SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V12I4P113

Volume 12 Issue 4, 142-147, April 2025

Original Article

Dynamic Hand Gesture Detection using CNN-based Keypoint Estimation

Rameez Shamalik[1], Shital pawar[2], D.B. Jadhav[3], Seema hadke[4], Kanchan Mahajan[5]

[1]*E&TC Department of Bharati Vidyapeeth's College of Engineering for Women, Pune, India.*
[2]*Computer Department of Bharati Vidyapeeth's College of Engineering for Women, Pune, India*
[3]*Mechanical Department at Bharati Vidyapeeth (Deemed to be University)College of Engineering, Pune, India.*
[4]*IT Department of Bharati Vidyapeeth's College of Engineering for Women, Pune, India*
[5]*E&TC Department of Bharati Vidyapeeth's College of Engineering for Women, Pune, India.*

[1]*Corresponding Author : shamalik1@gmail.com*

Received: 10 February 2025 Revised: 12 March 2025 Accepted: 13 April 2025 Published: 29 April 2025

Abstract *- Accurate and real-time hand gesture detection is crucial for advancing Human-Computer Interaction (HCI) applications. However, conventional methods often struggle with dynamic gestures due to factors such as motion blur, varying lighting conditions, and complex hand shapes. This research delves into developing a robust CNN-based hand gesture detection system to overcome these limitations. Trained and tested on real-life static and dynamic gesture datasets, the proposed model exhibits significant accuracy improvements over existing methods, achieving average precisions of 92.87% and 95.17%, respectively. This research presents a novel multi-layered CNN for accurate 3D hand poses estimation in real-time. By leveraging the power of CNNs and incorporating 3D key points, the proposed model achieves significant accuracy improvements over existing methods while maintaining real-time performance. This opens up new possibilities for hand gesture-based HCI applications, paving the way for more natural and intuitive interactions between humans and computers.*

Keywords *- CNN, Gesture detection, Skeletal representation, Video processing, 3D estimation.*

1. Introduction

Hand gesture recognition has emerged as a crucial component in Human-Computer Interaction (HCI) applications, facilitating natural and intuitive communication between humans and machines. Real-time hand gesture detection, however, poses significant challenges due to factors such as motion blur, varying lighting conditions, and complex hand shapes [1, 2].

Conventional methods have achieved promising results in static hand gesture recognition, but their performance often deteriorates when dealing with dynamic gestures [3, 4].

Existing hand gesture detection methods face several limitations. Firstly, they often rely on handcrafted features that require extensive domain knowledge and may not generalize well to diverse hand postures [5]. Secondly, they often lack robustness against lighting, background, and hand pose variations, leading to decreased accuracy in real-world scenarios [6, 7].

Finally, many methods struggle to handle complex hand shapes and occlusions, limiting their applicability in practical applications [8, 9]. This research proposes a novel multi-layered Convolutional Neural Network (CNN) architecture for real-time 3D hand pose estimation to address these limitations.

The proposed model utilizes a combination of palm detection and 3D key point estimation to track hand movements and identify gestures accurately. The model is trained and tested on two real-life gesture datasets, demonstrating significant accuracy improvements over existing methods.

1.1. Contributions of the Proposed Research

1.1.1. Accurate and Robust Palm Detection

The proposed method incorporates a dedicated palm detection module that effectively identifies and localizes the palm in real time, even under challenging lighting conditions. This robust palm detection serves as the foundation for subsequent hand pose estimation tasks.

1.1.2. High-Precision 3D Key Point Projection

The proposed method utilizes advanced techniques to project 3D key points onto monocular RGB frames, enabling accurate hand pose estimation even in low-light environments. This capability enhances the reliability of gesture recognition in real-world scenarios.

1.1.3. Real-time 3D Joint Estimation for Gesture Recognition

The proposed method employs a multi-layered CNN architecture to perform 3D joint estimation for gesture recognition. The CNN's lightweight and efficient design facilitates real-time performance, enabling seamless gesture recognition in interactive applications.

1.1.4. Comprehensive Comparison with Existing Models

The proposed method is evaluated against existing hand gesture recognition techniques, demonstrating significant improvements in accuracy and robustness. This comparison highlights the effectiveness of the proposed approach in real-time hand pose estimation.

2. Related Work

2.1. Skeletal Models

Researchers have explored using skeletal models to represent hand movements for gesture recognition. One approach combines Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) cells to capture the temporal dynamics of hand movements for action classification. This method effectively utilizes the sequential nature of hand gestures, enhancing gesture recognition accuracy. Without a depth image, this model relies heavily on hand-skeletal data. Furthermore, as illustrated in Figure 1, a 3D hand skeletal joint using key points is generated using an Intel RealSense camera.

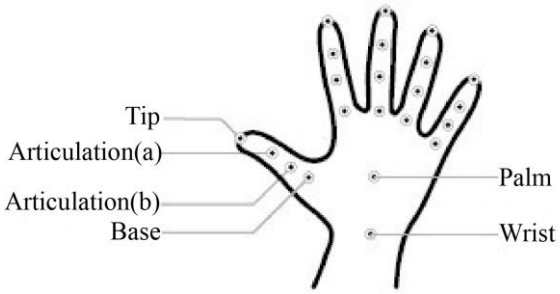


Fig. 1 Hand skeleton with 22 joints [11]

2.2. CNN-Based Approaches

Convolutional Neural Networks (CNNs) have gained popularity in hand gesture recognition due to their ability to learn complex features from image data. An approach employs a 3D-ResNet network for feature fusion to extract local spatiotemporal features, followed by a variant ConvLSTM for obtaining global spatiotemporal data [9]. This method effectively captures both local and global features from hand skeletons, which is crucial for recognizing complex gestures. A two-level hierarchical structure consisting of a detector and a classifier has been proposed for a CNN architecture that operates efficiently using the sliding window approach for real-time gesture detection [10]. This approach demonstrates promising results in efficient and accurate real-time gesture recognition.

2.3. Depth Estimation

Achieving accurate 3D hand pose estimation necessitates depth information from the input image. Monocular depth estimation, which estimates depth from a single image, poses a challenging task. However, it is essential for reconstructing the 3D hand pose. A Gated Multi-Scale Network has been proposed for monocular depth estimation, achieving state-of-the-art accuracy on benchmark datasets [11]. This approach contributes to improving the accuracy of 3D hand pose estimation.

2.4. Google's Hand Tracking Solution

Google's hand-tracking solution exemplifies integrating different techniques for accurate hand-tracking in real-time. It employs an ML pipeline consisting of two models: a palm detector and a hand landmark model. The palm detector locates palms in the image, while the hand landmark model extracts 2.5D landmarks from the palms. This combination enables precise hand tracking in real-world applications.

These related works collectively address the challenges of hand gesture detection by employing various techniques, including skeletal models for movement representation, CNN-based approaches for feature learning, depth estimation for 3D pose reconstruction, and the integration of multiple techniques for real-time hand tracking.

3. Methodology

3.1. Palm Detector

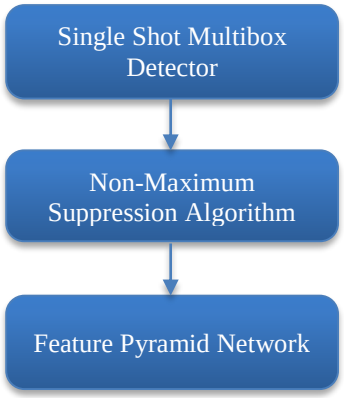


Fig. 2 Palm detector pipeline

As palms and fists are much easier than identifying hands featuring moving fingers, a palm detector is chosen rather than a hand detector, as shown in Figure 2. Furthermore, the non-maximum suppression technique works well for two-hand self-occlusion scenarios such as handshakes because palms are smaller objects. Furthermore, palms may be described as only cubic bounding boxes, including a Single-Shot Multibox Detector (SSD) [12].

3.2. CNN Training for Keypoint Estimation

The two-layered CNN for keypoint estimation is trained on a combination of static and dynamic hand gesture datasets.

3.2.1. Static Gestures

The FabDepth I dataset provides exceptional images for 21 hand gestures, totaling 2100 images. To enhance variability and foster a deeper understanding of hand gestures, a scaled version of the same gestures in real-world settings is also included, expanding the dataset by another 2100 images bringing the total to 4200 images.

3.2.2. Dynamic Gestures

The EgoGesture dataset [15] provides real-world dynamic hand gesture sequences with 21 key point annotations per frame. These dynamic gestures help the CNN learn the temporal dynamics of hand movements and their relationship to keypoint positions.

By combining static and dynamic gesture data, CNN can learn a more comprehensive representation of hand poses, enhancing its ability to accurately estimate key points in both static and dynamic scenarios.

3.3. Datasets and Annotations

The datasets used for CNN training and evaluation are:

3.3.1. FabDepth I Dataset

This dataset contains 4200 frames of static hand gestures, each with 21 keypoint annotations, including exceptional Foreground-Background (FGBG) Separation Images and Depth map predicted frames of the same gestures. The gestures represent a wide range of hand poses, including pointing, waving, and counting.

3.3.2. EgoGesture Dataset

This dataset contains over 11,500 video clips of real-world dynamic hand gestures, each with 21 key point annotations per frame. The gestures represent everyday interactions and activities, such as reaching, grasping, and manipulating objects. The keypoint annotations in both datasets provide precise 2D coordinates for the 21 key points, including fingertips, knuckles, and wrist landmarks. These annotations are essential for training the CNN to estimate keypoint positions and recognize hand gestures accurately.

The first layer of CNN is in charge of allocating 2D key points over the identified palm and fingers, as shown in Figure 3. A convex hull is built around them for this purpose, and the absolute distance between the extreme dimensions of a whole hand is measured. Depth map data is required to transform these important points into 3D, together with feature maps provided by the palm detector.

The former is supplied via a CNN trained on depth data from the EgoGesture dataset. Depth maps of captured hands aid in understanding the discrepancy. Delta maps, a mix of feature maps and depth maps, are important for the culminating operation of 3D joint estimation. Delta Maps are coupled with location maps to train a second CNN on 3D annotated data. As illustrated in Figure 3, adding these two maps yields accurate 3D coordinates of joints and key points. Equation 1 may be utilized to obtain a 3D representation of joints during the final stage.

$$\Theta = \sum J \times P \pm M \tag{1}$$

Where Θ, a measure of regeneration of maximal joints, J suggests optimal joint coefficients, P is the total number of principal components involved, and M is a mean vector constant. In this study, 21 key points are proposed to create a skeletal structure of a hand motion utilizing a basic monocular RGB camera rather than the typical 22 key points using a stereo camera. The second module is explained in a stepwise manner in algorithm 1.

Algorithm 1: 3D Key points & Joints estimation

1) Initialize: Collect feature maps from the palm detector
2) Project a convex hull around the palm and fingers
3) Calculate the absolute distance between extreme points
4) Assign 2D key points over selected frames
5) Process 2D key point images with Depth images
6) forming Delta maps
7) Add Delta maps with 3D Location maps
8) Extract 3D key points and joints from the given data
9) Return the gestures with the final output

4. Experimentation

This section discusses instrumentation, hyperparameters implemented, and datasets and their preprocessing steps for proposed research.

4.1. Instrumentation

As all the modules operate in real-time, a machine with an I5 processor with 8 GB of RAM is employed. The proposed method is implemented on a machine with an Intel Core i5-8250U processor running at 1.60 GHz with 8GB of RAM and an integrated Intel UHD Graphics 620 GPU. The operating system is Ubuntu 18.04 LTS. A Graphics Processing Unit (GPU) is not essential because the CPU with the requirements given works well enough with runtime performance of 30 Frames Per Second (fps).

A GPU can surely assist in speeding up model training and testing, as well as real-time processing, saving time. The input is a monocular RGB video stream from a basic webcam. The software for the proposed hand gesture recognition system is developed using TensorFlow, OpenCV, and MediaPipe.

4.2. Training Details

The hyperparameters are chosen for a tradeoff between potential results and model complexity. The CNNs are trained using the Adam optimizer with a learning rate 0.001 and a sigmoid activation function. The batch size for the first module is 64, and the batch size for the second module is 32. Both modules have 50 iterations. The annotations for both datasets are preprocessed to extract a maximum of 21 key points to describe the skeletal hand geometry. The data is split into 70% training, 15% validation, and 15% testing sets.

4.3. Frameworks

These three libraries or frameworks play crucial roles in implementing the proposed method: TensorFlow provides the foundation for building and training the neural networks, OpenCV facilitates image and video processing tasks, and MediaPipe enables the construction of a real-time pipeline for hand gesture recognition.

5. Results and Discussion

5.1. Quantitative Results

A collection of state-of-the-art models also produces their results on the jester dataset [16] to present a comparison, as shown in Table 1. The proposed model stands apart from all other models, focusing on real-time palm detection and feature extraction.

Table 1. Comparison of accuracy taken on jester dataset for static gestures with the proposed model

Model	Accuracy
HO-CP ConvNet-S [17]	85.4
ResC3D [18]	90.3
BN-Inception [19]	92.9
3D-MobileNetV2 [20]	94.59
Proposed Model	**95.17**

Table 2 provides a similar kind of comparison table of the latest techniques' performance on the EgoGesture dataset along with the proposed model.

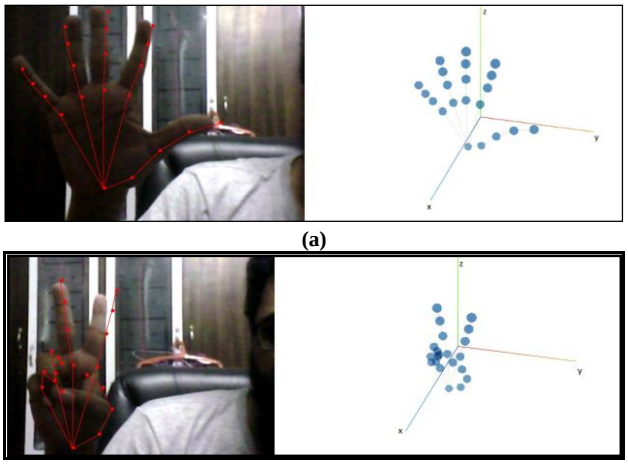
Table 2. Comparison of accuracy taken on the egogesture dataset for dynamic gestures with the proposed model

Model	Accuracy
VGG-16 [15]	62.4
VGG-16+LSTM [15]	76.2
C3D [10]	87.66
MTUT [21]	92.22
Proposed Model	**92.87**

5.2. Qualitative Results

Figures 3 (a) and (b) show how the proposed model detects the palm in the given video frame while subtracting the complex background, especially in low light. Although it is unable to track the exact key points on the index finger in 3 (a), it still provides a superior representation in the 3D graph.

In this case, joint estimation is crucial in projecting a skeletal figure of a palm and fingers, while its 3D key points clearly highlight the geometry and depth of gesture in a given space.



(a)

(b)

Fig. 3 (a) & (b) key points on the palm, in different gestures with its skeletal representation in XYZ axis

It also shows improved accuracy and precision in real-time gestures. The XYZ axis representation of hands can be rotated to analyze the gesture in a given video frame from multiple angles. Another approach to quantifying the hand gesture key points and joint estimation is the Percentage of Correct Key-points (PCK). If the difference between the anticipated and the genuine joint is less than a certain threshold, the detected joint is said to be accurate.

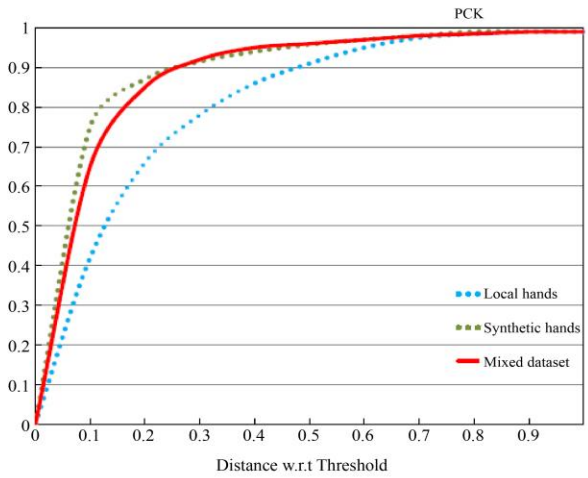


Fig. 4 PCK comparison between different datasets

The pixel lengths in each instance are normalized by 0.7 times the matching person's hand size to produce the distances with respect to the threshold used to create the PCK curves. The blue and green dotted lines represent the PCK on the local and synthetic hand gesture datasets, respectively, while the red line represents the PCK on the mixture of both datasets, as

shown in Figure 4. Thus, the tradeoff is clearly highlighted in terms of the practical application of hand gesture datasets.

6. Conclusion

The proposed method achieves high accuracy on both static and dynamic gesture datasets. On the Jester dataset for static gestures, the method achieves an accuracy of 95.17%, surpassing existing methods. For dynamic gestures on the EgoGesture dataset, the method achieves an accuracy of 92.87%.

These results demonstrate the effectiveness of the proposed approach, which utilizes palm detection and 3D key point estimation for efficient and accurate gesture recognition. The method's simplicity and real-time performance make it suitable for various applications, including virtual reality, augmented reality, and human-computer interaction.Future research might result in an effective method for foreground-background separation and 3D reconstruction of hand gestures for innovative applications.

References

[1] H. Pallab Jyoti Dutta et al., "Semantic Segmentation Based Hand Gesture Recognition Using Deep Neural Networks," *National Conference on Communications*, Kharagpur, India, pp. 1-6, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[2] Gu Lingyun, Zhang Lin, and Wang Zhaokui, "Hierarchical Attention-Based Astronaut Gesture Recognition: A Dataset and CNN Model," *IEEE Access*, vol. 8, pp. 68787-68798, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[3] Olena Vynokurova, and Dmytro Peleshko, "Hybrid Multidimensional Deep Convolutional Neural Network for Multimodal Fusion," *IEEE Third International Conference on Data Stream Mining & Processing*, Lviv, Ukraine, pp. 131-135, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[4] Nabeel M. Mirza et al., "Static Hand Gesture Angle Recognition via Aggregated Channel Features (ACF) Detector," *Signal Processing*, vol. 39, no. 3, pp. 939-944, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[5] Eric Fujiwara, Murilo Ferreira Marques dos Santos, and Carlos K. Suzuki, "Flexible Optical Fiber Bending Transducer for Application in Glove-Based Sensors," *IEEE Sensors Journal*, vol. 14, no. 10, pp. 3631-3636, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[6] Beom Jun Jo, Seok-Kyoo Kim, and SeongKi Kim, "Enhancing Virtual and Augmented Reality Interactions with a MediaPipe-Based Hand Gesture Recognition User Interface," *Information Systems Engineering*, vol. 28, no. 3, pp. 633-638, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] Umair Haroon et al., "A Multi-Stream Sequence Learning Framework for Human Interaction Recognition," *IEEE Transactions on Human-Machine Systems*, vol. 52, no. 3, pp. 435-444, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] Noor Fadel, and Emad I. Abdul Kareem, "Detecting Hand Gestures Using Machine Learning Techniques," *Information Systems Engineering*, vol. 27, no. 6, pp. 957-965, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[9] Danilo Avola et al., "2-D Skeleton-Based Action Recognition via Two-Branch Stacked LSTM-RNNs," *IEEE Transactions on Multimedia*, vol. 22, no. 10, pp. 2481-2496, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[10] Guillaume Devineau et al., "Deep Learning for Hand Gesture Recognition on Skeletal Data," *13th IEEE International Conference on Automatic Face & Gesture Recognition*, Xi'an, China, pp. 106-113, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[11] Yuqing Peng et al., "Dynamic Gesture Recognition Based on Feature Fusion Network and Variant ConvLSTM," *IET Image Processing*, vol. 14, no. 11, pp. 2480-2486, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] Okan Köpüklü et al., "Online Dynamic Hand Gesture Recognition Including Efficiency Analysis," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 2, no. 2, pp. 85-97, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[13] Lixiong Lin et al., "Efficient and High-Quality Monocular Depth Estimation via Gated Multi-Scale Network," *IEEE Access*, vol. 8, pp. 7709-7718, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[14] Wei Liu et al., "SSD: Single Shot MultiBox Detector," *Computer Vision-ECCV 2016: 14th European Conference*, Amsterdam, The Netherlands, pp. 21-37, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[15] Yifan Zhang et al., "EgoGesture: A New Dataset and Benchmark for Egocentric Hand Gesture Recognition," *IEEE Transactions on Multimedia*, vol. 20, no. 5, pp. 1038-1050, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[16] Joanna Materzynska et al., "The Jester Dataset: A Large-Scale Video Dataset of Human Gestures," *IEEE/CVF International Conference on Computer Vision Workshop*, Seoul, Korea (South), pp. 2874-2882, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[17] Jean Kossaif et al., "Efficient N-Dimensional Convolutions via Higher-Order Factorization," *arXiv*, pp. 1-11, 2019. [Google Scholar]

[18] Du Tran et al., "ConvNet Architecture Search for Spatiotemporal Feature Learning," *arXiv*, pp. 1-12, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[19] Okan Köpüklü, Neslihan Köse, and Gerhard Rigoll, "Motion Fused Frames: Data Level Fusion Strategy for Hand Gesture Recognition," *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, Salt Lake City, UT, USA, pp. 2184-21848, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[20] Okan Köpüklü et al., "Resource Efficient 3D Convolutional Neural Networks," *arXiv*, pp. 1-10, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[21] Mahdi Abavisani, Hamid Reza Vaezi Joze, and Vishal M. Patel, "Improving the Performance of Unimodal Dynamic Hand-Gesture Recognition With Multimodal Training," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, pp. 1165-1174, 2019. [CrossRef] [Google Scholar] [Publisher Link]