The Language of Hand Gesture in Bharatnatyam

Shravya[1], Sudarshan K[2], Ashvitha Fernandes[3], Jothsna P S[4], Krithi[5]
[1]*Student, Department of Computer Science and Design, Srinivas Institute of Technology, Mangalore, Karnataka, India*
[2]*Associate Professor, Department of Information Science and Engineering, Srinivas Institute of Technology, Mangalore, Karnataka, India*
[3,4,5]*Student, Department of Information Science and Engineering, Srinivas Institute of Technology, Mangalore, Karnataka, India*
doi.org/10.64643/IJIRTV12I12-206832-459

Abstract—This paper introduces a new computer vision and deep learning framework that can automatically recognize and analyze Hasta Mudras (hand gestures) used in Bharatanatyam, a classical Indian dance form. In the past, analyzing Mudras, which are a non-verbal language that is important for expressive storytelling, was a manual and subjective process. We use a two-stage pipeline in our proposed system to deal with the problems of scale, translation, and rotation invariance. The Media Pipe Hands model first finds 21 important hand landmarks in real time. Second, a custom feature engineering process normalizes these raw coordinates to the wrist to make an 84-dimensional invariant feature vector. A lightweight Multi-Layer Perceptron (MLP) model then sorts this vector. **The system has a high classification accuracy and a fast inference speed, which shows that it can be used in real time for training and digital archiving of cultural heritage.**

***Index Terms*—Bharatanatyam, Mudra recognition, Hand gesture analysis, Computer Vision, Media Pipe, Deep Learning, and Multi-Layer Perceptron (MLP) are some of the words used.**

I. INTRODUCTION

Bharatanatyam, a classical dance style from South India, uses complex hand gestures called mudras as the main way to tell stories and express feelings. There are 28 different types of Asamyukta (single-hand) mudras and 23 different types of Samyukta (doublehand) mudras. It is hard for both learners and automated systems to accurately identify mudras because they are so complicated and have so many small differences. Bharatanatyam is more than just a dance style; it is a complex way of communicating without words, with the Hasta Mudras (hand gestures) being the main words. The Natya Shastra, an ancient text, says that these 50+ standardized gestures are important for telling stories (Vakyartha) and showing feelings (Bhava). The accuracy and geometric Accuracy and correct geometry become of utmost importance when executing the dance form.

A. The Problem and the Research Gap:

While the significance of mudras should never be understated, it is through the particular angle taken by the teacher that there would be an opportunity to gain knowledge about and judge the performance of mudras. The reality that such an activity would depend highly on the above strategy means that the entire process is going to be totally subjective, thereby making the process of digital evaluation difficult. While trying to achieve the same in the past, the following problems were encountered:

1.Intrusive Hardware:
Early solutions used sensor gloves, which made it hard for the dancer to move naturally

2. Invariance Challenges:
Vision systems that don't use sensors often fail when the dancer moves closer to or farther away from the camera (scale variance) or rotates their hand. This makes them less reliable in a real-world studio setting.
Our study fills this gap by developing a system that is non-intrusive, operates in real-time, and exhibits high invariance to typical visual transformations.

B. Contributions

The main goal of this paper is to come up with a solution for static Mudra recognition that is both fast and reliable.

1. Invariant Feature Engineering:
We use a strong geometric transformation method to make the 21 MediaPipe hand landmarks normal, which gives us an 84dimensional feature vector that doesn't change much when the image is moved or scaled.

2. Real-Time Classification:
We show that a lightweight Multi-Layer Perceptron (MLP) can effectively classify 50 Mudra classes from the invariant feature vector. It does this faster than complex CNNs or RNNs.
High accuracy and use: The system gets more than 98% of the classifications right, showing that it is ready to be used in dance training and preservation.

II. LITERATURE REVIEW

The theoretical foundation of Mudra analysis is rooted in the Natya Shastra, which provides a categorical framework for hand gestures. Computational efforts to analyze these gestures initially involved sensor-based systems (e.g., data gloves), which provided high accuracy but were not practical for free movement. The shift towards non-intrusive, vision-based systems was initially challenging due to the difficulties in robust hand segmentation and feature extraction in varying real-world conditions.
The technological landscape changed significantly with the introduction of high-fidelity, real-time pose estimation. Modern approaches utilize powerful, pretrained models, such as MediaPipe Hands, which provide a streamlined pipeline for extracting 21 key hand landmarks in 3D space with high fidelity. This enables researchers to concentrate on the ensuing classification task instead of the laborious hand detection process. This allows researchers to focus on the subsequent classification task rather than the arduous process of hand detection.
The classification of these spatial data points requires addressing the problem of pose invariance. Research shows that normalizing landmark coordinates relative to a central point (the wrist) and scaling them using a characteristic distance (e.g., wrist-to-finger length) creates a stable, geometry-focused feature vector that resists changes in distance and rotation. For classifying these structured numerical features, dense networks, such as the MLP, are highly effective. They offer a strong combination of high inference speed and competitive accuracy when compared to more complex recurrent or convolutional architectures.

III. METHODOLOGY

This feature vector is fed into a fast MLP neural network to do efficient prediction.

Architecture of this neural network is specified as follows.
- Neurons in input layer: 84.
- Neurons in first hidden layer: 128 with activation function ReLU.
- Neurons in second hidden layer: 64 with activation function ReLU.
- Neurons in output layer: N with activation function Softmax.

The model is compiled using Adam optimizer and Categorical cross-entropy, and the output will be the classification of the feature vector into one of the Mudra classes.

Experimental Setup and Dataset:
1.Collecting the Dataset:
The dataset used is a collection of more than 15,000 images of three professional dancers performing Bharatanatyam. These images contained 50 commonly used Mudras (Asamyukta and Samyukta Hastas). Data augmentation in form of minor rotations and color saturations is used to improve generalization.

2.Training Details:
The dataset was split up into three equal parts; 70%, 15% and 15% of the train, validation and test sets respectively. Training was done over 50 iterations with a batch size of 64. Early Stopping was done depending on the loss value achieved during the validation phase. This research was developed using Python programming language, TensorFlow 2.0 and Keras.

IV. SYSTEM ARCHITECTURE

The designed system architecture is supposed to be implemented using a multiple stage pipeline approach where recognition and semantic interpretation of the Bharatanatyam mudras can be done in real time. In the first place, there is the Data Input module, responsible for data extraction either from a live video stream via HD camera connectivity or by processing still images uploaded in the system. To increase reliability in case of changes in lighting and environmental factors, all the incoming data frames will be preprocessed in terms of conversion into grayscale, normalization and resizing. The most important element of this stage is the use of ROI (region-of-interest) extractor.

After the preprocessing stage, the next layer involved in the analysis process includes feature extraction based on deep learning techniques. On the other hand, CNNs (convolutional neural networks) are used to recognize complicated skeletal signals and finger positions together with hand poses. The architecture of the neural network guarantees that there will be minimal delays and that the feature map can be easily generated with the necessary characteristics of the Mudra. However, it is small enough to be able to be successfully implemented. Consequently, the essential features related to finger poses for categorizing the Mudra like Mushti and Pataka are recognized.

This is our final module that has two significant modules called Inference and Output Display Module. The inference module makes use of the Softmax Activation Function in generating probability distributions with regards to the feature sets we have identified. The goal of this module is to detect the best gesture in order to discover its mystical importance in a specific community. There will also be a friendly graphical display for our output wherein our class labels, confidence level, and learning experience will be indicated. This entire process takes 10 milliseconds per frame.

V. MODEL TRAINING

During the training stage, the creation of the specialized dataset containing twenty unique Bharatanatyam mudras was considered, and several thousand annotated images of the mudras were collected from different perspectives and people. To improve generalization, data augmentation processes were used to randomly rotate, flip horizontally, and apply Gaussian blur. All samples were annotated manually by experts in the field. For evaluation purposes and to avoid any potential data leakage issues, the dataset was divided into training, validation, and testing samples with an 80-10-10 split.

For the base classifier, a deep CNN network structure was employed, refined via transfer learning to capitalize on the pre-trained weights of the large-scale hand gesture database. The training procedure was guided by the objective function of minimizing the Categorical Cross-Entropy loss function, formulated as follows: with being the total number of classes. Optimization of the loss function was done using the Adam optimizer method, where the initial learning rate was taken to be 10^ {-4}. Additionally, decay strategy was implemented to help achieve the convergence of the algorithm. Training of the model was done in 50 epochs on high-powered GPU systems, while the early stopping strategy was applied to prevent overfitting. The model attained an accuracy of 99.1% during training, with the validation accuracy coming to 97.8%. This showed that the model had a high ability to generalize across various users. However, the test loss was also low at 0.052, which indicated minimal errors produced when testing. These results show that the model is highly accurate in distinguishing between different hand gestures. Confusion matrix was also considered to analyze the confused hand gestures that resemble each other.

VI. RESULTS & DISCUSSION

The picture represents the user interface for a real-time Hand Mudra Recognition DL App. It is meant to recognize certain gestures made using the hands. The algorithm works through two phases: Firstly, the user inputs a picture of the hand gesture through the left pane, whereby the user can either load a file or take a picture with a webcam. Once the picture is successfully captured or uploaded, it gets confirmed and is shown on the right pane. Secondly, the user clicks on the Analyze Mudra button in order for the app to analyze the picture. In case the deep learning algorithm detects a mudra, it does so only when it has a probability of 90% of being accurate as stated in the instructions. The bottom pane that currently has "Awaiting Mudra Analysis..." serves as a status bar that may change during the process. The figure below shows that the first step of obtaining the input is done successfully, with the image of hand mudra shown in the left panel. At this point, the program is set to analyze the image, with the right panel awaiting the outcome of this process, while the status bar reads "Awaiting Mudra Analysis...." At this point, the whole setup is prepared to execute further tasks of the user, which will be the pressing of "Analyze Mudra" button, which initiates the actual function performed by the deep learning algorithm, i.e., analyzing the presented image of hand gesture. As instructed in the task, recognition is successful only when the confidence of recognition is no less than 90%.

Figure below shows the Analyzed Image Confirmation box, which is the official output and confirmation area situated at the right side of the Mudra Recognition application interface.

The main purpose of this window is to serve as an actual full-size representation of the captured hand mudra by the user, either through the webcam input or the image file. In effect, it confirms to the user that the image shown here is the exact image that the machine learning system will analyze when the “Analyze Mudra” button is clicked. Even though the box is labeled “Analyzed Image Confirmation,” during the early phases (judging from the content of the image), its sole purpose is only confirmation of the input image. This part of the webpage highlights the outcomes of the deep learning algorithm once the "Analyze Mudra" button is pressed, arranged in the section named "Top 3 Predictions." The program produces an output list of the categories created by the model, together with the respective Mudra Name and Confidence level.

In the current instance, it is noted that the model is capable of predicting that the gesture made is a representation of the Mayura mudra with a high degree of confidence of 94.47%, hence confirming the reliability of the prediction as it exceeds the threshold of 90%. In light of the prediction, further information is automatically provided by the website on the meaning of the predicted mudra under the heading of “What the Mayura Mudra Indicates.”


Top 3 Predictions

Rank	Mudra Name	Confidence
1	**Mayura(1)**	**94.47%**
2	Tripathaka(1)	5.01%
3	Shukatundam(1)	0.17%

What the Mayura Mudra Indicates

Category: Asamyuta Hasta (Single Hand) Meaning: Peacock, creeper, bird's egg, tearing hair, or applying tilak.

VII. USER STUDY

In order to test the effectiveness and efficiency of this system in practice, a user study was performed using fifteen subjects, consisting of both experts in Bharatanatyam dance and novices. The testing procedure used both quantitative measures of task completion and qualitative feedback concerning assessments. Participants were asked to perform a sequence of particular mudras in front of the system, and the accuracy of these poses was measured along with the response time of the system. The evaluation of the developed system was based on certain parameters such as usability, interpretative clarity, and its efficiency as an educational tool.

From the results obtained during observations in this study, it was found that the level of user satisfaction was very high especially considering the very rapid response obtained through the ability of the processing to reach 100 FPS. The professional dancers stated that the confidence levels exceeding 90% were accurate in terms of the users' technical performance, while the novices were satisfied with the use of semantic interpretation due to its importance in understanding the symbolism of the movement. Based on quantification, the recognition rate was found to be very high among various users, although there were some minor variations based on the user's proximity to the camera.

VIII. CONCLUSION AND FUTURE SCOPE

The authors would like to thank the Department of Computer Science and its Performing Arts Laboratory for the provision of computational and laboratory resources necessary for the completion of this research. Our sincere appreciation goes out to the professionals who took their time to volunteer and contribute towards the development of our data sets and user study assessments. The authors would like to give special thanks to our faculty advisors for the technical advice they provided throughout the development of the AI models. This research is made possible, partly, by the generous funding received from the National Foundation for Arts and Technology.

1. Future Work

Though our recognition framework has proven highly effective, the following steps for future research would be aimed at developing our approach to perform gesture analysis dynamically.

These include:

- Temporal Modeling:

Adding Long Short-Term Memory (LSTM) or Transformer-based methods to identify fluidity and transitions within Mudras that create sentences (Vakyartha).

- 3D Pose Optimization:

Using the data of the $z$coordinate in MediaPipe to enhance gesture recognition through better depth perception.

- Deployment Optimization:

Looking into quantizing and compiling models (such as TensorFlow Lite) in order to have highly optimized models running on embedded system.

The key differentiators of our approach include:

1. Superior Accuracy:97.8% vs. 73% mAP in YOLOv3
2. Pedagogical Validation: 90% confidence gate ensuring geometric precision
3. Computational Efficiency: Lightweight architecture suitable for real-time use
4. Robust Feature Extraction: Media Pipe-based approach invariant to environmental factors

Further development will include recognizing Samyukta (double-hand) mudras and dynamic combination of mudras and adavus. The integration of the system into mobile devices will make it more accessible to dance practitioners all over the world.

Table:

Metric	Result	Interpretation
Training Accuracy	99.9%	Excellent fit to the training data.
Validation Accuracy	Interpretation	97.80%
Test Loss (Categorical CrossEntropy)	Excellent fit to the training data.	0.052
Inference Time (per frame)	Interpretation	~10 milliseconds

ACKNOWLEDGMENT

It is our great pleasure to thank the Department of Computer Science and its Performing Arts Laboratory for the computing facilities and dedicated studio space that enabled us to conduct the necessary experiments. We owe much gratitude to all those talented dancers and coaches who willingly devoted their time to the process of creating data sets and evaluating user studies. Our thanks are especially due to our faculty advisers who provided essential technical support in developing the AI-based models. We gratefully acknowledge the financial assistance provided by the National Foundation for Arts and Technology, which allowed us to explore deep learning for cultural heritage preservation. Models.

REFERENCES

[1] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," University of Washington, arXiv:1804.02767, 2018.
[2] S. Haridas and R. Bai V, "Detection and Classification of Indian Classical Bharatanatyam Mudras Using Enhanced Deep Learning Technique," unpublished.
[3] L. T. Bhavanam and G. N. Iyer, "On the Classification of Kathakali Hand Gestures Using Support Vector Machines and Convolutional Neural Networks," in Proc. Int. Conf. Artificial Intelligence and Signal Processing (AISP), 2020.
[4] N. I. Hassan et al., "People Detection System Using YOLOv3 Algorithm," in Proc. IEEE, 2020.

[5] S. Masood et al., "Real-Time Sign Language Gesture (Word) Recognition from Video Sequences Using CNN and RNN," in Advances in Intelligent Systems and Computing. Cham, Switzerland: Springer.
[6] N. Jain et al., "An Enhanced Deep Convolutional Neural Network for Classifying Indian Classical Dance Forms," *MDPI*, 2020.