Journal of Ambient Intelligence and Humanized Computing (2021) 12:6793–6800
https://doi.org/10.1007/s12652-020-02314-2

ORIGINAL RESEARCH

Development of hand gesture recognition system using machine learning

Priyanka Parvathy[1] · Kamalraj Subramaniam[2] · G. K. D. Prasanna Venkatesan[3] · P. Karthikaikumar[4] · Justin Varghese[4] · T. Jayasankar[5]

Received: 10 May 2020 / Accepted: 9 July 2020 / Published online: 16 July 2020

Abstract

Human computer interaction (HCI) systems are increasing due to the demand for non-intrusive methods for communicating with machines. In this research article, vision based hand gesture recognition (HGR) System has been proposed using machine learning. This proposed system consists of three stages: segmentation, feature extraction and classification. The developed system is to be trained and tested using Sebastian Marcel static hand posture database which is available online. Discrete wavelet transform (DWT) along with modified Speed Up Robust Feature extraction technique has been used to extract rotation and scale invariant key descriptors. Then Bag of Word technique is used to develop the fixed dimension input vector that is required for the support vector machine. The classification accuracy of class 2 and class 4 which corresponds to the 'No' and 'grasp' gesture has reached 98%. The overall classification accuracy of the HGR system using SVM classifier is 96.5% with a recognition time of 0.024 s. Due to fast recognition time, this system can be employed in real time gesture image recognition system. Our HGR system addresses the complex background problem and also improves the robustness of hand gesture recognition.

Keywords Wavelet transform · Morphological operation · Support vector machine · Bag of word · Key descriptor · Hand gesture recognition

✉ Priyanka Parvathy
csepriyanka2001@yahoo.com

Kamalraj Subramaniam
kamalrajece@gmail.com

G. K. D. Prasanna Venkatesan
prasphd@gmail.com

P. Karthikaikumar
p.karthigaikumar@gmail.com

Justin Varghese
justin_var@yahoo.com

T. Jayasankar
jayasankar27681@gmail.com

1 Department of CSE, Karpagam Academy of Higher Education, Coimbatore, Tamilnadu, India

2 Department of ECE, Karpagam Academy of Higher Education, Coimbatore, Tamilnadu, India

3 Faculty of Engineering, Karpagam Academy of Higher Education, Coimbatore, Tamilnadu, India

4 Department of CSE, Karpagam College of Engineering, Coimbatore, Tamilnadu, India

5 Department of ECE, University College of Engineering, BIT Campus, Anna University, Tiruchirappalli, Tamilnadu, India

1 Introduction

The concept of HCI systems started with the development of personal computers. In earlier days, traditional methods of communication that used mechanical devices like the joysticks, keyboard and mouse was prevalent. But as the demand for more natural means of communication increased, researchers explored the area of speech and gestures for developing systems that are less cumbersome and are more users friendly. Hence, methods that depend on computational vision emerged, which has the ability to analyze and recognize human gesture (Manresa et al. 2000), especially hand gestures.

Hand gesture recognition is a broad area of research, which is sub categorized based on the context of the gesture and the technology used to input these gestures (Sergio Escalera et al. 2016). There are various taxonomies that affect the kind of HGR system that is designed: environmental factors, the person performing the gesture, the efficiency of the devices used to capture, the type of gesture-static or dynamic and also the application for which the system is designed. Different approaches for hand gesture analysis have been researched:

- Sensor based techniques that usually require the user to attach some sort of computer input device to their body (like a glove) (Sturman and Zelter 1994; Power Glove (2018); Chasing et al. 1994). These methods have an advantage that gesture identification is not distracted by the diverse backgrounds, but comes with a tradeoff of high cost, bulkiness and lack of natural interaction.
- Vision based techniques that use devices like camera or kinetic sensors to input information based on the way humans perceive their surrounding (Haitham Hasan et al. 2012). The efficiency of these methods are dependent on factors like number of cameras and its placement, visibility of the hand and how it is segmented from the image, efficient feature extraction and classification algorithms (LaViola et al. 1999).

Hand gesture segmentation usually involves a preprocessing stage that removes the unwanted noise. To remove noise, filters like morphological filter (Shubhangi et al. 2017; Dipak Kumar Ghosh 2011), Kalman filter (Xu et al. 2017) and other common filters like median and Gaussian filters have been used in many researches. Other image enhancement techniques using wavelets (Fu et al. 2015) and edge traversal (Ghotkar and Kharate 2012) have also been used as a pre-handling step. The actual process of gesture segmentation involves separating the hand gesture from its background image. The complexity of this step would depend on the type of background that we are dealing with. Skin color (Morrison and McKenna 2004) is a very distinctive characteristic of the human hand which can be used to separate it from the background. Color based segmentation usually uses histogram matching, lookup table approach or training to learn skin color data in different color spaces. Contour detection is another technique that is used for segmentation. Several systems use standard background subtraction and other edge detection techniques to locate the region of the hands. Some gesture recognition methods use the image edge information directly as a feature as well (Sun and Zhang 2010), but such techniques show poor performance in instances of variable lighting and other effects. Hybrid algorithms are also used that combine the background subtraction technique with skin color detection (Elsayed et al. 2015).

The feature extraction stage is a very crucial stage in the design of a HGR system. It is the stage where non redundant crucial information is derived from an initial set of data. Some of the earliest researches that used feature based technology for gesture classification are (Rubine et al. 1991; Sturman et al. 1992; Wexelblat et al. 1995); which used either a sensor glove or some other type of input sensory device. For vision based devices either Contour features (Kuo et al. 1996; Chen and Bui 1999; Stiene et al. 2018) or texture features (Triggs et al. 2005; Lowe et al. 2004) were commonly used.

Key points are salient image patches that include rich local information of an image. Some popular key point extractor techniques are Scale-Invariant Feature Transform (Ke and Sukthankar 2004), principal component analysis (PCA)—SIFT (Bay et al. 2008) and Speeded Up Robust Feature (Veltman et al. 1994). SIFT descriptor has the advantage of being scale and rotation variant and can also tolerate certain amount of view point changing. SURF was developed by Herbert Bay based on the concepts of SIFT, thus making it scale and rotation invariant which is a very desirable property for hand gesture recognition. But the 64 dimensional descriptor of SURF causes a very high computation cost. In this paper the SURF descriptor has been used and a novel approach has been proposed to reduce this computation cost.

The final stage of the HGR system is the gesture classification where a input feature vector derived from the feature extraction phase will be inputted into a suitable classifier for recognition. In the past several years, classifiers based on machine learning have gained popularity due to its versatility and tendency to learn behaviors. Artificial neural networks (ANN), hidden Markov model (HMM) and support vector machines (Dardas and Georganas 2011) and are highly researched classifiers that are used for classification tasks. Each classification technique has its own pros and cons; hence the performance of a classifier cannot be based on just the algorithm used. Some algorithms would perform well on a certain set of data but maybe completely unsuitable for a different data set.

Based on the literature review it was understood that most HGR systems that currently exist have some sort of prerequisites that has to be followed in order for the gesture to be recognized. Most gesture recognition systems that are scale or rotation invariant have a trade off in their time to recognize the gesture. So it is required to introduce techniques that reduce the overall computation complexity, at the same time keeping it rotation, scale and illumination invariant. This paper proposes a HGR system which accepts hand gestures with complex background and has shown to be rotation, scale and illumination invariant.

2 Materials and methods

2.1 Dataset

The proposed HGR system was trained and tested using Sebastian Marcel static hand posture database (Marcel 2019). The dataset has hand gestures from 10 different persons and on complex as well as plain backgrounds. Different pictures are subjected to different illumination and scale conditions. For the purpose of this research, it was imperative that we choose a dataset that had static gestures that are rotated as well as in different backgrounds, so this dataset was an appropriate match to test for these conditions. Four postures were chosen, mainly 'A', 'B', 'Five' and 'Point' and these were labelled as 'Grasp', 'Stop', 'Spread' and 'No'. For each gesture, it was made sure that different background and different rotation of the gesture was taken into consideration. Sample gesture images from the dataset are shown in Fig. 1

In order to ensure a thorough testing of the system, it is imperative that we have a varied dataset. Hence training and testing images were chosen in such a way that different illumination, background conditions are satisfied. For each gesture class, 150 images were taken for training purpose and 50 images were taken for testing purposes. A total of 600 images were used to train the classifier and 200 images were used to test the model.

2.2 Algorithm

Hand gesture recognition involves multiple stages of processing so it will be necessary to develop suitable algorithms for each stage of the process. The main stages can be divided into: hand segmentation, feature extraction and classification. The proposed algorithm is shown as in Fig. 2.

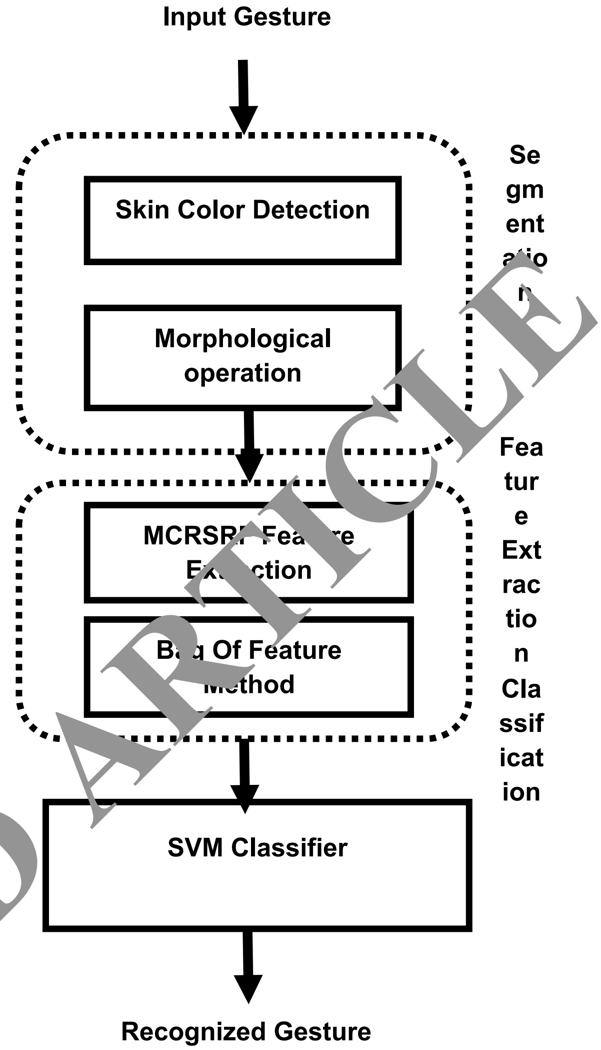

Fig. 2 Flow chart of proposed HGR system

2.2.1 Hand gesture segmentation

The first stage of the HGR system is the image segmentation where the hand will be isolated from its background and then the image will be prepared for its further stages. Experimental results in He and Zhang (2008) and Yang et al. (1998) show that skin color cluster closely in the color space; hence illumination invariance can be attained by removing the V and using only the H and S components. In this step, the color distribution is separated into skin color and non-skin color and then the threshold value is calculated using the histogram method. Firstly, the skin color is detected using hue–saturation–value (HSV) color model as it is shown to be robust against conditions like lighting, scale and rotation. The input images are in RGB color format and it is first transformed to the HSV format, using the following equations:

Fig. 1 Sample images from Sebastian Marcel dataset

$$H = \left\{ \begin{cases} \theta, G \geq B \\ 2\pi - \theta, G < B \end{cases} \right\} \tag{1}$$

$$S = \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B)} \tag{2}$$

The values of hue and saturation was shown to fall in the threshold range of $0 < H < 25°$ and $70° < S < 180°$.

The skin color segmentation can result in detection of other regions similar to skin color, so it will be necessary to identify the hand region from these. Morphological operation is applied to identify the hand gestures based on the connection property of skin regions. Here it is assumed that the hand region will be the only large region that corresponds to skin color so the connectivity property of the pixels within the hand region will be high as compared to all other skin colored pixels that is observed in the background. The morphological operation thus helps in breaking down the thin connections of the background skin pixels, The operation is performed by erosion and dilation process on a binary image where value '1' represents the skin color and value '0' represents non-skin pixels. Erosion is applied using structuring elements and then dilation is applied to grow the areas lost due to erosion. Effectively this stage will filter out any background noise which can be accidently identified as a skin region. After the morphological filtering, the binary image is labelled so as to identify the clustered group of image as a single region. The results of the segmentation stage can be seen as shown in Fig. 3.

2.2.2 Feature extraction methodology

Feature extraction stage is implemented using the Multi Coiflet Rapid SURF (MCRSRF. The standard SURF method analyzes the pixel distribution before extracting features and then the effective key points are detected based on the pixel orientation, direction and magnitude value. It is implemented in four steps:

1. The integral of the image is taken.
2. Key points are localized using Fast-Hessian.
3. Orientation assignment is performed.
4. Descriptors are extracted from the identified key points.

No change has been made in terms of the sequence of execution of the algorithm, but the speed of the algorithm has been improved in two ways:

1. 2D-DWT is applied before the SURF stage-this ensures a reduced size image without compromising on the image details.
2. The Hessian key point detector is replaced with Harris corner detector—this step ensures that fewer key points are detected (this will be called RSRF).

The 2D-DWT ensures image compression without compromising the image details and then these sub images are used for key point descriptor extraction using RSRF. The stages of the feature extraction methodology is shown in Fig. 4

2D-DWTcompresses the input image by splitting it into its wavelet coefficients and scaling functions. This is achieved as a multi layer resolution, by decomposing the image into the approximation and detail component. The detail component is the high frequency component of the images which will be the noises and edge information and the approximation details will have the main features of the original image. If the image has approximation pixel

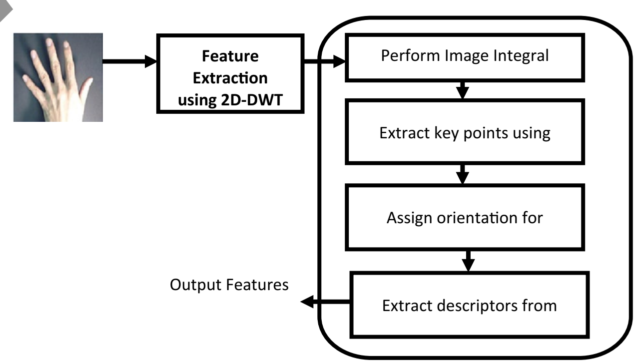

Fig. 4 Structure of multi-resolution Coiflet wavelet based feature extraction

Fig. 3 Hand gesture images with complex background after skin detection and morphological filtering

values as $= \boldsymbol{a}_1, \boldsymbol{a}_2, \dots \boldsymbol{a}_n$, length as $\boldsymbol{l} = \boldsymbol{l}_1, \boldsymbol{l}_2, \dots \boldsymbol{l}_n$ and high frequency component pixel values as $\boldsymbol{d} = \boldsymbol{d}_1, \boldsymbol{d}_2, \dots \boldsymbol{d}_n$. Then approximation and detail coefficient is calculated as follows,

$$\boldsymbol{a}_n = \frac{\boldsymbol{l}_{2n-1} + \boldsymbol{l}_{2n}}{\sqrt{2}} \; \boldsymbol{n} = 1, 2, 3, \dots \boldsymbol{.n}/2 \tag{3}$$

$$\boldsymbol{d}_n = \frac{l_{2n-1} - l_{2n}}{\sqrt{2}} \; \boldsymbol{n} = 1, 2, 3, \dots \boldsymbol{.n}/2 \tag{4}$$

The results of the DWT will depend on the mother wavelet that is used for the extraction. Here, Coiflet Wavelet has been used as it showed better erformance in terms of its entropy and energy components.

After this stage, the key point extraction is implemented using a modified version of the speed up robust feature (SURF). The first stage of this process is to compute the image integral at a point Y(x,y), given by

$$\boldsymbol{I}(\boldsymbol{Y}) = \sum_{m=1}^{m<x} \sum_{n=1}^{n<y} \boldsymbol{I}(\boldsymbol{m}, \boldsymbol{n}) \tag{5}$$

Here, the complexity of the computation is dependent on the size of the image. As we have reduced the image size by applying the 2D-DWT before doing the image inegral, the overall computation complexity is reduced (Priyanka Parvathy and Hema 2016; Priyanka Parvathy and Subramaniam 2019). Now, key point detection is performed by applying Harris corner detector, which identifies the key points that corresponds to the corners of the hand gestures. Since in this work, we have removed the background, the corners detected will be part of the sharp bends and turns of the hand gesture in the image. This step contributes to reduce the number of key points as the corners in an image will be fewer than the key descriptors. After this point we have used the steps of the original SURF algorithm. In order to achieve scale invariant points, the integral of the image is taken at different scales and then a search is conducted for locating scale invariant points in all the scaled images. This search is conducted using a $3\times3\times3$ region where the upper, actual and lower scale images are compared. So if we have a *m x n* image then in the worst case we might have to search until we reach the kernel size of $m \times n$. Once the orientation is assigned, a square window will be used for extracting the feature descriptor (Bag 2019; Lazebnik et al. 2006), which will be a 64-dimensional vector. As the number of key points generated will vary from image to image, it will be necessary to reduce the features to a fixed dimension collection. This step is mandatory if we have to use a support vector machine or artificial neural network as the classifier. The Bag of Feature (BoF) technique will be used to achieve this.

BoF model represents images as order less groups of local features (Hsu et al. 2010). As explained earlier, in the previous stage, after the MCRSRF the image will comprise of several patches whose number will depend on the number of key points that are detected. The BoF model will convert the variable feature space to a collection of same dimension, where the order of the vectors will not be important (Chen et al. 2007). K-mean clustering technique has been used to cluster the key points in the feature space (Varun and Annadurai 2020). For the Sebastian Marcel static hand posture database (Chung et al. 2009) that is used here, it was observed that the key points generated varied between a minimum of 10 and a maximum of 60. Based on this observation, the number of clusters was fixed as 60. This was used as the 'minimum' number of clusters that is required for generating the codebook. Thus the 64-dimensional feature space obtained after MCRSRF is divided into 60 (value of k) clusters:

- The algorithm calculates the Euclidean distance and each key point will be allotted to the cluster which has the shortest distance.
- Once the point is assigned, recalculation will be performed to find the new centroid, which will be the average value of all the points in the cluster.

These steps are continued until no more assignment can be done. So in the end, each image will be resized to a 1×60 dimension vector.

2.2.3 Gesture classification

After successfully extracting the required features and obtaining a fixed size input vector, the classification is implemented using a multiclass support vector machine. SVM's operate by separating labelled points by constructing a hyperplane. For classification, first the data will be separated into training and testing data. Every instance in the training set will have a desired output value which will be the class label and the features associated with that instance. The SVM after training will produce a model that will predict the target value when a test data with only the features is given as input (Picciali et al. 2019).

If we are given a set of $\boldsymbol{n}$ data points, $\boldsymbol{Z}=\{z_1, z_2, \dots z_n\}$, $z_i \in \boldsymbol{R}^l$ which has got labels $\boldsymbol{Y} = \{\boldsymbol{y}_1, \dots \dots \boldsymbol{y}_n\}$, $\boldsymbol{y}_i \in \{1, -1\}$, then we have to figure out the hyper plane that separates the points with $\boldsymbol{y}_i = 1$ from points with $\boldsymbol{y}_i = -1$. This is done by finding a solution to the optimization problem:

$$\min_{w,b,\xi} \frac{1}{2} \mathbf{W}^{\mathbf{T}}\mathbf{W} + \mathbf{C} \sum_{\mathbf{i}}^{\mathbf{l}} \xi_{\mathbf{i}} \tag{6}$$

$$\textbf{subject to } \mathbf{y_i}\left(\mathbf{w}^{\mathbf{T}}\phi\left(\mathbf{x_i}\right) + \mathbf{b}\right) \geq 1 - \xi_{\mathbf{i}}, \quad \xi_{\mathbf{i}} > 0 \tag{7}$$

The kernel function is what basically maps the data to the higher dimensional feature space. There are several kernels available: linear, polynomial, sigmoid and RBF kernel. This work has used Radial Basis Function is used which is defined by Chen et al. (2008):

$$\mathbf{K}\left(\mathbf{x_i}, \mathbf{x_j}\right) = \mathbf{exp}(-\boldsymbol{\gamma}\, \mathbf{x_i}\ \ \mathbf{x_j}^{\ 2}), \boldsymbol{\gamma} > 0 \tag{8}$$

where $\boldsymbol{\gamma}$ is the kernel operator. $\boldsymbol{C}$ and $\boldsymbol{\gamma}$ are the RBF kernel parameters, where $\boldsymbol{C}$ is the penalty parameter and $\boldsymbol{\gamma}$ is the kernel operator. Initially these values are unknown and the aim would be to identify those values with which the (Marcel 1999) classifier can successfully predict unknown data. Firstly, the dataset is separated into training and testing set and then the values are normalized so as to reduce numeric difficulties during computation. To determine the values of $\boldsymbol{C}$ and $\boldsymbol{\gamma}$, a heuristic grid search approach has been used, where various pairs of combination has been tried and the one with the best cross validation accuracy is fixed. All data will be first normalized and then fivefold Cross validation is used to find the best value for $\boldsymbol{C}$ and $\boldsymbol{\gamma}$. And then these parameters are used to train and test the SVM (Yun et al. 2009).

Here we have used a one-vs-all (OVA) SVM method, where each class will have two SVM's. For a given class C_y, there will be two SVM's-SVM_{y1} and SVM_{y2}. So to train SVM_{y1}, the matrix M_y is used where y is the same as the class that we are trying to classify and to train SVM_{y2}, the matrix M_y is used where y will be all other class that (Murthy and Jadon 2010) have to be classified negative. So for each class we have a positive labelled data which is the class that has to be classified as positive and negative labelled data which will be all other classes together that has to be classified as negative. The model of the training stage is as shown in Fig. 5.

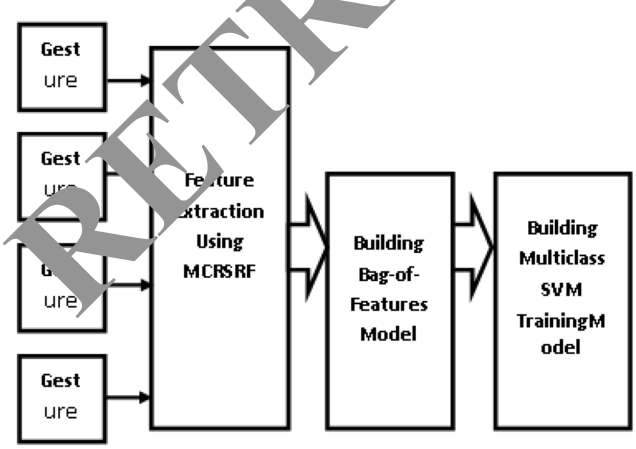

Fig. 5 Training of SVM for HGR with MCRSRF

3 Results and discussions

Two performance measuring criteria will be used to analyze the classifier results-classification accuracy and receiver operating characteristic (ROC) curve.

3.1 Classification accuracy

CA gives the measure of the images that have been rightly classified:

$$\boldsymbol{R} = \frac{\boldsymbol{n_c}}{\boldsymbol{N}} \tag{9}$$

where n_c represents the number of images that are correctly recognized and N is the total number of images used.

3.2 ROC curve

ROC curve is the plot of true positive (sensitivity) against False Positive rate. True Positive represents those images that are correctly classified into its true class and False Positive represents those images that are wrongly classified.

A Confusion matrix is a representation of the prediction results of a classifier on a set of data whose true values are known. The actual class is plotted against the predicted class as is as shown in Fig. 6.

3.3 Experimental results

The images from the Sebastian Marcel static gesture database are separated into training and testing images. Then each sample image in the database is subjected to the following steps:

1. Convert image to gray scale and then resize.
2. Extract approximation detail using 2d-DWT.
3. Extract key point descriptors using RSRF algorithm

	Positive	**Negative**
Positive	TP	FP
Negative	FN	TN

Fig. 6 Typical Confusion matrix

Table 1 Performance of SVM

Gesture	Support vector machine			
	Correct classification	Incorrect classificsation	Accuracy	Time (s)
Stop	188	12	94	0.024
No	196	4	98	0.024
Spread	192	8	96	0.024
Grasp	196	4	98	0.024

4. A unidimensional feature vector of size 1×60 is obtained using BoF.

The feature vectors of the training images are stored in a training feature matrix which is used for training the classifier to create a classifier model. The SVM classifier was first trained using 600 images, which is 150 images per gesture class. Training the classifier generates the training model on which the testing images are inputted. For testing purposes 50 images per class has been used and the accuracy is analyzed using the classification accuracy metric. The feature vectors of the testing images are stored in a testing feature matrix which will be used for testing on the trained classifier model. In this work, since we have four classes of data we will train 4 set of SVM classifiers. Each classifier tries to train one class from the other three classes, so that means each hand gesture class will be having a set of two SVM's (Table 1).

Through all testing, it was observed that the 'No' gesture showed the least misclassification and 'Stop' gesture showed maximum misclassification. This is because of the similarity of the Stop gesture with the Spread; many gestures have been misclassified between these two classes.

As can be seen, the time to recognize the gesture is very short and is 0.024 s. The performance comparison for each class can be clearly seen with the confusion matrix which is shown in Fig. 7.

The class accuracy of class 2 and class 4 which corresponds to the 'No' and 'Grasp' gesture is the highest at 98%. Class 1 (Stop gesture) showed maximum misclassification which is shown by the least accuracy of 94%. The overall classification accuracy of the SVM classifier is shown to be 96.5%.

4 Conclusion

In this research work, a novel hand gesture recognition system has been proposed which is tested and validated with hand gestures in complex background from the Sebastian Marcel static hand gesture database. In order to segment the hand gesture, skin color detection has been used and then morphological operation is applied to eliminate other possible detections of skin color areas. This is achieved by taking advantage of the connectivity property of the skin colored pixels within the hand gesture region. Further to this, essential key descriptors are extracted using the proposed MCRSRF algorithm and then classified using a multi class SVM. The system was tested with a wide range of images which are rotated as well as scaled and it has shown to perform with an overall accuracy of 96.5% with a very fast recognition time of 0.024 s. This clearly shows that the HGR system is rotation and scale invariant and can accept gestures from complex backgrounds. Due to the fast recognition time, this system can be enhanced by employing real time gesture images and developing the classifier accordingly.

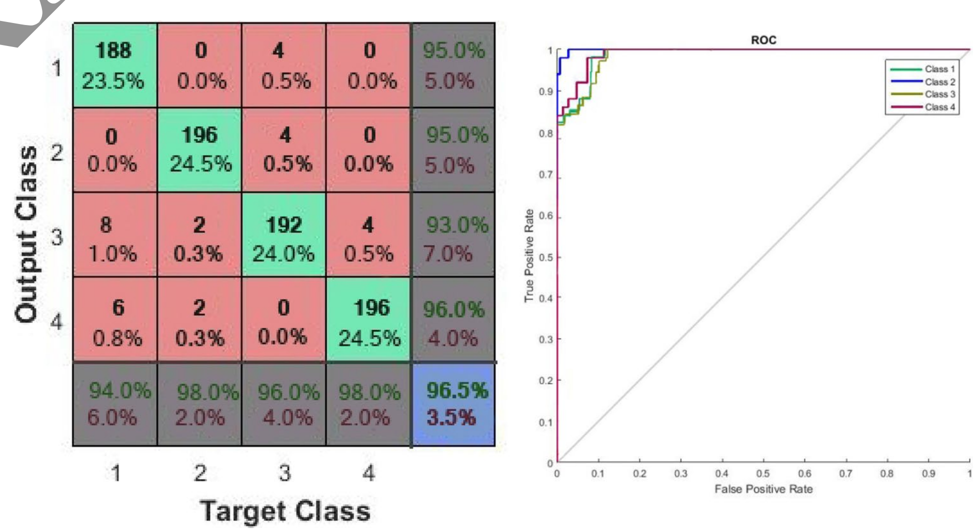

Fig. 7 Confusion matrix and ROC plot for the SVM

References

(1994) Chasing the colour glove: visual hand tracking. In: Chasing the colour glove: visual hand tracking

Bag-of-words model in computer vision. Wikipedia (Online). Available: https://en.wikipedia.org/wiki/Bag-of-words_model_in_computer_vision. Accessed May 2019

Bay H, Tuytelaars T, Van Gool L (2008) Speeded up robust features. In: Computer vision and image understanding, pp 346–359

Chen G, Bui TD (1999) Invariant fourier-wavelet descriptor for pattern recognition. Pattern Recogn 32:1083–1088

Chen Q, Georganas ND, Petriu EM (2008) Hand gesture recognition using Haar-like features and a stochastic context-free grammar. IEEE Trans Instrum Meas 57:1562–1571

Chen Q, Georganas ND, Petriu EM (2007) Real-time vision-based hand gesture recognition using haar-like features. In: IEEE instrumentation and measurement technology conference proceedings

Chuang GH, Kuo CC (1996) Wavelet descriptor of planar curves: theory and applications. IEEE Trans Image Process 5(1):56–70

Chung WK, Wu X, Xu Y (2009) A real-time hand gesture recognition based on Haar wavelet representation. In: IEEE int. conf. robot. biomimetics

Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: Computer vision and pattern recognition

Dardas NH, Georganas ND (2011) Real-time hand gesture detection and recognition using bag-of-features and support vector machine techniques. IEEE Trans Instrum Meas 60:3592–3607

Dipak Kumar Ghosh SA (2011) A static hand gesture recognition algorithm using K-mean based RBFNN. In: ICICS

Elsayed RA, Sayed MS, Abdalla MI (2015) Skin-based adaptive background subtraction for hand gesture segmentation. In: CECS

Escalera S, Athitsos V, Guyon I (2016) Challenges in multimodal gesture recognition. J Mach Learn Res 17(72):1–54

Fu X, Lu J, Zhang T, Bonair C, Coats ML (2015) Wavelet enhanced image preprocessing and neural networks for hand gesture recognition. In: 2015 IEEE international conference on Smart City/SocialCom/SustainCom (SmartCity), Chengdu, China

Ghotkar AS, Kharate GK (2012) Hand Segmentation techniques to hand gesture recognition for natural human computer interaction. Int J Hum Comput Interact 3(1):15–25

Haitham Hasan SK (2012) Static hand gesture recognition using neural networks. Springer Science+Business Media, Berlin

He J, Zhang H (2008) A real-time face detection method in human-machine interaction. In: International conference on bioinformatics and biomedical engineering

Hsu CW, Chang CC, Lin CJ (2003) A practical guide to support vector classification. (Online). Available: https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.224.4115

Ke Y, Sukthankar R (2004) Pca-sift: a more distinctive representation for local image descriptors. In: IEEE conf. on computer vision and pattern recognition

LaViola J (1999) A survey of hand posture and gesture recognition techniques and technology. Department of Computer Science, Brown University, Providence

Lazebnik S, Schmid C, Ponce J (2006) Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In: IEEE conference on computer vision and pattern recognition

Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60:91–110

Manresa C, Varona J, Mas R, Perales FJ (2000) Real-time hand tracking and gesture recognition for human computer interaction. Computer Vision Centre, University Autonomic, Barcelona

Marcel S (1999) Hand posture recognition in a body-face centered space. In: Proc. conf. human factors comput. syst

Marcel S (2019) Sebastian Marcel hand posture and gesture datasets. (Online). Available: The hand gestures are obtained from https://www.idiap.ch/resource/gestures/website. Accessed Mar 2019

Morrison K, McKenna SJ (2004) An experimental comparison of trajectory-based and history-based representation for gesture recognition. In: Gesture-based communication in human–computer interaction, pp 152–163

Murthy G, Jadon R (2010) Hand gesture recognition using neural networks. In: 2010 IEEE 2nd international advance computing conference (IACC), Patiala, India

Piccialli F, Cuomo S, di Cola VS, Casolla G et al (2019) A machine learning approach for IoT cultural data. J Ambient Intell Human Comput. https://doi.org/10.1007/s12652-019-01452-6

Power Glove (2018) (Online). Available: https://en.wikipedia.org/wiki/Power_Glove. Accessed 1 June 2018

Priyanka Parvathy D, Hema CR (2016) Hand Gesture Identification using Preprocessing, Background Subtraction and Segmentation Techniques. IJAER 11(5):3222–3228

Priyanka Parvathy DP, Subramaniam K (2019) Rapid speedup segment analysis based feature extraction of hand gesture image, multimedia tools and applications. Springer US, New York, pp 1–16

Rubine D (1991) Specifying gestures by example. In: SIGGRAPH

Shubhangi, Shinde G, Rajashri, Itkarkar R, Anilkumar V (2017) Gesture to speech conversion for sign language recognition. Int J Innov Adv Comput Sci 6(9)

Stiene S, Lingemann K, Nuchter A, Hertzberg J (2018) Contour-based object detection in range images. (Online). Available: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.73.640&rep=rep1&type=pdf. Accessed June 2018

Sturman DJ (1992) Whole-hand input. Massachusetts Institute, Cambridge

Sturman DJ, Zeltzer D (1994) A survey of glove-based input. IEEE Comput Gr Appl 14(1):30–39

Sun LJ, Zhang LC (2010) Static sign language recognition based on edge gradient direction histogram. Microelectron Comput 27(3)

Varun M, Annadurai C (2020) PALM-CSS: a high accuracy and intelligent machine learning based cooperative spectrum sensing methodology in cognitive health care networks. J Ambient Intell Human Comput. https://doi.org/10.1007/s12652-020-01859-6

Veltman SR, Prasad R (1994) Hidden Markov Models applied to onlinehandwritten isolated character recognition. EEE Trans Image Process 3:314–318

Wexelblat A (1995) An approach to natural gesture in virtual environments. ACM Trans Comput Hum Interact 2:179–200

Xu P (2017) A real-time hand gesture recognition and human-computer interaction system. (Online). Available: https://arxiv.org/pdf/1704.07296.pdf. Accessed June 2018

Yang J, Lu W, Waibel A (1998) Skin-color modeling and adaptation. In: ACCV

Yun L, Peng Z (2009) An automatic hand gesture recognition system based on Viola-Jones method and SVMs. In: Proc. 2nd int. workshop comput. sci. eng

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.