FMDB *Transactions on Sustainable Intelligent Networks*

Hierarchical Real-Time Gesture Interpretation of Bharatanatyam Mudras Using Bidirectional LSTM Networks

Advika Seetharaman[1], S. Vaishnavii[2,*], Anika Seetharaman[3], Prasanna Ranjith Christodoss[4], Premanand Jothilingam[5], Karthikeyan Sivanandi[6]

[1,2,3]Department of Computer Science and Engineering, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
[4]Department of Computing, Mathematics and Physics, Messiah University, One University Ave, Mechanicsburg, Pennsylvania, United States of America.
[5]Department of Life Cycle Service, Yokogawa Corporation of America, West Valley City, Utah, United States of America.
[6]Department of Research and Development, InfraSecure AI LLC, Casper, Wyoming, United States of America.
as1897@srmist.edu.in[1], vaish219@gmail.com[2], ani.seeth@gmail.com[3], prchristodoss@messiah.edu[4], premanand.j@yokogawa.com[5], admin@infrasecureai.com[6]

Abstract: Mudras, symbolic hand gestures, convey narrative and emotional content in Bharatanatyam, one of India's oldest classical dance genres. Sign language and human-computer interaction recognize gestures, but Bharatanatyam mudras are difficult to decipher. This work introduces the first hierarchical gesture recognition system that recognizes mudras and their meanings from video data. The MediaPipe Holistic architecture represents gestures as body and hand skeleton landmarks. The joint prediction uses a Bidirectional Long Short-Term Memory (BiLSTM) network with a common temporal backbone and two independent classification heads: a seven-class mudra head (Level 1) and a three-class semantic meaning head. Understanding that the Level 2 head anticipates one of up to three local indices of meaning defined within a mudra vocabulary rather than open-vocabulary semantic translation is crucial. The model was trained and tested on 2,400 gesture sequences from two practitioners, recorded under different settings, for 20 mudra–meaning pairs. On the held-out test set, BiLSTM scored 99.72% weighted F1 Score for mudra classification and 100.00% for semantic meaning prediction. The three-class meaning head distribution, tiny dataset, and two-performer constraint should be considered when interpreting the latter finding. These results beat a unidirectional LSTM model by 0.83 and 0.56 percentage points in mudra and meaning F1-scores, respectively. Hierarchical, context-aware gesture interpretation on common hardware supports dance teaching, cultural preservation, and human-computer interaction.

Keywords: Zero-Trust Architecture; Artificial Intelligence; Network Traffic Optimization; Deep Reinforcement Learning; Device Posture; End-User Behavior; App Sensitivity.

Received on: 29/05/2025, **Revised on:** 10/08/2025, **Accepted on:** 03/10/2025, **Published on:** 15/08/2026

Journal Homepage: https://www.fmdbpub.com/user/journals/details/FTSIN

DOI: https://doi.org/10.69888/FTSIN.2026.000734

Cite as: A. Seetharaman, S. Vaishnavii, A. Seetharaman, P. R. Christodoss, P. Jothilingam, and K. Sivanandi, "Hierarchical Real-Time Gesture Interpretation of Bharatanatyam Mudras Using Bidirectional LSTM Networks," *FMDB Transactions on Sustainable Intelligent Networks*, vol. 3, no. 3, pp. 137–150, 2026.

[*]Corresponding author.

1. Introduction

Bharatanatyam is one of India's most technically codified classical dance traditions, with roots in ancient Sanskrit texts on performance [2]. One of the key components of the expressive vocabulary of Bharatanatyam is the use of hasta mudras that have considerable narrative, emotional, and philosophical significance [4]. Each mudra has a distinct definition in terms of finger positions, palm orientation and body position. A number of these mudras are inherently polysemous in that a single mudra may point to different conceptual referents depending on the narrative context [8]. For example, the Alapadmam mudra may point to a flower, birth, or fruit depending on the narrative context, as discussed in Bharatanatyam viniyoga traditions and literature [10]. The polysemy of mudras poses a key challenge for computational gesture analysis. The dominant gesture recognition systems all follow a flat classification paradigm, where each gesture is classified into a single predefined category, without addressing the issue of contextual semantic meaning [5]. Such an approach is not appropriate for the analysis of Bharatanatyam dance, as the identification of a gesture's physical shape is only one aspect of its communicative content, akin to lexical identification alone in natural language processing [12]. A system that can simultaneously identify gesture and semantic meaning within a unified framework would be more appropriate for representing the communicative organization of the dance [1].

The pragmatic need for such a system is substantial, as a traditional dance performance offers no real-time explanation of gesture semantics, thereby creating a barrier for those unfamiliar with Bharatanatyam gesture symbolism [3]. An automated system for annotating gestures that can recognize and interpret them in real time may aid educators, students, and archivists in preserving and disseminating the art form [6]. In addition, high-accuracy gesture interpretation is directly relevant to human-computer interaction (HRI) and assistive technology research [7]. In this paper, a hierarchical gesture interpretation framework is proposed that jointly interprets gesture identification and semantic meaning from monocular video input in real time. The system analyses a live webcam video feed using the MediaPipe Holistic framework, which provides real-time pose and hand landmark detection [8]. This generates feature vectors, which are then fed into a sequence using a sliding-window approach and subsequently into a Bidirectional Long Short-Term Memory (BiLSTM) network with a shared backbone and two independent classification heads for predicting gesture category and semantic meaning, respectively. A dataset comprising seven mudra gesture categories and their respective meanings, totaling 20 unique combinations, was collected via structured webcam recordings with two practitioners under varying conditions [21]. The main contributions of this work are as follows:

- A two-level hierarchical classification model for interpreting Bharatanatyam gestures that simultaneously identifies mudra categories and their context-dependent semantic meanings from temporal landmark sequences [9].
- To the authors' knowledge, this is one of the first learning architectures that address this task in a unified manner [11].
- A BiLSTM model that has a shared temporal component and a dual classification head that can be trained using a compound categorical cross-entropy loss criterion for both classification levels concurrently [15].
- A novel dataset that comprises 2,400 gesture sequences from 20 different mudras and meaning pairs, collected from two practitioners in different lighting and positional conditions [17].
- A real-time inference pipeline that can perform confidence thresholding and temporal stability filtering for hierarchical gesture annotation in real-time using standard hardware [19].
- An ablation study that demonstrates the performance benefit of bidirectional over unidirectional modeling for this task [13].

2. Related Work

Research on computational hand gesture recognition spans several decades and multiple application domains. This section situates the proposed work within three relevant bodies of literature: the overall sign language and gesture recognition literature, the literature on Bharatanatyam mudra recognition systems, and the literature addressing the limitations of existing gesture recognition systems.

2.1. Sign Language and Gesture Recognition

Previous gesture recognition approaches were based on feature descriptor methods, such as Histogram of Oriented Gradients (HOG) or Gabor features, combined with simple classifiers, as implemented by Ghotkar et al. [9]. Feature-based approaches for Indian Sign Language recognition serve as baseline methods for classifying static hand configurations. These initial approaches demonstrated that simple spatial features are sufficient for static gesture recognition, but inadequate for more complex dynamic gesture recognition. With the emergence of deep learning techniques, the accuracy of gesture recognition has improved significantly using Convolutional Neural Networks (CNN) in various domains [10]. The recognition of dynamic gestures has further encouraged the implementation of Recurrent Neural Networks (RNN). The Long Short-Term Memory (LSTM) architecture has been shown to perform well at gesture classification. The Transformer architecture with relative

position encoding has achieved state-of-the-art results in sign language gesture recognition [12]. Yaseen et al [11] implemented MediaPipe hand tracking using the Inception-V3 model and a forward LSTM, thereby establishing a strong methodological foundation for landmark-based gesture recognition. However, the implementation of Bidirectional LSTM (BiLSTM) architectures for gesture recognition has been less explored, particularly within the context of classical dance gesture analysis.

2.2. Bharatanatyam Mudra Recognition

Computational studies on the recognition of Bharatanatyam mudras have shown significant growth over the past few years. Early works by Varsha and Pai [3] used SVM classifiers with HOG features on static mudra images, demonstrating the feasibility of computational classification with handcrafted representations. Kalaimani and Sigappi [4] proposed using convolutional networks for static mudra image classification with VGG architectures. They provided a comprehensive survey on gesture recognition techniques in the domain [1]. Madhura and Manjula [2] proposed a novel model to address intra-class variability caused by stylistic variations observed during gesture execution. Kumar et al. [13] proposed a novel architecture, the wavelet-based multi-head progressive attention network, to address the pose recognition problem. Sadhana et al. [14] proposed using deep learning and meta-learning algorithms to address the problem of mudra recognition. Recent studies have attempted to extend the concept of human recognition to video-based environments. Malavath and Devarakonda [15] used deep learning to classify mudras from video data. Raj et al. [16] have developed a systematic approach for tuning deep convolutional neural networks (CNNs) for the classification of Bharatanatyam mudras.

This was achieved by addressing domain-specific challenges in hyperparameter tuning. Subramanian et al. [17] developed a system for automatically detecting Bharatanatyam gestures using landmark-based features. This work demonstrates the system's viability for Bharatanatyam gesture detection. Haridas and Bai [18] used enhanced deep learning techniques for the detection and classification of mudras. The NATYA-AI framework was developed to integrate body pose, facial expression, and hand gesture for the interpretation of Bharatanatyam dance [19]. Paul et al. [20] developed a system for robust detection of Bharatanatyam gestures that combined hand detection and gesture classification. Baskar et al. [21] developed a system, called MudraGyaan, to extract features for classifying Bharatanatyam mudras. Nandeppanavar et al. [22] and Akarsha et al. [10] used deep learning for the categorization of Bharatanatyam mudras [23]. Beyond classification, Majumdar et al. [6] developed HasTA, a tutoring tool for Bharatanatyam dance based on learning theory, which validates the educational potential of automated mudra interpretation. Futane and Dharaskar [7] examined the interpretation of Indian sign gestures as a communication vocabulary, which is related to the semantic interpretation of gestures, as in the current research.

2.3. Limitations of Prior Approaches

There are two clear gaps in the current Bharatanatyam recognition systems that motivate the research. First, there is a lack of hierarchical semantic interpretation. This is the biggest gap in the current literature on Bharatanatyam recognition. The polysemy of the mudras is a significant gap that has been overlooked in current classifiers. A single gesture may have several meanings depending on the narrative context [15]. Therefore, to fill the gap in the current literature, the authors propose a joint multi-task learning approach. Secondly, most systems currently work on static images or short pre-recorded video clips rather than continuous video feeds. This restricts their usage for live performances or interactive teaching sessions [16]. Though Subramanian et al. [17] have demonstrated a system for real-time detection of Bharatanatyam dance, their method relies on gesture localization rather than interpretation. It does not employ hierarchical classification methods.

2.4. System Architecture

The system analyzes webcam videos step by step to identify hand gestures, known as mudras. The system does not just recognize the type of mudra being performed; it also offers insight into its symbolic meaning. The entire process, from recognizing the videos to interpreting the mudras, is depicted in Figure 1. This is a detailed analysis of the entire process. The video is captured using OpenCV's VideoCapture function at a resolution of 640x480 pixels and an approximate framerate of 30 frames per second. The frames are then passed through the MediaPipe Holistic model to detect face, body, and hand landmark locations. However, researchers are interested in the pose and hand-related information. The final feature vector has 258 dimensions. This includes 132 dimensions related to body pose information (33 points, 4 attributes each), 63 dimensions related to left-hand information (21 points, 3 attributes each), and another 63 dimensions related to right-hand information (21 points, 3 attributes each). When the hands are not visible, zero vectors are used to maintain the dimensionality. This allows us to process the video data efficiently. The sequence of consecutive feature vectors is grouped into fixed-length temporal sequences of $T = 30$ frames using a sliding-window buffer. This sequence is then passed to a bidirectional long short-term memory (BiLSTM) neural network to predict the probabilities of seven different hand gesture classes (mudras) and three possible meaning classes. However, to make the system more accurate and applicable to real-world scenarios, it considers a result valid only if the confidence level is at least 70%.

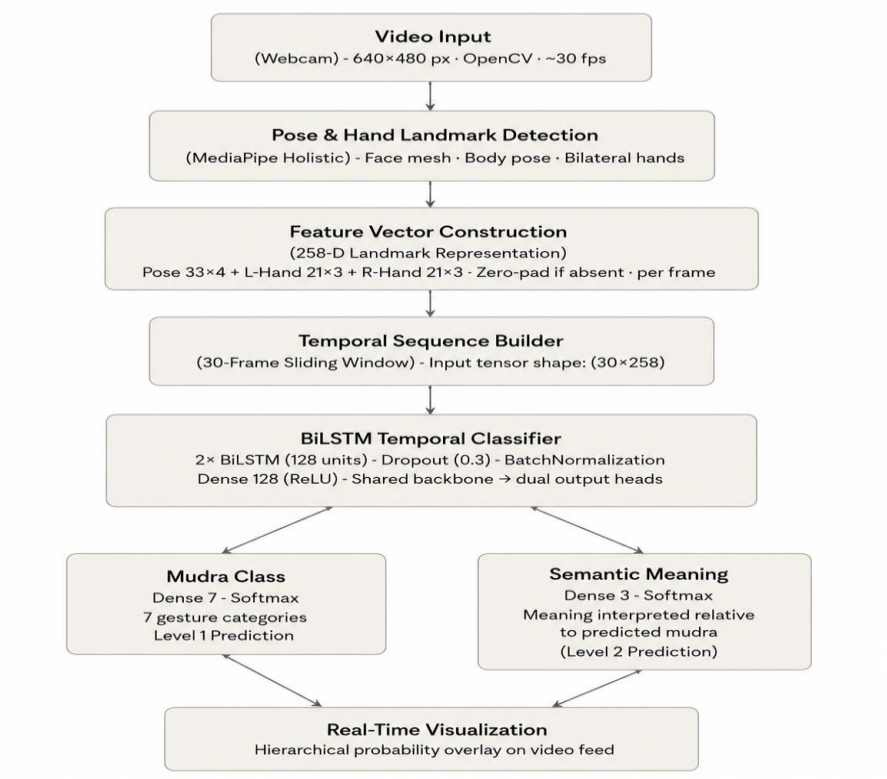

Figure 1: Architecture diagram

Additionally, the same gesture-meaning combination must be detected for 10 consecutive frames. This ensures incorrect classifications are filtered out. The results are then superimposed over the video feed to give the users an idea of what the system is interpreting.

3. Methodology

3.1. Dataset Construction

To create a dataset of Bharatanatyam mudras, researchers recorded two performers performing seven single-hand gestures using a webcam. The lighting, distance, and body positioning were varied to create a diverse dataset for training strong models. The seven single-hand gestures are Alapadmam, Ard- hachandra, Kartarimukha, Katakamukham, Mrighasrisham, Pathakam, and Shikaram, based on traditional Viniyoga descriptions of Bharatanatyam dance. The meaning of each mudra depends on the situation, yielding 20 unique combinations of mudras and their meanings, as presented in Table 1. For each of these combinations, researchers recorded 120 gesture sequences. In each sequence, researchers recorded 30 frames at approximately 30 frames per second, corresponding to one complete gesture. To ensure consistency, researchers displayed a 3-second countdown before each recording, allowing the performer to prepare for the gesture. This method enabled us to generate a holistic dataset that captures the subtleties of Bharatanatyam Mudras while accounting for variability across performance conditions. The complete dataset, therefore, contains:

$N_{seq} = 20 \times 120 = 2400$ gesture sequences (1)

$N_{frames} = 2400 \times 30 = 72,000$ frames (2)

The image is then converted into a 258-dimensional landmark feature vector, which serves as the final dataset for model training.

Table 1: Bharatanatyam mudras and their semantic meanings

Mudra	Semantic Meanings (Viniyoga)
Alapadmam	Pushpa (Flower), Janma (Birth), Phala (Fruit)

Ardhachandra	Ardha Chandra (Half Moon), Chintana (Thinking), Kati (Waist)
Kartarimukha	Nayanante (Eye), Lata (Creeper), Agami (Next)
Katakamukham	Muktashaangdhanam (Necklace), Gandha (Smell)
Mrighasrisham	Mriga (Deer), Griha (House), Bhaya (Fear)
Pathakam	Khadga (Sword), Ashirvada (Blessing), Sammarjanaya (Sweep)
Shikaram	Smarana (Recollection), Stambha (Pillar), Parirambhavidikrama (Hug)

3.2. Landmark Extraction using MediaPipe Holistic

Landmark detection is performed using MediaPipe Holistic. MediaPipe Holistic detects a face mesh, body pose, and both hand landmarks from a monocular video stream in real time. It provides three landmark sets for each frame. It provides 33 pose landmarks, 21 landmarks for the left hand, and 21 landmarks for the right hand. Pose landmarks carry four attributes per point:

(x, y, z, v) (3)

where x and y denote normalized image-plane coordinates, z represents relative depth, and v represents landmark visibility confidence. Hand landmarks carry three spatial coordinates:

(x, y, z) (4)

The per-frame feature vector is formed by concatenating flattened pose and hand landmark arrays:

$F = [Ppose, Pleft, Pright]$ (5)

where:

- $P_{pose} \in R^{33\times4}$ = 132 dimensions
- $P_{left} \in R^{21\times3}$ = 63 dimensions
- $P_{right} \in R^{21\times3}$ = 63 dimensions

MediaPipe Holistic also detects 468 facial mesh landmarks, which are not used in our features, since the Bharatanatyam mudras are more related to hand configuration and arm movement. In contrast, the facial movements are not relevant. Also, adding facial landmarks increases the dimensionality of our features considerably without a significant increase in discriminatory power.

3.3. Temporal Sequence Construction

Recognition of dynamic gestures also requires modeling temporal motion trajectories rather than static hand configurations. Per-frame feature vectors are grouped into fixed-size temporal sequences of T = 30 consecutive frames to construct input tensors of shape:

$$X \in R^{T\times D} = R^{30\times258} \quad (6)$$

In real-time inference, a sliding window buffer stores the last 30 feature vectors. The oldest vector is discarded, and the new vector is appended to the buffer for each incoming frame. A full sequence is guaranteed to be available after the initial 30-frame warm-up period.

3.4. Hierarchical Mudra-Meaning Labeling

The classification task is formulated as a two-level hierarchical prediction problem.

3.4.1. Level 1 - Mudra Classification

Each sequence is assigned a mudra label from a set of Cm = 7 classes, encoded as a one-hot vector:

$y_{mudra} \in \{0, 1\}^7$ (7)

3.4.2. Level 2 - Semantic Meaning Classification

Each mudra is associated with two or three semantic meanings. Meaning labels are represented using a shared three-class output space with Cmean = 3 classes:

$$y_{meaning} \in \{0, 1\}^3 \quad (8)$$

The number of available meaning labels varies for mudras (2 or 3), and the meaning class indices get remapped based on the predicted identity of the mudras from Level 1. For example, meaning class index 0 refers to 'Pushpa' (Flower) for Alapadmam, 'Khadga' (Sword) for Pathakam, and 'Mriga' (Deer) for Mrighasrisham. Therefore, the correctness of a predicted meaning depends on the identity of the mudras. Note that both prediction heads are trained jointly.

3.4.3. Dataset Split

The training and test splits of the dataset were determined using stratified sampling by mudra label, ensuring equal proportions of each class in both sets. The splits are 85%/15% train/test, for a total of 2,040/360 sequences.

3.5. Deep Learning Model Architecture

A bidirectional Long Short-Term Memory architecture describes the progression of gestures over time. Unlike a regular LSTM, which interprets the data step-by-step in one direction, a Bidirectional LSTM also interprets the data in the opposite direction. This way, the data from the later gestures helps to clarify the earlier gestures. Especially with hand gestures, the endpoint's position is a significant determinant of its meaning. The model accepts input tensors of shape $X \in R30\times258$. The shared temporal backbone consists of:

- BiLSTM Layer 1 (units = 128 in each direction, return_sequences=True): This layer processes the sequence in a bidirectional manner to generate a complete temporally ordered hidden state sequence.
- Dropout (rate = 0.3): This dropout layer is added after the first BiLSTM layer to avoid overfitting.
- BiLSTM Layer 2 (units = 128 in each direction, return_sequences=False): This layer generates a fixed-length sequence embedding.
- Dense layer (128 units, ReLU activation) + batch Normalization + Dropout (0.3): This layer generates the common latent representation used by the two classification heads.

This shared representation is then branched into two parallel output heads:

- Mudra Head: Dense layer with 7 units and SoftMax activation, producing $\hat{y}_{mudra} \in R7$ (Level 1 prediction)
- Meaning Head: Dense layer with 3 units and SoftMax activation, producing $\hat{y}_{meaning} \in R3$ (Level 2 prediction)

The full architecture is summarised in Table 2.

Table 2: BILSTM model architecture

Layer	Operation	Output Shape
Input	Temporal sequence	(batch, 30, 258)
BiLSTM L1	128 units	(batch, 30, 256)
Dropout	0.3	(batch, 30, 256)
BiLSTM L2	128 units	(batch, 256)
Dense + batch Normalization	128 units	(batch, 128)
Mudra Head	Dense (7), Softmax	(batch, 7)
Meaning Head	Dense (3), Softmax	(batch, 3)

3.6. Training Strategy

Here, training starts with the Adam optimizer, with a learning rate $\eta = 10{-}3$. Although they are two distinct heads, both play an equal role in determining the total loss. Both are responsible for adding a categorical cross-entropy term without any further weighting:

$$L = L_{mudra} + L_{meaning} \quad (9)$$

The term L_{mudra} refers to the categorical cross-entropy loss used for predicting one of seven mudras. Meanwhile, $L_{meaning}$ stands for the similar loss applied to distinguishing among three possible meanings.

Table 3: Training configuration parameters

Parameter	Value
Optimizer	Adam
Learning Rate	0.001
Batch Size	16
Epochs	150
Sequence Length	30 frames
Feature Dimension	258
Train/Test Split	85/15

Each of these losses operates using true labels encoded as one-hot vectors. Accuracy during both training and validation phases is tracked separately for every output branch. The training setup details are presented in Table 3.

3.7. Experimental Setup

3.7.1. Hardware Configuration

All experiments were conducted on a standard laptop equipped with a 10th-generation Intel Core i7 processor and 16 GB RAM. The video was captured using the laptop's webcam at 640 × 480-pixel resolution with an approximate frame rate of 30 FPS. No GPU acceleration, depth sensors, or any special motion capture hardware were used. This proves that the proposed system can be executed on common computing hardware.

3.7.2. Software Environment

The system was implemented using Python 3.9. The deep learning models were created using TensorFlow 2.10.1 with the Keras Functional API. Landmark detection was implemented using MediaPipe 0.10.10. The numerical computations were performed using NumPy 1.23.5. The video capture and display were done using OpenCV 4.13.0. The dataset splitting and evaluation metrics were calculated using scikit-learn 1.7.2. The training process was monitored using TensorBoard 2.10.1.

3.7.3. Real-Time Inference Configuration

During deployment, the trained model is loaded using the Keras load_model() method with compile set to False. Two filtering techniques are employed for model stability during real-time prediction. The first is a confidence threshold of $\theta = 0.7$ for the mudra head output. Any prediction that does not meet this threshold is filtered out. The second one is a temporal stability filter that requires a consistent prediction for the mudra-meaning over ten consecutive frames before the output is written to the buffer.

3.7.4. Qualitative Real-Time Validation

In addition to the stratified train-test evaluation approach on the recorded dataset, qualitative experiments for real-time validation of the system were conducted with several participants who were not part of the training dataset. This includes participants who are not formally trained in the classical dance form of Bharatanatyam. The trained model is deployed using the live inference pipeline as explained. Participants were asked to mimic the reproduced mudras while viewing the system's output. The system can detect a range of gestures from participants with varying levels of experience in the classical dance form. This indicates that the learned gesture representations might generalize beyond the two performers in the training data. However, these experiments are qualitative. The results are not formally evaluated as part of the quantitative evaluation metrics, as explained. This is a direction for future work. However, experiments with multiple participants indicate that the learned gesture representations generalize beyond the two performers in the training data.

4. Results and Discussion

4.1. Overall Classification Performance

The proposed hierarchical model was tested on a held-out test set containing 360 gesture sequences (15% of the full dataset), using stratified sampling to ensure proportional representation. This system simultaneously addresses two classification problems: seven-class mudra recognition and three-class semantic meaning prediction. The overall evaluation metrics for the proposed model are shown in Table 4. The provided metrics for the proposed semantic meanings must be viewed in the context in which they were designed. This is because the meanings are constrained in a three-class local output space relative to the corresponding mudra identity.

Table 4: Overall model performance on the test set

Task	Precision	Recall	F1-Score
Mudra Classification	0.9972	0.9972	0.9972
Meaning Prediction	1.0000	1.0000	1.0000

4.2. Training Convergence

The accuracy curves for both classification tasks converged quickly during the first 20-30 epochs and then stabilized until epoch 150.

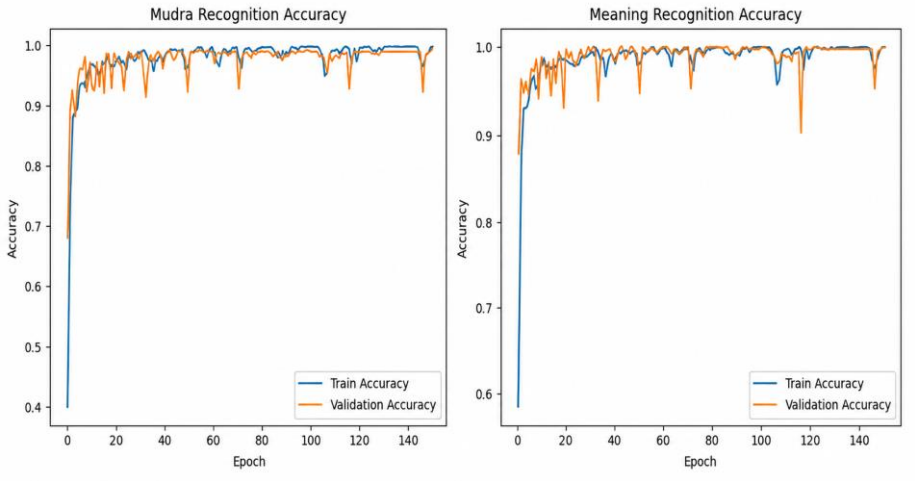

Figure 2: Training and validation accuracy curves for mudra recognition and semantic meaning prediction for BiLSTM

The high degree of alignment between the training and validation curves throughout training indicates that the proposed model fits the data without signs of overfitting, especially given the high number of model parameters relative to the dataset size. The training and validation curves for both classification tasks showed a general downward trend, with minor fluctuations typical of mini-batch optimization methods, as shown in Figure 2 and Figure 3.

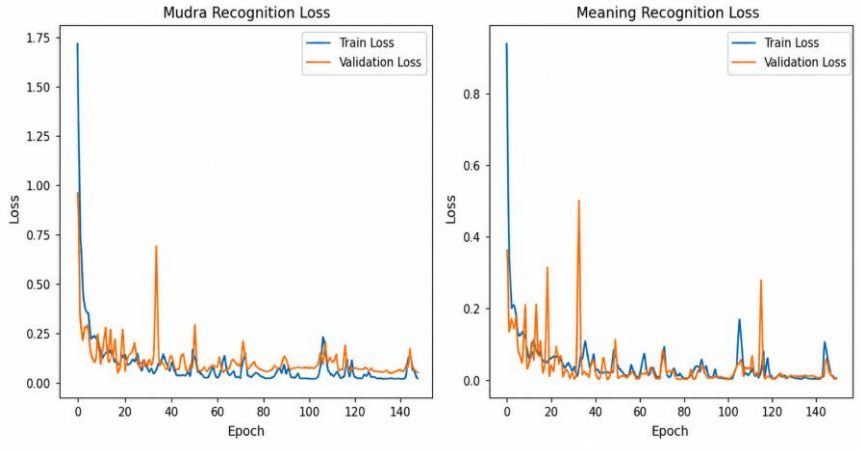

Figure 3: Training and validation loss curves for mudra recognition and semantic meaning prediction for BiLSTM

4.3. Per-Class Mudra Performance

Per-class classification performance for the seven mudra categories is presented in Table 5. Five of the seven classes, Alapadmam, Ardhachandra, Katakamukham, Mrighasrisham, and Shikaram, achieved perfect precision, recall, and F1-score across all test samples.

Table 5: Per-Class mudra classification performance

Mudra	Precision	Recall	F1-Score	Samples
Alapadmam	1.00	1.00	1.00	54
Ardhachandra	1.00	1.00	1.00	54
Kartarimukha	1.00	0.98	0.99	54
Katakamukham	1.00	1.00	1.00	36
Mrighasrisham	1.00	1.00	1.00	54
Pathakam	0.98	1.00	0.99	54
Shikaram	1.00	1.00	1.00	54

A single misclassification occurred in the Kartarimukha class, in which one test sample was misclassified as Pathakam. This error may reflect partial overlap in intermediate hand configurations between these two mudras during transition phases.

4.4. Semantic Meaning Prediction

Semantic meaning prediction achieves perfect weighted precision, recall, and F1-score (1.0000) across all three meaning classes on the test set. The test partition comprised 111 samples of meaning class 0, 131 samples of meaning class 1, and 118 samples of meaning class 2, providing a reasonably balanced evaluation across classes. The confusion matrix for semantic meaning prediction is shown in Figure 4, and that for the seven mudra classes is shown in Figure 5. Since meaning class indices are interpreted relative to the predicted mudra identity, these results suggest that the BiLSTM can model temporal variations associated with different semantic enactments of the same mudra within the present dataset. When evaluating semantic meaning accuracy conditioned on the related mudra identity, the score should not be interpreted as autonomous end-to-end semantic translation accuracy. The meaning head is dealing with a three-class local classification problem, with the three classes known to be closed per mudra, and the dataset is derived from two practitioners. It is an empirical question whether the nuanced trajectory distinctions apply to other performers. Thus, the result can be regarded as strong within the distributional performance on a constrained predictive task, not as an indication of deep semantic comprehension.

4.5. Ablation Study: Unidirectional LSTM versus BiLSTM

To determine the value added by bidirectional empirical modeling, an ablation study was conducted.

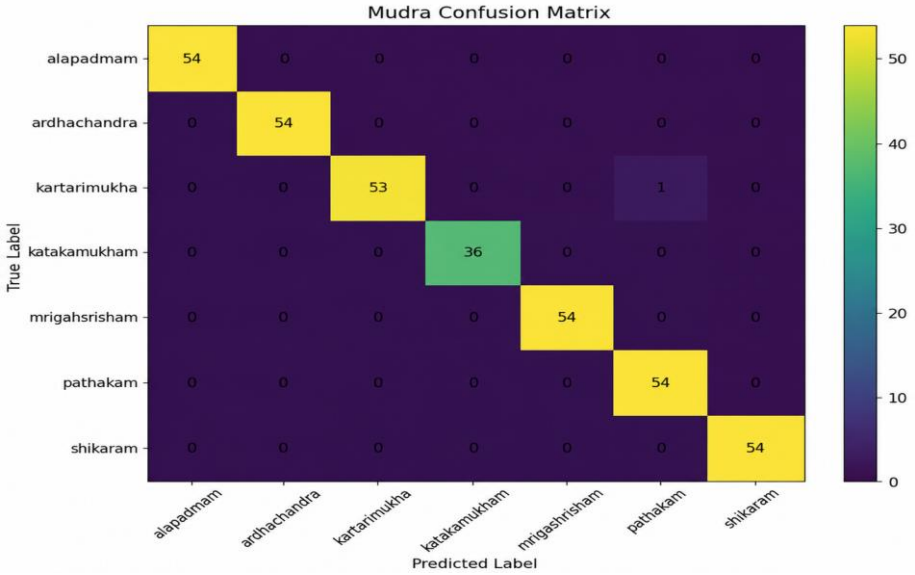

Figure 4: Confusion matrix for mudra recognition across the seven gesture classes for BiLSTM

The baseline keeps the same architecture as the BiLSTM model, unmodified (in terms of layer counts and dimensions, Dropout rate, Batch Normalisation, loss functions, optimizers, and training hyperparameters), and only modifies the BiLSTM layers to bidirectional LSTM layers of the same dimensionality (128 hidden units per layer). The results are shown in Table 6.

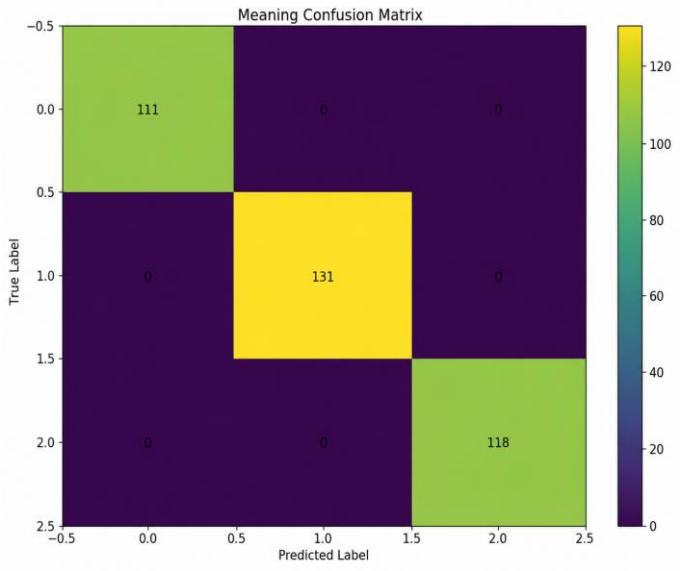

Figure 5: Confusion matrix for three-class semantic meaning prediction for BiLSTM

Scores across both tasks, improving the F1 score of mudras by 0.83 and of meanings by 0.56. BiLSTMs also outperform LSTMs across all gestures.

Table 6: Ablation study: LSTM vs BiLSTM architecture

Model / Task	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Parameters
LSTM – Mudra Classification	98.89	98.91	98.89	98.89	~348K
LSTM – Meaning Prediction	99.44	99.45	99.44	99.44	~348K
BiLSTM – Mudra Classification	99.72	99.73	99.72	99.72	~674K
BiLSTM – Meaning Prediction	100.00	100.00	100.00	100.00	~674K

For instance, BiLSTMs achieved perfect recall for Alapadmam and Mrighasrisham, while LSTMs achieved 0.9630 and 0.9815, respectively.

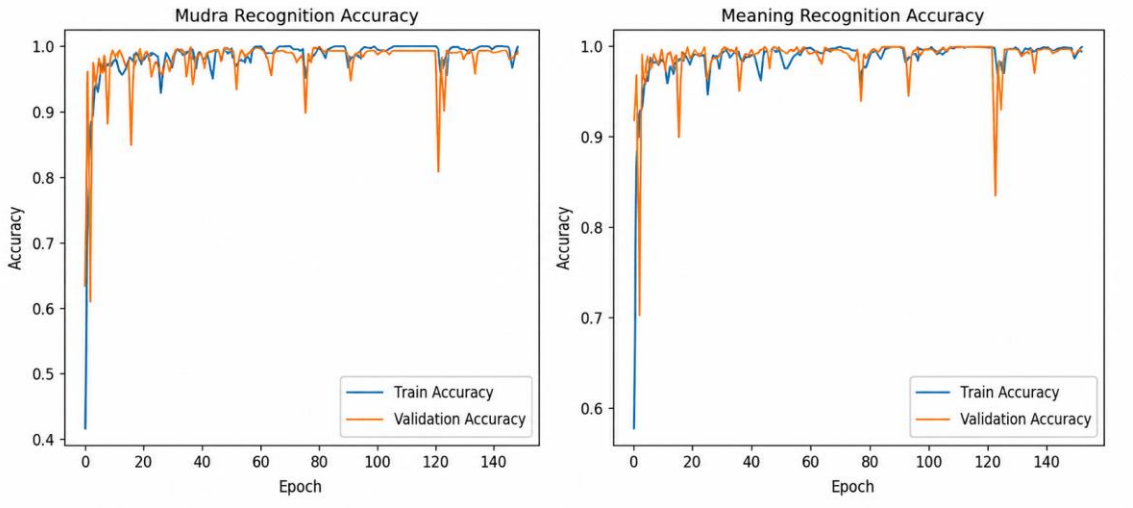

Figure 6: Training and validation accuracy curves for mudra recognition and semantic meaning prediction for LSTM

These gestures include precise finger movements in advance of the gesture, and the backward temporal pass of the BiLSTM enables the model to incorporate information from the completed gesture while forming a representation of prior frames. The training and validation convergence behavior for both architectures is illustrated in Figure 6 and Figure 7.

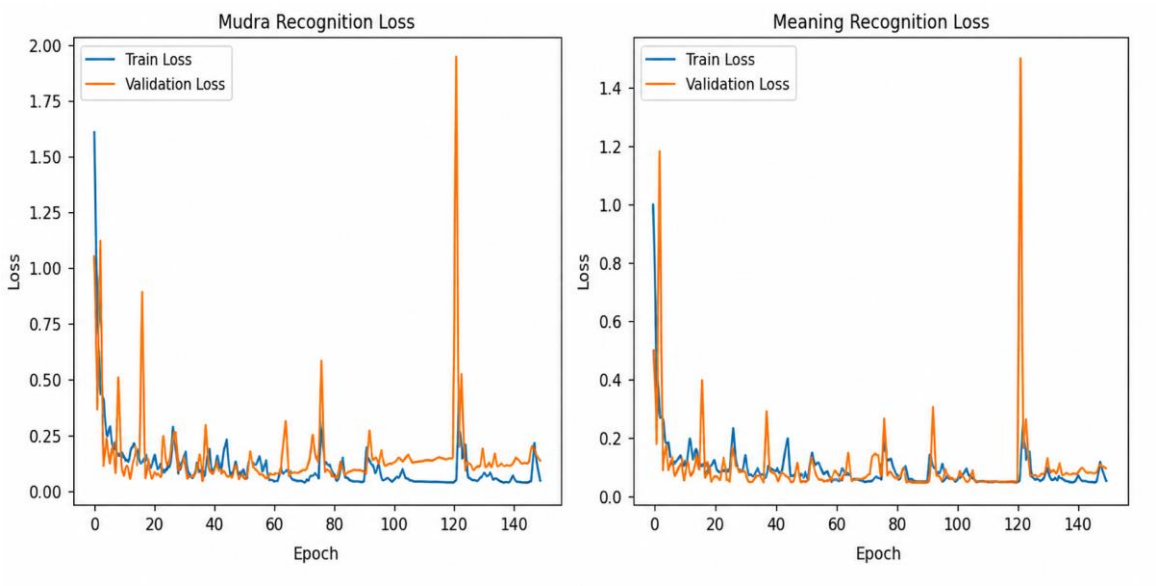

Figure 7: Training and validation loss curves for mudra recognition and semantic meaning prediction for LSTM

At the same time, the class-wise prediction distributions are further visualized in the confusion matrices shown in Figure 8 and Figure 9. The roughly 93% increase in the number of parameters for BiLSTM compared to LSTM constitutes an improvement in model complexity (674k vs 348k). Thus, the LSTM baseline model, with 98.89% mudra and 99.44% meaning accuracy, remains an acceptable option for implementation in low-resource environments.

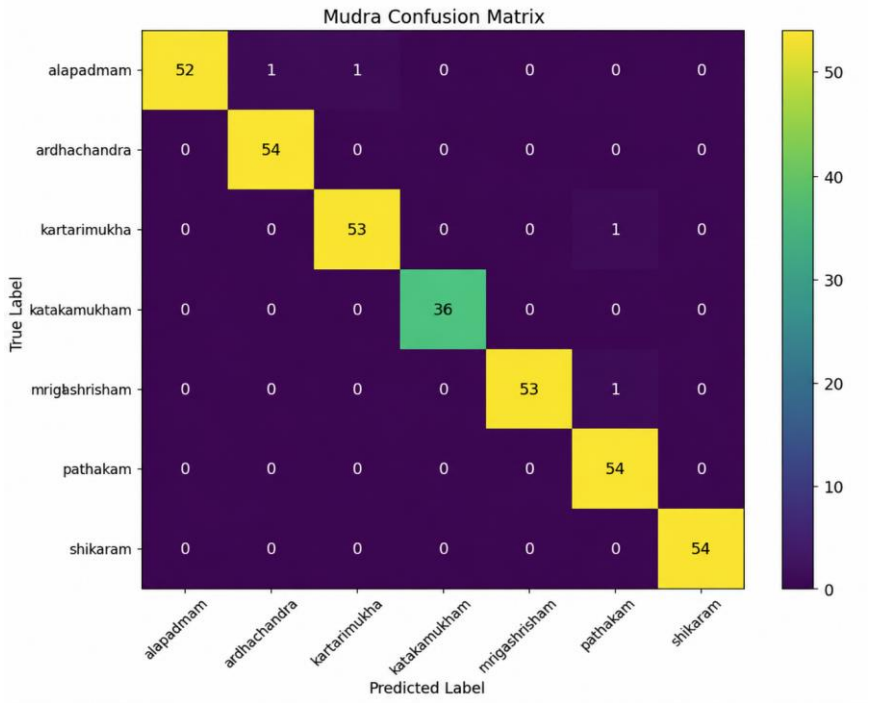

Figure 8: Confusion matrix for mudra recognition across the seven gesture classes for LSTM

Look to the future for model compression techniques, such as knowledge distillation for post-training quantization, to compress the BiLSTM without compromising its advantages.

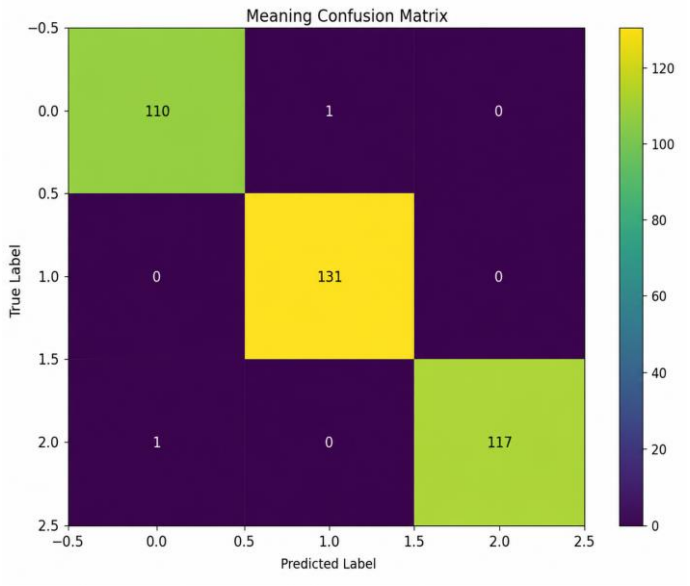

Figure 9: Confusion matrix for three-class semantic meaning prediction for LSTM

4.6. Applications and Impact

4.6.1. Cultural Preservation and Documentation

Bharatanatyam has developed a complex vocabulary of gestures that has assimilated many centuries of the philosophical, mythological, and aesthetic corpus of Indian civilization. The types of computational analyses proposed here would facilitate systematic documentation and archiving of classical dance performances by producing machine-readable semantic annotations for video recordings. These annotations make cultural archives searchable, allowing researchers and practitioners access to specific narrative fragments in extensive performance repositories. Furthermore, the system's real-time feedback capability also fosters intergenerational transmission by providing objective gesture evaluations to novices.

4.6.2. Dance Education and Pedagogical Tools

The proposed system's real-time gesture interpretation will be particularly useful in Bharatanatyam teaching. Because of gesture recognition, Bharatanatyam students will be able to evaluate their own practice by executing gestures at the desired level and receiving feedback from the model. This gesture recognition technology enables self-practice and eliminates the need for constant expert supervision to sustain practice. With gesture recognition, remote teachers will be able to provide automated grading and feedback for gestures. This will enable classical dance pedagogy to reach students in remote locations.

4.6.3. Accessibility and Audience Engagement

The system can be modified for use in theatres, museums, and streaming services, and greatly helps those who lack familiarity with Bharatanatyam's symbolic language. The use of video performances with real-time annotations of mudras and their meanings can provide the audience with contextual understanding without hindering the performance. This is particularly relevant to the hearing-impaired audience, as communicating through gestures is highly important and highly expressive.

4.6.4. Human-Robot Interaction

The gesture recognition system proposed here can be implemented in human-robot interaction (HRI) and gesture- driven interfaces. The classical hand gesture systems, with their spatial and semantic consistency, provide a means to formulate command structures for robotic control. The suggested multi-output configuration also allows for control of collaborative robotics in situations where verbal commands are not possible.

5. Conclusion

This paper introduces a hierarchical framework for interpreting gestures, focusing on monocular video-based recognition of Bharatanatyam hand mudras and their context-dependent semantic meanings. The framework integrates a Bidirectional Long

Short-Term Memory (BiLSTM) architecture with MediaPipe Holistic landmark extraction (Zhang et al. [8]) to perform gesture recognition and context analysis. The model was tested on a custom dataset of 2,400 gesture sequences, balanced over 20 mudra–meaning pairs. From the dataset, 360 sequences were held out for testing. The model attained a weighted F1-score of 99.72% for seven classes of mudra recognition and 100.00% for three classes of semantic meaning prediction. The ablation study confirmed the advantage of using a bidirectional LSTM over a unidirectional LSTM, especially on gesture classes that include subtle anticipatory motions. The framework delivers real-time predictions on standard computing devices and generates hierarchical probability visualizations for educational and archival purposes. Researchers found no prior work that simultaneously predicts Bharatanatyam mudra identity and contextually relevant meanings using a unified learning architecture. It should be mentioned that the Level 2 meaning head is designed to predict local semantic indices within the semantic range of each mudra's vocabulary, and the scope of the contribution is purposefully limited. This work paves the way for future studies on culturally aware semantic classical dance gestures, with the potential to contribute to the preservation of cultural heritage, dance teaching, and gesture-based HCI.

5.1. Limitations and Future Work

The proposed system demonstrates strong reliability on the given dataset; however, the current results are constrained by several limiting factors and outline the primary focus areas for future work. While the dataset involved two practitioners trained in a controlled setting, further informal prompt testing of the gestures with several participants, including those with no formal dance training, suggests that the learned gesture representations are robust. Nonetheless, the achieved high test accuracy on the held- out partition may still be a result of the attributes associated with the specific recording conditions and performer population. Thus, demonstrating formal quantitative cross-performer generalization through leave-one-performer-out validation and extending the dataset to performers of varying body types, experience, and stylistic interpretations is the most critical for future work. The existing gesture vocabulary consists of seven asamyukta hasta mudras.

The complete Bharatanatyam repertoire includes a vastly larger number of single- and double-hand gestures, and to extend the framework to accommodate this larger vocabulary, it will be necessary to use more comprehensive datasets and potentially more sophisticated sequence modeling approaches, such as Transformer models that can effectively capture long-range temporal dependencies. The 30-frame temporal window creates a minor latency when predictions are made during real-time inference. While this design increases recognition stability, future designs may consider alternative models that incorporate predictive causal architectures that predict before the temporal window elapses. The existing feature set does not leverage facial landmarks because mudra interpretation is solely hand- and arm-focused. However, adding facial expression data as a separate input stream may improve the disambiguation of gestures reinforced by facial expressions, as suggested by multimodal studies. Furthermore, fusion with natural language generation models can automate the narrative descriptions of performance sequences, facilitating large-scale digital archiving.

Acknowledgment: The authors sincerely acknowledge the academic support, research facilities, and professional collaboration provided by SRM Institute of Science and Technology at Ramapuram, Messiah University, Yokogawa Corporation of America, and InfraSecure AI LLC, which significantly contributed to the successful completion of this research work.

Data Availability Statement: The data for this study are available upon request from the corresponding author.

Funding Statement: This manuscript and research paper were prepared without any financial support or funding

Conflicts of Interest Statement: The authors have no conflicts of interest to declare. This work represents a new contribution by the authors, and all citations and references are appropriately included based on the information utilized.

Ethics and Consent Statement: This research adheres to ethical guidelines and obtains informed consent from all participants. Confidentiality measures were implemented to safeguard participant privacy.

References

1. M. Kalaimani, B. Latha, and A. N. Sigappi, "Survey on Hand Gesture Recognition in Bharathanatyam Mudras," *Telematique*, vol. 21, no. 1, pp. 7561–7577, 2022.
2. K. Madhura and H. M. Manjula, "A Hybrid Model Proposal to Recognize the Hand Gestures of Bharatanatyam Mudras," in 2023 *International Conference on New Frontiers in Communication, Automation, Management and Security (ICCAMS),* Bangalore, India, 2023.

3. K. S. Varsha and M. L. Pai, "Bharatanatyam Hand Mudra Classification Using SVM Classifier with HOG Feature Extraction," in Innovations in Computer Science and Engineering, ser. Lecture Notes in Networks and Systems. *Springer*, Singapore, 2020.
4. M. Kalaimani and A. N. Sigappi, "Implementation of VGG Models for Recognizing Mudras in Bharatanatyam Dance," *International Journal of Intelligent Systems and Applications in Engineering,* vol. 12, no. 3S, pp. 306–319, 2024.
5. S. Naaz and K. B. S. Kumar, "Integrated Deep Learning Classification of Mudras of Bharatanatyam: A Case of Hand Gesture Recognition," *Science Temper,* vol. 14, no. 4, pp. 1374–1380, 2023.
6. R. Majumdar, P. Bhawar, S. Sahasrabudhe, and P. Dinesan, "HasTA: Hasta Training Application—Learning Theory-Based Design of Bharatanatyam Hand Gestures Tutor," in *2014 IEEE 14th International Conference on Advanced Learning Technologies (ICALT),* Athens, Greece, 2014.
7. P. R. Futane and R. V. Dharaskar, "Hasta Mudra: An Interpretation of Indian Sign Hand Gestures," in *2011 3rd International Conference on Electronics Computer Technology (ICECT),* Kanyakumari, India, 2011.
8. F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C. L. Chang, and M. Grundmann, "MediaPipe Hands: On-Device Real-Time Hand Tracking," *arXiv preprint arXiv:2006.10214*, 2020. [Accessed by 12/03/2025].
9. A. S. Ghotkar, R. Khatal, S. Khupase, S. Asati, and M. Hadap, "Hand Gesture Recognition for Indian Sign Language," in *2012 International Conference on Computer Communication and Informatics (ICCCI)*, Coimbatore, India, 2012.
10. D. P. Akarsha, C. Hansitha, N. Padma Priya, S. K. Suresh, and K. S. Vaaruni, "Bharatanatyam Hastas Recognition Using CNN and Computer Vision," in *2025 5th International Conference on Pervasive Computing and Social Networking (ICPCSN)*, Salem, India, 2025.
11. Y. Yaseen, O. J. Kwon, J. Kim, S. Jamil, J. Lee, and F. Ullah, "Next-Gen Dynamic Hand Gesture Recognition: MediaPipe, Inception-V3 and LSTM-Based Enhanced Deep Learning Model," *Electronics*, vol. 13, no. 16, pp. 1–11, 2024.
12. N. Aloysius, M. Geetha, and P. Nedungadi, "Incorporating Relative Position Information in Transformer-Based Sign Language Recognition and Translation," *IEEE Access*, vol. 9, no. 10, pp. 145929–145942, 2021.
13. D. A. Kumar, P. V. V. Kishore, and K. Sravani, "Deep Bharatanatyam Pose Recognition: A Wavelet Multi-Head Progressive Attention," *Pattern Analysis and Applications*, vol. 27, no. 5, p. 53, 2024.
14. P. Sadhana, N. Ravishankar, and S. Palaniswamy, "Bharatanatyam Mudra Recognition Using Deep Learning and Meta-Learning Techniques," in *2024 1st International Conference on Communications and Computer Science (InCCCS)*, Bangalore, India, 2024.
15. P. Malavath and N. Devarakonda, "Natya Shastra: Deep Learning for Automatic Classification of Hand Mudra in Indian Classical Dance Videos," *Revue d'Intelligence Artificielle,* vol. 37, no. 3, pp. 689–701, 2023.
16. R. J. Raj, S. Dharan, and T. T. Sunil, "Systematic Approach to Tuning a Deep CNN Classifying Bharatanatyam Mudras," in Proceedings of the Satellite Workshops of ICVGIP 2021, ser. Lecture Notes in Electrical Engineering. *Springer*, Singapore, 2022.
17. R. R. Subramanian, V. Devamani, S. Prasanna, U. Sravanthi, V. Pranitha, and M. S. Hamsika, "Automated Real-Time Hand Gesture Detection for Bharatanatyam Dance," in *2025 International Conference on Computational Robotics, Testing and Engineering Evaluation (ICCRTEE),* Virudhunagar, India, 2025.
18. S. Haridas and V. R. Bai, "Detection and Classification of Indian Classical Bharathanatyam Mudras Using Enhanced Deep Learning Technique," in *International Conference on Innovations in Science and Technology for Sustainable Development (ICISTSD),* Kollam, India, 2022.
19. G. Veena, M. G. Thushara, G. K. P. K. Nambiar, and N. M. Kumar, "NATYA-AI: A Cultural AI Framework for Multimodal Interpretation of Bharatanatyam," *IEEE Access,* vol. 13, no. 12, pp. 207320–207339, 2025.
20. S. Paul, G. Sagar, P. P. Das, and K. Sreenivasa Rao, "Two-Stage Pipeline-Based Robust Hand Gesture Recognition from Bharatanatyam Dance Images," *Multimedia Tools and Applications*, vol. 84, no. 8, pp. 39667–39691, 2025.
21. S. Baskar, W. J. Hans, V. R. Anuprapaa, V. S. Solomif, and R. Arthi, "MudraGyaan: A Novel Feature Extraction Algorithm for Machine Learning–Based Bharatanatyam Mudra Classification," in *2024 International Conference on Advancement in Renewable Energy and Intelligent Systems (AREIS),* Thrissur, India, 2024.
22. A. S. Nandeppanavar, S. S. Kallur, P. Thotad, and V. A. Sankannavar, "Bharatanatyam Hasta Mudra Categorization Using Deep Learning Approaches," in *2023 IEEE North Karnataka Subsection Flagship International Conference (NKCon)*, Belagavi, India, 2023.
23. D. P. Akarsha, B. S. Monisha, K. L. Bhoomika, and V. D. Rao, "Bharatanatyam Mudra Classification Using CNN," in *2024 First International Conference on Software, Systems and Information Technology (SSITCON)*, Tumkur, India, 2024.