

KALANETRA — A Mudra Detection Model for Mohiniyattam

Dr. G Naresh¹, Pendyala Siri², M Manideep³, Ch Teja⁴

*1Assistant Professor, Department of Computer Science and Engineering,
Methodist College of Engineering and Technology, Abids, Hyderabad, Telangana, 500001, India.
234Students, Department of Artificial Intelligence and Data Science,
Methodist College of Engineering and Technology, Abids, Hyderabad, Telangana, 500001, India.*

Abstract—This paper presents KALANETRA, a real-time mudra recognition system designed for Mohiniyattam, a prominent classical dance tradition in India. Mudras, the complex hand gestures that are at the heart of this art form, have a lot of meaning, but figuring out how to recognize them automatically has not been a big focus of research. Our system uses a keypoint-based method to get 21 hand landmarks in real time using MediaPipe Hands. It then uses these landmarks to find geometric features like the distances between keypoints, the angles of joints, and the relative orientations of fingers. These features go into a small machine learning classifier that doesn't need a GPU to work, so it can run on regular consumer hardware. The system achieves 75–80% classification accuracy while running at interactive frame rates after thorough normalization to handle differences in hand size, position, and orientation. KALANETRA has been tested with many users and in different lighting conditions, and it works well where other image-based options fail. This work goes beyond its technical contributions by helping to preserve classical dance heritage in digital form and opening up new ways for interactive, self-guided dance education.

1. INTRODUCTION

Indian classical dance is a language all its own. Mudras are the most important part of it. They are carefully planned hand gestures that show emotion, tell a story, and have spiritual meaning. In Mohiniyattam, these hand movements require a lot of accuracy. Even a tiny misalignment of a finger or wrist can completely change what is being said. It takes years of hard work and practice with an experienced guru to learn this vocabulary.

The issue is that this kind of expert help isn't always easy to find. Students who are learning from home or in areas where there aren't any trained teachers can't be sure that their gestures are correct. Small mistakes in how the fingers are placed or the wrist is turned can go unnoticed, which could make bad habits stick early on in a learner's journey.

Recent progress in computer vision shows a promising way forward. Frameworks such as MediaPipe Hands can extract 21 three-dimensional landmarks from a hand in real time [2], [4], yielding a detailed geometric representation that captures the subtleties of gesture structure. Keypoint-based methods are much more resistant to changes in lighting and background than traditional image-based methods, and they use a lot less computing power [8], [14].

Landmark-based gesture recognition has already been shown to work in related fields, especially sign language recognition and human-computer interaction [5], [7]. But not much of this research has looked at Indian classical dance, where the gestures are culturally specific, have a lot of meaning, and are hard to see.

KALANETRA fills this need. We suggest a lightweight, real-time recognition system just for classical mudras. It should be easy to use, strong, and easy to set up in schools.

2. LITERATURE SURVEY

Over the past ten years, the field of recognizing hand gestures has grown a lot. Landmark-based pipelines built on MediaPipe are a popular choice because they work well in real time for a lot of different tasks [2], [9]. Researchers have shown that preprocessing and normalizing these landmarks carefully makes them much better at generalizing across different users and environments [3]. They have also shown that even simple classifiers can do well with clean geometric features [13].

Pose estimation techniques have been particularly effective in the field of sign language recognition for capturing the intricate hand configurations that differentiate various signs [5], [7]. More recent research has gone even further, looking into transformer architectures and continuous gesture recognition pipelines for dealing with dynamic, multi-gesture sequences [6, 10].

The particular challenge of Indian classical dance gestures has garnered relatively minimal focus. Pose2Gest [1] is a notable exception, as it uses few-shot learning to recognize gestures in South Indian classical dance. The HMD dataset also added 3D pose-based annotations for Bharatanatyam mudras [16], which made it possible to use data-driven methods. But these systems tend to be complicated and need either a lot of data, a lot of computing power, or both.

Deep learning techniques have been utilized for mudra classification [15]; however, the substantial data requirements and computational intensity of these models restrict their practical implementation for real-time, on-device deployment. Keypoint embedding methods are a better option because they encode gesture structure in a smaller space without losing discriminative power [14].

The existing literature indicates a distinct opportunity for a lightweight, keypoint-based system specifically designed for classical mudra recognition. That is exactly what this work suggests.

3. SYSTEM ARCHITECTURE

KALANETRA is set up as a modular pipeline, with each stage doing a specific transformation that flows smoothly into the next. The design puts a lot of emphasis on being easy to understand, efficient, and able to work in real time.

A webcam or mobile camera sends live video as input to the system. MediaPipe Hands looks at each frame to find the hand area and get its 21 three-dimensional keypoints [2], [4]. These landmarks mark important anatomical points on the hand, such as the fingertips, interphalangeal joints, metacarpal joints, and wrist. They make up a detailed geometric skeleton of the hand.

The raw landmarks then go through a preprocessing step in which they are normalized for scale, position, and orientation. This is a necessary step; without it, the same mudra done by two people with different hand sizes would create very different feature vectors [3].

We use the normalized landmarks to find a group of geometric features:

- The distances between keypoints in pairs
- The angles of the joints at each knuckle and finger joint
- Vectors for the relative orientation of the fingers

These features encode the hand's structural configuration in a way that is both compact and discriminative. They are small enough to be processed right away, but expressive enough to tell the difference between visually similar mudras.

A lightweight classifier gets the feature vector and gives back a predicted mudra label and a score of how sure it is. Both are shown in real time on the live video feed, so users can see how their gesture is being received right away.



Fig.1: Architecture

4. IMPLEMENTATION

4.1 Data Acquisition and Hand Detection

The system can use a regular webcam to get live video. MediaPipe Hands can find hands and extract landmarks at interactive frame rates [2], which makes it a good choice for use on consumer-grade hardware.

4.2 Landmark Preprocessing

After being extracted, the 21 landmarks are translated so that the wrist is at the origin and then scaled so that the hand fits within a set bounding area. This normalization takes away the effect of where the hand is in the frame and how big it is, letting the classifier focus only on the shape of the gesture [3].

4.3 Feature Engineering

Using normalized landmarks, we figure out three kinds of geometric features: the Euclidean distance between pairs of landmarks, the joint angles, and the finger orientation vectors. These two things together show the overall meaning of a gesture and the small differences that make mudras different from each other, even if they look the same at first.

4.4 Classification

A lightweight classifier connects the feature vector to a mudra label. The model was chosen because it has a low inference latency and can run well on regular hardware. This was done on purpose to make sure the system can be used in real-world educational settings.

4.5 System Environment

The implementation uses MediaPipe to find landmarks and NumPy to do math. It runs on Python 3.10. You don't need a GPU; the system works well on a regular laptop or desktop computer.

5. METHODOLOGY

The end-to-end pipeline changes raw video into a mudra prediction by going through a series of clearly defined steps.

5.1 Input Acquisition

The system records video frames all the time and processes them one at a time, so it can respond to changes in gesture without any noticeable delay.

5.2 Landmark Extraction

MediaPipe Hands looks at each frame, finds the hand, and returns 21 three-dimensional landmarks that describe how the hand is shaped [2].

5.3 Normalization

The raw landmarks are moved to the middle of the wrist and made to be the same size. This makes sure that the next feature representation is always the same, no matter how the user holds their hand in front of the camera.

5.4 Feature Extraction

We use the normalized landmarks to find the distances, joint angles, and orientation vectors of the mudra. This gives us a fixed-length feature vector that represents the mudra.

5.5 Classification

The trained classifier takes the feature vector and puts it in one of the known mudra categories. It also makes a guess about how sure it is.

5.6 Output Generation

The system displays the anticipated mudra name along with its corresponding accuracy score on the same video feed. Such instantaneous information gives the user the chance to verify the efficiency of the system and the accuracy of the process employed.

6. RESULTS AND DISCUSSION

We tested KALANETRA on a custom dataset of static mudras from several participants, making sure to change the size of the hands, the color of the skin, and the recording environment to see how well the system could generalize.

6.1 Accuracy

The system got about 75–80% of the tested mudra classes right overall. This result is good news for a lightweight, keypoint-only method, especially since some Mohiniyattam mudras have very small geometric differences.

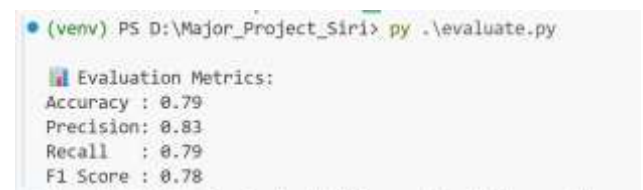


Fig-2: Accuracy metrics



Fig-3: K-fold Validation

6.2 Robustness

KALANETRA worked well in a variety of lighting and background situations, which is a big benefit of using a geometric approach instead of a pixel-based one. These results are consistent with previous research on landmark-based systems in general [8], [14].

6.3 Real-Time Performance

The system kept interactive frame rates during testing, and the latency was low enough to allow for real-time feedback. Users could see that their gesture had been recognized within a few milliseconds.

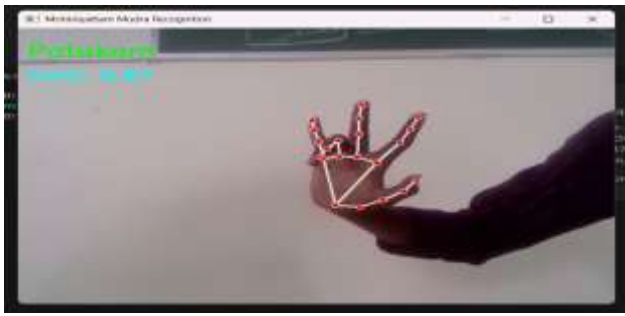


Fig-4: Patakam detected with 67% confidence

6.4 Limitations

There was some confusion between mudras that look very much alike geometrically. For example, gestures that differ only in the exact angle of one finger joint. This suggests two areas for future work: making feature representations more detailed so they can pick up on more subtle geometric cues, and getting bigger, more varied training datasets so the classifier has more examples to learn from.

7. CONCLUSION

KALANETRA shows that it is possible to recognize mudras in real time and with little weight without using deep neural networks or special hardware. We get 75–80% accuracy by using principled geometric feature engineering on top of MediaPipe landmarks. This keeps inference fast and the deployment footprint small.

The real-world effects are much more than just the numbers for accuracy. This kind of system can help more people learn classical dance by letting students practice their mudras and get automatic feedback from anywhere, even if there isn't an instructor there. On a larger cultural level, KALANETRA helps to digitally preserve Mohiniyattam and the classical dance traditions it represents.

There are many exciting ways that the system can be made better in the future. For instance, it could add dynamic gesture sequences, make feature design better at picking up on small differences, and add more performers and conditions to the training dataset.

8. FUTURE SCOPE

- Being able to see dynamic mudras that change over time, not just still pictures
- Making geometric modeling more accurate by using 3D landmark depth data

- Getting more people of all ages and levels of experience to join the dataset
- Allowing two-handed gesture recognition for mudras that need both hands at the same time
- Linking the system to AR/VR platforms to create very realistic self-guided dance learning spaces [12].

REFERENCES

1. K. Raju, N. J. Warriar, M. Madhavan, S. C., A. B. Warriar, and T. Kumar, "Pose2Gest: A Few-Shot Model-Free Approach Applied in South Indian Classical Dance Gesture Recognition," *arXiv preprint arXiv:2404.11205*, 2024.
2. P. Gil-Martín, J. Ortega-Garcia, and D. Toledano, "Hand Gesture Recognition Using MediaPipe Landmarks," *IEEE Access*, 2025.
3. M. Manseri, L. Zekte, and N. Alioua, "Preprocessing MediaPipe Joint Annotation for Robust Sign Language Recognition," *Journal of Imaging Science*, 2025.
4. Alsharif and Y. Alghamdi, "Real-Time American Sign Language Interpretation Using Hand Landmarks," *Sensors*, vol. 25, no. 2, 2025.
5. R. Myagila, K. Omondi, and D. Kimani, "Framework for Detecting and Recognizing Sign Language Using Pose Estimation," *Expert Systems with Applications*, 2025.
6. Emporio and L. Rastegar, "Continuous Hand Gesture Recognition: Benchmarks and Methods," *Pattern Recognition Letters*, 2025.
7. P. Rastogi, M. Singh, and S. Rane, "Advanced Gesture Recognition in Indian Sign Language Using Pose Estimation," *Scientific Reports*, 2025.
8. R. Jalayer, M. Yazdani, and O. Bayat, "A Comprehensive Review on Vision-Based Hand Detection and Gesture Recognition in Human–Robot Interaction," *Robotics and Autonomous Systems*, 2025.
9. S. Navendu, P. Jain, and M. Singh, "Word-Level Sign Language Recognition Using MediaPipe and Deep Learning," *Procedia Computer Science*, 2024.
10. R. Nandhini and K. Aravind, "Dynamic Gesture Recognition Using a Transformer and

- MediaPipe Holistic Landmarks," Neural Computing & Applications, 2024.*
11. X. Meng and L. Zhou, "A Real-Time Hand Gesture Monitoring Model Based on Triple Loss," *IEEE Transactions on Multimedia, 2024.*
 12. R. Jayanthi, A. Narayanan, and S. Sundar, "AI and Augmented Reality for 3D Indian Dance Pose Identification," *Scientific Reports, 2024.*
 13. Y. Wang and M. Liu, "Personalized Gesture Recognition Using Lightweight Neural Networks," *Applied Sciences, 2024.*
 14. De Coster, F. Braffort, and S. Hadjadj, "Keypoint Embeddings for Sign Language Recognition," *Pattern Recognition, 2023.*
 15. M. Muthusamy and S. Venkatesh, "Deep Learning for Automatic Classification of Bharatanatyam Hand Mudras," *International Journal of Computer Vision and Signal Processing, 2023.*
 16. IIT Hyderabad, "HMD: A 3D Hand Mudra Dataset for Bharatanatyam," *Technical Report, 2021.*
 17. Khan, "Real-Time Sign Language Recognition from Webcam Video Stream Using Hand Landmarks," *International Journal of Advanced Computer Science, 2021.*
 18. IJRAR, "Sign Language Detection Using MediaPipe and Machine Learning," *International Journal of Research and Analytical Reviews, 2024.*
 19. IJCRT, "Hand Gesture Recognition Using MediaPipe and Machine Learning," *International Journal of Creative Research Thoughts, 2024.*
 20. Dagde and S. Patra, "A Hand Sign Recognition-Based Signal System for Mute Individuals," *International Journal of Computer Applications, 2025.*