



## **REAL-TIME IMPLEMENTATION FOR DIGIT RECOGNITION USING RASPBERRY PI**

Snehal R. Kalbhor<sup>1</sup>, Ashwini M. Deshpande<sup>2</sup>

**Abstract-** The identification of sign language is a bridge between the normal people and people with hearing and speech problems. Most of the people are unable to recognize the gestures made by these people. Computer vision and machine learning techniques can be used to remove this barrier by creating a working model which automatically detect the gesture performed by them. In this paper, a real-time implementation for digit recognition based on the Raspberry Pi with camera module is presented. Raspberry Pi is programmed with Python programming language supported with OpenCV library. In this system template matching, Linear Discriminant Analysis-logistic Regression (LDA-LR), Principal Component Analysis Logistic regression (PCA-LR), K-Nearest Neighbors (K-NN) and Support Vector Machine (SVM) learning based methods are used for digit recognition and their performance is compared. In template matching method, the template of the gesture is matched with the database image by using correlation function while in LDA-LR, PCA-LR, KNN and SVM based systems, the distance between the palm centroid and fingertip is considered as a feature and the features are trained with LR, KNN and SVM Models. The proposed system achieved the accuracy of 63.76% for LDA-LR, 72.46% for PCA-LR, 77.77% for KNN and 83.09% for SVM approach. Template matching algorithm shows 100% matching result only if it is compared with exactly same template.

**Keywords** –Contour, Convex hull, K-NN, LDA-LR, PCA-LR, SVM, OpenCV, Raspberry Pi

### **1. INTRODUCTION**

Sign language recognition is one of the important fields of research. The hand gesture is one of the methods used in sign language for non-verbal communication. It is commonly used by deaf and dumb people who have hearing or speech problems to communicate with each other or with normal people. Advances in computer vision and machine learning techniques can use to develop a working prototype which automatically detects the gesture performed by the impaired people.

Various techniques are developed for static and dynamic sign language recognition. The techniques are designed on the basis various features of the hand. Contour, convex hull, centroid, moments, hand shape, skin color, hand motion and palm image extraction [1] are the various features of the hand. These extracted features are classified using various classifiers such as SVM, K-NN, NN, ANN and PCA-LR.

Skin color is a convenient feature because it is human skin color [2] which encountered first while detecting the hand gesture. Also, color is selected as a feature because of its computational simplicity. YCbCr and HSV are the best skin color model for finding the hand region from the image. But, the variability of skin color under different lighting conditions may lead to failures in detecting skin color. Sometimes light skin color is harder to detect in bright lighting condition while dark skin color might be similar to their background in dim lighting condition. So, for further work, we use additional features such as contour and convex hull. From that, we extract the features like centroid, fingertip and Euclidean distance. Then PCA-LR, LDA-LR, K-NN and SVM are used for classification purpose.

### **2. LITERATURE SURVEY**

Amiraj Dhawan et al. [3] implemented work on hand detection techniques for human-computer interaction. They calculated the contour, convex hull and convexity defects of the input image and extracted the features like finger counting, hand orientation and finger tracking. But this technique does not have good background subtraction. This is a technique used in gesture-controlled robot and picks place robot.

Ashish S. Nikam et al. [4] proposed gesture recognition technique for sign language recognition. They designed real-time system for recognition of hand gesture. They used a hand-based feature like orientation, the center of mass, centroid, finger status and thumb in the position of raised or folded finger of the hand. This technique recognizes the gesture 1 to 5.

<sup>1</sup> Department of Electronics and Telecommunication Engineering, MKSS's Cummins College Of Engineering for Women, Pune, Savitribai Phule Pune University, Maharashtra, India

<sup>2</sup> Department of Electronics and Telecommunication Engineering, MKSS's Cummins College Of Engineering for Women, Pune, Savitribai Phule Pune University, Maharashtra, India

HsiangYaech et al. [5] designed real-time dynamic hand gesture recognition. They used YCbCr skin color model for detection of hand. Using the contour-convex hull technique, they found the tips of the finger and measure the angle between the finger spacing. The accurate recognition rate achieved more than 95.1%

Mandeep kaur Ahuja et al. [6] designed system for PCA based static hand gesture recognition. Skin color and thresholding for feature extraction. In this method, template matching is done by using PCA. The system is implemented for recognizing 4 gestures with 5 different poses. It shows average accuracy rate 91.25 %.

GaorArva et al. [7] proposed work on embedded video surveillance detection using a raspberry pi. Harr cascade and local binary pattern are used for detection of face and hand. As a result, they concluded that object detection using LBP classifier is three times faster than the Harr feature based.

Ali A. Abed et al. [8] proposed a python-raspberry pi based system for hand gesture recognition. Contour, convex-hull and convexity defects are used as features. They used five actions of the finger for controlling a mobile robot in real time.

Amitkumar Shinde et al. [9] proposed Marathi sign language recognition. The design system is based on computer vision. They used features like the center of gravity, Euclidean distance and HSV skin color model for hand image extraction. Lastly, template matching is used for comparing input image with database image. Their system gives 100% recognition results for offline detection.

### 3. METHODOLOGY

#### 3.1 Template Matching based digit gesture recognition

Template matching is a simple method for digit gesture recognition. Here, mainly template matching algorithm is used for static hand gesture recognition. It is used to find the location of the template in the large image. The RGB image is first converted to a grayscale image. This grayscale image shows how each template image pixel matches with input image pixel. The database stores the appropriate number of gesture templates. Next, template matching algorithm is applied. Steps used for template matching are as follows:

Algorithm:

Read the input image

Assign the template folder path

Assign the label for each template image

Convert input and template RGB image into a grayscale image

Apply template matching algorithm using cv2.matchtemplate()

Finding correlation between template and input image by using cv2.TM\_CCOEFF\_NORMED

Set threshold value for correlation.

Compare matching value  $\geq$  threshold value

Initially set flag=1

When flag=1 then the template is a perfect match

The keyword is detected.

The threshold value depends on the accuracy with which we want to detect the template in the input image. Threshold value should be high (ideally 1) for the identical template.

#### 3.2 Contour-based digit gesture recognition

A general block diagram of the digit gesture recognition system is shown in Fig. 1. It contains various sub-blocks such as frame extraction, pre-processing, feature extraction and database storage in training phase. In testing phase, additionally, feature database, sign recognition and its corresponding label display are the blocks.

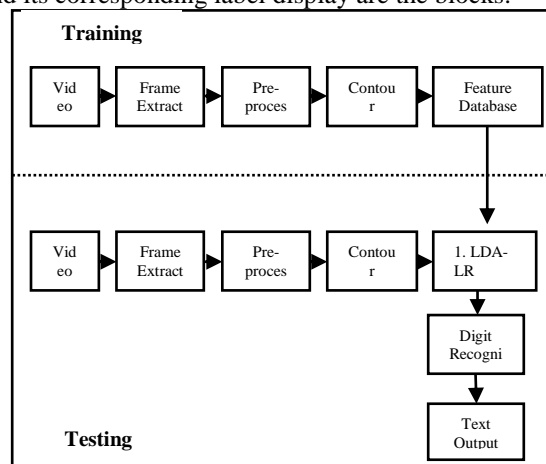


Fig. 1. Block diagram for contour-SVM based digit gesture recognition

In the hand gesture signal recognition system for digit recognition, a video is captured from the raspberry pi camera. This video is then passed sequentially through various blocks as shown in Fig. 1 for digit recognition.

Contours are those curves that are included in the continuous points with the boundary. The points are of the same intensity. The contour is drawn around the hand image. The convex hull is established in the Euclidean space. These points are connected to contours. The convex hull is drawn around the contour. Contours points lie within the convex hull. Convex hull works as envelop around the hand [10].

Steps used in contour-based digit gesture recognition are as follows-

Capture the video using USB Samsung camera having resolution 720X480  
 Converted recorded video into frames  
 Read frames  
 Apply pre-processing techniques.  
 Extract hand region using the bounding box  
 Convert RGB image to grayscale image  
 Apply thresholding (Otsu's method) to obtain a binary image  
 Extract the Region of Interest(ROI)  
 Apply a Gaussian filter for blurring the image  
 Find the contours  
 Contours, hierarchy = cv2.findContours (thresh1, cv2.RETR\_TREE, cv2.CHAIN\_APPROX\_SIMPLE)  
 Find the largest contour  
 Draw convex hull  
 Hull = cv2.convexHull (cnt)  
 Find Moment for finding the centroid of hand palm.  
 Find convexity defects  
 Apply cosine rule to find angles for all defect between the fingers  
 Find the fingertip position  
 Find the distance between centroid and fingertip point and stored as a feature  
 Train the LDA-LR, PCA-LR, K-NN and SVM model using training feature set  
 Load the testing gesture  
 Extract the features  
 Test with LDA-LR, PCA-LR, K-NN and SVM model  
 Display the recognized digit

Feature extraction is carried out in training phase. During the training phase, extracted features are stored in the database. In the testing phase, different classifiers such as LDA-LR, PCA-LR, K-NN and SVM are used for classification of features extracted from testing images. In the feature extraction stage, we have more information about the fingertips, finger counting the palm center position. For good feature detection of hand, we have used the relative distance between the fingertips and the hand center by using following formula.

$$\text{Euclidean distance} = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2} \quad (1)$$

Point (x1, y1) is a fingertip position and (x0, y0) is the center of hand palm. Depending upon the number of defects and length we classify the corresponding gesture using LDA-LR, PCA-LR, K-NN and SVM. The output of recognition system will be displayed in the form of labels (1-5 digits).

### 3.3 Linear Discriminant Analysis-Logistic Regression (LDA-LR)

LDA is linear transformation technique [11]. It is most commonly used for dimensionality reduction. Dimensionality reduction helps to avoid overfitting problem. LDA is described as "supervised" algorithm. It is used to find the axes that maximize the separation between two classes. Logistic regression (LR) algorithm is used for solving multiclass classification. LR model takes the feature values and calculates the probability using softmax function.

### 3.4 Principal Component Analysis-Logistic Regression (PCA-LR)

PCA is linear transformation technique [11]. It is most commonly used for dimensionality reduction. It is an "unsupervised" algorithm. It is used to find the axes that maximize the variation in the dataset. Logistic regression (LR) algorithm used for solving multiclass classification. LR model takes the feature values and calculates the probability using softmax function.

### 3.5 K- Nearest Neighbors (K-NN)

K-NN is supervised and non-parametric machine learning algorithm [12]. There are only two parameters required implementing the K-NN i.e. k and distance function. It calculates the distance between a new data point and all other training

data points. Then it selects the k-nearest data points. Then, it assigns the data points to the class to which the majority of the k data points fit in.

### 3.6 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a set of supervised learning techniques [10]. It is used for multiclass classification purpose. It is a flexible non-parametric machine learning algorithm. SVM algorithm implemented using kernel. Radial basis function (RBF) used as the kernel. It is used to avoid numerical difficulties. SVM model takes the feature values. Then, predict the target values for test data attributes.

## 4. EXPERIMENTAL RESULTS

### 4.1 Database

The database for 0-9 digit is taken from Sign-language-digit dataset (SLD) [12]. Some samples of the database are as shown in Fig. 2.



Fig. 2. SLD database for 1-5 signs

Details of the dataset are as follows:

Image size: 100 x 100 pixels

Color space: RGB

Number of classes: 10 (Digits: 0-9)

Number of participant students: 218

Number of samples per student: 10

Software Platform

Implementation of digit recognition system is carried out using OpenCV and Python on Intel core TM i5 Processor.

### 4.2 Results using template matching

Offline template matching results are as shown in Fig. 4. Template matching algorithm gives 100% result when it is compared with exactly same template. But, when the template is compared with other same gesture image given, accuracy is only 4.32% on SLD database [13].

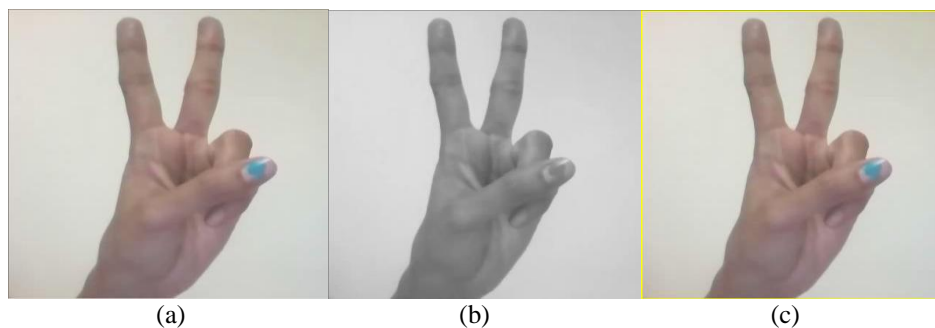


Fig. 3. Template matching based digit gesture recognition method: (a) Input image, (b) Grayscale image, (c) Final detected image

Fig. 3 shows the input image and its grayscale image. Fig. 3 (c) shows the detected template of the image corresponding to the input image extracted from the database. Detected digit gesture label is displayed as shown in Fig. 4.

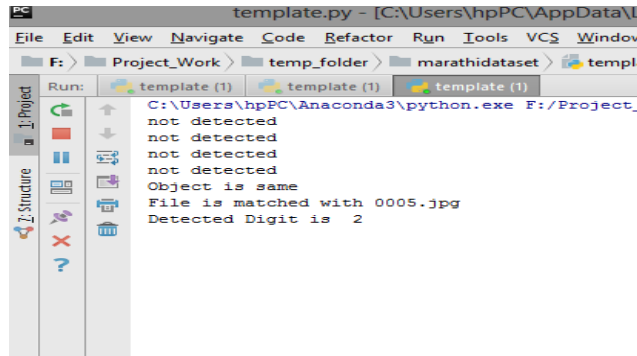


Fig. 4. Detected digit gesture output label (Template matching result)

Observation shows template matching is variant to rotation and scaling. The output of the system depends on how a gesture is presented by the person. It also depends on the use of proper training gestures. The templates are then stored in the database and used for recognition purpose. So that other users will have lower recognition rate than original person.

#### 4.3 Results using the contour-based method

Here, we compare the recognition performance using four classifiers such as LDA-LR, PCA-LR, K-NN and SVM. The results for all four techniques are given in Table I. It shows that SVM classifier gives better accuracy than other methods. The LDA-LR is used to find the components that maximize the separation between different classes. Feature values for some classes are nearly close to each other which gives less accuracy than other methods. The PCA-LR method is used to find the components that maximize the variance in our dataset which cause the misclassification.

K-NN algorithm does not work well with high dimensional data. Because it is difficult to calculate the distance in each dimension. In SVM classifier, RBF kernel is used for non-linear classification. So that, SVM classifies the multiple classes.

Table I. Comparison of four methods for 1-5-digit recognition

No.	Method	Digit recognition accuracy
1.	Contour-LDA-LR	63.76%
2.	Contour-PCA-LR	72.46%
3.	Contour-K-NN	77.77%
4.	Contour-SVM	83.09%

Apart from the database, system performance is tested on the live video input. Table I shows that the Contour-SVM based digit recognition system gives the highest accuracy 83.09% than other methods. Hence, we implemented the contour-SVM method on Raspberry Pi for digit gesture recognition.

#### 4.4 Hardware Platform

A pictorial view of hardware setup is as shown in Fig. 5. The hardware platform of the system consists of Raspberry Pi 3B model and Samsung USB camera having resolution 720x480. There is an onboard Raspbian operating system for supporting the real-time implementation [7].



Fig. 5. A pictorial view of hardware setup

The proposed method counts the number of fingertips from convex defect points between the corresponding fingers. It also finds the distance between palm center and fingertips. From these features, the system recognizes the digit gesture for 1-5 digits from the raspberry pi board.

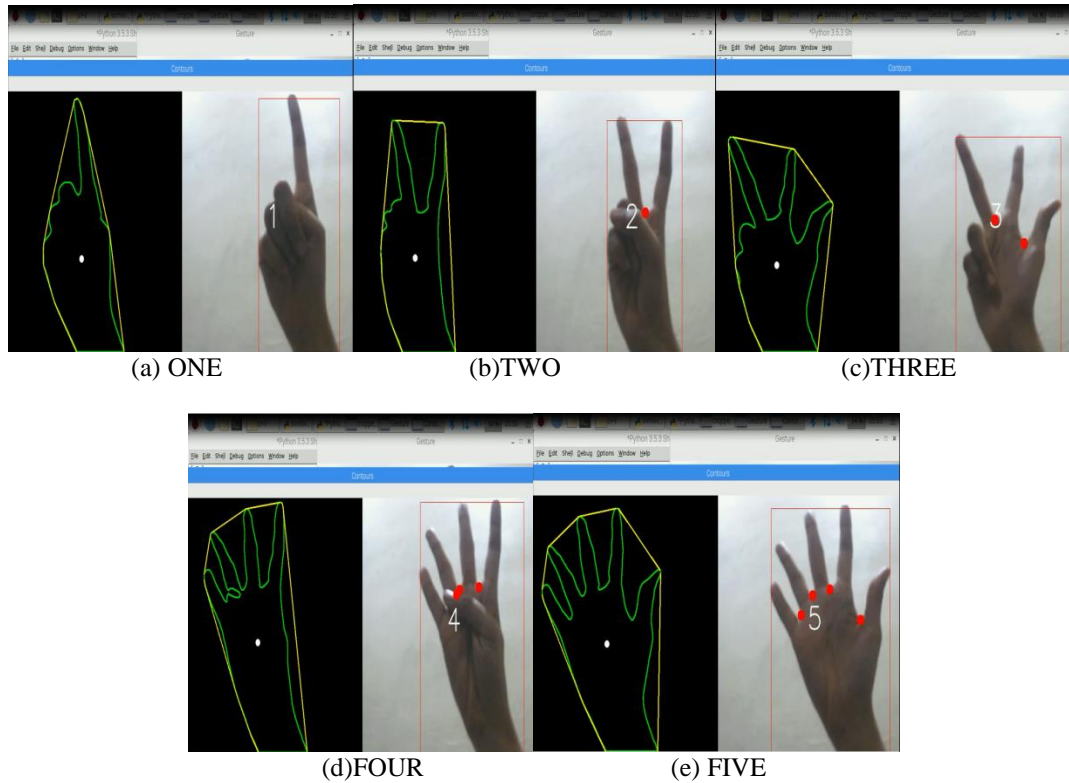


Fig. 6. Results of digit recognition on Raspberry Pi using contour-SVM method

Fig. 6 shows the results of digit recognition system on the raspberry pi using contour-SVM method. Table II shows the result of recognition rate for 1-5-digit gesture recognition using the contour-SVM method on the raspberry pi. It shows that the average accuracy is nearly 82% for digit gesture recognition.

Table II. Recognition Rate of 1-5 digits gestures on raspberry pi

Sign	Number of Images	Recognized gesture images	Recognition Rate
1	10	7	70%
2	10	10	100%
3	10	9	90%
4	10	8	80%
5	10	7	70%
		Average	82%

## 5. CONCLUSION

This paper demonstrates a real-time implementation of hand gestures for digit recognition using Raspberry Pi. An experimental evaluation is carried out for computing recognition accuracy using template matching, contour with LDA-LR, PCA-LR, K-NN and SVM based recognition methods. Template matching method does not work for rotated and different scale version of images. As size and shape of object changes template will give a false match.

The contour-LDA-LR method does not support for multiple classes. Thus, it gives accuracy 63.76%. The contour-PCA-LR method finds the axes with maximum variances which ignore the class labels. It gives the accuracy 72.46%. Contour-K-NN based method gives accuracy 77.77% with  $k=5$ .

The contour-SVM based method gives the accuracy of 83.09% for software implementation. SVM is very powerful for multiclass classification. Hence, when we implemented it on raspberry pi for 1-5 digits gesture recognition, it gives 82% accuracy on Raspberry Pi.

## 6. FUTURE SCOPE

The proposed system is implemented for recognition of five different digits. It can be extended for the entire set of digits as well as other language constructs (words, sentences etc.). Support Vector Machine can be modified for reduction of complexity which can lead to computation time. Recognition for occluded images is one of the challenging cases and it can be handled as a future scope.

## 7. REFERENCES

- [1] Imran Hussain, Anjan Kumar Talukdar and Kandarpa Kumar Sarma, "Hand gesture recognition system with real-time palm tracking", Annual IEEE India Conference (INDICON) 2014.
- [2] S. Kolkur, D. Kalbande, P. Shimpi, C. Bapat and J. Jatakia, "Human skin detection using RGB, HSV and YCbCr color models", Advances in Intelligent Systems Research, 2017.
- [3] Amiraj Dhawan, Vipul Honrao, "Implementation of hand detection based techniques for human-computer interaction", International Journal of Computer Applications, June 2013.
- [4] Ashish S. Nikam, Aarti G. Ambekar, "Sign language recognition using image-based hand gesture recognition techniques", IEEE Online International Conference on Green Engineering and Technologies (IC-GET), November 2016.
- [5] Hsiang-Yueh Lai, Han-Jheng Lai, "Real-time dynamic hand gesture recognition", Computer, Consumer and Control (IS3C), 2014 International Symposium on, June-2014.
- [6] Mandeep Kaur Ahuja and Amardeep Singh, "Static vision-based hand gesture recognition using principal component analysis", IEEE 3rd International Conference on MOOCs, Innovation and Technology in Education (MITE), October 2015.
- [7] Gabor Arva, Tomas Fryza, "Embedded video processing on raspberry pi", 27th International Conference Radio elektronika (RADIOELEKTRONIKA), April 2017.
- [8] Ali A. Abed, Sarah A. Rahman, "Python-based raspberry pi for hand gesture recognition", International Journal of Computer Applications, September 2017.
- [9] Amitkumar Shinde, Ramesh Kagalkar "Advanced Marathi sign language recognition using computer vision", International Journal of Computer Applications, May 2015.
- [10] Houssein Lahiani, Mohamed Elleuch, Monji Kherallah, "Real-time hand gesture recognition system for Android devices", 15th International Conference on Intelligent Systems Design and Applications (ISDA), 2015 IEEE.
- [11] Aleix M., Martõ Áñez, Avinash, C. Kak, "PCA versus LDA", IEEE Transactions on Pattern Analysis and Machine Intelligence, February 2001.
- [12] Shruthi S, Sona KC and Kiran Kumar S, "Classification on hand gesture recognition and translation from Real Time video using SVM-KNN", International Journal of Applied Engineering Research ISSN, 2016.
- [13] <https://www.kaggle.com/ardamavi/sign-language-digits-dataset/version/2#>