Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Highly Accurate Trimesh and PointNet based algorithm for Gesture and Hindi air writing recognition

Jayash Kumar Sharma
Rajasthan Technical University
Rajeev Gupta
Rajasthan Technical University
Shakti Sharma (shakti.sharma.mathura@gmail.com)
Bennett University
Vinay Pathak
Chhatrapati Shahu Ji Maharaj University
Manoj Sharma
Bennett University

Research Article

Keywords: Gesture Recognition, PointNet, leap motion controller (LMC), trimesh

Posted Date: March 22nd, 2023

DOI: https://doi.org/10.21203/rs.3.rs-2702018/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Highly Accurate Trimesh and PointNet based algorithm for Gesture and Hindi air writing recognition

Jayash Kumar Sharma[1], Rajeev Gupta[1†], Shakti Sharma[2*†], Vinay Kumar Pathak[3†] and Manoj Sharma[4†]

[1*]Department of Electronic and Communications, Rajasthan Technical University, Akelgarh, Kota, 324010, Rajasthan, India.
[2]School of Computer Science Engineering and Technology, Bennett University, Techzone II, Greater Noida, 201310, Uttar Pradesh, India.
[3]Chhatrapati Shahu Ji Maharaj University, Kalyanpur, Kanpur, 208024, Uttar Pradesh, India.
[4]Department of Electrical Engineering, Bennett University, Techzone II, Greater Noida, 201310, Uttar Pradesh, India.

*Corresponding author(s). E-mail(s): shakti.sharma.mathura@gmail.com;
Contributing authors: jayash.sharma@gmail.com; rajeev_eck@yahoo.com; vinay@vpathak.in; mksnith@gmail.com;
†These authors contributed equally to this work.

Abstract

Air writing and gesture recognition has been one of the most comprehensive and intensively researched areas due to their wide variety of applications in several domains of Human-Computer Interaction (HCI), such as gaming, medicines, and automobiles. This article presents a highly accurate point-based algorithm for identifying dynamic air writing and free hand gesture recognition for Devanagari Hindi characters. Usually, the written gestures are limited in nature. In order to create a training data set, we are using Leap Motion Controller (LMC) that generates twenty-four sample points from the gesture drawn. In the pre-processing stage, a trimesh is generated using these sample points; the developed trimesh is then fed as input to PointNet. The algorithm developed measures the similarity of the drawn gesture with the training samples to identify and recognize the gesture drawn using Mahalanobis distance. The validation test indicates that the approach is quite accurate, giving a recognition rate of more than 97%. Comparative studies also show that the methodology performs better when trimesh is used as input for the PointNet and more training samples are used.

Keywords: Gesture Recognition, PointNet, leap motion controller (LMC), trimesh

1 Introduction

With increasing demand and applications of touch less interaction in our daily life, Human-Computer Interaction (HCI) has become really important and plays a crucial role in various fields. Touch-less HCI can be broadly categorized into voice recognition-based interaction and gesture or writing-based interaction. In gesture or air writing recognition, hand or hand-held equipment is utilized to make gestures or signs which are then recognized using gesture recognition systems.

The technology of interpreting and recognizing gestures or signs by making use of sensors is termed gesture recognition. Gesture recognition or air writing is one of the aspects of HCI and with increasing demands, the research to make this interaction as natural as possible has become intensive. Hands are the easiest way to make gestures or to represent ideas, actions, and words using different hand shapes that can be identified by gesture or writing recognition systems [1]. Gesture recognition requires sensing techniques to acquire gesture input from the user [1, 2]. The techniques to acquire inputs can be classified as vision-based techniques, glove-based techniques, and depth-based techniques. The vision-based techniques are based on the visuals or images that are captured using the cameras or webcams to gather the input and analyze them to identify the gestures [3, 4]. These are highly affected in the case of occlusions, light sensitivity, skin color variations, camera distance, and poor brightness. Glove-based methods make use of hardware devices to gather input, which is placed on hands or joints to extract gestures by measuring the joint angles and hand and finger positions in real-time. This method is very fast and accurate however, these devices are costly and equipment blocks the free movement of hands, making the system difficult to operate and complex [1, 5, 6].
Depth-based techniques use 3D depth sensors to generate depth-related data which overcomes the limitation of vision-based and glove-based techniques, Some of the popularly used sensors are the Kinect sensor and Leap Motion Controller [7]. Kinect generates a stream of data with body movement, depth information, and skeletal motion and directly determines the position of hands fingertips [8], while leap motion controller developed and designed by Leap Motion is a consumer-grade infrared sensor useful for hand gesture and finger

position detection [9]. Leap Motion Controller traces the user's hand movements, and generates more compact and limited information, not a complete depth map, only a few key points instead [10].
Classification of the data is one of the most important aspects of gesture recognition. From time to time several methods of classification and optimization have been proposed by researchers [11, 12] but PointNet is one of the most popular techniques for point cloud classification and segmentation. PointNet network architecture uses point clouds as input. Point clouds are a set of points representing the 3D shape of the object. Point clouds are geometric data structures having an irregular format that often require conversion to 3D voxel grids or collections of images resulting in the rendering of unnecessary voluminous data [13].
To avoid this problem PointNet network can be used in a way that directly consumes point cloud data as it shows stronger performances than most state-of-the-art techniques. PointNet requires sufficiently large sample points to correctly train the network. However, most of the economic devices capture less number of data points for each character. To resolve this challenge a pre-processing step of creating a trimesh layer has been proposed in this research. In this work 'point cloud' is first converted to a 3D object and then re-sampling from these objects has been used as input to PointNet. The proposed methodology of incorporating a trimesh layer sampling has reduced the issue of low sampling points and has provided highly accurate results.

2 Literature Review

In this section, a short review of many previous related works about air writing and hand gesture recognition has been discussed. Air writing and gesture recognition has been used in a wide variety of applications and may require different recognition techniques depending on various factors such as accuracy, speed, and use cases. For decades, researchers have worked on static and dynamic gesture recognition and achieved a highly accurate recognition rate, we have discussed a few of the past works in this section.
D. Frolova et al.[14] proposed a modified LCS technique termed Most Probable LCS that classifies the trajectory for dynamic free-air hand gesture recognition, which is an extension to the longest common sub-sequence (LCS) classification algorithm. In this technique, a learning preprocessing stage is performed to generate a probabilistic 2-D template for each and every gesture that takes different trajectory distortions with different probabilities into consideration. The methodology measures the similarity between the sample template and the gesture sample and the matches are decided on the basis of length and probability of extracted sub-sequence providing a recognition rate of approximately 98%.
P. Kumar et al.[15] proposed a framework on a dataset consisting of twenty-eight isolated manual signs and twenty-eight finger-spelling gestures to recognize finger-based spellings and manual signs using Leap Motion Sensors. In

order to differentiate finger-based spelling and manual gestures, the Support Vector Machine classifier has been utilized. Further, two BLSTM-NN classifiers are used for recognizing manual signs and finger spelling gestures that comprise sequence classification and sequence transcription. The overall accuracy of 63.57% was achieved in real-time.

C. Chuan et al. [16] designed a compact and low-cost 3D motion sensor to identify American Sign Language that applied k-nearest neighbor and support vector machine for identifying the English alphabet in American Sign Language using the derived features from the sensory data. An average classification rate of 79.83% was achieved via Support Vector Machine Classification while the k-nearest neighbor gave an accuracy of 72.78%. The designed sensor gave a much more portable and economical solution than Cyberglove or Microsoft Kinect.

P. Kumar et al.[17] presented a 3D text recognition method utilizing Hidden Markov Model (HMM) and Bidirectional Long Short-Term Memory Neural Networks (BLSTM-NNs). This model was tested on a data set of 560 Latin sentences drawn by ten participants on Leap Motion Sensor and gave an accuracy of 78.2% in word recognition while an accuracy of 86.88% was recorded in word recognition using BLSM-NN while HMM classifier gave an accuracy of 81.25% in word recognition.

B. Khelil et al.[18] came up with a novel approach for pattern recognition of symbols of Arabic Sign Languages. Sparse data generated by Leap Motion Controller Sensor form the basis of this approach in which significant characteristics from data like angles between the fingers are utilized to achieve higher accuracy. The proposed approach can successfully recognize digits 0 to 9 and twenty-eight static hand gestures of ArSL.

C. Naidu et al.[19] came up with a technique that uses Leap Motion Controller for hand gestures to recognize Indian Sign Language consisting of ten digits (1 to 10), twenty-six alphabets (a to z) and 9 words tested with 10 different signatories. The author proposed and compared four different similarity measures to recognize the sign language that is Jaccard Similarity, Cosine similarity, Dice Similarity, and Euclidean distance measure giving an accuracy of 86%, 90%, 83.11%, and 88.22% respectively.

Y. Chen et al.[20] came up with a recognition system for recognizing the performance of human-computer interaction on 100 samples of Arabic dataset that consisted of 26 letters from A to Z and ten digits from 0 to 9, with the implementation of this system an average recognition rate of 82.4% was achieved.

C. Agarwal et al.[8] came up with a methodology of word recognition and extraction from recorded cursive handwritten text lines by Leap Motion Controller from which large gap information between the start and end position of successive text lines were found by using segmentation methodology of continuous 3D strokes into words and lines analytically. An accuracy of 77.6% was achieved using the methodology.

K. Fok et al. [21] developed a multi-sensory real-time recognition system useful for American sign language (ASL) on data generated by Leap Motion sensors which were fused via multiple sensor data fusion (MSDF) techniques and Hidden Markov Model was implemented for recognition thereby giving recognition rate of 93.14%.

R. Mapari et al.[9] presented an approach for recognizing Indian Sign Language. The data was collected using a Leap Motion sensor. Cosine similarity and Euclidean distance were used to measure the similarity. Testing on ISL alphabets with 10 different signers was done and recognition accuracy of 90.32% and 88.39% was achieved respectively.

M. Mohandes et al.[22] developed an approach for recognition of Arabic Sign Language that makes use of 2 Leap Motion Controllers thereby preventing occlusion of a finger by another finger or hand. The controllers extract the features and are fused together. For recognition, Linear Discriminant Analysis (LDA) classifiers were used that give 97.7% classification accuracy, and an accuracy of 91.1% is achieved with classifier level fusion. The use of 2 LMC gives higher accuracy in comparison to a single LMC.

G. Marin et al.[23] used an ad-hoc feature that computes orientation and placing of fingertips which serve as input for multi-class Support Vector Machine classifier to recognize gestures. The features extracted to measure similarity are distance, orientation, and fingertip angle giving an accuracy of 91.28%.

S. Ameur et al.[24] came up with an approach that extracts spatial feature descriptors based on the position of palm center and fingertips and utilized it as input for support vector machines classifier to recognize the performed gestures, The approach was designed explicitly for 3D dynamic gesture recognition target to leap motion data. The results show the approach to be very effective in recognizing modeled gestures with an accuracy rate of nearly 81%.

W. Lu et al.[25] proposed a novel feature vector-based approach suited for identifying dynamic gestures of hands with Leap Motion Controller. The computed feature vector along with depth information is fed as input to the Hidden Conditional Neural Field (HCNF) classifier to recognize hand gestures. The framework includes 2 main steps of feature extraction followed by classification using a Hidden Conditional Neural Field (HCNF) classifier to identify dynamic hand gesture datasets. An accuracy of 89.5% was observed for the Leap Motion data set while the data set from Handicraft gestures gave 95% accuracy, clearly showing the proposed method gave better accuracy in the case of Handicraft gestures.

M. Chen et al.[26] developed a window-based approach to automatically detect and extract air writing events in case of a continuous motion data stream that may contain stray finger movements having no relation with the writing. Continuous writing events are converted to segments. The proposed algorithm was found to be very accurate with an error rate of 1.15%.

A.Saif et al.[27] presented an effective method of gesture recognition by making a robust feature vector. In the proposed methodology, the feature vectors are created by the selection of distinctive locations using blob detection based on Hessian Matrix approximation, and then feature vector and edges detection

are done followed by computing edge orientation. After this, 1D and 2D mapping of templates are done to compare candidates with the prototype using an adaptive threshold. A recognition rate of 98.33% is achieved in this approach. Won-Du Chang et al.[28] proposed an artificial neural network method to recognize characters written in the air by utilizing uni-stroke-designed characters. The proposed method provided 91.06% accuracy on Arabic numbers and English alphabets drawn by 18 people.
Prabhat et al. [29] proposed a generic webcam-based method using HSV color space and morphological operations for detection and CNN based classifier for recognition of virtual characters drawn in the air using 99.75%, 99.73%, 99.13%, 99.97%, and 99.81% accuracy for 26 English alphabets, MNIST dataset (10 different classes), Devanagari characters dataset, Devanagari Digits (0-9) and traditional digits (0-9) respectively.
C. Qi et al. [13] proposed a network, known as PointNet, This network has a unified architecture and is useful for a wide range of applications such as object classification, part segmentation, parsing of scene semantics, etc.

3 Proposed Methodology

In this section, a brief discussion about the technology and methods has been done. The block diagram presented in Figure 2 gives an idea about the approach and implementation of the methodology.

PointNet

The architecture of the PointNet classification network has been shown in Figure 1. The PointNet classification network takes the point cloud in pre-segmented form as input or input points can be sampled directly from shapes for classification purposes. A set of 3D points $\{P_i|i = 1, ..., n\}$, represents a point cloud, and each point in set Pi is a vector of its (x, y, z) coordinates and additional feature channels such as color, brightness, etc. The proposed deep network gives an output of k scores for all the k candidate classes. The input can be a single object for part region segmentation or a sub-volume from a $3D$ scene for object region segmentation. The proposed model gives a score of '$n \times m$' for each 'n' point and each 'm' semantic subcategory. The classification network applies input transformation followed by feature transformation on 'n' input points and uses max pooling to accumulate point features. The output is classification scores for k classes.

An image of the Leap Motion Controller is shown below in Figure 3. Leap Motion controller is a small USB peripheral device that consists of 3 infrared LEDs and 2 Infrared monochromatic cameras, giving a range of roughly 1m [30]. The compact size of the sensor makes it easy to use.

In the proposed system (Figure 2), Leap Motion Controllers are used to acquire input by tracking hand gestures or movement of fingers or hands in 3D digital format which are mapped in terms of a set of 24 points. A trimesh network is developed using these 24 generated points. The trimesh formed is then

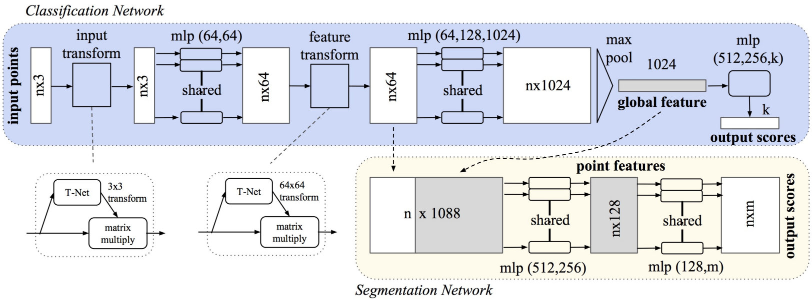

Fig. 1 PointNet Classification Network Architecture

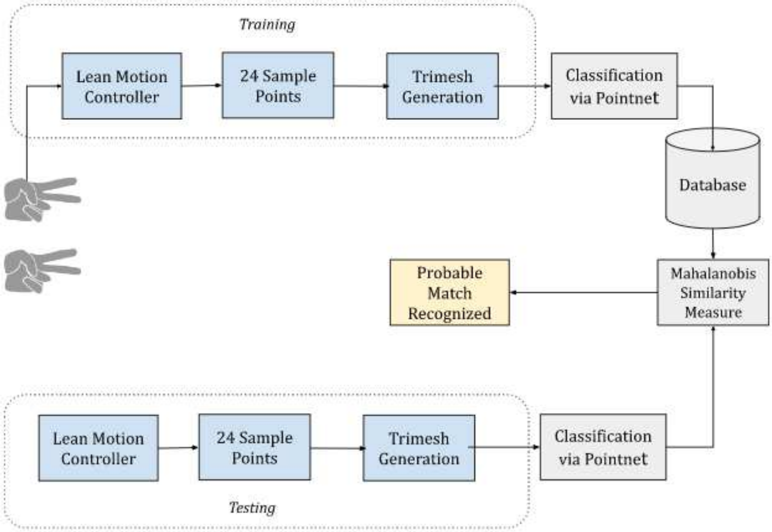

Fig. 2 Block Diagram of Proposed model

fed as input to the PointNet architecture for classification. PointNet can perform classification directly taking point clouds as input directly without any requirement of trimesh or 3D shape reconstruction; however, the number of sample points is less. By developing trimesh and using them as input for PointNet, sufficient sample points are available thereby giving a better recognition rate.

1. **Data Acquisition**
   Gestures made are captured via Leap Motion Controllers that stream frames which are stored and represent a few key points corresponding to each hand gesture [19]. For a single alphabet, a set of eight different inputs are captured, Corresponding to each input twenty-four key feature points are generated and stored. A trimesh is developed using these 24 generated sample points, which are fed as input to the PointNet cloud architecture

Fig. 3 Lean Motion Controller

for classification purposes. These sets of data are stored as training samples for our system. Figure 4 shows generated training data samples of Devnagari characters.

2. **Classification of Gesture**
While testing, a current gesture is captured generating twenty-four key sample points which are processed to develop a trimesh, and classification is done using PointNet architecture. The similarity of current gesture data is measured with stored samples in the database using Mahalanobis distance using the equation 1 written below. Here, 'd' represents the Mahalanobis distance, and (p,q) represents the point coordinates.

$$d\left(p,q\right)=\sqrt{\sum_{i=1}^{n}\left(p_i-q_i\right)^2} \tag{1}$$

4 Result and Analysis

The pointnet with trimesh system is tested for gestures of Hindi Devanagari alphabets, The experimental results obtained are shown in the images and discussed in this section. There was a total of ten characters considered for this analysis (Figure 5). The model loss and model accuracy has been calculated and the graph has been plotted with 60% training samples (Figure 6), 65% training samples (Figure 7), 70% training samples (Figure 8), 75% training samples (Figure 9), and 80% training samples (Figure 10). The accuracy of

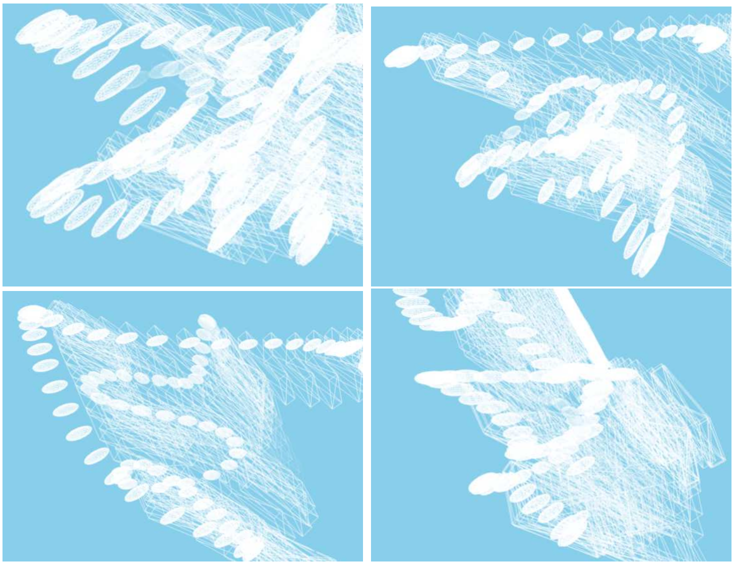

Fig. 4 Training Samples with proposed model

अ	इ	उ	ए	ओ
आ	ई	ऊ	ऐ	औ

Fig. 5 Devanagari characters used for training and classification

the proposed method after running fifty iterations on random distribution with different training testing distributions is shown in table 1. It should be noted that training and test samples were created by the same team members in the experiment. The proposed Model shows difficulties when training and testing samples are made by different personnel. The accuracy of PointNet suddenly drops to 68% when tested on samples of entirely different ways of drawing the same character, however, the proposed model still manages to get 91% accuracy in this case. Domain Adaptation and transfer learning still show challenges for character recognition, but after adding a few demo trials of a user-proposed method shows improvements over PointNet and produces an accuracy of 95%.

Equations 2 and 3 have been used to calculate model accuracy and model loss. where TN: True Negative, TP: True Positive, FN: False Negative, FP: False Positive

$$Model\ Accuracy = \frac{TP + FP}{TP + FP + TN + FN} \tag{2}$$

$$Model\ Loss = \frac{TN + FN}{TP + FP + TN + FN} \tag{3}$$

Table 1 Impact of training sample size on classification accuracy of the proposed method

S.No	Percentage of training distribution	Accuracy	Error
1	60	81.74	6.292
2	65	89.52	3.719
3	70	96.66	0.504
4	75	97.84	0.064
5	80	99.01	0.003

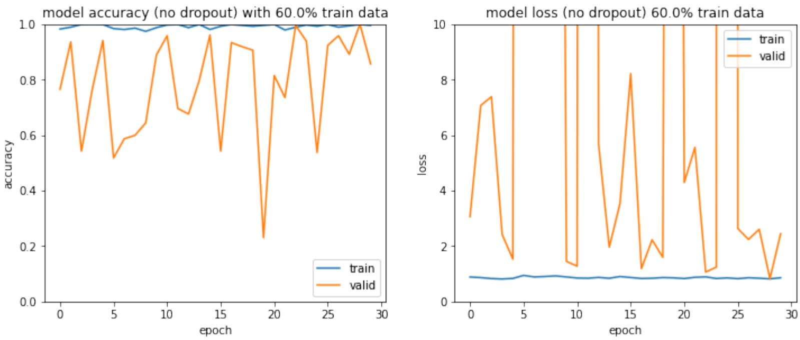

Fig. 6 Model accuracy and model loss with 60% training samples

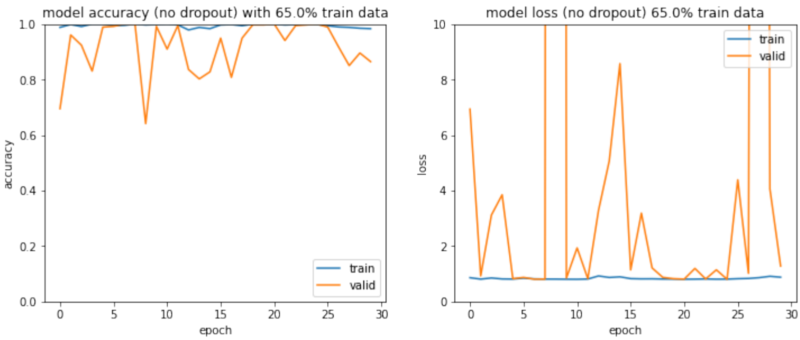

Fig. 7 Model accuracy and model loss with 65% training samples

Analysis of model accuracy and model loss is done and it has been observed that with the increase in the percentage of training samples, the model accuracy improves, as soon as 70% of the training samples are available, high model accuracy is achieved, which is close to 97%.

Similarly, model loss reduces with an increase in the availability of training samples, at 70% availability of training samples, model loss becomes insignificant. It can be seen in Table 2 that the proposed method is highly accurate when compared to other state-of-the-art methods.

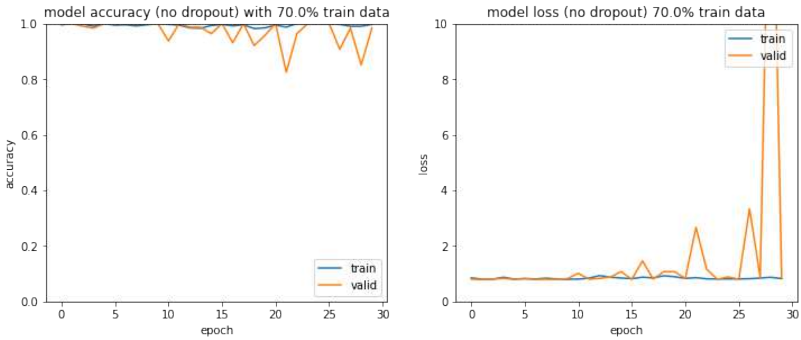

Fig. 8 Model accuracy and model loss with 70% training samples

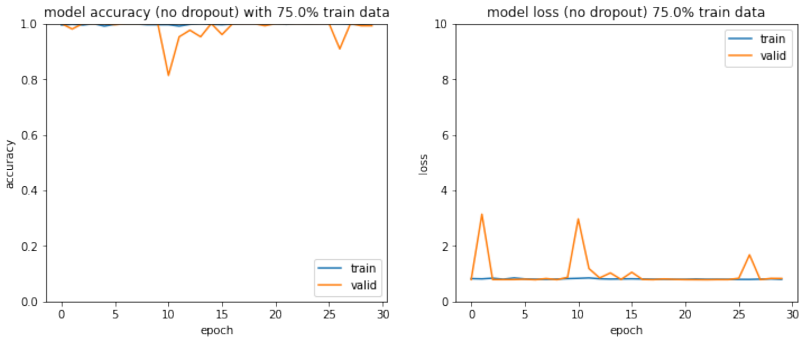

Fig. 9 Model accuracy and model loss with 75% training samples

The confusion matrix of the trained model presented in Figure 13 shows that the model is able to classify and recognize most of the gestures accurately. However for a few gestures, the model was not able to recognize, and it was observed that the letters showing similarity (an example of the such case has been shown in Figure 12) in curves are sometimes inaccurately recognized.

Proposed model using trimesh as input for PointNet method is compared with classification using PointNet without trimesh and results are shown in Figure 11. Analysis of model accuracy and model loss shows that the performance of the proposed model is better in comparison to only the PointNet approach. A comparison of the proposed method with PointNet is also shown in Table 3.

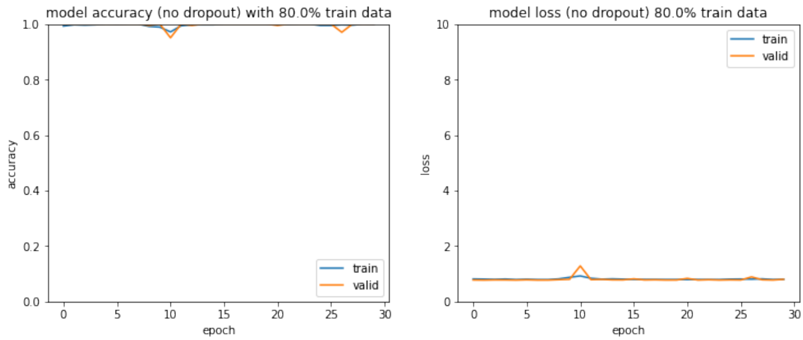

Fig. 10 Model accuracy and model loss with 80% training samples

Table 2 Comparison of the proposed method with other state of art methods

S.No	**Method Name**	**Language (words/alphabets)**	**Character recognition rate (%)**
1	k-Nearest Neighbor with SVM[16]	English alphabet in American Sign Language	79.83
2	BLSTM-NN [17]	Latin words	86.88
3	HMM[17]	Latin words	81.25
4	MSDF with HMM[21]	American sign language words	93.14
5	LDA[22]	Arabic sign Language	91.1
6	HCNF[25]	Handicraft gestures	95
7	Convolutional neural network (CNN)[29]	Devanagari handwritten characters	99.13
8	PointNet	Hindi alphabet	92.19
9	Proposed Method	Hindi alphabet	97.84

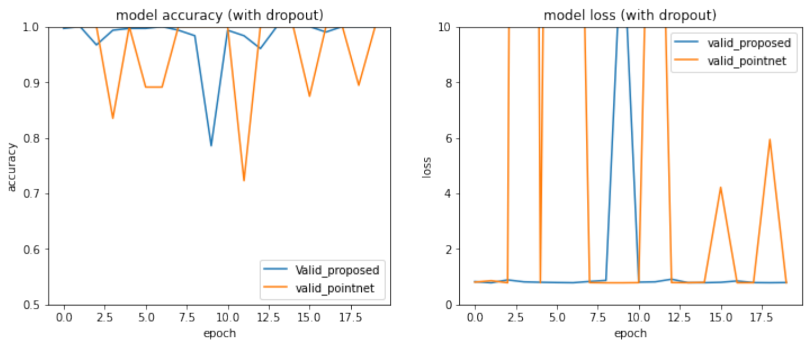

Fig. 11 Model accuracy & model loss comparison with trimesh and without trimesh

Table 3 Comparison between Validation accuracy and error produced by proposed method and PointNet

S.No	Training sample percentage	Accuracy of proposed method	Error of proposed method	Accuracy of PointNet	Error of PointNet
1	70	96.66	0.504	92.19	5.53
2	75	97.84	0.064	92.87	1.18

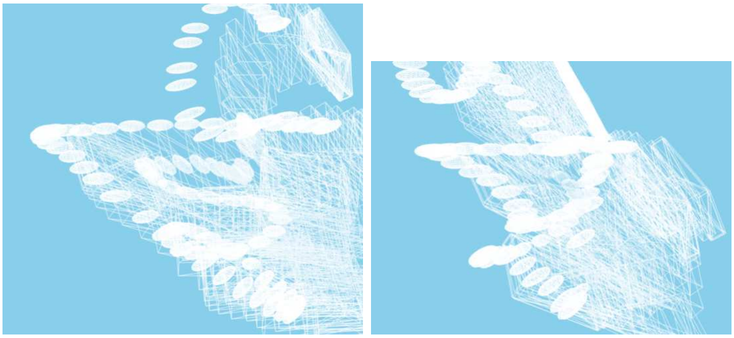

Fig. 12 Training Samples of letters with similar curves

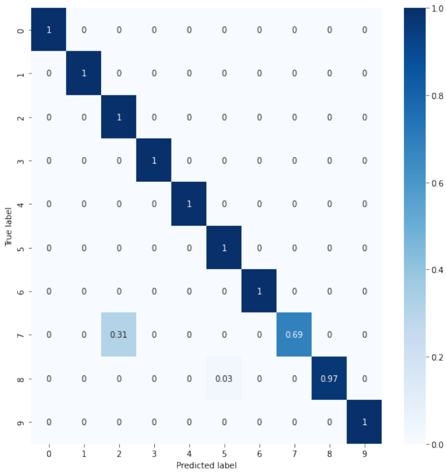

Fig. 13 Model accuracy & model loss comparison with Trimesh and without trimesh

5 Conclusion and Scope of Future Work

In this article, a novel method capable of recognizing Hindi Devanagari gestures with high accuracy was proposed. In the proposed methodology, data points generated from Leap Motion Controller are used for creating trimesh which is used as input for point-net classification. The measure of similarity for recognition between training samples and real-time gestures is determined using Mahanobolis distance. Re-sampling from 3D objects has significantly improved the accuracy when compared to directly feeding inputs in the PointNet. Due to the re-sampling layer, the proposed method is highly usable for devices with low sampling rates. It has been observed that character recognition accuracy improves with an increase in training samples. However, it can be concluded from the analysis that the trained models may sometimes give incorrect results, especially in the case of similar letters with rounded curves. Another major challenge is the difference in the writing styles of different individuals. If sufficient training samples of different individuals are not provided, there are high chances of over-fitting. A wider dataset or more sampling points will overcome this error. In the future, multi-agent datasets and a wider range of languages and alphabets can be tested. Further, the impact of the extreme difference in scale and distance while making symbols in the air is yet to be tested.

6 Data Availability

Datasets used in this study are available upon request from the corresponding author.

7 Conflicts of Interest

There are no conflicts of interest regarding the publication of this article.

8 Acknowledgments

This work did not receive any external funding.

References

[1] Wang, Q., Xu, Y.-R., Bai, X., Xu, D., Chen, Y.-L., Wu, X.: Dynamic gesture recognition using 3d trajectory. In: 2014 4th IEEE International Conference on Information Science and Technology, pp. 598–601 (2014). IEEE

[2] Warhade, A.V., Misal, P.K.: An approach to hand gesture recognition for devanagari sign language using image processing tool box. International Research Journal of Engineering and Technology **3**(9), 1246–1250 (2016)

[3] Priyal, S.P., Bora, P.K.: A robust static hand gesture recognition system using geometry based normalizations and krawtchouk moments. Pattern Recognition **46**(8), 2202–2219 (2013)

[4] Shanableh, T., Assaleh, K.: User-independent recognition of arabic sign language for facilitating communication with the deaf community. Digital signal processing **21**(4), 535–542 (2011)

[5] Wilson, A.D., Bobick, A.F.: Parametric hidden markov models for gesture recognition. IEEE transactions on pattern analysis and machine intelligence **21**(9), 884–900 (1999)

[6] Sun, L.-j., Zhang, L., Guo, C.-l.: Technologies of hand gesture recognition based on vision. Computer Technology and Development **18**(10), 214–216 (2008)

[7] Singh, S.P., Dhiman, G., Tiwari, P., Jhaveri, R.H.: A soft computing based multi-objective optimization approach for automatic prediction of software cost models. Applied Soft Computing **113**, 107981 (2021)

[8] Agarwal, C., Dogra, D.P., Saini, R., Roy, P.P.: Segmentation and recognition of text written in 3d using leap motion interface. In: 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR), pp. 539–543 (2015). IEEE

[9] Mapari, R.B., Kharat, G.: Real time human pose recognition using leap motion sensor. In: 2015 IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), pp. 323–328 (2015). IEEE

[10] Bassily, D., Georgoulas, C., Guettler, J., Linner, T., Bock, T.: Intuitive and adaptive robotic arm manipulation using the leap motion controller. In: ISR/Robotik 2014; 41st International Symposium on Robotics, pp. 1–7 (2014). VDE

[11] Srinivasu, P.N., Bhoi, A.K., Jhaveri, R.H., Reddy, G.T., Bilal, M.: Probabilistic deep q network for real-time path planning in censorious robotic procedures using force sensors. Journal of Real-Time Image Processing **18**(5), 1773–1785 (2021)

[12] Srinivasu, P.N., Sandhya, N., Jhaveri, R.H., Raut, R.: From blackbox to explainable ai in healthcare: Existing tools and case studies. Mobile Information Systems **2022** (2022)

[13] Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 652–660

(2017)

[14] Frolova, D., Stern, H., Berman, S.: Most probable longest common subsequence for recognition of gesture character input. IEEE transactions on cybernetics **43**(3), 871–880 (2013)

[15] Kumar, P., Saini, R., Behera, S.K., Dogra, D.P., Roy, P.P.: Real-time recognition of sign language gestures and air-writing using leap motion. In: 2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA), pp. 157–160 (2017). IEEE

[16] Chuan, C.-H., Regina, E., Guardino, C.: American sign language recognition using leap motion sensor. In: 2014 13th International Conference on Machine Learning and Applications, pp. 541–544 (2014). IEEE

[17] Kumar, P., Saini, R., Roy, P.P., Dogra, D.P.: Study of text segmentation and recognition using leap motion sensor. IEEE Sensors Journal **17**(5), 1293–1301 (2016)

[18] Khelil, B., Amiri, H., Chen, T., Kammüller, F., Nemli, I., Probst, C.: Hand gesture recognition using leap motion controller for recognition of arabic sign language. In: 3rd International Conference ACECS, vol. 16, pp. 233–238 (2016)

[19] Naidu, C., Ghotkar, A.: Hand gesture recognition using leap motion controller. International Journal of Science and Research (IJSR) ISSN (Online) **2319**, 436–441 (2016)

[20] Chen, Y., Ding, Z., Chen, Y.-L., Wu, X.: Rapid recognition of dynamic hand gestures using leap motion. In: 2015 IEEE International Conference on Information and Automation, pp. 1419–1424 (2015). IEEE

[21] Fok, K.-Y., Ganganath, N., Cheng, C.-T., Chi, K.T.: A real-time asl recognition system using leap motion sensors. In: 2015 International Conference on Cyber-enabled Distributed Computing and Knowledge Discovery, pp. 411–414 (2015). IEEE

[22] Mohandes, M., Aliyu, S., Deriche, M.: Prototype arabic sign language recognition using multi-sensor data fusion of two leap motion controllers. In: 2015 IEEE 12th International Multi-Conference on Systems, Signals & Devices (SSD15), pp. 1–6 (2015). IEEE

[23] Marin, G., Dominio, F., Zanuttigh, P.: Hand gesture recognition with leap motion and kinect devices. In: 2014 IEEE International Conference on Image Processing (ICIP), pp. 1565–1569 (2014). IEEE

[24] Ameur, S., Khalifa, A.B., Bouhlel, M.S.: A comprehensive leap motion

database for hand gesture recognition. In: 2016 7th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), pp. 514–519 (2016). IEEE

[25] Lu, W., Tong, Z., Chu, J.: Dynamic hand gesture recognition with leap motion controller. IEEE Signal Processing Letters **23**(9), 1188–1192 (2016)

[26] Chen, M., AlRegib, G., Juang, B.-H.: Air-writing recognition—part ii: Detection and recognition of writing activity in continuous stream of motion data. IEEE Transactions on Human-Machine Systems **46**(3), 436–444 (2015)

[27] Saif, A., Mahayuddin, Z.R., et al.: An efficient method for hand gesture recognition using robust features vector. Journal Information System and Technology Management (JISTM) (2021)

[28] Chang, W.-D., Matsuoka, A., Kim, K.-T., Shin, J.: Recognition of uni-stroke characters with hand movements in 3d space using convolutional neural networks. Sensors **22**(16) (2022). https://doi.org/10.3390/s22166113

[29] Kumar, A.S. P Chaudhary: A cnn based air-writing recognition framework for multilinguistic characters and digits. SN Computer Science **3**(6) (2022)

[30] Ultraleap: Digital Worlds that feel human (2022). https://www.leapmotion.com/