Soft Computing (2020) 24:15239–15248
https://doi.org/10.1007/s00500-020-04860-5

METHODOLOGIES AND APPLICATION

An efficient method for human hand gesture detection and recognition using deep learning convolutional neural networks

P. S. Neethu[1] · R. Suguna[2] · Divya Sathish[3]

Published online: 23 March 2020

Abstract

The physical movement of the human hand produces gestures, and hand gesture recognition leads to the advancement in automated vehicle movement system. In this paper, the human hand gestures are detected and recognized using convolutional neural networks (CNN) classification approach. This process flow consists of hand region of interest segmentation using mask image, fingers segmentation, normalization of segmented finger image and finger recognition using CNN classifier. The hand region of the image is segmented from the whole image using mask images. The adaptive histogram equalization method is used as enhancement method for improving the contrast of each pixel in an image. In this paper, connected component analysis algorithm is used in order to segment the finger tips from hand image. The segmented finger regions from hand image are given to the CNN classification algorithm which classifies the image into various classes. The proposed hand gesture detection and recognition methodology using CNN classification approach with enhancement technique stated in this paper achieves high performance with state-of-the-art methods.

Keywords Hand gesture · Recognition · Mask · Fingers · Segmentation

1 Introduction

At present, human–machine interaction is very important for operating the machines in a remote manner by the commands which are received from humans. In this regard, gestures are playing an important role in operating the machine at a distant mode (Yasukochi et al. 2008). The machines capture the gestures from the human and recognize it for operating the machines. The gestures are different types of modes as static and dynamic. The static gestures do not change their position, while the machine is operated, and the dynamic gestures change their positions during the machine is operated (Elmezain et al. 2010; Mitra and Acharya 2007). Hence, the identification or recognition of dynamic gestures is very important than the static gestures (Yrk et al. 2006; Tauseef et al. 2009). Initially, the camera, which is connected with machine, captures the gestures which are generated by humans. The background of the detected gestures is removed, and the foreground of the gesture is captured. The noises in the foreground gesture are detected and removed by filtering techniques (Manresa-Yee et al. 2005). These noise removed gestures are compared with pre-stored and trained gestures for verifying the sign of the gestures. Figure 1 shows the different types of gestures which are generated by humans.

The automotive sectors and many consumer electronics division use the gesture-based machine operating system without any human interaction. Besides the static and dynamic gestures, the gestures of human are also classified into online and offline gestures. The offline gestures operate the icons on the machine, and they are not able to alter the position of the items in the menu or system. The online gestures operate the icons in the machine to different positions or inclinations (Yao and Fu 1935; Liu et al. 1898). The online gestures are very much useful in real-time machine operating systems than the offline gestures.

Communicated by V. Loia.

✉ P. S. Neethu
ps.neethu@gmail.com

1 Department of Information and Communication Engineering, Anna University, Chennai, India

2 Department of CSE, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, Chennai, India

3 Department of CSE, SKR Engineering College, Chennai, India

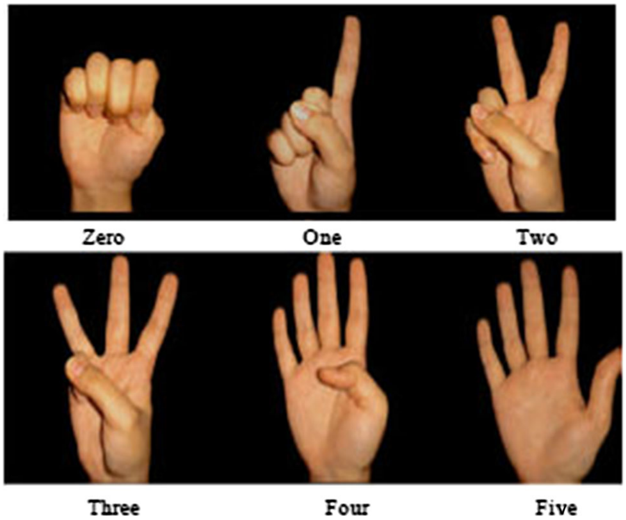


Fig. 1 Different hand gesture postures (Kawulok et al. 2012)

Park et al. (2012) and Ren et al. (2013) used Naïve Bayes classifier and support vector machine (SVM) methodologies for gesture recognition. These methods did not support large training dataset, and it also required high number of training samples. This drawback is eliminated by proposing CNN classifier in this paper. It does not require high number of samples in training mode, and the complexity level of this algorithm is low. The novelty of this proposed work is to implement deep learning algorithm in hand gesture recognition system with novel segmentation technique.

The paper is organized as follows: Sect. 2 deals the conventional methods of hand gesture recognition, Sect. 3 proposes CNN classification-based hand gesture recognition, and Sect. 4 discusses the simulation results of proposed hand gesture recognition system using Python programming language. At the end, Sect. 5 concludes the paper.

2 Literature survey

Zuocai Wang et al. (2018) proposed hand gesture recognition system using particle filtering approach. The authors applied this filtering approach on hand gesture images with same background. The authors obtained 92.1% of sensitivity, 84.7% of specificity and 90.6% of accuracy. Suguna and Neethu (2017) extracted shape features from hand gesture image for the classification of hand gesture images into various classes. Then, these extracted features were trained and classified using k-means clustering algorithm. Marium et al. (2017) proposed hand gesture recognition system using convexity algorithm approach. The authors applied this filtering approach on hand gesture images with same background. The authors obtained 90.7% of sensitivity, 82.1% of specificity and 87.5% of accuracy. The main limitation of this approach is that the proposed algorithm produced optimum results if the background of the hand gesture image is static. Ashfaq and Khurshid (2016) used Gabor filtering approach for converting the spatial domain format hand gesture image into multi-class domain format image. Then, the authors applied both Bayesian and Naïve Bayes classifier on Gabor transformed hand gesture image in order to classify the test hand gesture image into different classes. The authors obtained high level of classification accuracy on Naïve Bayes classifier than the Bayesian classification methodology due to its simple architecture pattern.

Rahman and Afrin (2013) used support vector machine (SVM) classification approach for classifying the hand gesture images into various classes. The authors achieve 89.6% of sensitivity, 79.9% of specificity and 85.7% of accuracy. The error rate was high in this method, and this is not suitable for fast moving background and foreground object images. Rao et al. (2009) developed hand gesture recognition system using hidden Markov model. The authors constructed Markov model for foreground fingers in hand gesture image. This Markov model was used in both training and testing modes of binary classification approach. The authors produced 90.1% of sensitivity, 82.6% of specificity and 90.6% of accuracy. The classification time is high in this methodology as the mail limitation.

The following points are limitations of the conventional methods for gesture recognition.

- Conventional gesture recognition method used SVM and Naïve Bayes classifier, which required high number of training samples for gesture pattern recognition.
- The complexities of these algorithms are quite high in nature.

3 Proposed methodology

In this paper, the human hand gestures are detected and recognized using CNN classification approach. This process flow consists of hand ROI segmentation using mask image, fingers segmentation, normalization of segmented finger image and finger recognition using CNN classifier. Figure 2 shows the proposed flow of hand gesture recognition system.

The proposed algorithm for hand gesture recognition system is given in the following.

Start;

Step 1: Segment hand region ROI using hand mask images;

Step 2: Segment fingers using Connected Component Analysis algorithm;

Step 3: Classify the segmented fingers using CNN classification algorithm;

End;

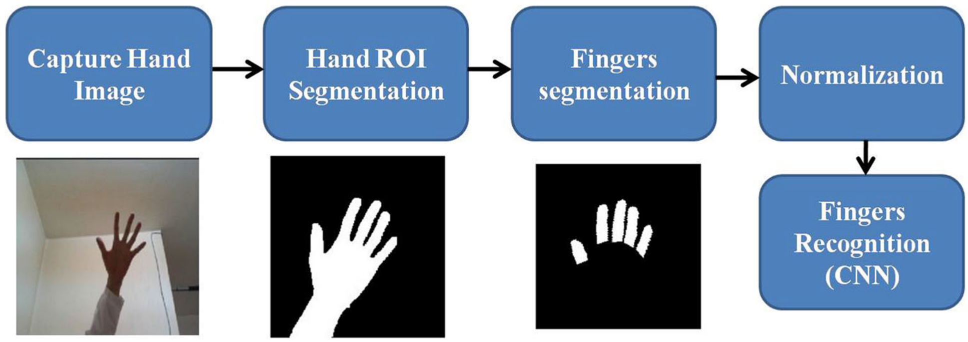


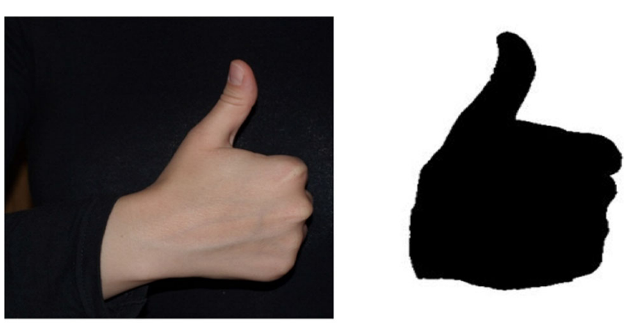
Fig. 2 Proposed hand gesture recognition system

3.1 Hand ROI segmentation

The hand region of the image is segmented from the whole image using mask images available in open access dataset (Rautaray and Agrawal 2015). The mask image is inverted, and it is convolved with the hand image which produces convolved image. The gray-level threshold is applied on the convolved image which produced hand region segmented image. Figure 3a shows the hand image, and Fig. 3b shows the mask image.

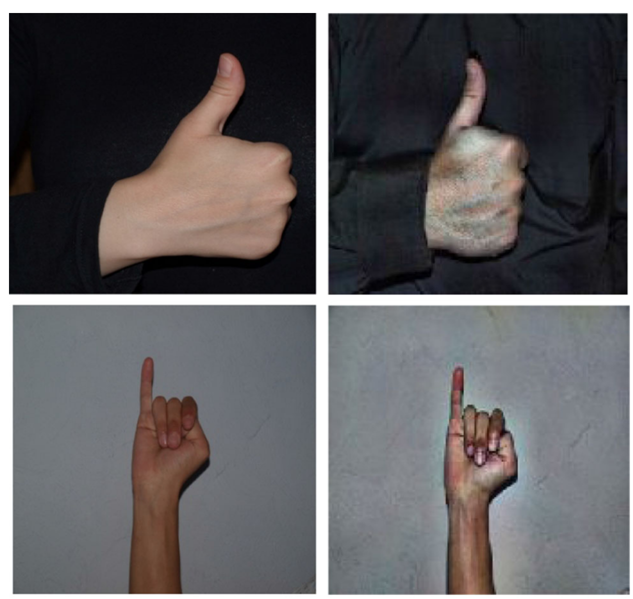
The parameter recognition rate is important for analyzing the performance of the proposed hand gesture recognition system. The adaptive histogram equalization (AHE) method (Kapil and Bhattacharya et al. 2016) is used as enhancement method for improving the contrast of each pixel in an image. In histogram equalization method, the entire image region is split into 3 * 3 patterns and the center pixel of each 3 * 3 patterns is contrast-enhanced by accumulating the maximum histogram count to the center pixel. In the case of AHE method, the 3 * 3 patterns are overlapped with each other and then each center pixel of 3 * 3 patterns is contrast-enhanced.

Figure 4a shows the different hand gesture images, and Fig. 4b shows the enhanced hand gesture images.



(a) (b)

Fig. 3 **a** Hand image and **b** mask image



(a) (b)

Fig. 4 **a** Source hand gesture images and **b** enhanced hand gesture images

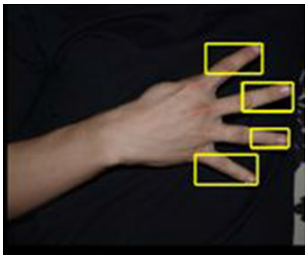


Fig. 5 Segmented finger region in hand image

3.2 Fingers segmentation

In this paper, connected component (CC) analysis algorithm is used in order to segment the finger tips from hand image. The finger tip from each hand image is detected and noted as finger peak. The aim of this finger tips segmentation is to count the number of fingers opened in gesture posture. To detect the finger tip in hand image, the m * n window is placed over the hand image which starts from left most and passed over the image toward the right. The value of 'm' and 'n' is chosen as odd numbers in order to improve the accuracy of the detection. For each scanning from left to right, the number of '1's is detected and counted in window. The number of counted ones is stored in an array. The number of ones in window is threshold as 10. The window scanning is stopped when the count goes beyond the value of 10. From top to bottom, the region is marked as finger. Figure 5 shows the segmented finger region in hand image, and the algorithm used for finger region segmentation is illustrated in the following algorithm.

3.3 Deep learning convolutional neural network classifier

The segmented finger regions from hand image are given to the CNN classification algorithm which classifies the image into various classes (Hong et al. 2012). This classification algorithm is explained in the following sections.

Over fitting and back propagation are the present problems of neural networks, and hence, it is not able to produce high classification accuracy for pattern recognition. In order to overcome such limitations in neural networks, CNN is used for obtaining high classification accuracy for hand gesture recognition. CNN has only forward path which does not have any feedback path for classifications. CNN is a kind of deep learning classification methodology which has many successful records in image analysis and classifications tasks. In this paper, CNN classification methodology is proposed to classify each input hand sign gesture image into various classes with high classification accuracy.

Figure 6 shows the developed CNN architecture which is used in this paper for hand gesture image classifications. It consists of convolutional filters which perform multiplication of kernel with input image (7 * 7 size), pooling and fully connected layers as shown in Fig. 6. Figure 7 shows the detailed internal architecture of proposed CNN classifier for hand gesture recognition which depicts convolution layers and fully connected layers. The fully connected layers produce *N* number of output classes.

The proposed CNN architecture consists of five convolutional layers and one fully connected layer with 1024 units. The first convolutional layer is designed with Gaussian filter of size 3 * 3 for 32 filters, second convolutional layer is designed with Gaussian filter of size 3 * 3 for 64 filters, third convolutional layer is designed with

Inputs: Hand gesture image and m*n window;
Output: Fingers detected image;

Start
Step 1: Move Width (m)* Height (n) window on hand gesture image from the position (i,j) as set to (0,0);
Step 2: Check its center pixel p(i,j) on m*n window as either 1 or 0;
Step 3: If (p (i,j)==1)
{
Check its neighboring pixel as 1
Set new label 'a' if condition is not satisfied;
Else
Set parent label of previous pixel;
}
Else
Step 4: Move m*n window by one pixel towards right;
End

Gaussian filter of size 3 * 3 for 128 filters, fourth convolutional layer is designed with Gaussian filter of size 3 * 3 for 256 filters, and fifth convolutional layer is designed with Gaussian filter of size 3 * 3 for 512 filters. The final fully connected layer is a standard feed forward neural network. This fully connected layer produces final classification responses.

3.3.1 Convolutional layers

Each convolutional layers act as feature extractors, and they extract individual feature set from the input source image for classification process. The feature map is constructed by integrating the neuron factors which are obtained from convolutional layers. All neurons within a feature map have weights that are constrained to be equal; however, different feature maps within the same convolutional layer have different weights so that several features can be extracted at each location.

The feature map at ith level can be determined using the following equation,

$$Y_i = f(W_i * I)$$

where W is internal weight and I is the input source image.

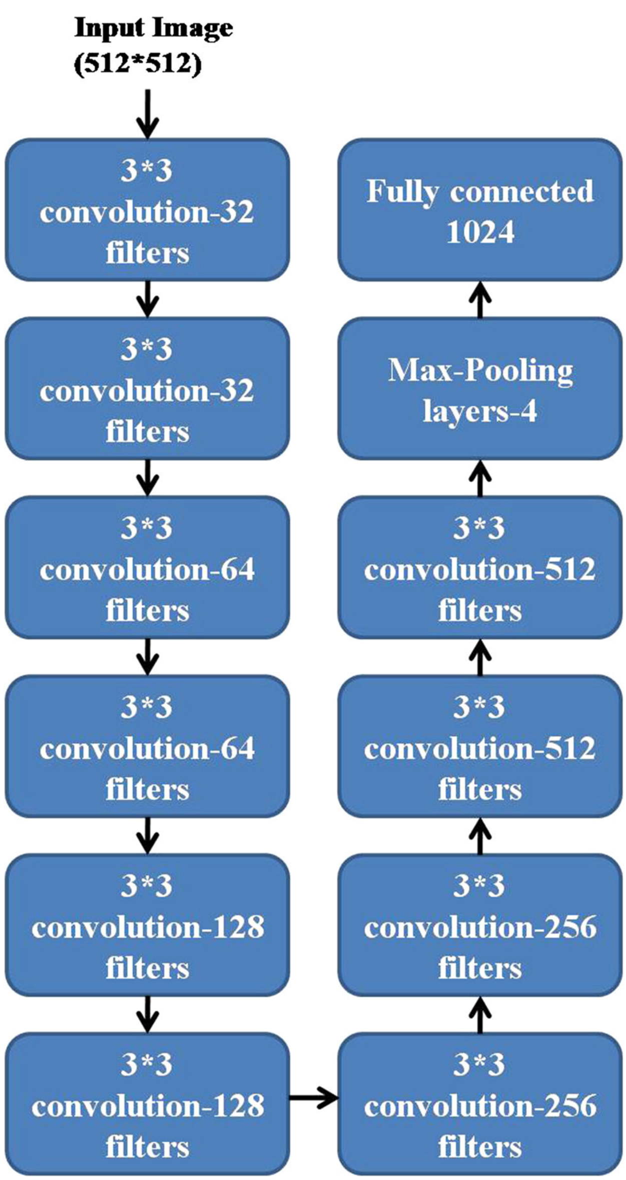


Fig. 6 Developed CNN architecture used in hand gesture recognition

3.3.2 Pooling layers

The purpose of the pooling layers is to reduce the spatial resolution of the feature maps which are obtained from convolutional layers. There are two pooling techniques used in pattern recognition as average pooling and max pooling, as depicted in Fig. 8. The average pooling demolishes the originality of the source image pixel, and the max pooling retains the original pixel value in source hand gesture image. Hence, in this paper, max pooling aggregation methodology is used which determines the maximum value from each feature set map and passes these maximum feature set values to the next layer. This can be illustrated in the following equation.

$$P_i = \mathrm{Max}(Yi)$$

In this paper, max pooling with a filter of size 2×2 with a stride of 2 is commonly used in practice. This paper uses 4 numbers of max pooling layers in order to obtain the optimum classification accuracy.

3.3.3 Fully connected layers

The fully connected layers that follow the convolutional and pooling layers interpret these feature representations and perform the function of high-level reasoning.

The first convolutional layer receives training samples along with test samples. The training set along with test sample forms the test vector which can be further convolved with 3 * 3 mask Gaussian filter for producing its corresponding hypotheses. Each hypothesis is considered as new test samples. The second convolutional layer receives this new test sample along with new training patters which was produced by the previous convolutional layer as shown in Fig. 9. The same procedure is followed until test outcomes received from last convolutional layer.

The output from CNN classification algorithm produces eight different classes, and each class represents the individual hand gesture. Figure 10 shows the training hand gesture images with different postures and backgrounds.

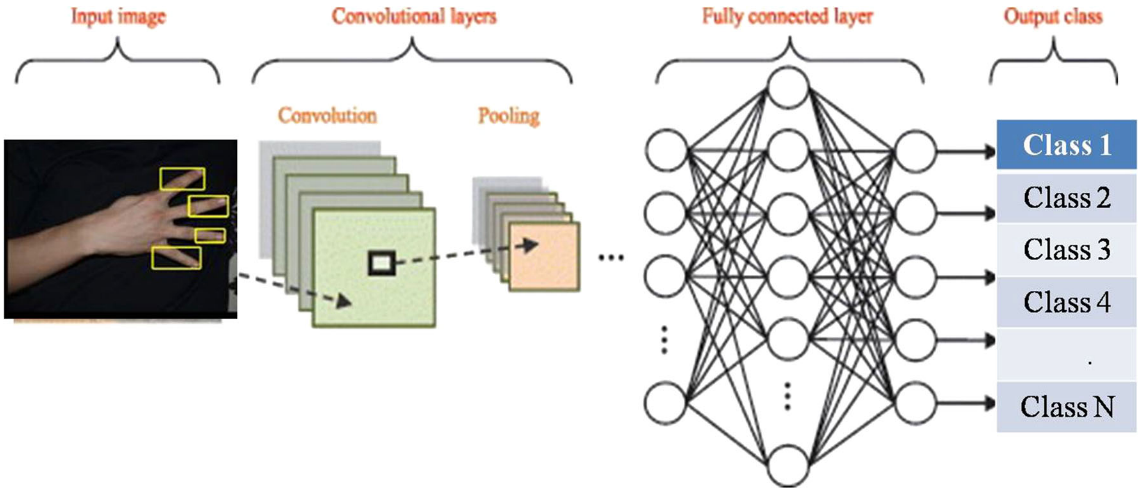


Fig. 7 Internal architecture of proposed CNN classifier for hand gesture recognition

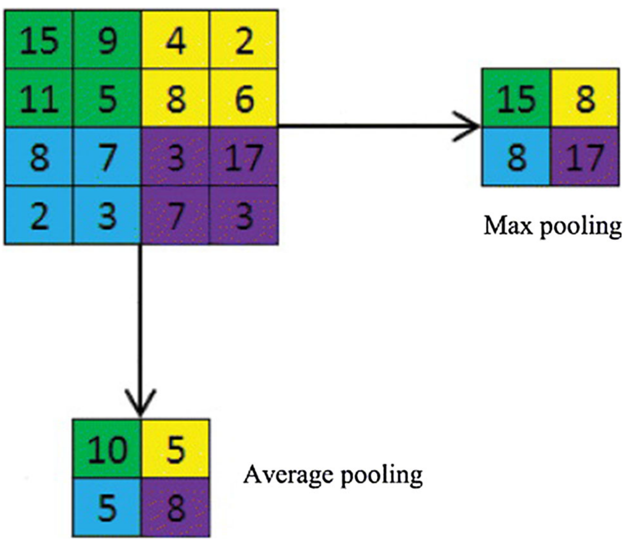


Fig. 8 Illustrations of average and max pooling

4 Results and discussion

The proposed hand gesture detection and recognition methodology is simulated using Python version 2.7 open-source simulation software. This open-source software is authorized by Python scientific distributions. The Python software package includes spyder, Anaconda, keras, panda and theano modules. These modules are license free and available as open tools. Each module is integrated in Python kernel, and Python programming language is used to simulate the proposed work. The Python software is installed in windows 8 with 4 GB internal memory and executed in core i3 processor.

The proposed hand gesture detection and recognition methodology is applied on the images which are openly available in Kawulok et al. (2012) dataset. This dataset consists of large number of gesture patterns which were taken at different angles of orientation. The gesture images in this dataset were obtained or captured with different background environment under different lighting conditions. In this paper, eight different gestures and 200 images for each gesture posture are used. These 1600 gesture images from this open-source dataset are used as training set in this paper. The testing set contains 800 images which represent eight different gestures. The hand gesture images in both training and testing dataset are independent with each other. This dataset also contains 700 non-hand gesture images. The following parameters are used to evaluate the out performance of the proposed work which is stated in this paper.

$$\mathrm{Sensitivity(Se)} = \mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$$
$$\mathrm{Specificity(Sp)} = \mathrm{TN}/(\mathrm{TN}+\mathrm{FP})$$
$$\mathrm{Accuracy(Acc)} = (\mathrm{TP}+\mathrm{TN})/(\mathrm{TP}+\mathrm{FN}+\mathrm{TN}+\mathrm{FP})$$
$$\text{Recognition rate} = \text{Number of images correctly classified/Total number of images}$$

where TP is the true positive which represents the total number of correctly recognized hand gesture images (154 images) and TN is the true negative which represents the total number of correctly recognized non-hand gesture images (50 images). FP is the false positive which represents the total number of wrongly recognized hand gesture images (six images), and FN is the false negative which represents the total number of wrongly recognized non-hand gesture images (five images). The value of sensitivity, specificity and accuracy lies between 0 and 100, and they are determined in %. Higher values of these parameters show that the efficiency of the proposed hand gesture detection and recognition methodology is high. Table 1 shows the analysis of recognition rate of proposed method

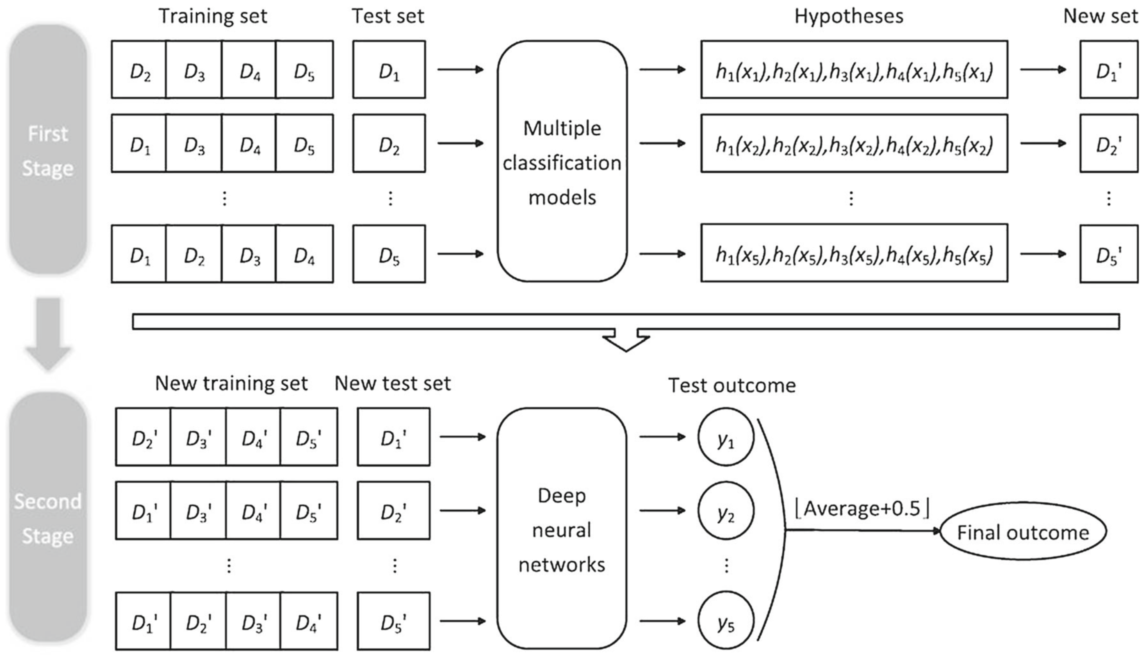
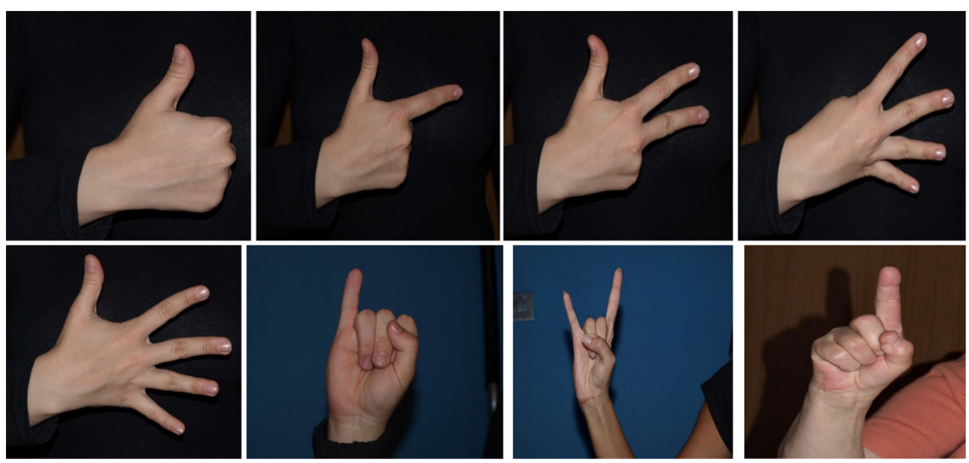


Fig. 9 Deep learning classification processes

Fig. 10 Training hand gesture images with different postures and backgrounds



with respect to different gesture class images and pooling techniques.

Table 2 shows the performance analysis of proposed hand gesture recognition method in terms of sensitivity, specificity and accuracy with recognition rate. The proposed hand gesture detection and recognition methodology using CNN classification approach stated in this paper achieves 96.8% of sensitivity, 89.2% of specificity, 94.8% of accuracy and 96.2% of recognition rate.

The proposed hand gesture detection methodology is applied only with CNN classification approach, and this methodology achieved 91.5% of sensitivity, 82.7% of specificity and 91.6% of accuracy with 90.7% of recognition rate, as stated in Table 2. These simulation results are not suitable for real-time recognition of hand gesture images for different applications. Hence, the CNN classification approach is combined with CC algorithm in order to improve the performance of the hand gesture detection system. This integrated approach achieved 96.8% of sensitivity, 89.2% of specificity and 94.8% of accuracy with 96.2% of recognition rate.

The CNN with CC methodology improved 5.4% of sensitivity rate from CNN without CC methodology. The CNN with CC methodology improved 7.8% of specificity rate from CNN without CC methodology. The CNN with CC methodology improved 3.4% of accuracy rate from CNN without CC methodology. The CNN with CC methodology improved 6% of recognition rate from CNN without CC methodology. The proposed hand gesture recognition system (AHE + connected component

Table 1 Analysis of recognition rate of proposed method with respect to different gesture class images and pooling techniques

Gesture class	Number of gesture images	Number of gestures correctly recognized	Recognition rate (%)	
			Using average polling	Using max polling
Class 1	200	196	98	99
Class 2	200	195	97.5	98.5
Class 3	200	194	97	98.5
Class 4	200	194	97	99
Class 5	200	193	96.5	99
Class 6	200	192	96	98.5
Class 7	200	193	96.5	98.5
Class 8	200	193	96.5	98.5
	1600	1550	96.8	98.7

Table 2 Performance analysis of proposed hand gesture recognition method

Performance analysis parameters	Experimental results (%)
Sensitivity	98.1
Specificity	93.4
Accuracy	96.2
Recognition rate	98.7

Table 3 Analysis of proposed hand gesture detection methodologies using CNN

Performance analysis parameters (%)	CNN classification approach	Connected component analysis + CNN classification approach	AHE + connected component analysis + CNN classification approach
Sensitivity	91.5	96.8	98.1
Specificity	82.7	89.2	93.4
Accuracy	91.6	94.8	96.2
Recognition rate	90.7	96.2	98.7

analysis + CNN classification approach) obtained 98.1% of sensitivity, 93.4% of specificity and 96.2% of accuracy with 98.7% of recognition rate. From Table 3, it is very clear that the proposed hand gesture detection methodology using CNN classification approach combined with CC algorithms provides optimum results when compared with CNN classification approach alone.

The proposed methodology is also analyzed using SVM classification approach. The methodology with AHE + connected component analysis+ SVM classification approach achieved 92.1% of sensitivity, 89.9% of specificity and 93.5% of accuracy with 91.5% of recognition rate, as depicted in Table 4.

Table 4 Analysis of proposed hand gesture detection methodologies using SVM

Performance analysis parameters (%)	SVM classification approach	Connected component analysis + SVM classification approach	AHE + connected component analysis + SVM classification approach
Sensitivity	89.1	90.5	92.1
Specificity	78.7	87.5	89.9
Accuracy	87.5	91.6	93.5
Recognition rate	88.2	90.5	91.5

The proposed CNN-based hand gesture recognition system is compared with SVM classifier. The kernel of this SVM classifier is categorized into two types as linear and nonlinear. The regression pattern of the linear kernel SVM is exponentially curved than the regression pattern of the nonlinear kernel SVM classifier. In this paper, the CNN classification results are compared with linear kernel SVM type.

Table 5 shows the performance comparisons of proposed hand gesture detection and recognition methodology with state of arts in terms of sensitivity, specificity and accuracy. The proposed hand gesture detection and recognition methodology stated in this paper is compared with the conventional methods as Wang et al. (2018), Marium et al. (2017), Rahman and Afrin (2013) and Rao et al. (2009). Wang et al. (2018) used particle filtering method for hand gesture recognition and achieved 92.1% of sensitivity, 84.7% of specificity and 90.6% of accuracy. Marium et al. (2017) used convexity algorithm for hand gesture recognition and achieved 90.7% of sensitivity, 82.1% of specificity and 87.5% of accuracy. Rahman and Afrin (2013) used support vector machine for hand gesture recognition and achieved 89.6% of sensitivity, 79.9% of specificity and 85.7% of accuracy. Rao et al. (2009) used hidden Markov model for hand gesture recognition and achieved 90.1% of sensitivity, 82.6% of specificity and 90.6% of accuracy.

The analysis work is performed using recognition time which can be computed by the total execution time for recognizing the single hand gesture image for an automated process. It is essential for real-time applications for processing the hand gestures under different environmental conditions. Table 6 shows the performance comparisons of proposed methodology with state of art in terms of recognition time (s).

The proposed CNN classification approach based on CC algorithm with enhancement technique stated in this paper

Table 5 Performance comparisons of proposed methodology with state of art

Authors and year	Methodology	Sensitivity (%)	Specificity (%)	Accuracy (%)
Proposed methodology (in this paper)	CNN classification approach based on CC algorithm with enhancement technique	98.1	93.4	96.2
Sharma et al. (2019)	Static hand gesture algorithm	93.2	91.1	91.8
Wang et al. (2019)	Super-pixel	94.2	90.8	92.6
	Earth mover's distance classification algorithm			
Wang et al. (2018)	Particle filtering method	92.1	84.7	90.6
Chaikhumpha and Chomphuwiset (2018)	Hidden Markov models	90.5	91.7	91.6
Marium et al. (2017)	Convexity algorithm	90.7	82.1	87.5
Rahman and Afrin (2013)	Support vector machine	89.6	79.9	85.7
Rao et al. (2009)	Hidden Markov model	90.1	82.6	90.6

Table 6 Performance comparisons of proposed methodology with state of art in terms of recognition time (s)

Authors and year	Methodology	Recognition time (s)
Proposed methodology (in this paper)	CNN classification approach based on CC algorithm with enhancement technique	0.356
Proposed methodology (in this paper)	SVM classification approach based on CC algorithm with enhancement technique	0.674
Sharma et al. (2019)	Static hand gesture algorithm	0.810
Wang et al. (2019)	Super-pixel	0.768
	Earth mover's distance classification algorithm	
Wang et al. (2018)	Particle filtering method	0.789
Chaikhumpha and Chomphuwiset (2018)	Hidden Markov models	0.973
Marium et al. (2017)	Convexity algorithm	1.372
Rahman and Afrin (2013)	Support vector machine	2.102
Rao et al. (2009)	Hidden Markov model	1.357

consumed 0.356 s as recognition time, whereas the conventional methods in Wang et al. (2018) consumed 0.789 s, Chaikhumpha and Chomphuwiset (2018) consumed 0.973 s, Marium et al. (2017) consumed 1.372 s, Rahman and Afrin (2013) consumed 2.102 s and Rao et al. (2009) consumed 1.357 s. From Table 4, it is very clear that the proposed methodology for gesture recognition consumed less recognition rate when compared with conventional methodologies.

5 Conclusions

In this paper, deep learning convolutional neural network-based hand gesture detection and recognition methodology is proposed. This proposed method segments the finger tips from the hand gesture image, and then, this finger tips are given as input to the CNN classifier. The CNN classification approach trains and classifies the test hand gesture image which is obtained from open access image dataset. The performance of the proposed hand gesture detection and recognition methodology is analyzed in terms of sensitivity, specificity, accuracy and recognition rate. The proposed hand gesture detection and recognition methodology using CNN classification approach stated in this paper achieves 98.1% of sensitivity, 93.4% of specificity, 96.2% of accuracy and 96.2% of recognition rate.

Compliance with ethical standards

Conflict of interest All authors state that there is no conflict of interest.

Human and animal rights Humans/animals are not involved in this work.

References

Ashfaq T, Khurshid K (2016) Classification of hand gestures using gabor filter with bayesian and naïve bayes classifier. Int J Adv Comput Sci Appl 7(3):276–279

Chaikhumpha T, Chomphuwiset P (2018) Real—time two hand gesture recognition with condensation and hidden Markov models. In: 2018 international workshop on advanced image technology (IWAIT), Chiang Mai, 2018, pp 1–4

Elmezain M, Al-Hamadi A, Michaelis B (2010) A robust method for hand gesture segmentation and recognition using forward spotting scheme in conditional random fields. In: Proceedings of the 20th international conference on pattern recognition (ICPR'10), August 2010, pp 3850–3853

Hong J, Kim ES, Lee H-J (2012) Rotation-invariant hand posture classification with a convexity defect histogram. In: IEEE international symposium on circuits and systems (ISCAS) 20–23 May 2012, pp 774–777

Kapil Y, Bhattacharya J (2016) Real-time hand gesture detection and recognition for human computer interaction. In: Berretti S, Thampi S, Srivastava P (eds) Intelligent systems technologies and applications. Springer, Cham, pp 559–567

Kawulok M, Grzejszczak T, Nelapa J, Knyc M (2012) Database for hand gesture recognition. http://sun.aei.polsl.pl/~mkawulok/gestures/. Accessed 12 May 2018

Liu K, Chen C, Jafari R, Kehtarnavaz N (2014) Fusion of inertial and depth sensor data for robust hand gesture recognition. Sens J IEEE 14(6):1898–1903

Manresa-Yee C, Varona J, Mas R, Perales FJ (2005) Hand tracking and gesture recognition for human–computer interaction. Electronic letters on computer vision and image analysis 5(3):96–104

Marium A, Rao D, Crasta DR, Acharya K, D'Souza R (2017) Hand gesture recognition using webcam. Am J Intell Syst 7(3):90–94

Mitra S, Acharya T (2007) Gesture recognition: a survey. IEEE Trans Syst Man Cybern Part C Appl Rev 37(3):311–324

Park S, Yu S, Kim J, Kim S, Lee S (2012) 3D hand tracking using Kalman filter in depth space. EURASIP J Adv Signal Process 2012(1), 36

Rahman MH, Afrin J (2013) Article: hand gesture recognition using multiclass support vector machine. Int J Comput Appl 74(1):39–43

Rao J, Gao T, Gong Z, Jiang Z (2009) Low cost hand gesture learning and recognition system based on hidden markov model. In: Proceedings of the 2009 2nd international symposium on information science and engineering (ISISE '09), IEEE, Shanghai, China, December 2009, pp 433–438

Rautaray SS, Agrawal A (2015) Vision based hand gesture recognition for human computer interaction: a survey. Artif Intell Rev 43(1):1–54

Ren Z, Yuan J, Meng J, Zhang Z (2013) Robust part-based hand gesture recognition using kinect sensor. IEEE Trans Multimed 15(5):1110–1120

Sharma S, Jain S, Khushboo (2019) A static hand gesture and face recognition system for blind people. In: 2019 6th international conference on signal processing and integrated networks (SPIN), Noida, India, 2019, pp 534–539

Suguna R, Neethu PS (2017) Hand gesture recognition using shape features. Int J Pure Appl Math 117(8):51–54

Tauseef H, Fahiem MA, Farhan S (2009) Recognition and translation of hand gestures to Urdu alphabets using a geometrical classification. In: Proceedings of IEEE international conference on visualisation (VIZ), Barcelona, Spain, pp 213–217

Wang Z, Chen B, Wu J (2018) Effective inertial hand gesture recognition using particle filtering based trajectory matching. J Electric Comput Eng 2018, 6296013

Wang Y, Jung C, Yun I, Kim J (2019) SPFEMD: super-pixel based finger earth mover's distance for hand gesture recognition. In: ICASSP 2019–2019 IEEE international conference on acoustics, speech and signal processing (ICASSP), Brighton, United Kingdom, 2019, pp 4085–4089

Yao Y, Fu Y (2014) Contour model-based hand-gesture recognition using the Kinect sensor. IEEE Trans Circ Syst Video Technol 24(11):1935–1944

Yasukochi N, Mitome A, Ishii R (2008) A recognition method of restricted hand shapes in still image and moving image as a man–machine interface. In: IEEE conference on human system interactions, pp 306–310

Yrk E, Konukoglu E, Sankur B, Darbon J (2006) Shape-based hand recognition. IEEE Trans Image Process 15(7):1803–1815