

# **Optimized Feature Extraction for Precise Sign Gesture Recognition Using Self-improved Genetic Algorithm**

Rajesh Kaluri\*, Pradeep Reddy CH

School of Information Technology and Engineering, VIT University, Vellore, India.

Received 04 February 2017; received in revised form 25 April 2017; accepted 04 May 2017

## **Abstract**

Over the past two years, gesture recognition has become the powerful communication source to the hearing-impaired society. Furthermore, it is supportive in creating interaction between the human and the computer. However, the intricacy against the gesture recognition arises when the environment is relatively complex. In this paper, a recognition algorithm with feature selection based on Self-Improved Genetic Algorithm (SIGA) is proposed to promote proficient gesture recognition. Furthermore, the recognition process of this paper includes segmentation, feature extraction and feed-forward neural network classification. Subsequent to the gesture recognition experiment, the performance analysis of the proposed SIGA is compared with the conventional methods as reported in the literature along with standard Genetic Algorithm (GA). In addition, the effects of optimization and the feature sensitivity are also demonstrated. Thus, this method makes aggregate performance against the conventional algorithms.

**Keywords:** gesture sign recognition, GA, SIGA, feed- forward neural network

## **1. Introduction**

In recent times, with the quick growth of computer technology and increasing reputation of the human - Computer interface (HCI) have made the development in gesture recognition of sign languages. Basically, sign languages are the prime means of communication of Deaf people [1-3]. Those sign languages are interpreted into speech that enables the hearing people to understand the deaf people by developing an appropriate tool. The functionality of the corresponding tool is referred as Sign Language Recognition SLR [4]. This tool also has the complement function to transform the text or speech into sign languages that make the deaf people to understand. The main applications of SLR are routing and management in virtual environment, communication in video conferencing, somatosensory game, robot control and so on [5-7].

Research against the SLR tool has begun in about twenty years ago, for Indian, Australian, American, Japanese and African sign languages [8-10]. Since then, many techniques have been developed using a diverse method such as signal processing, sensor fusion, image processing, and pattern recognition and so on have with different characteristics of gesture [4]. Still, recognition of sign language remains a far-reaching issue because of the complexity and visual analysis and quickness against the variations in the signed gestures. Moreover, all countries have their own version of sign language, as the gestures are not-universal [11-12]. Therefore, it is necessary to develop an efficient sign recognition system which should be supportive towards the internationalizing sign-based communications [13-14].

---

\* Corresponding author. E-mail address: rajeshkaluri@yahoo.com

Basically, the sign gesture recognition was implemented using signal processing (2D) and image processing (3D) approaches. In case of signal processing, surface electromyographic sensors (sEMG) extract the electrical activity of the skeletal muscles [15]. It provides rich information of the co-activation and coordination of multiple muscles with various sign gestures. However, there are many factors that affect the characteristics of EMG signals. For example, the individual difference and electrode displacement gives change in measurement of a specific individual and task of gesture at different event [16-18]. On the other hand, in terms of image / video processing, many algorithms show its effectiveness in the particular data base of gestures, though they suffer from degradation of video background, deviation of hand positions etc. [13, 19]. In general, the computer vision based- gesture recognition mainly includes the extraction and selection, description and classifier, learning and training [20]. Through the selection of the gesture characteristics, the accuracy of the recognition gets varied. Consequently, in case of classification, the neural network based classifier given by [21]; it requires extensive training with explicit specifications of the starting and end of the gesture. To solve this, self-organizing map (SOM) is proposed in [22] where it requires training the system for recognition of initial classes. Additionally, the system may suffer from complex classification when the number of gestures increases, complex background image and unstable light conditions using other conventional classifiers [23-28]. Consequently, the improvement of exceedingly valuable picture/video handling based-motion acknowledgment with successful classifier is fundamental in current gesture recognition system [28].

## 2. Literature Review

### 2.1. Related works

In 2015, Kumud Tripathi et al. [29] have proposed the gradient based- key frame extraction scheme to solve the problems arising in the Indian Sign Language (ISL) gesture recognition system. Those key frames were used to split the gestures of sign language into series of signs, and further, each gesture was considered as an isolated gesture which also has removed the similar frames. The combination of Orientation Histogram (OH) with Principal Component Analysis (PCA) has applied to extract the features of the pre-processed gestures. Finally, the experimental result was compared using classifiers Hidden Markov Model (HMM), Support vector machine (SVM) with various distance metrics like Euclidean distance, and have proved that Euclidean distance and correlation provides better accuracy.

Wang Liang et al. [20] have adopted the multi- feature fusion based recognition algorithm for gestures. The initial process was the image segmentation stage, where it extracts the interested region of gestures in terms of color and depth through the mixture of depth data. The features such as weighted Hu invariant moments of depth map and Histogram of oriented gradients (HOG) of the color image were extracted, and the fusion of the two features was performed. Further, the corresponding features were applied to the HMM classifier, and the performance was carried out. The experimental results have proved that, the proposed method can adapt with the skin object influence, moving of multiple objects, background interference and performs the operation even in real time.

In 2015, R. S. Rokade and D. D. Doye [13] have put forth a novel system for sign language gesture recognition. There were three main steps in this technique includes detection of region of interest, detection of key frames and recognition of gestures. From the uniform and non-uniform backgrounds, the segmentation algorithm has distinguished the regions of interest. Further, the implementation results were analyzed, and have stated that it is a simple and fast algorithm that can handle temporal illuminations.

Shanableh et al. [4] have suggested the spatio-temporal feature extraction techniques for gesture recognition of Arabic Sign Language. Using forward, backward and bidirectional predictions, the temporal features of the video- based gesture were extracted. The motion of the sequence was represented from which the prediction errors are threshold and accumulated into a

single image which was then followed by the feature extraction. Additionally, the linear separability of the extracted features was evaluated which was then applied to the classification techniques such as K nearest neighbor and Bayesian classifier. The experimental results have revealed the proposed method provides the requirements of the computation and storage of the classifier as the features are precise and concise.

In 2015, Shao-Zi Li et al. [30] have developed the feature learning approach that consists of sparse auto-encoder (SAE) and PCA in order to recognize the human actions in case of RGB-D images. The components of the feature learning process were divided into two steps. In case of first component [20], the SAE with conventional neural network were learned the features from the image and depth channels where as in case of second component [4], the final features were obtained by concatenating the feature which was then applied to the multi-layer PCA. The testing results have assured that the proposed method is effective by improving the recognition rate.

## 2.2. Review

The literature reveals the significance of different methodologies of sign gesture recognition. These methodologies have dealt with the initialization of feature extraction techniques such as OH with PCA [29], HOG [20], spatio-temporal feature extraction [4], SAE with PCA [14], respectively. Accordingly, the classification of different features was examined with various classifiers such as HMM [20, 29], neural network, SVM [29], K nearest neighbor and Bayesian classifier [4] etc. However, they need remarkable improvements and have to be subjected to handle the challenges yet. Those challenges under the feature extraction include avoiding non-linear mapping [29], avoiding problem of extracting only global features [20], make occurrence of steerability property [13], avoiding computational complexity [4, 29, 30]. Hence, the literature review stated the state-of-the-art of sign gesture recognition requires additional perfection in image/video processing.

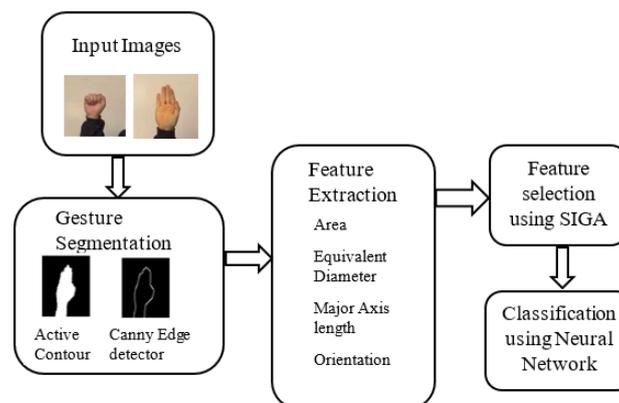


Fig. 1 Block diagram of proposed Gesture Recognition

## 3. Process of Gesture Recognition

The gesture recognition using EMG based signal processing [16-18] still suffers from the remarkable issues in terms of varied signal measurement in various times and its cost inefficiency. Hence, this proposal intends to adopt the image based-gesture recognition. This methodology will include four basic steps such as image segmentation, feature extraction and classification. In the course of segmentation process, active contour model is used to extract the region of the gesture and canny edge detector [30] is used to extract the edge of the corresponding gesture. Subsequently, 13 features are extracted under the feature extraction process. Further, the required rules are selected under the feature selection process using SIGA. The proposed self-improvement mechanism effectuates the performance of the conventional GA, so that precise feature information for each gesture can be extracted. Finally, the extracted features will be trained under the Neural Network to recognize the appropriate sign of the gesture.

## 4. Self-Improved GA Based Gesture Recognition

### 4.1. Gesture segmentation

Let us consider the input gesture image  $I_{gesture}(x, y)$  which is applied to the two segmentation processes such as active [14] contour model and canny edge detector.

**Active Contour model:** The region of the gesture is segmented through the active contour model. According to active contour model, a set of coordinates should be initialized to make the mask or contour of the object to be segmented. This created contour is moved over the image by the driven forces of the image to the boundaries of the desired object. There are two types of forces utilized in this process such as internal energy and external energy. The model is set as smooth amid the twisting model utilizing inner vitality though the model is moved toward the limit of the protest utilizing outer strengths. Thus, the set of coordinates useful to make the contour is parametrically represented in Eq. (1) where 'S' refers to the normalized index of the control points of the contour and (x, y) refers to the coordinates of the contour.

$$C(s) = (x(s)y(s)); C(s) \in I_{gesture}(x, y) \quad (1)$$

The overall energy of the deformation model is expressed in Eq. (2) where  $E^{internal}$  refers to the internal energy of the curve,  $E^{image}$  refers to the energy of the image and  $E^{con}$  refers to the exterior restrictions.

$$E_{snake}^* = \int_0^1 (E^{internal} C(s) + E^{image} C(s) + E^{con} C(s)) ds \quad (2)$$

The summation of elastic energy and bending energy is called as internal energy which is represented in Eq. (3)

$$E^{internal} = E^{elastic} + E^{bend} = \alpha(s) \left| \frac{dv}{ds} \right|^2 + \beta(s) \left| \frac{d^2v}{ds^2} \right|^2 \quad (3)$$

$$E^{elastic} = \alpha(C(s) - C(s-1))^2 ds \quad (4)$$

$$E^{bend} = \beta(C(s-1) - C(s) + (C(s+1)))^2 ds \quad (5)$$

Thus, the segmented gesture by the active contour model is referred as  $S_{active}(x, y)$ .

**Canny edge detector:** The edge of the gesture image is extracted using the canny edge detector. Since the canny detector is more sensitive to noise, Gaussian filter is applied to remove the noise which is represented in Eq. (6).

$$Ga(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (6)$$

Thus, the convolution of the input image with the Gaussian filter is expressed in Eq. (7).

$$V(x, y) = Ga(x, y) * I_{gesture}(x, y) \quad (7)$$

The gradient magnitude and directions are computed after convoluting the input image with the Gaussian filter which is represented in Eq. (10) and Eq. (11) where  $G(x)$  and  $G(y)$  refers to the gradient magnitude of the convolved image in  $x$  and  $y$  directions respectively.

$$G(x) = (V(x+1, y) - V(x, y) + V(x+1, y+1) - V(x, y+1)) / 2 \quad (8)$$

$$G(y) = (V(x+1, y) - V(x, y) + V(x+1, y+1) - V(x, y+1)) / 2 \quad (9)$$

$$G = \sqrt{G(x)^2 + G(y)^2} \quad (10)$$

$$\theta = a \tan 2(G(x), G(y)) \quad (11)$$

Thus, the edge extracted image of the gesture after the computation of the gradient magnitude and directions can be denoted as  $S_{canny}(x, y)$ .

#### 4.2. Feature extraction

The feature extraction technique is applied to both the segmented images by active contour model  $S_{active}(x, y)$  and canny edge detector  $S_{canny}(x, y)$ . The features such as area, equivalent diameter, major axis length, minor axis length, euler number, bounding box, centroid, extent, orientation, convexarea, extrema, solidicity and eccentricity are computed in this experimentation are described below.

**Area:** It is defined as the total count of pixels in the region  $R$  which is expressed in Eq. (12) where  $r, c$  refers the size of the image,  $S(x, y)$  refers the segmented image which is either  $S_{active}(x, y)$  or  $S_{canny}(x, y)$ .

$$A = \sum_{r,c \in R} S(x, y) \quad (12)$$

**Equivalent Diameter:** The equivalent diameter of the region can be expressed in Eq. (13).

$$D = \text{sqrt}(4 * A / \pi) \quad (13)$$

**Major axis length:** The number of pixels or the length of the major axis  $M(x, y)$  of the ellipse which has similar normalized central moments is called as major axis length in Eq. (14).

$$L_{major} = \sum M(x, y) \quad (14)$$

**Minor axis length:** The number of pixels or the length of the minor axis  $m(x, y)$  of the ellipse which has similar normalized central moments is called as major axis length in Eq. (15).

$$L_{minor} = \sum m(x, y) \quad (15)$$

**Euler number:** It's the difference between the number of objects in the region  $S$  and the number of holes in the particular region  $H$  which is expressed in Eq. (16).

$$E_N = S - H \quad (16)$$

**Bounding box:** The smallest rectangle suited to cover the region is called as bounding box.

**Centroid:** The center of mass of regions or the average of  $x$  and  $y$  coordinates of the vertices are termed as centroid. The formulation of centroid calculation for two vertices is expressed in Eq. (17).

$$C = \left( \frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2} \right) \quad (17)$$

**Extent:** It is defined as the total count of pixels in the region  $A$  to the total number of pixels in the bounding box  $B_A$  which is expressed in Eq. (18).

$$E_x = \frac{A}{B_A} \quad (18)$$

**Orientation:** It is defined as the angle between the X-axis and the major axis of the ellipse which has alike second-moments as the region.

**Convex Area:** It is defined as the total count of pixels in convex image  $C_i(X, Y)$  which is expressed in Eq. (19).

$$C_o = \sum C_i(x, y) \quad (19)$$

**Extrema:** The extremal points in the particular region of the image are called as extreme.

**Solidity:** It is the degree to which the given region is convex or concave which is represented in Eq. (20) where  $K$  refers to the convex hull area of the shape.

$$S_i = \frac{A}{K} \quad (20)$$

**Eccentricity:** The ratio of the distance between the foci of the ellipse  $f_d$  to semi-major axis length  $a$  is called eccentricity in Eq. (21).

$$Ec = \frac{f_d}{a} \quad (21)$$

#### 4.3. Feature selection

The resultant output after the feature extraction technique includes the 13 features of the segmented image by active contour model as well as the 13 features of the segmented image by canny detector. Accordingly, a feature selection method include SIGA is utilized to select the required features. The extracted feature set from both active contour model and canny edge detector is referred as  $F_{active}$  and  $F_{canny}$ . These features set are commonly referred as,

$$X = \{F_{active}, F_{canny}\} \quad (22)$$

**Conventional GA:** GA [31-32] is one of the meta-heuristic algorithms which are operated through the method of natural selection based on the Darwin theory of natural evolution “the survival of fittest”. GA algorithm was first introduced by John Holland in 1975. The high-quality solutions to the problems are basically solved by the genetic optimization and the search problems depends on the genetic operators such as selection, cross over and mutation.

**Selection:** The reproduction is performed by selection two best chromosomes through the selection operator. Consequently, the best chromosomes are more often selected for reproduction.

**Crossover:** This operator combines the two-selected chromosome to produce offspring which holds the characteristics of the parent chromosomes. In cross over, a locus is selected in both the selected chromosome and afterwards exchanges the bits after and before the locus to generate two offspring.

**Mutation:** The declining of entire solution in the population into local optimum of solved problem is reduced the mutation operation. This operator makes changes in the newly generated offspring by randomly flipping the bits from 0 and 1 which is referred as *rand*. Here the probability of mutation  $P(m)$  is fixed at low value (usually 0.05).

*Algorithm 1: Pseudocode of Conventional Ga*

- (1) Generate a population (feature set)  $X$  contains  $P$  number of chromosomes
- (2) Compute the fitness  $f$  of all chromosomes
- (3) Production of offspring  $X$ .
- (4) Perform the selection and crossover operation.
- (5) Perform the mutation operation and find  $rand$ .
- (6) Fix probability of mutation  $P(m)$  as 0.05.
- (7) If  $rand < P(m)$   
 $X = X + (P(m) * rand)$   
 End
- (8) Put back the current population with new population.
- (9) Continue the process till stopping condition.

**Proposed SIGA:** As per the conventional GA, the mutation operation is only performed if the  $rand < P(m)$ . Rather than in proposed SIGA, it is unavoidable to perform the mutation operation where the concerned condition is given in the pseudo code. In addition, the formulation of the probability of mutation is expressed in Eq. (23)

$$P(m) = 0.9 \times f / \max(f) + 0.1 \quad (23)$$

*Algorithm 2: Pseudocode of Proposed SIGA*

- (1) Generate a population (feature set)  $X$  contains  $p$  number of chromosomes.
- (2) Compute the fitness  $f$  of all chromosomes
- (3) Production of offspring  $X$ .
- (4) Perform the selection and crossover operation.
- (5) Perform the mutation operation and find  $rand$ .
- (6) Compute the probability of mutation  $P(m)$  using Eq. (23)
- (7) If  $rand < P(m)$   
 $X = X + (P(m) * rand)$
- (8) Else  
 $X =$  create a random solution
- (9) Put back the current population with new population.
- (10) Continue the process till stopping condition.

The description of the pseudo code and flowchart of the proposed SIGA algorithm is illustrated below.

- (1) The population of the feature set is initialized as  $X$  which contains  $p$  number of chromosomes or features.
- (2) The genetic operators such as selection cross over and mutation is performed and subsequently, computes  $rand$ .

- (3) The probability of mutation  $P(m)$  is determined.
- (4) The offspring  $X' = X + (P(m) \times rand)$  is generated if  $rand < P(m)$ ,
- (5) The offspring is generated by through the creation of random solution if  $rand > P(m)$
- (6) The current population is replaced with the newly generated population
- (7) The steps are repeated until the stopping condition is reached.

#### 4.4. Classification

The optimally selected feature set by GA is applied to the feed-forward neural network [33] for classification. Thus, the feature set of the training data base is represented in Eq. (24) where  $N^{opt}$  refers the number of optimally selected features.

$$X' = [f_1, f_2, f_3, f_4 \dots f_{N^{opt}}] \quad (24)$$

The weight of the network model  $w$  is optimally determined by Levenberg - Marquardt (LM) algorithm. The model of feed-forward neural network is represented in Eq. (25) where the resultant output from  $j^{th}$  node of layer  $l$  is denoted as  $Y_j^{(l)}$ , the input to the layer is denoted as  $X_i^{(l)}$ ,  $F_l(\bullet)$  refers to the non-linear function of layer  $l$ , the total count of input to the  $l^{th}$  layer is denoted as  $N_{opt}^{(l)}$ ,  $b_l$  represents the input bias to the  $l^{th}$  layer, the variables  $u$  and  $v$  represents the coefficient of weight to fit the model with the original data,  $w$  refers to the optimal weight,  $|w|$  refers to the cardinality of  $w$ ,  $z$  refers to the actual output and the function  $|\cdot|$  refers the absolute error.

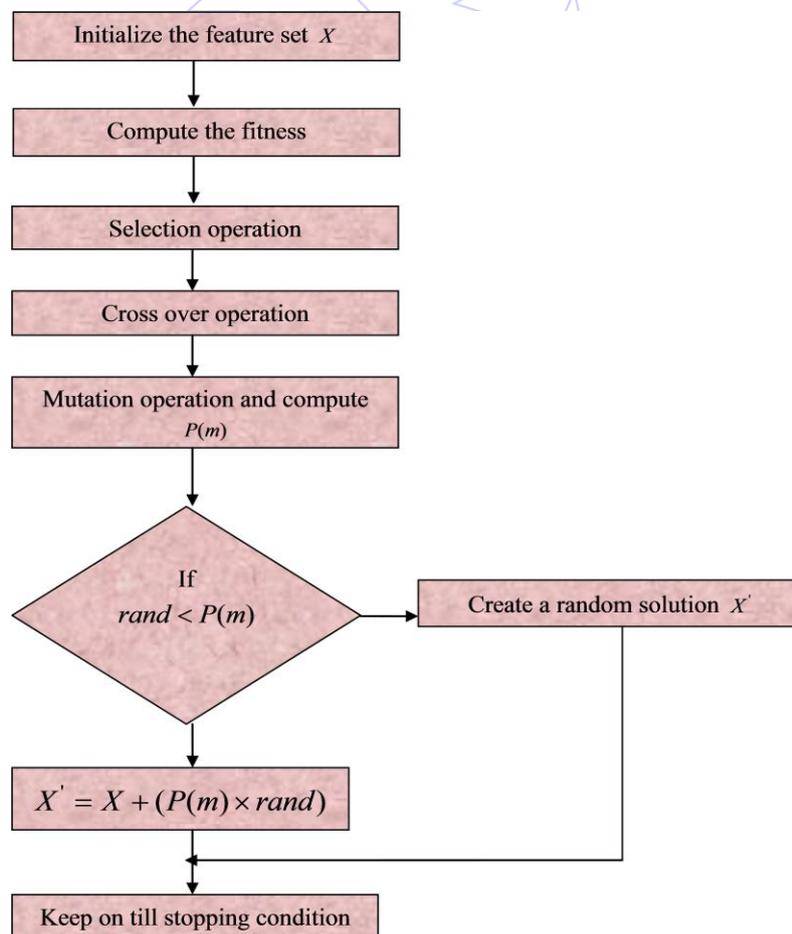


Fig. 2 Flowchart of SIGA

$$Y_j^{(l)} = F_l \left[ u_j^{(l)} b_l + \sum_{i=1}^{N_{opt}^{(l)}} X_i^{(l)} v_{ij}^{(l)} \right] : 0 \leq l \leq k-1 \tag{25}$$

$$w = [u; v] \tag{26}$$

$$w^* = \arg \min_{w=[u;v]} |z - \hat{z}| \tag{27}$$

$$|w| = \sum_{l=1}^{k-1} (N_{opt}^{(l-1)} \times N_{opt}^{(l)}) + N_{opt}^{(l)} \tag{28}$$

## 5. Results and Discussions

### 5.1. Experimental setup

The experimentation for the gesture recognition is conducted using MATLAB 2015a. The sample images of the gesture showing different signs are downloaded from the URL <http://www.idiap.ch/resource/gestures/>. Here 30 images are taken for training and 25 images are taken for testing the data.

### 5.2. Segmented output

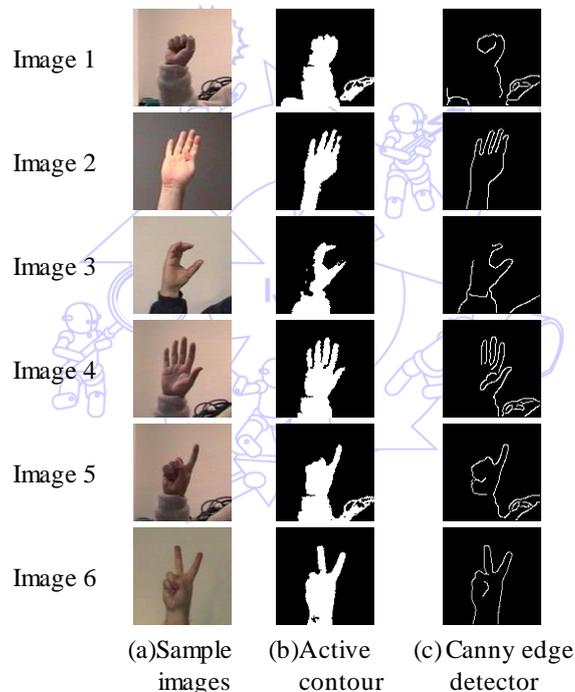


Fig. 3 Experimental results of active contour segmentation and Canny edge detector segmentation for six sample images

The resultant output of the two types of segmentation processes such as active contour model and canny edge detector is shown in Fig. 2. The result is taken for five sample images showing different signs.

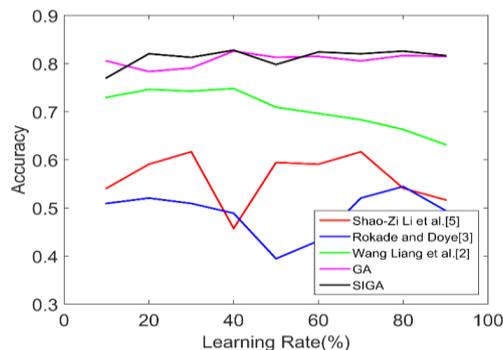


Fig. 4 Line chart representing accuracy of proposed SIGA with conventional methods for various learning rate

Table 1 Comparative analysis of proposed SIGA method with conventional methods

Metrics	Accuracy	Sensitivity	Specificity	Precision	FPR	FNR	NPV	FDR	F1-score	MCC	Average rank	Final rank
Shao-Zi Li et al. [5]	0.51(4)	0.2(5)	0.57(4)	0.08(5)	0.42(4)	0.8(5)	0.57(2)	0.91(5)	0.12(5)	-0.17(5)	4.4	5
Rokade and Doye [3]	0.50(5)	0.46(3)	0.51(5)	0.16(4)	0.48(5)	0.5(3)	0.51(1)	0.83(4)	0.23(4)	-0.01(4)	3.8	4
Wang Liang et al. [2]	0.63(3)	0.36(4)	0.68(3)	0.18(3)	0.31(3)	0.6(4)	0.68(3)	0.81(3)	0.25(3)	0.042(3)	3.2	3
GA	0.74(2)	0.57(1)	0.77(2)	0.33(2)	0.22(2)	0.4(1)	0.77(4)	0.66(2)	0.42(2)	0.28(2)	2	2
SIGA	0.75(1)	0.56(2)	0.79(1)	0.35(1)	0.20(1)	0.4(2)	0.79(5)	0.64(1)	0.43(1)	0.30(1)	1.6	1

### 5.3. Comparative analysis

The tabulation demonstrating the comparative analysis of the proposed SIGA method with conventional methods by Shao-Zi Li et al. [30], Rokade and Doye [13], Wang Liang et al. [20] and GA is shown in Table 1. Here the efficiency of the proposed method is validated by analysing the performance measures such as accuracy, sensitivity, specificity, precision, False Positive Rate (FPR), False Negative Rate (FNR), Negative Prediction Value (NPV), False Discovery Rate (FDR), F-Score and Matthews Correlation Coefficient (MCC). Accordingly, these measures are determined based on the positive and negative images. As per the classification, true positive (TP) specifies the gesture correctly classified as gesture, true negative (TN) specifies the non-gesture correctly classified as non-gesture, false positive (FP) specifies the non-gesture incorrectly classified as gesture and false negative (FN) specifies the gesture incorrectly classified as non-gesture images. The performance of each method is precisely shown in terms of ranking. Thus, in figuring the overall analysis, the final rank is better for the proposed SIGA method.

The estimation of accuracy of proposed SIGA with conventional methods for different learning rate is represented graphically in Fig. 4. The training data for learning is varied from 10 to 90% and the accuracy is calculated for the proposed in this section. Here the accuracy of the proposed SIGA is 77% for 10% of training data, 82% for 20% of training data, 81% for 30% for training data, 82% for 40% of training data, 78% for 50% of training data, 81% for 60%, 70%, 80% and 90% of training data. These above-mentioned accuracy analyses of the proposed SIGA are better than the conventional algorithms.

### 5.4. Effect of optimization

As discussed in the methodology, feature selection is a part of the gesture recognition in this paper. Since, more features are obtained under the feature extraction; feature selection method is used to select the optimum features which are fit for the experimentation. Subsequently, performance of the features selected from the proposed SIGA based feature selection is compared dropped features as well as the total features. The effect of optimization based feature selection is shown in Table 2. The performance analysis is approved by determining the performance metrics. Under such circumstance, accuracy of the SIGA algorithm is 14% better than dropped features and 13% better than the total features whereas the sensitivity is high for the SIGA method. Further, the specificity of the SIGA method is 17% better than the dropped features and 19% better than the total features whereas the precision is 33% and 19% better than the dropped features and total features. Moreover, FPR and FDR provide less performance on SIGA and FNR and NPV provides improved performance. On verifying the F-score and MCC, F-score of SIGA is 21% better than dropped features and 62% better than total features whereas MCC is 59% and 12% superior to the dropped and total features.

### 5.5. Feature sensitivity

The effect of optimization based feature selection is shown in Table 1. Generally, more features are extracted during the feature extraction process. Thus, SIGA based feature selection method is used to optimally select the significant features. These features are computed both for the training and testing data. Further, the training data is varied from 10 to 90% and the selected

features are estimated. Based on the overall analysis, centroid, extend and convex area is considered as the best features which are mostly selected. Subsequent to this features, major axis length, minor axis length and bounding box are the next imminent features. Further, Euler number is also selected less than the aforesaid features. Moreover, area, equivalent diameter, orientation, extrema and eccentricity are less selected featured.

Table 2 Effect of optimized features on gesture recognition over the conventional features

Metrics	Accuracy	Sensitivity	Specificity	Precision	FPR	FNR	NPV	FDR	F1-score	MCC
SIGA	0.74	0.577	0.7733	0.33766	0.22	0.42	0.77	0.66	0.42	0.289
Dropped features	0.64	0.57778	0.66	0.25366	0.34	0.42	0.66	0.74	0.35	0.182
Total features	0.65741	0.68889	0.65111	0.28311	0.34	0.31	0.65	0.71	0.40	0.258

Table 3 Sensitivity of features with respect to the rate of training

Features	Training rate									
	10%	20%	30%	40%	50%	60%	70%	80%	90%	Count
Area	1	0	0	1	0	0	0	1	0	3
Equivalent Diameter	0	0	0	0	0	1	1	0	1	3
Major Axis Length	1	1	0	0	0	1	1	1	0	5
Minor Axis Length	1	0	1	0	1	1	0	1	0	5
Bounding Box	0	1	0	0	1	1	0	1	1	5
Euler Number	0	0	0	1	1	0	0	1	1	4
Centroid	1	1	0	1	1	1	1	0	0	6
Extent	0	1	1	0	1	0	1	1	1	6
Orientation	0	1	0	1	0	0	0	0	0	2
Convex Area	0	1	1	1	1	1	1	0	0	6
Extrema	0	0	0	1	0	0	1	0	0	2
Solidity	0	1	1	0	0	1	0	0	0	3
Eccentricity	1	0	0	0	0	0	0	0	1	2

## 6. Conclusions

The upcoming challenges under the gesture recognition are due to the influence from the skin object and intrusion from the gestures in complex background. This paper has extended the novel gesture recognition algorithm using four basic processes such as gesture segmentation, feature extraction, feature selection and classification. Under the feature selection process, SIGA has proposed in this paper to endorse the admirable gesture recognition process. Later, the performance of the proposed SIGA was compared with conventional algorithm in the literature and the GA method. Moreover, the effectiveness of the SIGA method is affirmed by certifying the effect of optimization along with the sensitivity of features. In due course, it was obvious that the proposed SIGA provides maximum recognition rate than conventional methods.

## References

- [1] K. Tripathi, N. Baranwal, and G. C. Nandi, "Continuous indian sign language gesture recognition and sentence formation," *Computer Science*, vol. 54, pp. 523-531, 2015.
- [2] L. Wang, G. Liu, and H. Duan, "Dynamic and combined gestures recognition based on multi-feature fusion in a complex environment," *The Journal of China Universities of Posts and Telecommunications*, vol. 22, no. 2, pp. 81-88, April 2015.
- [3] R. S. Rokade and D. D. Doye, "Spelled sign word recognition using key frame," *IET Image Processing*, vol. 9, no. 5, pp. 381-388, April 2015.
- [4] T. Shanableh, K. Assaleh, and M. Al-Rousan, "Spatio-temporal feature-extraction techniques for isolated gesture recognition in arabic sign language," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 37, no. 3, pp. 641-650, June 2007.

- [5] S. Z. Li, B. Yu, W. Wu, S. Z. Su, and R. R. Ji, "Feature learning based on SAE-PCA network for human gesture recognition in RGBD images," *Neuro computing*, vol. 151, pp. 565-573, March 2015.
- [6] J. L. Rahejaa, S. Subramaniyamb, and A. Chaudhary, "Real-time hand gesture recognition in FPGA," *Optik - International Journal for Light and Electron Optics*, vol. 127, no. 20, pp. 9719-9726, October 2016.
- [7] R. C. B. Madeo, S. M. Peres, and C. A. de Moraes Lima, "Gesture phase segmentation using support vector machines," *Expert Systems with Applications*, vol. 56, pp. 100-115, September 2016.
- [8] T. H. S. Li, M. C. Kao, and P. H. Kuo, "Recognition system for home-service-related sign language using entropy-based K-means algorithm and ABC-based HMM," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 1, pp. 150-162, January 2016.
- [9] M. R. Abid, E. M. Petriu, and E. Anjadian, "Dynamic sign language recognition for smart home interactive application using stochastic linear formal grammar," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 3, pp. 596-605, March 2015.
- [10] G. Fang, W. Gao, and D. Zhao, "Large-vocabulary continuous sign language recognition based on transition-movement models," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 37, no. 1, pp. 1-9, January 2007.
- [11] R. Xu, S. Zhou, and W. J. Li, "MEMS accelerometer based nonspecific-user hand gesture recognition," *IEEE Sensors Journal*, vol. 12, no. 5, pp. 1166-1173, May 2012.
- [12] J. Gałka, M. Maşior, M. Zaborski, and K. Barczewska, "Inertial motion sensing glove for sign language gesture acquisition and recognition," *IEEE Sensors Journal*, vol. 16, no. 16, pp. 6310-6316, August 15, 2016.
- [13] G. Pradhan, B. Prabhakaran, and C. Li, "Hand-gesture computing for the hearing and speech impaired," *IEEE MultiMedia*, vol. 15, no. 2, pp. 20-27, April-June 2008.
- [14] M. H. Yang, N. Ahuja, and M. Tabb, "Extraction of 2D motion trajectories and its application to hand gesture recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, pp. 1061-1074, August 2002.
- [15] Y. Zhou, X. Chen, D. Zhao, H. Yao, and W. Gao, "Adaptive sign language recognition with exemplar extraction and MAP/IVFS," *IEEE Signal Processing Letters*, vol. 17, no. 3, pp. 297-300, March 2010.
- [16] H. D. Yang, S. Sclaroff, and S. W. Lee, "Sign language spotting with a threshold model based on conditional random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 7, pp. 1264-1277, July 2009.
- [17] J. S. Kim, W. Jang, and Z. N. Bien, "A dynamic gesture recognition system for the Korean sign language (KSL)," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 26, no. 2, pp. 354-359, April 1996.
- [18] J. F. Lichtenauer, E. A. Hendriks, and M. J. T. Reinders, "Sign language recognition by combining statistical DTW and independent classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 11, pp. 2040-2046, November 2008.
- [19] M. B. Waldron and S. W. Kim, "Increasing manual sign recognition vocabulary through relabeling," *Neural Networks, IEEE World Congress on Computational Intelligence*, pp. 2885-2889, vol. 5, 1994.
- [20] P. K. Simpson, "Fuzzy min-max neural networks. I. Classification," *IEEE Transactions on Neural Networks*, vol. 3, no. 5, pp. 776-786, September 1992.
- [21] S. S. Fels and G. E. Hinton, "Glove-Talk: a neural network interface between a data-glove and a speech synthesizer," *IEEE Transactions on Neural Networks*, vol. 4, no. 1, pp. 2-8, January 1993.
- [22] A. Kumar, K. Thankachan, and M. M. Dominic, "Sign language recognition," *2016 3rd International Conf. Recent Advances in Information Technology (RAIT)*, pp. 422-428, July 2016.
- [23] M. K. Bhuyan, M. K. Kar, and D. R. Neog, "Hand pose identification from monocular image for sign language recognition," *2011 IEEE International Conf. Signal and Image Processing Applications (ICSIPA)*, pp. 378-383, 2011.
- [24] X. Chen, V. Lantz, K. Q. Wang, Z. Y. Zhao, X. Zhang, and J. H. Yang, "Feasibility of building robust surface electromyography-based hand gesture interfaces," *2009 Annual International Conf. IEEE Engineering in Medicine and Biology Society, Minneapolis*, pp. 2983-2986, 2009.
- [25] B. Hudgins, P. Parker, and R. N. Scott, "A new strategy for multifunction myoelectric control," *IEEE Transactions on Biomedical Engineering*, vol. 40, no. 1, pp. 82-94, January 1993.
- [26] H. Cheng, Z. Dai, Z. Liu, and Y. Zhao, "An image-to-class dynamic time warping approach for both 3D static and trajectory hand gesture recognition," *Pattern Recognition*, vol. 55, pp. 137-147, July 2016.
- [27] E. M. Taranta, A. N. Vargas, and J. J. LaViola Jr., "Stream lined and accurate gesture recognition with Penny Pincher," *Computers & Graphics*, vol. 55, pp. 130-142, 2016.

- [28] C. Chansri and J. Srinonchat, "Hand gesture recognition for Thai sign language in complex background using fusion of depth and color video," *Procedia Computer Science*, vol. 86, pp. 257-260, 2016.
- [29] R. Elakkiya and K. Selvamani, "An active learning framework for human hand sign gestures and handling movement epenthesis using enhanced level building," *Procedia Computer Science*, vol. 48, pp. 606-611, 2015.
- [30] L. Raheja, M. Minhas, D. Prashanth, T. Shah, and A. Chaudhary, "Robust gesture recognition using Kinect: A comparison between DTW and HMM," *Optik - International Journal for Light and Electron Optics*, vol. 126, pp. 1098-1104, June 2015.
- [31] E. P. Ijjina and K. M. Chalavadi, "Human action recognition using genetic algorithms and convolutional neural networks," *Pattern Recognition*, vol. 59, pp. 199-212, November 2016.
- [32] M. Abdelaziz, "Distribution network reconfiguration using a genetic algorithm with varying population size," *Electric Power Systems Research*, vol. 142, pp. 9-11, January 2017.
- [33] K. Bhatnagar and S. C. Gupta, "Investigating and modeling the effect of laser intensity and nonlinear regime of the fiber on the optical link," *Journal of Optical Communications*, vol. 38, no. 3, pp. 1-13, August 2017.

