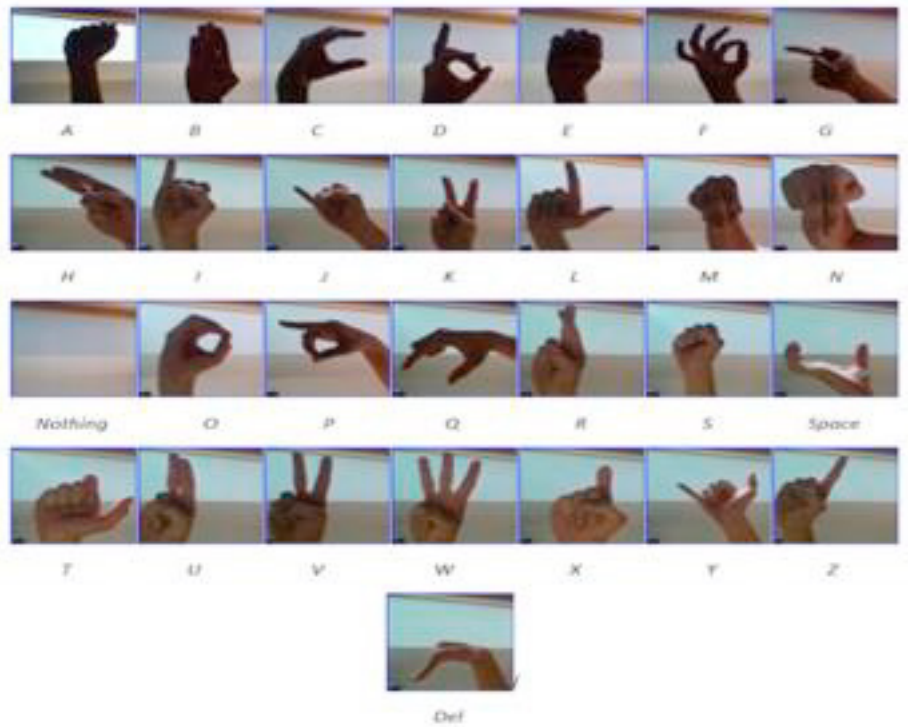


Fig 1. ASL character dataset

This paper involves six segments: segment 2 analyses the related work made in the area of gesture recognition. The background and methodology are showcased in segment 3 and segment 4 respectively. The results and analysis have been performed in segment 5. In Segment 6, the conclusion has been drawn along with the future scope. The last Segment contains the list of references used in this paper.

## 2. Related Work

Various types of techniques can be used to implement the classification and recognition of images using machine learning. Apart from recognizing static images, work has been done in the field of depth-camera sensing and video processing. A diversity of processes embedded in the system was developed using various other programming languages to implement the procedural techniques for the final system's maximum efficacy.

The problem can be solved and systematically organized into three similar approaches which are, firstly using static image recognition techniques and preprocessing procedures, secondly by using deep learning models and thirdly by using Hidden Markov Models.

Various research papers used different kinds of combinations of preprocessing and feature extraction techniques, like using MATLAB as the main programming language and using scale-invariant feature transform (SIFT), Histogram of Gradients (HOG) in order to extract the features descriptors from the image and then using

Support Vector Machine in order to classify these into various alphabets [1]. Webcam images have also been used to identify gestures [2]. Xbox Kinect camera to retrieve the images and then transformed these images to YCbCr format as an image pre-processing step and then used HOG for feature extraction and SVM for classification of the images to their respective letters [3]. Combination of self-developed algorithm assisted by the OpenCV library in which video sequencing is used to get images and conversion into YCbCr image format, followed by the hand being further segmented to achieve reduced noise images for better recognition by using mathematical techniques [4]. An improved and optimized Genetic algorithm along with adaptive filtering has been used to sign gestures [5]. Various signal processing methods approaches have also been used to develop an accurate model such as a combination of DWT and F-ratio [6]. Video as the input with a combination of k-nearest neighbours (KNN) and Bayesian algorithm to classify the images proving to be an innovative recognition system [7]. Self-developed hand segmentation technique and HOG for feature extraction and Principal Component Analysis (PCA) for classification of the images [8]. Different types of SVM kernels and Multi-Class SVM accompanied by their comparison helped to elucidate the change in kernel effect on the image recognition system [9]. Different types of Segmentation Graph kernels have also been compared to find its effect on the dataset of images related to American sign language [10]. A similar approach but utilizing labeled graph kernel is also present, used to study the behavior [11].

State of art techniques focused upon utilizing the deep learning models to get better accuracy and less execution time. Model using special hardware components such as depth camera has been used to get the information about the depth variation in the image to find an additional feature for comparison, and then developed a Convolutional Neural Network (CNN) for getting the results [12]. An innovative technique of not requiring a pre-trained model for executing the system was developed by the use of a capsule network and adaptive pooling [13]. Moreover, reducing the layers of CNN which uses a greedy method to do that and developing a deep belief network, it was found that this showed better results compared to other simple approaches [14]. Feature Extraction using SIFT and classification using Neural Networks (CNN) was developed, to get the desired results [15]. In one of the models, the images were converted into an RGB scheme, the information was developed using the motion depth channel and finally using 3D recurrent convolutional neural networks (3DRCNN) to develop the working system [16]. HMM models for sign language recognition is another prominent technique [7], [17], [18]. Since all these papers use different datasets of different numbers and types of images, the comparison among these becomes a tedious task. It was found that most of the projects utilized images that were nearly free of noise.

## 3. Background

Four major feature extraction techniques being used widely in computer vision and gesture recognition models have been developed in order to compare with the novel approach proposed. The techniques are as follows:

1. Histogram of Gradients (HOG)
2. Principal Component Analysis (PCA)
3. Local Binary Patterns (LBP)

### *3.1 Histogram of Gradients:*

Images were processed and features were used effectively by the utilization of Histogram of Gradients (HOG) technique. With its help, the feature descriptors were cloistered from the image. Since our dataset contains an immense number of images (87000 images), the feature descriptors were not calculated from each image using the conventional inbuilt values, but parameters had to be modified and changed to get a pragmatic number of features isolated descriptors on which further processing could be performed in efficient ways.

It is a technique that is widely used for object detection. Various techniques have been developed and on a comparison between some eminent ones like HAAR, upon researching it was found [19] that HOG outperformed HAAR, as HAAR would result in less accuracy and it will also generate more false positives. The problem with HOG is that if the object to be detected is entirely covering most of the image or if it is a very small part of the image, then HOG would not give desired results. But since the dataset taken had the hand section within the desired limits, HOG was selected. HOG utilizes mathematical calculation and it was essential to find out techniques of optimizing it [20].

As HOG is a feature descriptor, its main task is to take up the image and extract the essential part out of the image and then reject all the redundant parts left. To calculate the feature descriptors, the size of the image that is the width and height of the image along with the 3 channels are taken into account. Using this information, a feature vector is calculated and then this information is passed into the classifier to recognize. To calculate the HOG feature descriptor, first, the horizontal and vertical gradients have to be calculated. Some steps can also be taken to optimize the HOG approach like using gamma correction [21]. The image is sectioned into cells in which each cell pixel has 2 values attributed to it, which are direction and magnitude, this helps in the compact representation of the image.

*3.2 Principal Component Analysis (PCA):*

PCA is essentially a dimension reduction approach utilized over a large set of variables to convert it into a narrower feature vector which still retains most of the information in the large set. Therefore, our desired outcome of PCA was to display a feature space, from our dataset consisting of 200 × 200 images onto a smaller subspace which can represent our data well. Notable applications of PCA include Image Compression, Data Visualization, and Page Rank Algorithms.

*3.3 Local Binary Pattern:*

Local Binary Patterns is an effective texture operator that labels pixels of an image by defining a threshold and approximating the neighbouring pixels of concerned pixels [22]. The result is considered as a binary string. LBP popularly works with gray-scale images, such that only one depth is considered, however, this technique can we worked with coloured images such that RGB pixels can be used to produce three resultant binary numbers with could be represented as one feature vector with some modifications.

## 4. Methodology

A standard dataset has been used; it is approximately 1 gigabyte. The dataset contains 29 components each of 3000 images each, consisting 26 alphabets, nothing character (a special image which is used to recognize the absence of sign as default prediction), the space character and deletes character making it a total set of 87000 images The proposed methodology consists of four major steps - segmentation, feature extraction, generation of the histogram of visual vocabulary and classification as shown in the flowchart in Fig 2. In this proposed approach, a pre-processing technique was used to effectively obtain the feature descriptors in an image. The approach makes use of ORB (Oriented FAST and Rotated BRIEF) feature detection technique and K-means clustering algorithm to create the bag of features model for all descriptors.

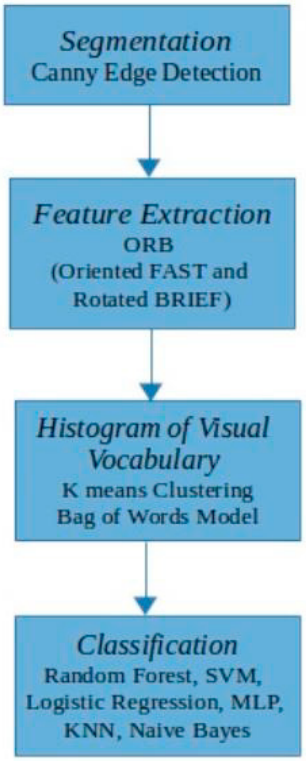


Fig 2. Flowchart of ORB Approach

### *4.1. Segmentation*

The RGB image (as shown in Fig 3) is first transformed into a grayscale image of a single channel. On the converted grayscale image, canny edge detection is used here to generate only strong edges present in the image (as shown in Fig 4). Canny edge detection is an effective and widely used technique for the detection of the edges in an image. It uses a multi-stage algorithm to distinguish sharp discontinuities or edges. This helps in reducing the background noise so that further techniques can be effectively applied.

### *4.2. Feature Extraction*

Feature Detection and Extraction is performed through Oriented Fast and Rotated Brief (ORB). ORB is an efficient feature detection and matching alternative to SIFT or SURF [23]. ORB is made up of two well-known descriptors FAST (Features from Accelerated and Segments Test) and BRIEF (Binary Robust Independent Elementary Features) with several modifications to boost the performance.

It firstly uses the FAST keypoint detector technique to compute key points along with the orientation which is calculated by computing the direction of the vector from the located corner point to the intensity weighted centroid of the patch. The orientation is not a part of FAST features so ORB uses a multi-scale image pyramid. The key points are effectively located at each pyramid level which contains the downsampled version of the image.

For computing descriptors, rBRIEF or rotated BRIEF technique is used since BRIEF performs poorly with rotation. In the BRIEF algorithm, the image is smoothened using a Gaussian kernel to reduce noise sensitivity and increasing the stability of descriptors. A matrix containing coordinates of feature pixels is defined and then using the orientation of patch, it's rotated(steered) matrix is calculated. Considering the key point orientation is consistent enough, the steered matrix can be used to compute essential descriptors with the help of a lookup table comprising

of predetermined BRIEF patterns.

ORB feature detector is used to detect patches (as shown in Fig 5) from the image and a 32-dimensional vector for each of the patches generated. Thus, for every image belonging to a set of a single class of sign images, a 32-dimensional vector of features is produced.

Fig 3. Original Image

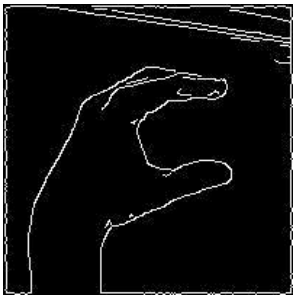

Fig 4. Edges detected

Fig 5. Features detected using ORB

### *4.3. Generation of Histogram of Visual Vocabulary*

For each class of sign image, a set of key descriptors with identical features is produced (as shown in Fig 6) through the above steps. This set is used to create a bag of feature models for all training images. K-means clustering is implemented to obtain K clusters having similar descriptors [24] (as shown in Fig 7). Each patch in the image is mapped to the closest cluster. Next, for each image, all the descriptors are mapped to their its closest cluster and a histogram of codewords is generated. A vocabulary of codewords is produced using a bag of feature histograms (as shown in Fig 8). In the proposed approach, K is taken to be 150, i.e. 150 codewords for each image are produced.

### *4.4. Classification*

In the final step of the technique, the above-generated feature vector of 150 codewords for each image is trained through various classifiers such as Random Forest, Naïve Bayes, Support Vector Machine, Logistic Regression, K Nearest Neighbour and Multi-Layer Perceptron. Accuracy for each model is calculated by passing them through the testing set (containing 17400 images).

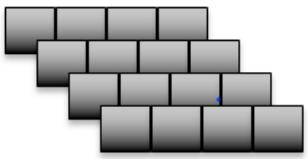

Fig 6. Descriptors

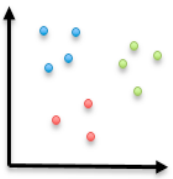

Fig 7. K-means Clustering

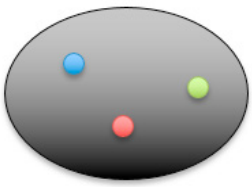

Fig 8. Vocabulary

## 5. Result and Discussion

To determine the suitable combination of decision-making pipeline with our method, various classification algorithms with ORB feature extraction as well as other feature extraction techniques in consideration. ORB

produces substantially accurate predictions as compared to other systems as its feature vector represents the visual features of various hand gestures in a compressed but informative manner. As compared to higher feature representation such as HOG and PCA, ORB is computationally efficient with inexpensive training of machine learning models both in terms of time and memory. Also, ORB appears to be a more natural approach as compared to LBP or PCA as it is intuitively based on encompassed visual characteristics of the image as opposed to texture analysis of the image or statistical study of data.

ORB presents better results as when it is combined with canny edge detection and bag of words, the feature vector provides better average results as the number of features has been reduced (using K-means clustering) and is much less than the features of both PCA and HOG. The following figures and comparison describe the performance of our solution as compared to other selected existing approaches (as shown in Fig 9).

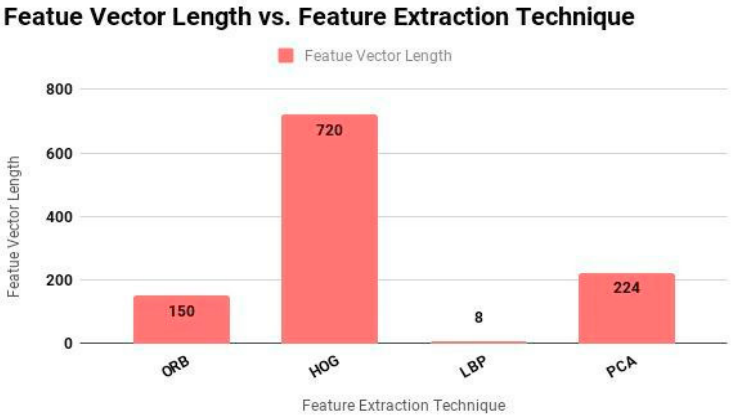


Fig 9. Feature Vector Graph

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 |

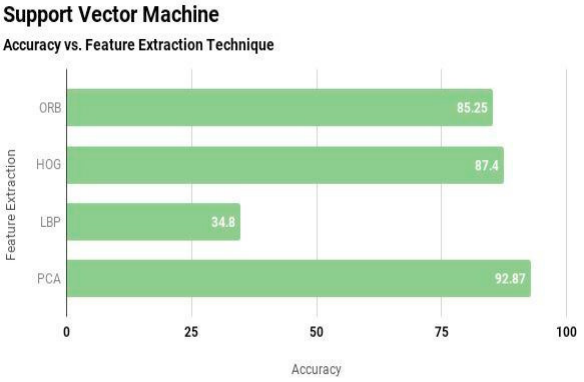


Fig 10. Accuracy graph for SVM

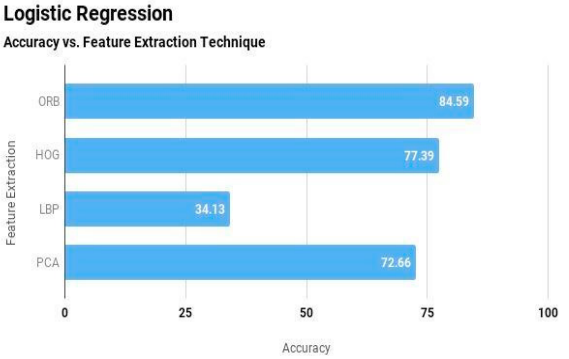


Fig 11. Accuracy graph for Logistic Regression

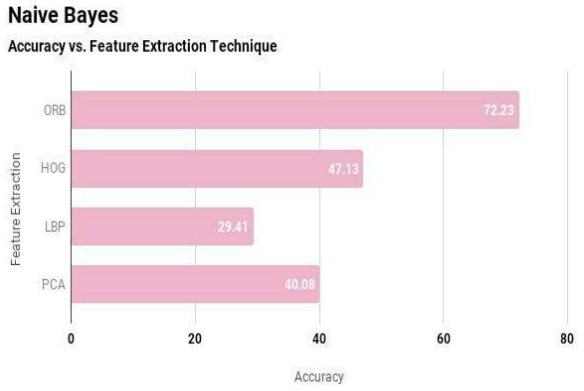


Fig 12. Accuracy graph for Naïve Bayes

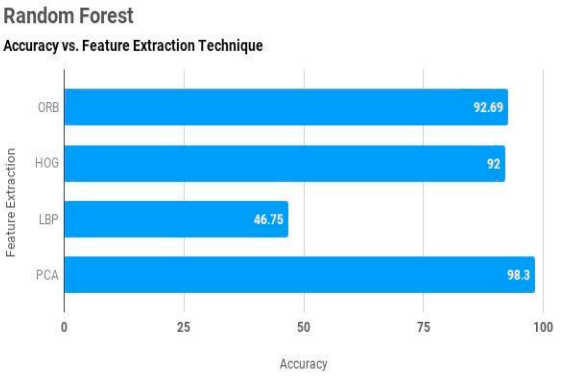


Fig 13. Accuracy graph for Random Forest

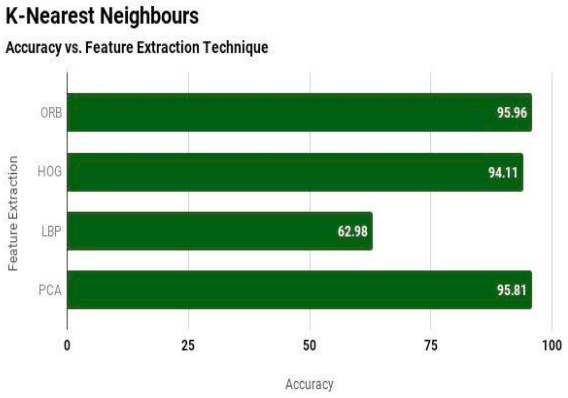


Fig 14. Accuracy graph for K-Nearest Neighbours

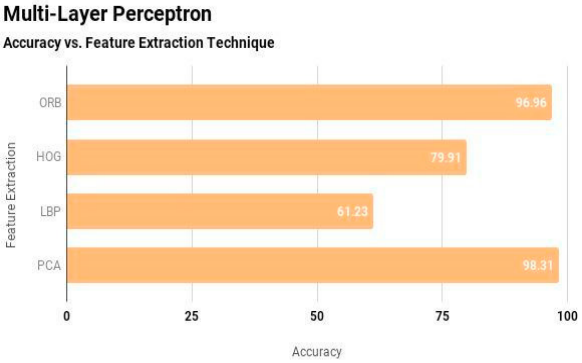


Fig 15. Accuracy graph for MLP

## 6. Conclusion and Future Scope

In the presented paper, the proposed technique of ORB feature extraction has been tested against many different pre-processing techniques such as Histogram of Gradients, LBP and PCA on the same dataset. These approaches have been successfully passed through various prominent classifiers such as KNN, SVM, Random Forest, Naïve Bayes, Logistic Regression and Multi-Layer Perceptron. The proposed technique outperforms all the other pre-processing techniques for Naïve Bayes, Logistic Regression and KNN classifiers while PCA outperforms all the other techniques for MLP, Random Forest and SVM classifiers. Although the approach gives substantially high accuracy for recognition of gestures.

The system is currently only tested against static gesture images and can be further extended to recognize dynamic gestures in videos in real-time. The system can be modified to recognize RGBD images detected from Kinetic Sensors. The paper can be further extended by using deep learning techniques such as modified convolutional neural networks, by optimizing through the use of quantum computing and evolutionary algorithms for feature selection further after feature extraction. The model can be trained on physical hand models containing sensors utilizing graph theory to provide extra data which can be studied for improving the accuracy.

## 7. References

[1] Anita Jadhav, Rohit Asnani, Rolan Crasto, Omprasad Nilande & Anamol Ponkshe. (2015). Gesture Recognition using Support Vector Machine. International Journal of Electrical, Electronics and Data Communication, 36-41.

[2] Ruslan Kurdyumov, Phillip Ho & Justin Ng. (2011). Sign Language Classification Using Webcam Images.

[3] Lahoti, S. K. (2018). Android based American Sign Language Recognition System with Skin Segmentation and SVM. International Conference on Computing, Communication and Networking Technologies (ICCCNT) (p. 6). IEEE.

[4] Rajeshree S. Rokade & Dharmpal D. Doye. (2015). Spelled sign word recognition using key frame. IET Image Processing, 381-388.

[5] Rajesh Kaluri & Ch. Pradeep Reddy. (2016). A framework for sign gesture recognition using improved genetic algorithm and adaptive filter. Cogent Engineering.

[6] Jaya Prakash Sahoo , Samit Ari & Dipak Kumar Ghosh. (2018). Hand gesture recognition using DWT and F-ratio based feature descriptor. IET Image Processing, 1780-1787.

[7] Tamer Shanableh, Khaled Assaleh & M. Al-Rousan. (2007). Spatio-Temporal Feature-Extraction Techniques for Isolated Gesture Recognition in Arabic Sign Language. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 641-650.

[8] Marwa Elpeltagy, Moataz Abdelwahab, Mohamed E. Hussein; Amin Shoukry; Asmaa Shoala & Moustafa Galal. (2018). Multi-modality-based Arabic sign language recognition. IET Computer Vision, 1031-1039.

[9] Islam, M. R. (2018). Hand Gesture Feature Extraction Using Deep Convolutional Network for Recognizing American Sign Language. International Conference on Frontiers of Signal Processing (ICFSP) (p. 5). IEEE.

[10] Harchaoui, Z., & Bach, F. (2007). Image Classification with Segmentation Graph Kernels. 2007 IEEE Conference on Computer Vision and Pattern Recognition.

[11] Ruiqi Zhao & Aleix M. Martinez. (2015). Labeled Graph Kernel for Behavior Analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1640-1650.

[12] Bin Xie, Xiaoyu He & Yi Li. (2018). RGB-D static gesture recognition based on convolutional neural network. The Journal of Engineering, 1515-1520.

[13] Jalal, M. A. (2018). American Sign Posture Understanding with Deep Neural Networks. International Conference on Information Fusion (FUSION) (p. 7). IEEE.

[14] Ma, M. X. (2018). Design and Analyze the Structure based on Deep Belief Network for Gesture Recognition. International Conference on Advanced Computational Intelligence (ICACI) (p. 5). IEEE.

[15] Shanta, S. S. (2018). Bangla Sign Language Detection Using SIFT and CNN. International Conference on Computing, Communication and Networking Technologies (ICCCNT) (p. 6). IEEE.

[16] Ye, Y., Tian, Y., Huenerfauth, M., & Liu, J. (2018). Recognizing American Sign Language Gestures from Within Continuous Videos. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2145-214509, IEEE.

[17] Daniel Kelly, John Mc Donald & Charles Markham. (2011). Weakly Supervised Training of a Sign Language Recognition System Using Multiple Instance Learning Density Matrices. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 526-541.

[18] Jeroen F. Lichtenauer, Emile A. Hendriks & Marcel J.T. Reinders. (2008). Sign Language Recognition by Combining Statistical DTW and Independent Classification. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2040-2046.

[19] Pablo Negri, X. C. (2007). Benchmarking haar and histograms of oriented gradients features applied to vehicle detection., (p. 6). Paris.

[20] Muhammed Jamshed Alam Patwary, S. P. (2015). Significant HOG-Histogram of Oriented Gradient Feature Selection for Human Detection. International Journal of Computer Applications, 5.

[21] Triggs, N. D. (2005). Histograms of Oriented Gradients for Human Detection. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). San Diego: IEEE.

[22] Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., & Feuston, B. P. (2003). Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. Journal of Chemical Information and Computer Sciences, 43(6), 1947–1958.

[23] Rublee, Ethan & Rabaud, Vincent & Konolige, Kurt & Bradski, Gary. (2011). ORB: an efficient alternative to SIFT or SURF. Proceedings of the IEEE International Conference on Computer Vision.

[24] F. Yasir, P. W. C. Prasad, A. Alsadoon and A. Elchouemi. (2015). SIFT based approach on Bangla sign language recognition. IEEE 8th International Workshop on Computational Intelligence and Applications (IWCIA), Hiroshima.

[25] Jin, C. M., Omar, Z., & Jaward, M. H. (2016). A mobile application of American sign language translation via image processing algorithms. 2016 IEEE Region 10 Symposium (TENSYMP).