ISSN 2278-3091
Volume 9, No.5, September - October 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse61952020.pdf
https://doi.org/10.30534/ijatcse/2020/61952020

# Deep Learning based Dynamic Hand Gesture Recognition with Leap Motion Controller

**Dr.V .Elizabeth Jesi[1], Dr. Shabnam Mohamed aslam[2], Ms. Ruhi Fatima[3]**
[1]Department of Information Technology, college of Engineering and Technology, Faculty of Engineering and Technology, SRM Institute of Science and Technology, SRM Nagar, Kattankulatur, 603203, Kancheepuram, Chennai, TN, India . jesiv@srmist.edu.in
[2]Department of Information Technologies, College of Computer and Information Sciences, Majmaah University, Al Majmaah, 11952, KSA . s.aslam@mu.edu.sa
[3]Department of Computer Sciences, College of Computer and Information Science, Majmaah University, Al Majmaah, 11952 , KSA. ruuhiwahaj@yahoo.co.in

## ABSTRACT

Dynamic hand gesture recognition (DHGR) is an important yet difficult thing in the pattern recognizing groups and research communities. The latest establishment of new methods for acquisitions like Leap Motion and the Kinect permits deriving a vital data about hand pose which could be utilized for precisely recognizing the gestures. The previous system designed a Hidden Conditional Neural Field (HCNF) classifier with Leap Motion Controller (LMC) for DHGR. The descriptive information about the gestures by hand is obtained with the assistance of LMC. It keeps a note of the movement of hands and fingers in a digitalized manner and provides some points related to every gesture. Training and recognition is done with the help of this. However, HCNF has issue with overfitting problem due to the great expressive power, especially when trained on a small corpus. It reduces the classification accuracy. To solve this problem the proposed system designed a deep learning approach for DHGR. This system has 2 main phases such as feature extraction and classification. Initially, the acquired data from LMC is taken as an input and perform feature extraction from it. The given feature vector single-finger features and double-finger features. Based on extracted feature vector, Weighted Bias Mean based Convolutional Neural Network (WBMCNN) is utilized for DHGR. The demonstrated technique is analyzed on 2 sets of data of dynamic hand gesture accompanied by frames got with a Leap Motion Controller. The LeapMotion-Gesture3D recognizes with an accuracy of92.8% and Handicraft-Gesture dataset recognizes to an accuracy of 96.7%. This method can be specifically used for DHGR. The experimental result shows that it has a better capability in comparison with the previous methods in terms of accuracy and execution time.

**Key words:** Deep learning, Hand gestures**,** Leap Motion controller (LMC), Weighted Bias Mean based Convolutional Neural Network (WBMCNN).

## 1. INTRODUCTION

In the day today communication among people gesturing has an important part and hence it is crucial in augmented reality (AR)/Virtual Reality (VR) interaction. Humans commonly communicate their thoughts and feelings using gestures [1-2]. Gesture identification is the procedure by which the systems can identify the gestures of the humans so that it can respond to it. These actions can be done by the body part such as head, face, arms, hands, and so on. Yet often we utilize our hands for expressing like waving the hands, greeting someone. Hand gestures are broadly used in multiple systems such as Human – Computer Interaction (HCI), robotics, sign language, human machine interaction. Due to development in the field of technology Human- Computer Interaction is now a potential field in past few years. Since long this tools are trying to use hand gestures in the place of joystick, mouse and keyboard [3-5].

Gestures can be of two types, static and dynamic. The first one is simple and doesn't require much computational skills while the latter is complicated but is more appropriate for real-time gestures [6].DHGR plays a vital part in recognizing the human actions. However, this is a demanding activity as there is a huge variation of structure and the serious occlusion in-between fingers. The monocular video recording is not as effective method as it may not be able to capture much of the real-time hand gestures. The latest depth sensors like LMC [7] and Microsoft Kinect sensor. These creates three-dimensional (3-D) depth data of the scene, have provided a lot to object segmentation as well as three-dimensional hand gesture recognition. The below figure shows how to make use of the LMC. This sensor could identify the movement of finger and palm above the sensors. The data which has been tracked contains the details about the placement of finger, direction and pace.

**Figure 1:** Use of Leap Motion. Leap Motion controller is the object in the center. It is connected to the system on the right side. Hand's interaction with the virtual object is tracked with the Leap motion.

Various strategies have been implemented to obtain data for hand gesture recognition system [8].Often some accessory hardware like data gloves, and color markers are needed for quick and precise DHGR. Apart from that other techniques derive the necessary features with the help of skin color to divide the hands and obtain the essential features. This is a simple, natural and cost-effective methods in comparison with earlier methods

The paper covers the following details: section 2tells the importance of systems for recognizing hand gesture that includes segmentation, features extraction, and recognition. The proposed systems are presented in third section. Section 4 provided experimental results. At the end, conclusion is given in fifth section.

## 2. LITERATURE REVIEW

The author and his colleagues (2011)proposed a method for recognizing hand gestures with the path of hand motion with the help of classifier called Hidden Markov Model (HMM). They created 8 types of hand gestures with 1 hand or both the hands. In their design, initially face is localized later palm of the hand from skin region is located with the help of maximum circle plate mapping and in the process of feature extraction orientation is used as an important characteristic. This test result shows 96% accuracy with HMM recognition model [9].

The author and his colleagues (2013) presented a method with Histrograms of Oriented Gradients (HOG) to eliminate the disturbances created due to cluttered background while modelling and localizing the hand Standard database is developed using the oriented gradients' histrograms. First the hands are located then temporal gesture's motion trajectory are derive and finally it is developed. To recognize Mahalanobis distance is applied. This method is examined on six common gestures and they got a mean accuracy of 91.7%. The mentioned system can be applied for complicated gestures [10].

The author and his colleagues (2013) proposed a powerful part-based method with the help of kinect sensor. In their method initially color images as well as depth maps of that that image is recorded with kinect sensor. The depth maps help to identify hand in spite of cluttered background with the help of depth thresholding. The hand is then depicted by its finger parts with the help of time series curve. Finger-Earth Mover's Distance (FEMD) is utilized for recognizing gestures. FEMD is a dissimilarity measure that is capable of recognizing noisy hand contours in comparison with other methods used for recognizing. It is also capable enough to vary in scale, orientation, local distortions as well as background conditions. This method is tested with a dataset of ten persons, ten gestures and ten cases/gestures and obtains an accuracy of 93.2% and it is applied on2 day today applications [11].

Marin et al (2014) designed new method for recognizing hand gestures with Leap Motion as well as kinect sensors. An ad-hoc feature set depending on placements of the tip of finger is analyzed and used as an input for multi-class Support Vector Machine (SVM) classifier for recognizing the gestures done. By merging the 2 sets, it would be able to gain an enhanced accuracy on daily basis. Distance, angle between tip of fingers as well as orientation of the hand are derived. This gained an accuracy of 91.28% [12].

The author (2017) designed a technique driven by sparsity of micro-Doppler analysis of DHGR using radar sensors. Initially, sparse representations of the echoes that are sent back from the gestures that are dynamic are collected with the Gaussian-windowed Fourier dictionary. After that, the orthogonal matching pursuit algorithm is utilized to derive the hand gestures' the micro-Doppler features. At last, the nearby or adjacent classifier is merged to the modified Hausdorff distance for DHGR with the sparse micro-Doppler features. Test results using real radar data prove that it can recognize with an accuracy rate of more than 96% under average disturbances [13].

The author (2017) designed a novel motion feature built upon Recurrent Neural Network (RNN) for skeleton-based DHGR. The finger motion features are derived to depict movement of fingers and global motion features are used to explain the overall skeletal change in position of the hand. Later they are used as input for bidirectional RNN accompanied by sequences of skeletal images, that can increase the motion features for Recurrent Neural Network and enhance the capability of classification. Experimental results show that the designed strategy attains better results [14].

## 3. PROPOSED METHODOLOGY

In this proposed research work, Weighted Bias Mean based Convolutional Neural Network (WBMCNN) is utilized for DHGR. The systematic framework is included in 2 steps: first one is feature extraction; the other is using WBMCNN classifier for classifying. The flow diagram of the proposed work is shown in figure 2.

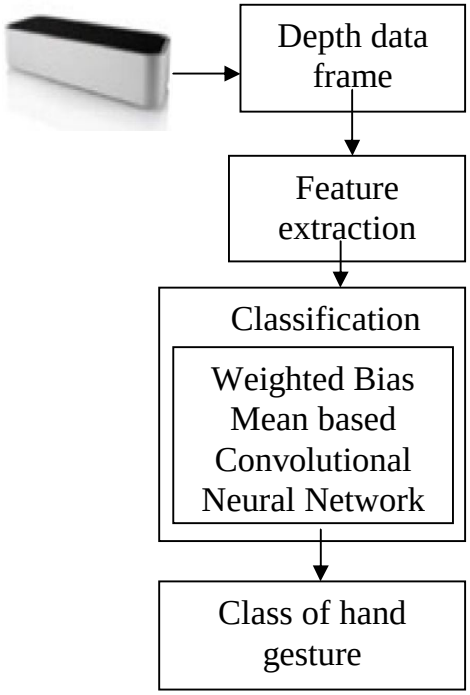


**Figure 2:** Flow chart of the events

### 3.1 Feature Extraction

Contrary to the Kinect sensor, the Leap Motion controller gives depth data frames which has details about position of the fingers, position of the hand, scaling data, frame timestamp, rotation, etc. Hence, the time taken for extracting features is decreased in Leap Motion Control in comparison to Kinect sensor.

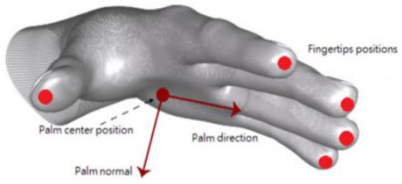


**Figure 3:** figure representing the direction of palm, Palm normal, Fingertips positions, and position of the center of the palm.

In this article we utilized features such as palm direction, palm normal, position of the fingertip and the data about center of the palms presented in Figure 3), which includes
1) $\vec{D}$ is the palm direction or unit direction vector. It starts from position of the palm to fingers.
2) $\vec{N}$ is the vector of normal posture of the palm.
3) $F_i$ depicts position of fingertips, i equals to 1,..., 5, depicts the three dimensional positions of each fingertips.
4) C is the position of finger tips that depicts the position of the center of the palm in three-dimensional space.

The single-finger and double finger feature vectors are demonstrated here. The single-finger features are derived to describe the association among the tip of fingers, we demonstrate double-finger features. Every features values normalized to the interval [0, 1].
The following are the kinds of features:

**1. Single-Finger Features:**

a) Fingertip-distances $Df_i = ||F_i - C||/M$, i = 1,..., 5. This depicts Euclidean distance from the tip of finger to the palm center. M is the Euclidean distance from center of the palm to tip of middle finger. Keep in mind, dividing by M normalizes the distances of the tip of finger to the interval [0, 1], and simultaneously this helps in making the strategy appropriate for hands in varying dimensions. The M is the scale factor. It can be calculated with the fully open palm as the person initiates the system.

b) Fingertip-angles $Af_i = \angle(F_i^P - C, \vec{D})/\pi$, i = 1,..., 5. Here $F_i^P$ is the projection of $F_i$ on the plane depicted by $\vec{N}$, are the angles that corresponds to the orientation of the projected fingertips to the direction of the palm $\vec{D}$. The angle between tip of finger is normalized with π.
c) Fingertip-elevations $Ef_i = sgn((F_i - F_i^P) \cdot \vec{N}) ||F_i - F_i^P)/M$,i = 1,..., 5. These are the distance of the tip of the finger from the plane that corresponds to the palm surface.

**2. Double-Finger Features:**

a) Adjacent fingertip-distances $Daf_i = ||F_i - F_{i+1}/M$, i = 1,..., 4. This depicts the Euclidean distances from one finger tip to the next.

b) Adjacent fingertip-angles $Aaf_i = \angle(F_i - F_{i+1})/\pi$, i = 1,..., 4. This depicts the absolute angles from one fingertip to the next.
The demonstrated feature vector has 2 important uses. One is single-finger features assists to give solution for issues of mislabeling that are commonly created due to doing the dynamic hand gesture in varying positions. The other is double-finger features assist in differentiating the various kinds of association among neighboring tips of fingers.

### 3.2 Classification using Weighted Bias Mean based Convolutional Neural Network (WBMCNN)

Based on the extracted feature vector, the intrusions are classified by using Weighted Bias Mean based Convolutional Neural Network (WBMCNN). Convolutional Neural Networks (CNNs) is a well-known deep learning architecture. The CNN which can add many hidden layers performing convolution as well as sub sampling for extracting the features ranging from minimum to maximum. Usually, this variety of networks is comprised of three layers [15-17]. The layers are convolution layers, sub sampling or pooling layers, as well as full connection layers [18]. The below

picture is an example of the architecture of a Convolutional Neural Network (CNN).

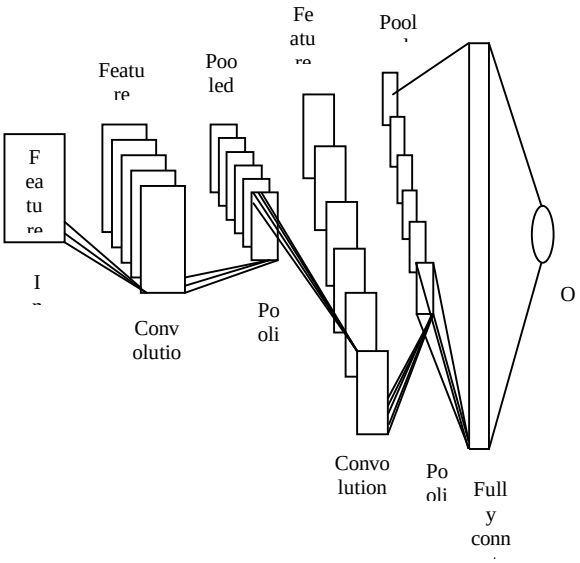


**Figure 4 :**The CNN architecture

**Convolution layer**

Here, extracted feature vectors are considered as an input and it is convolved with a kernel (filter). A pixel in the output is created by convolution of the input matrix accompanies by kernel. The convolution of the input feature and kernel generates results which can be utilized to create n output image features. Filter is the convolution matrix's kernel is commonly known as a filter and feature maps of size i∗i.is the output generated by the convolution of kernel and the input images.

The Convolution Neural Network can have various convolutional layers, the feature vectors the inputs and outputs of succeeding convolutional layers. Every convolution layer consists of a group of n filters. They are convolved alongside the input, and the depth of the created feature maps ($n*$) is equal to the numeric value of filters employed in the convolution operation.

$C_j^{(l)}$ is the $l^{th}$ convolution layer's output. It is comprised of feature maps. It is calculated as

$$C_i^{(l)} = B_i^{(l)} + \sum_{j=1}^{a_i^{(l-1)}} K_{i,j}^{(l-1)} * C_j^{(l)} \quad (1)$$

Here $B_i^{(l)}$ depicts bias matrix; $K_{i,j}^{(l-1)}$ depicts convolution filter or kernel of size a∗a which links the j-th feature map in layer $(l-1)$ to the i-th feature map in the same layer. The output $C_i^{(l)}$ layer is comprised of feature maps. In (2), initial convolutional layer $C_i^{(l-1)}$ is the space for input, that is,$C_i^{(0)} = X_i$ .

The kernel creates feature map. Next to the convolution layer, we can apply convolution function for the activation for nonlinear transformation of the outputs of the convolutional layer:

$$Y_i^{(l)} = Y(C_i^{(l)}) \quad (2)$$

Here ,$Y_i^{(l)}$depicts the output generated for activation function and $C_i^{(l)}$depicts input that it gets.

Sigmoid, tanh, and rectified linear units (ReLUs) are often used function for activating. Here, rectified linear units that is depicted as $Y_i^{(l)} = \max(0, Y_i^{(l)})$ are applied. As this function assists in decreasing the interaction and non-linear effects, it is commonly utilized in deep learning models. Rectified linear units gives the output as 0, in case anegative input is received and it gives the input value itself in case it is positive. The important benefit of the activation function is that due to the least error derivative in the saturating region, it helps in faster training; hence, the updates of the weights decrease or disappear. It is the vanishing gradient problem.

**Sub sampling Layer**

The sub sampling layer comes after the convolutional layer. The prime objective is to decrease the spatial dimensionality of the attribute maps derived from the preceding convolution layer. For this task, a mask of size b∗b is chosen, and the sub sampling operation among the mask and the attribute maps is done.

$$X_j^l = f(\beta_j^l \text{down}(X_j^{l-1}) + b_j^l) \quad (3)$$

Here, down (·) depicts a sub-sampling function. Every separate n-by-n feature will be added to this in the set of data inputs . Hence the output is smaller by n-times among both spatial dimensions. Every output map has its self-multiplicative bias $\beta$ as well as additive bias b.

In this right value of the b are provided from the weighted bias mean of feature map. The weighted bias mean is the sum of bias values of the entire feature with weight value divided by the total weight value of the feature vector.

$$\text{Weighted Bias Mean (WBM)} = \frac{\sum_{i=1}^{n} w_i bias_i}{\sum_{i=1}^{n} w_i} \quad (4)$$

Where, $x_i \in x$ depicts feature vector demonstrated in the preceding section.

$w_i$ −weight value of the feature vector

**Full Connection**

The Softmax activation function is used by the output layer:

$$y_i^{(l)} = f(z_i^{(l)}), \text{ where } z_i^{(l)} = \sum_{i=1}^{m_i^{(l-1)}} w_{i,j}^{(l)} y_i^{(l-1)} \quad (5)$$

Where, $w_{i,j}^{(l)}$ are the weights which has to be set by the overall fully linked layer so as to depict every class; *f* depicts transfer function that shows the nonlinearity. The proposed system classifies the input feature vector into class of hand gesture.

## 4. EXPERIMENTAL RESULTS

The performance of the proposed Weighted Bias Mean based Convolutional Neural Network (WBMCNN) based hand gesture recognition system is evaluated by using MATLAB simulation tool. The 2 types of sets of data of dynamic hand gesture that is LeapMotion-Gesture3D dataset as well as Handicraft-Gesture dataset, with a Leap Motion Controller are utilized for the proposed work. Entire depth data frames of each sets of data obtained are accompanied by the Leap Motion Controller's unique API.

**1. LeapMotion-Gesture3D Dataset:**

In the current scenario many sets of dynamic hand gesture were obtained using Kinect sensor like the MSRGesture3D dataset. For comparing the capability of the technique for recognizing hand gestures with other methods, we constructed a dataset with an Leap Motion Controller, known as LeapMotion-Gesture3D, which is similar to the MSRGesture3D dataset. It has a subgroup of actions depicted by ASL. Overall, twelve gestures in this set of data have been added. They are bathroom, blue, finish, green, hungry, milk, past, pig, and store, where, *j*, z.

**2. Handicraft-Gesture Dataset:**

For evaluating our technique with more practical gestures, a set of data known as Handicraft-Gesture has been constructed. This consists of 10 gestures that are obtained from pottery skills. The gestures are poke, pinch, pull, scrape, slap, press, cut, circle, key tap, mow. They are shown in Figure5.The depth data in both sets of data is obtained with sixty frames per second. Overall ten people were assisting to construct the datasets and all gestures are done thrice. Hence, the LeapMotion-Gesture3D has 360depth data and Handicraft-Gesture dataset has set of depth data.

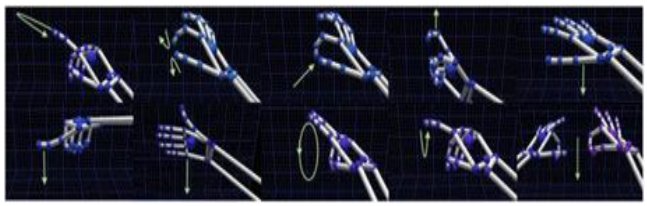

**Figure 5:**Representation of the 10 hand gestures from the pottery skill data. Left to right, top to bottom: Poke, Pinch, Pull, Scrape, Slap, Press, Cut, Circle, Key Tap, and Mow. The arrows represent the motion trajectories of the hand/fingers.

The performance of the proposed Weighted Bias Mean based Convolutional Neural Network (WBMCNN) and existing Hidden Conditional Random Field (HCRF) and Hidden Conditional Neural Field(HCNF) classifiers [19] are evaluated with respect to accuracy and execution time that is represented in table 1.

**Table 1:** Performance comparison

| Metrics | LeapMotion-Gesture3D dataset | | | Handicraft-Gesture dataset | | |
|---|---|---|---|---|---|---|
| | HCRF | HCNF | WBMCNN | HCRF | HCNF | WBMCNN |
| Accuracy | 87.8 | 89.5 | 92.8 | 95 | 95.8 | 96.7 |
| Execution time | 20 | 16 | 14 | 22 | 18 | 13 |

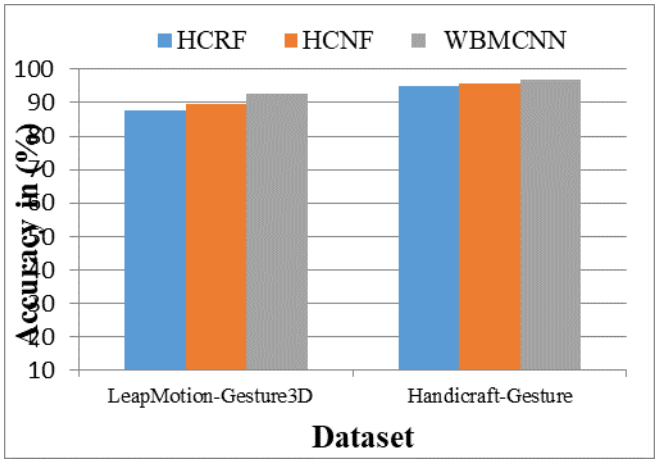


**Figure 6:** Accuracy comparison

The accuracy comparison of the proposed Weighted Bias Mean based Convolutional Neural Network (WBMCNN) method is compared with the existing Hidden Conditional Random Field (HCRF) and Hidden Conditional Neural Field(HCNF) classifier which is depicted in figure 6. The datasets are used in x-axis and accuracy is used as y-axis. In this proposed research work, Weighted Bias Mean based Convolutional Neural Network (WBMCNN) is used for

DHGR. In WBMCNN, bias values are updated optimally by using Weighted Bias Mean (WBM). It enhances the accuracy rate. The test results show that the demonstrated system achieves 92.8% of accuracy where as other methods such as HCRF and HCNF attains 87.8% and 89.5% respectively for LeapMotion-Gesture3D dataset. For Handicraft-Gesture dataset, WBMCNN attains 96.7% of accuracy while other methods such as HCRF and HCNF achieve 95% and 95.8% respectively.

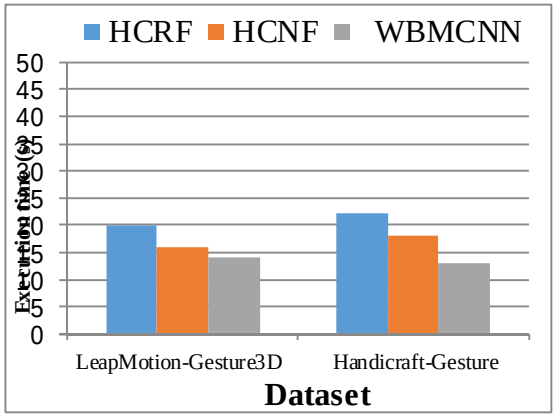


**Figure 7: Execution time comparison**

From the above figure 7, the graph demonstrates that the execution time comparison of the proposed Weighted Bias Mean based Convolutional Neural Network (WBMCNN) and existing Hidden Conditional Random Field (HCRF) and Hidden Conditional Neural Field(HCNF) approaches. Datasets are used in the x-axis and execution time is used in y-axis. From the test results, we can conclude that the proposed system obtains14s of execution time while other methods such as HCRF and HCNF achieves 20s and 16s respectively for LeapMotion-Gesture3D dataset. For Handicraft-Gesture dataset, the WBMCNN achieves 13s whereas other methods such as HCRF and HCNF attain 22s and 18s respectively.

## 5. CONCLUSION

The proposed system designed a Weighted Bias Mean based Convolutional Neural Network (WBMCNN) for DHGR. In this study, designed a new feature vector that can be used for DHGR. There are 2 important advantage of feature vector which is comprised of single-finger features, double finger features. One is single-finger features resolve the issue of mislabeling that happens because dynamic hand gesture is executed in various positions. The other is, double finger features assists in differentiating the various kinds of association among tips of neighboring fingers. Based on the extracted feature vector, hand gesture classification is done by using Weighted Bias Mean based Convolutional Neural Network (WBMCNN). The experimental result demonstrates that the proposed system has better capability compared to the system that are already present with respect to accuracy and execution time.

## REFERENCES


1. Rautaray, S. S., &Agrawal, A. **Vision based hand gesture recognition for human computer interaction: a survey**. Artificial intelligence review, vol. 43, no. 1, pp.1-54, 2015
2. Li, Y. **Hand gesture recognition using Kinect**. IEEE International Conference on Computer Science and Automation Engineering , pp. 196-199. IEEE, June, 2012.
3. Stergiopoulou, E., &Papamarkos, N. **Hand gesture recognition using a neural network shape fitting technique**. Engineering Applications of Artificial Intelligence, vol. 22, no. 8, pp.1141-1158, 2009.
4. Suk, H. I., Sin, B. K., & Lee, S. W. **Hand gesture recognition based on dynamic Bayesian network framework**. Pattern recognition, vol. 43, no. 9, pp. 3059-3072, 2010.
5. Kim, Y., &Toomajian, B. **Hand gesture recognition using micro-Doppler signatures with convolutional neural network**, *IEEE Access*, vol. *4*, pp. 7125-7130, 2016.
6. S.Mitra, and T. Acharya. **Gesture Recognition: A Survey**, IEEE Transactions on systems,Man and Cybernetics, Part C: Applications and reviews, vol. 37 no. 3, pp. 311- 324, 2007.
7. Guzsvinecz, T., Szucs, V., &Sik-Lanyi, C. **Suitability of the Kinect sensor and Leap Motion controller—a literature review**. *Sensors*, vol. *19*, *no.* 5, pp. 1072-1079, 2019.
8. Simei G. Wysoski, Marcus V. Lamar, SusumuKuroyanagi, Akira Iwata, **A Rotation Invariant Approach On Static-Gesture Recognition Using Boundary Histograms And Neural Networks**, IEEE Proceedings of the 9th International Conference on Neural Information Processing, 2002.
9. Chang-Yi Kao,Chin-ShyurngFahn, **A Human-Machine Interaction Technique: Hand Gesture Recognition Based on Hidden Markov Models with Trajectory of Hand Motion**, Procedia Engineering vol. 15, pp. 3739 – 3743, 2011.
10. Jing Lin, Yingchun Ding, **A temporal hand gesture recognition system based on hog and motion trajectory**, Optik vol. 12, no. 4, pp.6795– 6798, 2013.
11. Z. Ren, J. Yuan, J. Meng, and Z. Zhang, **Robust part-based hand gesture recognition using Kinect sensor**, IEEE Trans. Multimedia, vol. 15, no. 5, pp. 1110–1120, Aug. 2013.
12. Giulio Marin, Fabio Dominio and PietroZanuttigh, **Hand Gesture Recognition With Leap Motion And Kinect Devices**, International Conference on Image Processing (ICIP), 2014, pp. 1565-1569.
13. Li, G., Zhang, R., Ritchie, M., & Griffiths, H. **Sparsity-driven micro-Doppler feature extraction for dynamic hand gesture recognition**, *IEEE*

*Transactions on Aerospace and Electronic Systems*, vol. *54*, no. 2, pp. 655-665, 2017

14. Chen, X., Guo, H., Wang, G., & Zhang, L. **Motion feature augmented recurrent neural network for skeleton-based dynamic hand gesture recognition** In *2017 IEEE International Conference on Image Processing (ICIP)* IEEE. 2017, pp. 2881-2885.
15. Salamon, J., & Bello, J. P. **Deep convolutional neural networks and data augmentation for environmental sound classification**, *IEEE Signal Processing Letters*, vol. *24*, no. 3, pp. 279-283, 2017.
16. Meghana, A. S., Sudhakar, S., Arumugam, G., Srinivasan, P., & Prakash, K. B. **Age and Gender prediction using Convolution, ResNet50 and Inception** ResNetV2. *International Journal of Advanced Trends in Computer Science and Engineering*, vol. *9, no.* 2, pp. 1328-1334, 2020
17. Selvarathi, Indu, Kavysdharshini, logesh kumar and Mohamed yasher, **Human computer interaction using hand gesture recognition**, International Journal of Advanced Trends in Computer Science and Engineering, vol. 9, no. 2, pp. 1674 – 1677, 2020
18. Africa, A. D. M., Bulda, L. R., Marasigan, M. Z., & Navarro, I.F. **A study on number gesture recognition using neural network**, *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8, no. 4, pp. 1076-1082, 2019.
19. Lu, W., Tong, Z., & Chu, J. **Dynamic hand gesture recognition with leap motion controller**, IEEE Signal Processing Letters, vol. 23, no. 9, pp. 1188-1192, 2016.