ASL GESTURE RECOGNITION USING VARIOUS FEATURE EXTRACTION TECHNIQUES AND SVM

Siddharth Kaslay[1], Tejal Kesarkar[2], Kanchan Shinde[3]

[1,2,3]*Datta Meghe College of Engineering, Navi Mumbai*
[4]*Under the guidance of* ***Dr. Nita Patil*** *(Assistant Professor)*

---***---

Abstract— Gestures are one of the primary means of communication when people of different languages meet, and no one knows in which language they should express their feelings or needs. A gesture is defined as a movement of part of your body, especially your hands or head, to show what you mean or how you feel. One of the ways to express these gestures is through Sign Language. The sign language is a way of communication for deaf-dumb people and in sign language, each gesture has a specific meaning. Gesture recognition refers to the mathematical interpretation of human motions using a computing device. In this paper performance evaluation of features and classifier is presented. Color moment, Hu moment and Gray Level Co-occurrence Matrix (GLCM) are the features used in the proposed system. Support Vector Machine (SVM) is employed for classification. The proposed system is implemented American Sign Language (ASL) alphabet image dataset. The system performance is evaluated using predictive measures of precision, recall and f1-score. Our proposed system achieved and accuracy of 87%.

***Keywords*— Gestures, Sign Language, ASL, SVM**

1. INTRODUCTION

Dumb and Deaf people have difficulty in interacting with normal people so they depend on vision-based communication for interaction. If there is a common interface that converts the sign language to text, then gestures can be easily understood by other people, so research has been made for a vision-based interface system where dumb and deaf people can enjoy communication without really knowing each other's language. The aim is to develop a user-friendly human-computer interface (HCI) where the computer understands the human sign language. There are various sign languages all over the world, namely American Sign Language. American Sign Language (ASL) is a natural language that serves as the predominant sign language for deaf and dumb people.

Our aim is to develop a system where we provide input using bare hands to form gestures which will then be recognized by our system. In our project, we basically focus on producing a model which can recognize fingerspelling-based hand gestures and convert it into text.

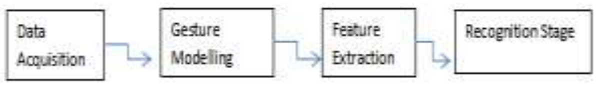

Fig-1: Block diagram of Gesture Recognition System

The general block diagram of Gesture Recognition System is shown in fig 1. It includes following broad steps.

Data Acquisition: This step is responsible for collecting the input data which are the hand, Face or Body gestures and classifier classifies the input tested gesture into required one of classes.

Gesture Modeling: This employed the fitting and fusing the input gesture into the model used; this step may require some pre-processing steps to ensure the successful unification.

Feature Extraction: This step is responsible for successful modelling of input data/gesture. The feature extraction should be smooth since the fitting is considered the most difficult obstacle. These features can be hand/palm/fingertips location, joint angles, or any emotional expression or body movement. The extracted features might be stored in the system at training stage as templates or maybe fused with some recognition devices such as neural network, HMM, or SVM.

Recognition Stage: This stage is considered to be a final stage for gesture system and the command/meaning of the gesture should be declared and carried out, this stage usually has a classifier that can attach each input testing gesture matching class.

The rest of the paper is organized as follows: Section II gives a brief overview of previous work in gesture recognition system. Section III presents the different kinds of feature extraction techniques used in this paper. It also provides information about the classifiers used. Section IV presents experimental results. Finally, Section V concludes the paper.

2. PREVIOUS WORK

Basically, there are two approaches for sign recognition vision based and sensor-based gesture recognition. Lots of study has been done on sensor-based approaches like gloves, wires, etc. [1] [2]. but due to disadvantage of wear it continuously is not possible, therefore further work is concentrated on Image based approaches [3]. Gesture recognition systems use various feature extraction techniques. Sana Jadwaa [4] presented a review of various feature extraction methods for hand gesture recognition. They discussed Zernike Moments, Local orientation histogram, Local Brightness, Local Brightness Binary Object Features for feature extraction. P.S.Edirisinghe et al. [5] used features like hu moments, edge histogram descriptor and circularity shape parameter values for understanding British Sign Language alphabet (BSL). Sulochana M. Nadgeri et al. [6] used orientation histogram for feature extraction. Orientation histogram is used to produce the shape of the object at different positions in the image. They have used it for recognition of American Sign Language alphabet. There are various algorithms used in Gesture Recognition systems. T. Yang et al. [7] used Hidden Markov Model (HMM) for the classification of the gestures. This model deals with the dynamic aspects of gestures. Gestures are extracted from a sequence of video images by tracking the skin-color blobs corresponding to the hand into a body– face space centered on the face of the user. Pujan Ziaie et al. [8] used Naïve Bayes Classifier which is an effective and fast method for static hand gesture recognition. It is based on classifying the different gestures according to geometric based invariants which are obtained from image data after segmentation. Thus, unlike many other recognition methods, this method is not dependent on skin color. The gestures are extracted from each frame of the video, with a static background. Nagashree R N et al. [9] used Support Vector Machines for classification of gestures. They used a predefined dataset of gestures containing images of 1 to 20 numbers.

3. PROPOSED SYSTEM

The proposed system is divided into two phases training phase and testing phase.

Training Phase:

In training phase, we have extracted the features of the images from the ASL alphabet dataset. Then we have normalized the extracted features. We have used the SVM classifier to train these features.

Testing Phase :

The initial step is to capture the hand gesture via webcam. The next step is frame extraction. The next step is feature extraction. The next step is Classification using SVM. Finally, the gesture is recognized and converted to text and the text is displayed.

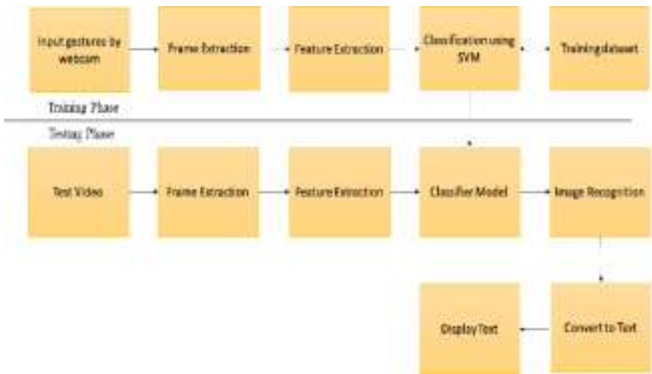

Fig.-2: Block Diagram of Proposed System

1) Frame Extraction:

Each individual image is called a frame which is where you see the term frames per second (FPS). A video file on a computer simply stores all the frames together and plays them in order, and the total frames stored for a typical movie reaches into the hundreds of thousands.

In our system, we have extracted the frames from the video and provided them as input to the next block. These frames were helpful for extracting features for classification.

2) Feature Extraction:

Feature extraction efficiently represents interesting parts of an image as a compact feature vector. This approach is useful when image sizes are large and reduced feature representation is required to quickly complete a task such as an image matching and retrieval. The feature extraction methods which we have used in our system are as follows:

1.Color feature: Color acts as a discriminative feature for understanding image or keyframe content. Color feature is independent of image size and orientation. For example, blue color is prominent in beach or sky concept whereas brown color is dominating in desert or sunset concepts.

a) Color moment

Color moments are measures that can be used to differentiate images based on their features of color. Once calculated, these moments provide a measurement of color similarity between images. These values of similarity can be compared to the values of images indexed in a database for tasks like image retrieval.

Equation 1 and 2 are the formulae to calculate mean and standard deviation. Color moment returns a six-dimensional feature vector.

$$E_i = \sum_{j=1}^{N} \frac{1}{N} p_{ij} \tag{1}$$

$$\sigma_i = \sqrt{\left(\frac{1}{N} \sum_{j=1}^{N} (p_{ij} - E_i)^2\right)} \tag{2}$$

b) Hu moment

The most notable are Hu Moments which were used to describe, characterize, and quantify the shape of an object in the image. Hu Moments are normally extracted from the silhouette or outline of an object in an image. By describing the silhouette or outline of an object, we were able to extract a shape feature vector to represent the shape of the object. We then compared two feature vectors using a similarity metric or distance function to determine how "similar" the shapes were. Hu moments return a seven-dimensional feature vector.

2. Texture feature: Texture is an important visual feature used in domain-specific applications. It can give us information about the content of an image efficiently. It is a repeated pattern of information or arrangement of the structure with regular intervals. It quantifies the properties such as smoothness, coarseness and regularity in an image. The texture feature used in our system is:

a) Gray Level Co-Occurrence Matrix (GLCM):

A statistical method of examining texture that considers the spatial relationship of pixels is the gray-level co-occurrence matrix (GLCM), also known as the gray-level spatial dependence matrix. The GLCM functions characterized the texture of an image by calculating how often pairs of the pixel with specific values and in a specified spatial relationship occur in an image, creating a GLCM, and then extracting statistical measures from this matrix. GLCM has four properties namely: Contrast, Correlation, Energy, Homogeneity. GLCM returns a four-dimensional feature vector.

3) Classifier

Classification is the process of predicting the class of given data points. Classification algorithms help us in predicting the accuracy of our training data. In our system, we have used Support Vector Machines (SVM) for classification.

1. Support Vector Machine

A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperplane. In other words, given labelled training data (supervised learning), the algorithm outputs an optimal hyperplane which categorizes new examples. The output of a SVM can be a classification map that contains class labels for each pixel (or object), or a probability map that contains probability estimates for each pixel (or object) to belong to the assigned class. In this method, the one-versus-rest approach is used for the multiclass SVM output. The one versus-rest approach builds n SVMs (where n is the number of concept classes), each of which is able to separate one class from all the others. In this experiment, one-vs-rest multi-class classifier provided by sklearn module in python is used. SVMs are having kernel with parameters, box constraint constant C and a third constant depending on the type of kernel function used. The parameter C is the regularization parameter fine-tuned with the help of cross-validation. It defines the tradeoff between margin maximization and error minimization. Radial Basis Function (RBF) is used as kernel function. For SVM parameter tuning, normalized training and test data has been used. SVM is trained using training dataset with parameters obtained after cross validations.

a) Dataset Description:
We have used American Sign Language(ASL) alphabet Dataset. Dataset contain two folders one is training dataset and other is testing dataset. The training dataset contains 3000 images for each alphabet along with some additional movements. In total, it contains 87000 images.

b) Evaluation Metrics: The benchmark metric for classifier evaluation includes classification precision, classification recall and F1-score. They are defined as:

i) Precision: Precision defines the ratio of the total number of the retrieved video or keyframes which matches the query to the sum of the relevant videos matching and not matching the query from the video database. The precision can be accurately expressed by equation (3)

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (3)$$

ii) Recall: Recall defines the ratio of the total number of the retrieved video or keyframes which matches the query to the sum of the retrieved videos matching the query in the video database. Equation (4), defines the precise expression for the recall parameter.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (4)$$

iii) F1-score: F1-score defines the harmonic mean, weighted average of the precision and the recall parameters. Since both measures are important, usually classifier with f1-score is followed. F1-score is expressed by equation (5).

$$\text{F1 Score} = 2 \times \frac{\text{Precision}*\text{Recall}}{\text{Precision+Recall}} \quad (5)$$

Where TP is the set of true positive samples that are related to the corresponding semantic concept and are classified correctly, TN is the set of true negative samples that are irrelevant to the corresponding semantic concept and are classified correctly, FP is the set of false positive samples that are related to the corresponding semantic concept but are miss classified.

4. EXPERIMENTAL RESULTS

Performance metrics for each alphabet

Alphabet	Class	Precision	Recall	F1-Score	Support
A	1.0	0.91	0.72	0.80	600
B	2.0	0.90	0.89	0.90	600
C	3.0	0.84	0.97	0.90	600
D	4.0	0.90	0.91	0.90	600
E	5.0	0.87	0.84	0.85	600
F	6.0	0.94	0.91	0.93	600
G	7.0	0.96	0.84	0.90	600
H	8.0	0.87	0.98	0.92	600
I	9.0	0.89	0.92	0.90	600
J	10.0	0.92	0.98	0.95	600
K	11.0	0.94	0.95	0.95	600
L	12.0	0.91	0.91	0.91	600
M	13.0	0.86	0.93	0.89	600
N	14.0	0.88	0.93	0.90	600
O	15.0	0.85	0.90	0.87	600
P	16.0	0.93	0.94	0.94	600
Q	17.0	0.80	0.93	0.86	600
R	18.0	0.89	0.73	0.80	600
S	19.0	0.84	0.89	0.86	600
T	20.0	0.92	0.74	0.82	600
U	21.0	0.80	0.71	0.75	600
V	22.0	0.80	0.78	0.79	600
W	23.0	0.82	0.90	0.86	600
X	24.0	0.90	0.83	0.87	600
Y	25.0	0.82	0.74	0.78	600
Z	26.0	0.81	0.82	0.82	600
DELETE	27.0	0.85	0.88	0.87	600
NOTHING	28.0	1.00	1.00	1.00	600
SPACE	29.0	0.80	0.92	0.85	600

Fig-3: Performance measures for each class

Confusion Matrix of the SVM classifier

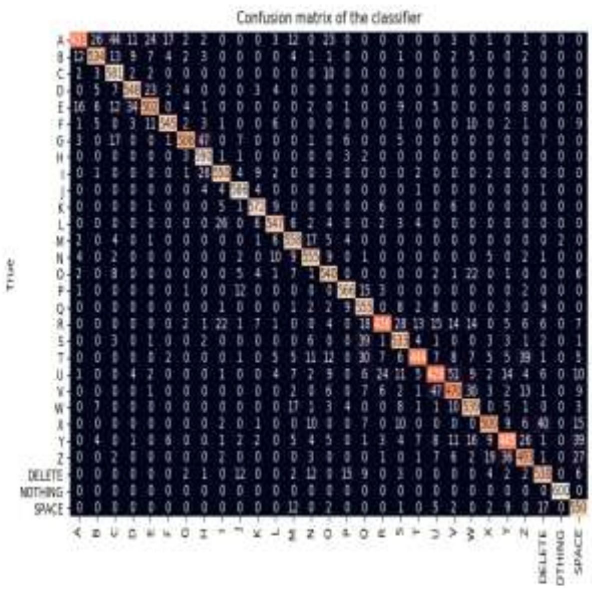

Fig-4: Confusion matrix of SVM Classifier

5. CONCLUSION

Our Project Gesture Sign Language Recognition is aimed to be used by dumb and deaf (disabled) people to communicate with normal people. We aim to solve this problem by taking input of gesture through webcam which is widely available for use. Our system uses the Frame Extraction technique. In Frame Extraction the frames were extracted from the video and the output from frame extraction was given as an input to feature extraction. In feature extraction data was extracted on the basis of color, shape and texture. The SVM classifier was used to classify the ASL alphabet dataset. SVM classifier models were created. With SVM, we achieved an accuracy of 87 percent to recognize the gesture. In the testing phase, the image was captured from the webcam. The gesture was extracted from the image by providing a graphical user interface to the user. This GUI will allow the user to pause the frame of the webcam and crop the hand gesture which is required. The features of the image are extracted. These features are normalized and provided to SVM for prediction. The trained SVM classifier returns the matching alphabet to the gesture.

6. REFERENCES

[1] Luigi Lamberti and Francesco Camastra, "Real-Time Hand Gesture Recognition Using a Color Glove", International Conference on Image Analysis and Processing,2011, Italy.

[2] Dalal Abdulla, Shahrazad Abdulla and Rameesa Manaf," Design and Implementation of a Sign-to speech/Text System for Deaf and Dumb People ", 2016 5th International Conference on Electronic Devices, Systems and Applications (ICEDSA), January 2017, United Arab Emirates.

[3] Apoorva M A, Harshitha M S, Chaitra S, Akshitha V, Rajath A N," An Efficient and Robust System for Hand Gesture Recognition and Interpretation", International Journal of Engineering Research & Technology (IJERT),2017.

[4] Sana'a Khudayer Jadwaa, "Feature Extraction for Hand Gesture Recognition Review," International Journal of Scientific & Engineering Research, Volume 6, Issue 7, July 2015.

[5]E.M.P.S. Edirisinghe, P.W.G.D.Shaminda, I.D.T.Prabash, N.S.Hettiarachchige, L.Seneviratne and U.A.A.Niroshika,"Enhanced feature extraction method for hand gesture recognition using support vector machine," IEEE 8th International Conference on Industrial and Information Systems, ICIIS 2013, Aug. 18-20, 2013, Sri Lanka.

[6] Sulochana M. Nadgeri, Dr. S.D. Sawarkar, Mr.A.D.Gawande, "Hand gesture recognition using CAMSHIFT Algorithm", Third International Conference on Emerging Trends in Engineering and Technology,IEEE , 2010, India.

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

[8] Pujan Ziaie, Thomas M¨uller , Mary Ellen Foster , and Alois Knoll"A Naïve Bayes Classifier with Distance Weighting for Hand-Gesture Recognition ",Munich,Dept. of Informatics VI, Robotics and Embedded Systems,Germany.

[9] Nagashree R N, Stafford Michahial, Aishwarya G N, Beebi Hajira Azeez, Jayalakshmi M R, R Krupa Rani, "Hand gesture recognition using support vector machine", The International Journal of Engineering and Science (IJES), June 2015.