Single-Hand Gesture Recognition of Manipuri Classical Dance of India using a Supervised Machine Learning Approach

Mampi Devi[1†], Dipanjoy Majumder[2†], Alak Roy[3*†]

[1]Department of Cyber Security and Digital Forensic, National Forensic Sciences University, Tripura Campus, Agartala, 799001, Tripura, India.
[2]Department of Computer Science and Engineering, ICFAI University Tripura Kamalghat, Mohanpur, 799210, Tripura, India.
[3]Department of Information Technology, Tripura University (A Central University), Suryamaninagar, 799022, Tripura, India.

†These authors contributed equally to this work.

Abstract

Hand gesture recognition plays a pivotal role in human-computer interaction and cultural heritage preservation. This paper presents an innovative machine learning approach for recognizing single-hand gestures in Manipuri classical dance, a prestigious Indian classical dance form. Manipuri dancers utilize 25 distinct single-hand gestures (Asamyukta mudra) to convey meaning and emotions. Leveraging a Support Vector Machine (SVM) classifier, we develop a dataset comprising 750 images to recognize these gestures, achieving an impressive test accuracy of 80.76%. Our contributions include the creation of a dedicated dataset for Manipuri classical dance single-hand gestures and the proposal of an SVM-based recognition method. This work not only advances the field of hand gesture recognition but also highlights the potential of machine learning in preserving and promoting cultural heritage.

Keywords: Human computer interaction, Hand gestures recognition, Manipuri classical dance, Support vector machine, Sci-kit learn.

1 Introduction

Gestures are a non-verbal means of communication. All kinds of gestures have some meaning which is formed by hand movements, finger movements, head movements, or any other body part movements in harmonic manner [1]. Gesture recognition is a cross-disciplinary area of study. And, Dance gesture recognition is a sub-field of gesture recognition which is used to identify meaningful human expressions as a dance and drama medium of communication [2]. Dance gesture recognition can play an important role in cultural heritage preservation. Indian classical dance has a central role in ensuring the cultural heritage of India and thus plays an essential role within the research field.

Manipuri classical dance is an ancient classical dance form that originated in the Indian state of Manipur, situated in the North-Eastern region. It is more restrained than the other Indian classical dances where the artist does not make any eye contact with the audience. Manipuri classical dance has devotional themes and is performed on religious occasions and in temples. The dancers of Manipuri classical dance uses 37 hand gestures to communicate any meaning or message to another and to interact with the audience. Out of those 37 hand gestures, 25 of them are single-hand gestures (Asamyukta mudra) namely Ahitunda, Ankura, Alapallava, Ankusha, Ardhapataka, Ardhyachandra, Bhringa, Chatura, Dhenu, Hansapakshya, Hamsasya, Kartarimukha, Khatakamukha, Koraka, Kangula, Mrigashirsha, Musthi, Padmakosha, Pataka, Samdamsya, Shardulashaya, Shikhara, Suchimukha, Tripataka, Trishula and 12 of them are double hand gestures (Samyukta mudra) namely Karkata, Chakra, Rambhasuma, Shankha, Anjali, Arakhya, Pasha, Samputaha, Kokila, Pushpaputa, Shuka, Swastika [3] as shown in Figure 1. These hand gestures add aesthetic beauty to the dance, convey emotions, and preserve the cultural and spiritual heritage of Manipur, making it a captivating and culturally rich art form.

In the present scenario of the digital world, it is very much essential to preserve and bring up the beauty, devotion, and craftsmanship of Manipuri classical dance of India by evaluating its traditional relationship while embracing technological change. By doing so, can create a traditional cultural bridge between the present and surpassed, which displays our traditional and cultural heritage. This research represents the dedication to uphold tradition while embracing modernity, motivating researchers to use cutting-edge technology to reveal the hidden potential of ancient art, and bringing the essence of Manipuri dance to the world in a fresh and significant way.

Some of the issues and challenges faced for the recognition of single-hand gestures of Manipuri classical dance are highlighted as follows:

- *Selection of Machine Learning algorithm for recognition of single-hand gestures of Manipuri classical dance:* The traditional machine learning (ML) methods of image classification and recognition always faces several limitations. These traditional methods always rely on manual feature engineering which is very much time consuming and requires expertise to identify the features which is relevant. There are problems with the high dimensionality of data and scalability with large datasets which lead to computational complexities. Machine learning methods like KNN, Decision Tree, Random Forest and Naive Bayes face limitations in image classifications. They face scalability, overfitting, complex feature relationship problems [1, 2].

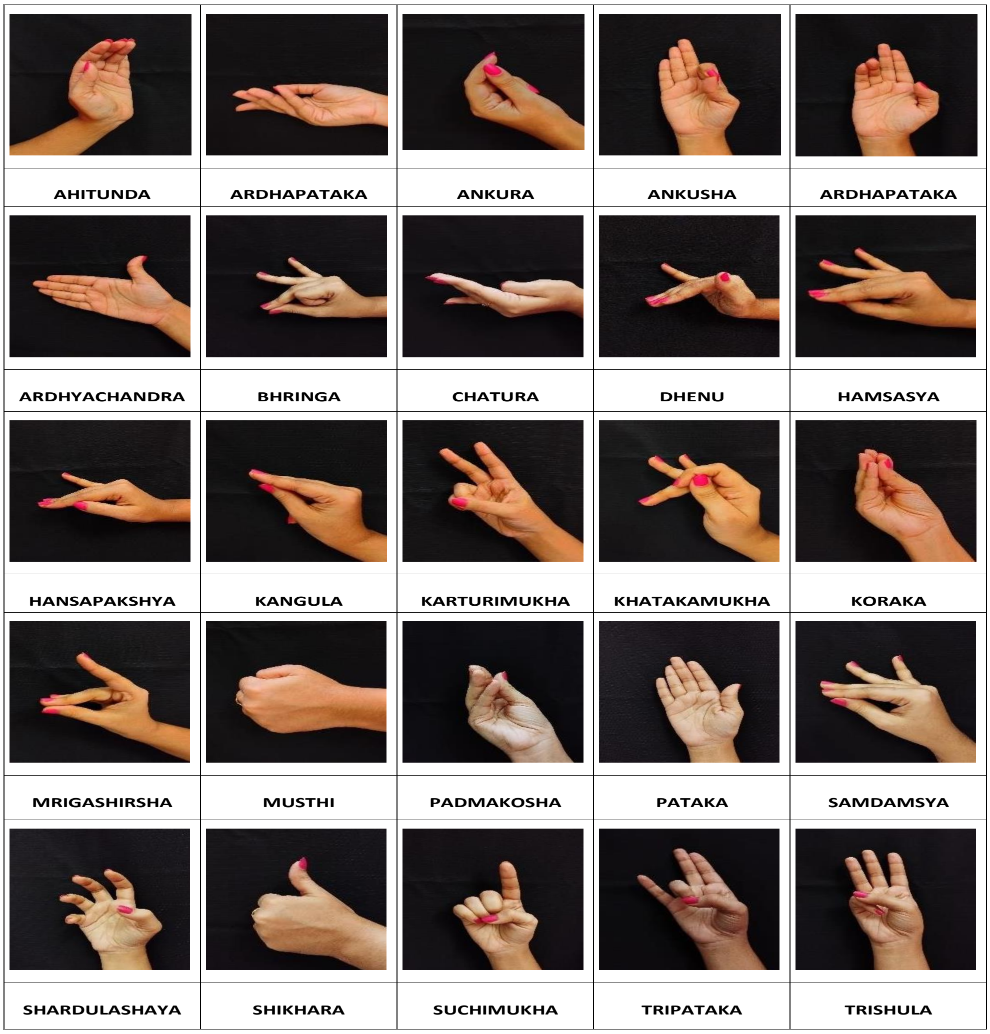

Fig. 1: 25 numbers of single-hand gestures of Manipuri classical dance.

- *Dataset for Manipuri classical dance:* To the best of the knowledge of the authors, there is no dataset available for Manipuri classical dance form in the recent literature. It is not possible to apply machine learning algorithms without a proper dataset. Some of the challenges for the creation of a dataset are as follows:

1. To acquire high-quality data images from trained Manipuri classical dance experts is a challenging task.
2. Environmental conditions like lighting, which is not possible every time to maintain during data collection. Background noise and camera placement also have an impact on hand gesture classification. Also, if there is a background noise available in the gesture image it might affect the hand gesture classification.

- *Feature extraction:* Since the dance poses of different experts of Manipuri classical dancer varies, so to extract the relevant features from the single-hand gestures and using these features to recognize the single-hand gestures of Manipuri classical dance is a challenging task.

Based on the issues and challenges discussed above, the objectives of this paper are finalized as: i) to develop a dataset for single-hand gestures of Manipuri classical dance, and, ii) to propose an approach using a Machine Learning classifier to recognize the 25 single-hand gestures of Manipuri classical dance.

The main contributions of this paper are highlighted as follows:

- Since no dataset is available for single-hand gestures of Manipuri classical dance form in the recent literature. The paper contributes single-hand gestures to the Manipuri Classical dance dataset, which is comprised of 750 images.
- The paper proposes a supervised machine learning approach using a Support Vector Machine (SVM) classifier within sci-kit-learn to classify 25 single-hand gestures of Manipuri classical dance.

The rest of the paper is organized as follows. Section 2, provides a literature survey on hand gesture recognition with different methods of feature extraction and classification. Section 3, discusses the proposed approach with a flowchart and methodology. Section 4, represents the experimental results. Section 5, concludes the paper with highlights of future directions.

2 Literature Survey

This section describes previous state-of-the-art related to research work of this paper as follows.

In [4], this research paper focuses on the classification of single-hand gestures for Sattriya dance of Assam, India. It proposes a two-level classification method to enhance hand gesture recognition. In the first level, a SVM is employed to categorize an unknown hand image into one of three groups. Subsequently, a Decision Tree classifier is utilized in the second level to identify the specific "hasta" within the group. The performance evaluation indicates that SVM achieves superior results compablack to other classifiers, achieving an accuracy of 97.24% when utilizing an RBF kernel and conducting 10-fold cross-validation on extracted feature sets. The overall classification accuracy of the proposed method is noted as 76.45%.

In [5], proposes a novel hand gesture recognition system on the basis static hand gesture database created by Sebastian Marcel. Skin colour detection was used for hand gesture segmentation, and morphological operations were used to eliminate the other detection areas of skin color. This is accomplished through the connectivity property of the skin colour pixels. Following that, essential key components are extracted using the MCRSRF proposed algorithm and classified using a multi-class SVM with a 96.5 % accuracy. One disadvantage of this paper is that some misclassifications occurblack during the testing phase.

In [6], the classification of Kathakali hand gestures using SVM and CNN is the main topic of this paper. The research suggests a feature extraction technique to identify the unique traits of Kathakali hand movements. For classification tasks, SVM and CNN models are used, and their performance is assessed. The findings demonstrate that Kathakali hand motions can be accurately classified using both SVM and CNN. The absence of discussion of the proposed classification systems' limits or potential difficulties in practical implementations is a drawback of this work.

In [7], their work is to capture 3D hand and finger motions with a sensor-based motion tracking system. They propose a novel angular velocity method that is directly applied to the sensor-based system's real-time 3D motion stream. The experimental results show a high level of precision. The issue is that the meaning of some gestures may differ depending on the country.

In [8], this paper proposed a reliable and efficient method for recognising hand gestures in real time. The method involves triggering the hand detection and tracking process with a specific gesture, followed by segmentation of the hand using motion and colour. The approach has been shown to achieve excellent performance and meet the requiblack standards.

In [9], a dynamic hand gesture detection approach based on neural networks with short-term sampling is proposed. To accurately recognise dynamic motions, the method includes taking and analyzing short-term samples of hand gesture images. The suggested approach achieves excellent accuracy by taking both the spatial and temporal details of gestures. The approach's usefulness and efficiency in real-time hand gesture recognition applications are shown through experimental findings.

In [10] explores the realm of Human-Computer Interaction (HCI) with a focus on hand gesture recognition in challenging environments. It highlights the significance of gesture recognition in enhancing HCI experiences. The paper discusses the difficulties faced in implementing gesture recognition systems, covering issues like variability in gestures, background noise, lighting conditions, and real-time processing. Various computer vision techniques and machine learning algorithms are exploblack to address these challenges. Despite its potential, the paper also points out drawbacks, including hardware requirements, user learning curve, environmental constraints, privacy concerns, gestural fatigue, and recognition latency. The authors propose future research directions to overcome these limitations and improve gesture recognition systems. The drawbacks of hand gesture recognition include specialised hardware requirements, a learning curve for users, environmental constraints, privacy concerns, gestural fatigue, and recognition latency. Improvements are needed for wider adoption and better user experience.

Paper [11], reviews machine learning methods for Indian dance forms classification. It discusses recent advances and challenges in the field and classifies the literature into dance classification and tasks classified within dances. A wide range of models is reviewed. It also discussed the available datasets, preprocessing, and feature extraction techniques. The paper also presents a MoveNet-based approach for the classification of Indian classical dance and gives the challenges and possible solutions in this domain.

In [12], the authors presents a CNN based Kathakali characters recognition. Also, a dataset on Kathakali images representing six different characters are presented. A web platform has been developed where one can upload Kathakali images for character identification, embedding AI algorithms. This platform is for the accessibility and visibility enhancement of Kathakali to show how the advancements in technology will promote traditional art forms.

Table 1 summarizes the methods and results found in the previous state-of-the-art related to the research work of this paper.

Table 1: Summary of Methods and Results

Ref.	Year	Method	Purpose	Pros	Cons
[4]	2016	Two-level classification	To recognize single-hand gestures for Sattriya classical dance of Assam, India	Overall classification accuracy is 76.45%	Misclassification of gestures
[5]	2020	Skin colour detection, Morphological operation, MCRSRF feature extraction, SVM Classifier.	To solve the problem of human-computer interaction.	Fast recognition time, high accuracy (96.5%), addresses the complex background problem.	Some misclassification in testing phase
[6]	2020	SVM model, CNN model.	Create a dataset of hand gestures of Kathakali dance, classification of hand gestures of Kathakali dance.	Machine learning, Deep learning both was used to compare the result.	In the case of SVM, the accuracy is low (40%)
[7]	2020	Angular velocity method.	To develop and evaluate a real-time hand gesture recognition system for interactive applications.	High accuracy (97.3%), The speed of recognition is high and does not require pre-processing, annotation and training.	Meaning of some gestures are different in other countries.

6

[8]	2020	Short-term sampling neural networks	To develop a dynamic hand gesture recognition system	Effective for recognizing gestures with temporal variations.	Requires training neural networks, which can be resource-intensive.
[9]	2021	Involves data collection and processing hand gestures to train machine learning models	To develop hand gesture recognition for improved gesture detection and classification.	Applicable to various interactive systems and devices	Requires substantial data and computational resources for training
[10]	2023	Explores possible solution for hand gestures recognition in difficult conditions such as poor lighting and complex backgrounds	To develop hand gesture recognition for challenging environments.	Robust and reliable	Complexity in implementation
[11]	2024	Explores machine learning methods for classifying dance	Presents a MoveNet-based approach for the classification of Indian Classical Dance	Geometric models are observed	Lack of standardized datasets, computational issues
[12]	2024	Explores CNN based method for identifying Kathakali characters	Presents a dataset on Kathakali images and a web platform has been developed	cloud based web environment	slow processing

The following sections briefly explain the different methods of feature extraction and existing possible machine-learning algorithms that can be applied to the recognition of single-hand gestures in Manipuri classical dance available in the literature [4–10] discussed above.

2.1 Methods of Feature Extraction

Feature extraction extracts pertinent information from the input data (such as images or video sequences) to accurately represent the hand gestures. There are various feature extraction techniques related to hand gesture recognition available in the literature [4–10]. Some of them are discussed as follows.

Geometric features: Geometric features are used to calculate the area using the contour of the hand region, perimeter using the contour perimeter of the hand region and solidity using the ratio of contour area and the convex hull area of the hand region.

Spatial feature: Static qualities of hand gestures are captublack by spatial features. The positions, orientations, and shapes of the fingers and palm, as well as the spatial distribution of the hand, are all described by these characteristics. The location of fingertips, hand contours, hand shape descriptors (like convex hull or bounding box), and segmentation of the hand region are examples of spatial features.

Edge Features: This Edge feature is used to detect the edges of the hand images using the Canny edge detector algorithm.

Temporal features: The temporal properties of hand gestures throughout time are represented as their dynamic components. When producing a motion, the hand's movement and shape are determined by these features. Examples of temporal aspects include the trajectory of hand movements, speeds, accelerations, and changes in hand orientation. They make the smoothness, speed, and temporal dynamics of the gesture visible.

Hierarchical Features: Hierarchical features capture the structural relationships between different hand parts and their interactions during gesture execution. These features can represent the hierarchical organization of the hand, such as the relationships between fingers, palm, and wrist. Hierarchical features can help capture the spatial configuration and relative positions of hand parts, providing contextual information for gesture recognition.

Joint and Pose Features: Joint and pose features describe the joint angles and orientations of the hand. These features provide information about the hand's articulation and pose, representing the hand's configuration in 3D space. They can be obtained using techniques like hand pose estimation or skeletal tracking, which track the positions and orientations of key hand joints, such as knuckles or wrists.

Motion Features: Motion features capture the movement patterns exhibited during gesture execution. These features describe how the hand gestures evolve, including the direction, speed, and smoothness of hand motion. Optical flow, which represents the apparent motion of pixels between consecutive frames, or tracking specific hand landmarks over time can be used to derive motion features.

2.2 Machine Learning Algorithms

As per the literature survey [4–10], various machine-learning algorithms can recognize single-hand gestures of Manipuri classical dance, some of them are briefly discussed as follows:

Convolutional Neural Network (CNN): CNNs are made up of several layers, including convolutional, pooling, and fully connected ones. Convolutional layers use convolutional filters to extract local patterns and characteristics from the input image. The convolved feature maps are downsampled by pooling layers, lowering spatial dimensions while keeping critical information. Fully linked layers connect the retrieved characteristics to the classification layer. One of the most significant features of CNNs is their capacity to detect spatial hierarchies and local correlations within images.

Support Vector Machine (SVM): The SVM is a classical supervised machine learning algorithm that is used for image classification. SVM is a classical machine learning algorithm that can be used for the classification of images. In the context of image classification, SVMs are often used with handcrafted features rather than raw pixel data. To apply SVMs to image classification, a process called feature extraction is typically performed. Features that are extracted are used as input to train SVM classifiers. The SVM learns to classify the images by finding an optimal decision boundary that maximally separates the different classes based on the feature representations [13].

Random Forest (RF): RF is the strongest algorithm for tree learning techniques in ML. It works by making a certain number of Decision Trees in the training phase. Using the random subset of the data set, each tree is constructed to calculate the random feature subset in each division. This randomness and variation in every tree shows the danger of data overfitting and enhances the overall performance of prediction. In practice, the algorithm sums up the output of all trees, either by voting (in the task of classification) or by calibrating. (in the task of regression). This consensus decision-making approach, which is facilitated by many trees with their experiences, yields a consistent example and accurate results. Above all, Random forests are applied widely for classification and regression purposes, which are renowned for having the capability of processing complex data, overfitting and producing sound predictions in various environments.

K-Nearest Neighbour (KNN): KNN is the very simplest algorithm for image classification. It classifies an object with respect to the majority class of the k-nearest neighbor. KNN is a supervised learning technique of ML. It is very easily available in real-world situations as it is non-parametric, i.e., it makes no assumptions about the underlying data distribution.

Logistic Regression (LR): LR is a supervised algorithm for machine learning that is employed in the classification task where the principal task is to predict the probability that an instance belongs to some specified class Or not. Logistic algorithm is a numerical algorithm that looks at the association between two data variables. The articles cover the basics of logistic regression, its categories and implementation. For instance, imagine two classes class 0 and class 1 if the value of the worth of the logistic function for input is more than 0.5 (threshold value) then it is classified as class 1 otherwise as class 0. It is called reversion because it's an extension of linear regression but it is principally applied to the problem of classification.

3 Proposed Approach

This paper proposes a supervised machine learning approach using a SVM [13] classifier available in sci-kit-learn to recognize 25 single-hand gestures of Manipuri classical dance. The various phases involved in this approach include i) dataset creation, ii) feature extraction, and iii) gesture recognition as shown in Figure 2.

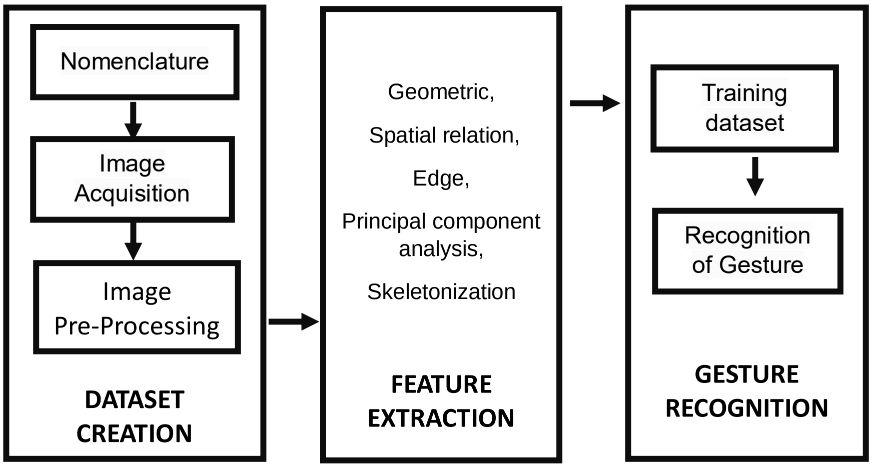

Fig. 2: Block diagram for single-hand gestures recognition of Manipuri classical dance.

The phases involved in the recognition of single-hand gestures of Manipuri classical dance have been discussed as follows:

3.1 Dataset Creation

The dataset creation step includes the following sub-phases, viz., i) image acquisition, ii) nomenclature, and, iii) image pre-processing which are explained as follows:

3.1.1 Image Acquisition

For any classification and recognition technique, image acquisition plays a vital role throughout the classification. This section discusses the details of how the single-hand gesture images have been captublack. The environment setup for image acquisition [4] is discussed as follows:

1. The hand gestures images are captublack from three volunteer dancers of Manipuri classical dance. Each dancer provided ten images for each class from the direction of the different angles. So, a total of thirty images for each class have been captublack to create the dataset.
2. All three volunteers are professional female dancers of Manipuri classical dance.
3. To capture the images a mobile camera of 50 pixels of blackme Note 11 specification was used.
4. The captublack images are the size of 9612 X 6912 pixels. However, the images has been resized during the time of computation.

3.1.2 Nomenclature

Nomenclature is a systematic way of naming things in technical and scientific fields. This paper introduces a simple rule for nomenclature of images for dataset creation which is as follow: GN-VN.jpg, where GN is for gesture name (eg: Alapalla, Ahitunda) and, VN is for variation number. The reason for selection of jpg format for image file type is because it can deal with complex images. Moreover, jpg format is made to effectively hold high-quality digital photographs with lots of color and detail. An example of nomenclature is ALAPALLAVA-1.jpg

3.1.3 Image Pre-processing

In the image preprocessing part, the images are resized to the resolution of (64*64) pixels to gain speed in the computational process. Resizing of the images of hand gestures is done using OpenCV's 'cv2.resize' function for better feature extraction and less computational complexity.

3.2 Feature Extraction

For the classification and recognition of the gesture, there is a need for features that need to be extracted from the hand gesture images which are used to train the gesture classification model. In this paper, multiple features such as Geometric features, Spatial relations between the features such as distance, angle between the fingers, different edges, and features from skeletonization images have been extracted.

Geometric features (x_{geo}): The following steps have been followed to find the geometric features, viz., i) Find the contours from hand images, ii) Select the largest contour based on area, iii) Calculate the areas, perimeter and solidity of the largest contour form the hand image.

Spatial relation ($x_{spatial}$): This part extracts spatial relations like distance and angles. To extract spatial features, the following steps have been followed, , viz., i) Generates all pairs of landmarks to calculate distances, ii) Generates all triplets of landmarks to calculate angles, iii) Computes the Euclidean distance between pairs of points.

Edge detection (x_{edge}): The discriminant edges of the hand images were extracted using the Canny edge detection algorithm. To apply this canny edge detection algorithm on the hand images necessary modification of threshold value has been done with experiment.

Principal Component Analysis (x_{pca}): The following steps have been followed to extract these features, viz., i) Computes the mean of the image pixel, ii) Performs Singular Value Decomposition (SVD) on the mean-centeblack image, iii) Takes the first two principal components and flattens them into a single array.

Skeletonization ($x_{skeleton}$): To extract the skeletonization of the images, the RGB images are first converted to binary images and then the skeletonization is extracted. After that different features such as the central point, and medial axis were extracted.

After extracting all the features from the hand gesture images, the feature vector "X" has been computed by concatenating all the extracted features using the formula: $X = [x_{geo}, x_{spatial}, x_{edge}, x_{pca}, x_{skeleton}]$.

3.3 Gestures Recognition

The gesture recognition phase is completed into two sub-phases, viz., i) training dataset and ii) recognition of gesture. In the first sub-phase, training the dataset is done in the SVM model using the Sci-Kit Learning approach, and in the next sub-phase recognition of gesture is done with an unknown dataset i.e., testing with real-life hand gestures.

3.3.1 Training dataset

By concatenating the gatheblack features from the single-hand gesture images, train the machine using the extracted features in the SVM model using the Sci-Kit learning approach. Here, among the three kernels of the SVM, the linear kernel has been trained on the training data using "kernel = linear". This means that the linear kernel of the SVM classifier function of sci-kit learning will be used throughout this experiment. Which means SVM will be able to find a linear hyperplane that best separates the gesture classes in future space. The equation of the hyperplane in a SVM is given by [13].

$$w^T x + b = 0$$

where w is the weight vector, x is the feature vector, b is the bias term.
Finally, the training of the SVM model is done by using the formula:

$$\text{svm\_model.fit}(\mathbf{X}_{\text{train}}, \mathbf{Y}_{\text{train}})$$

The equation of trained model is as follows:

$$\text{svm\_model} = \text{SVC(kernel='linear', C=1.0)}$$

Here, C=1.0 parameter in the SVM model is a regularisation parameter that controls how much we should focus on margin maximization (width of the decision boundary) versus how much we are willing to tolerate individual point misclassification. A small value of "C" leads to a larger margin but potentially more classifications.

After training the SVM model using a linear kernel with a regularization parameter of C=1.0, the trained model is then saved for future use.

3.3.2 Recognition of Gesture

After training the SVM model using the Sci-kit Learn approach, this phase tests the recognition of unknown single-hand gestures from Manipuri classical dance. For real-time hand-gesture recognition, the saved model is loaded and applied it to the input data. At first, the dataset is split into the ratio of 80:20. That means 80% of the data is selected for training and 20% of the data is selected for testing. This data has been evaluated by calculating Precision, Recall and F1 Score values by generating the confusion matrix as shown in the next section. The overall precision value is calculated using the precision score of the 25 classes which is 0.8427. The overall recall value is also calculated by averaging the recall value of the 25 classes, which is 0.8104; the F1 score is calculated using the overall precision and recall value of 0.8075. The higher value of F1 Score denotes that the SVM is a good classifier for recognizing single-hand gestures of Manipuri classical dance. Thus, the overall accuracy achieved for recognition of single-hand gestures of Manipuri classical dance is 80.75%.

4 Experimental Results and Analysis

This section presents different equations and their values such as confusion matrix, precision, recall and F1 score to evaluate the performance of the SVM classifier. Also, a ROC curve is used to represent the performance of the classifier across the various threshold values. In addition, the real-time single-hand gesture of Manipuri classical dance output performance has been represented.

4.1 Confusion Matrix

A confusion matrix as shown in Table 2 is a tool for measuring the performance in machine learning, especially in classification jobs. A confusion matrix is used to appraise the performance of the classification algorithm by outlining the number of correct pblackictions and incorrect pblackictions by the trained model on a dataset. Here, TP = True Positive, TN = True Negative, FP = False Positive, FN = False Negative.

Table 2: Confusion Matrix

	Pblackicted positive	Pblackicted negative
Actual Positive	TP	FN
Actual Negative	FP	TN

The confusion matrix for testing phase as shown in Figure 3 consists of 25 rows and 25 columns, which means that the classifier is pblackicting one of the 25 possible gesture classes. Here, the rows of the confusion matrix shows the actual true class, and columns of the confusion matrix shows the true pblackicted by the classifier.

Based on the values of TP, TN, FP, FN of different classes of single-hand gestures of Manipuri classical dance the precision, recall, and F1-Score values are calculated using various formulas as discussed in the following subsections.

4.2 Precision

Precision is a type of metric that is used in machine learning works to evaluate the classification accuracy of a model. It is a measurement of correctly pblackicted positive outcomes (True positive (TP)) out of the number of total outcomes as positives which consists of True positives (TP) and False positives (FP). The formula for the precision to calculate its value is:

$$\textbf{Precision} = \frac{TP}{TP + FP}$$

In this work, we calculated Average precision which gives equal weight to each class, regardless of class imbalance. Average precision is useful when there is a need to calculate the performance of the classifier across all the classes without considering the

13

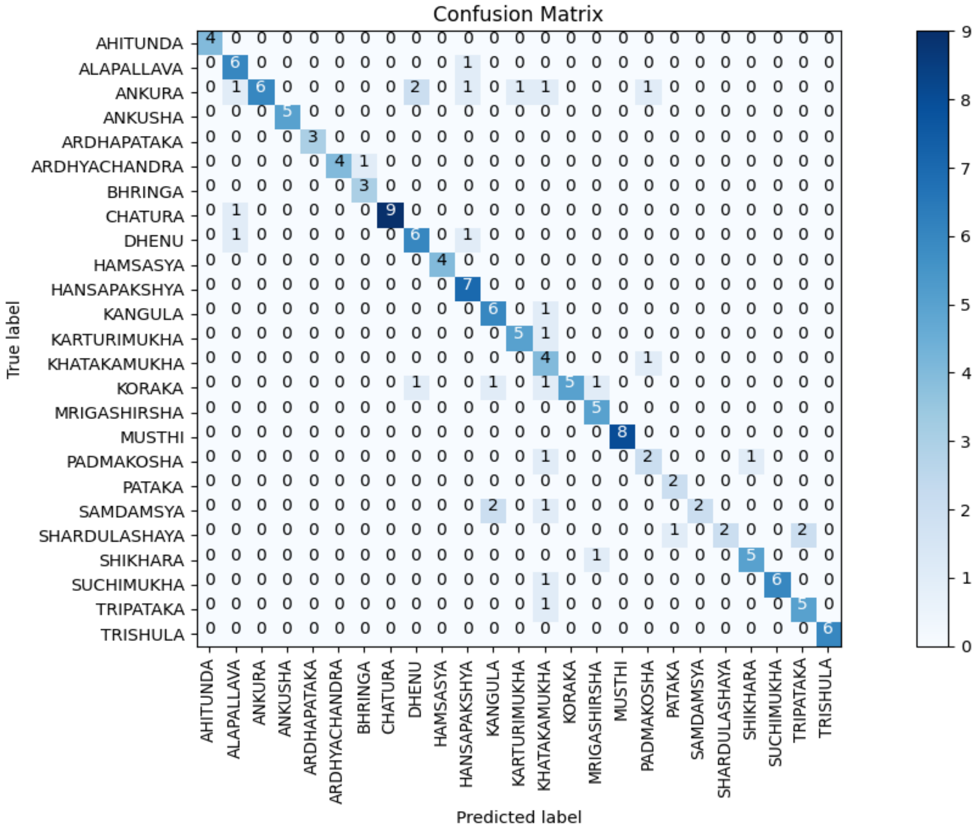

Fig. 3: Confusion matrix of testing phase

distribution of classes.

$$\textbf{Average Precision} = \frac{sum_{i=1}^{N} Precision_i}{N} = 0.842761905$$

Where, N is the number of classes.

4.3 Recall

Recall is a type of metric that measures the correct positive pblackictions out of all the positive pblackictions that could have been made. Given below is the formula by which one can calculate recall values:

$$\textbf{Recall} = \frac{TP}{TP + FN}$$

14

Table 3: Precision, Recall and F1-Score values

GESTURES	TP	FP	FN	TN	PRECISION	RECALL	F1-SCORE
AHITUNDA	4	0	0	146	1	1	1
ALAPALLAVA	6	3	1	140	0.666666667	0.857142857	0.75
ANKURA	6	0	7	137	1	0.461538462	0.631578948
ANKUSHA	5	0	0	145	1	1	1
ARDHAPATAKA	3	0	0	147	1	1	1
ARDHYACHANDRA	4	0	1	145	1	0.8	0.888888889
BHRINGA	3	1	0	146	0.75	1	0.587142857
CHATURA	9	0	1	140	1	0.9	0.947368421
DHENU	6	3	2	139	0.666666667	0.75	0.705882353
HAMSASYA	4	0	0	146	1	1	1
HANSAPAKSHYA	7	3	0	140	0.7	1	0.823529412
KANGULA	6	3	1	140	0.666666667	0.857142857	0.75
KARTURIMUKHA	5	1	1	143	0.857142857	0.857142857	0.857142857
KHATAKAMUKHA	4	8	1	137	0.333333333	0.8	0.461538462
KORAKA	5	0	4	141	1	0.555555556	0.714285714
MRIGASHIRSHA	5	2	0	143	0.714285714	1	0.833333333
MUSTHI	8	0	0	142	1	1	1
PADMAKOSHA	2	2	2	144	0.5	0.5	0.5
PATAKA	2	1	0	147	0.666666667	1	0.8
SAMDAMSYA	2	0	3	145	1	0.4	0.571428571
SHARDULASHAYA	2	0	3	145	1	0.4	0.571428571
SHIKHARA	5	1	1	143	0.833333333	0.833333333	0.833333333
SUCHIMUKHA	6	0	1	143	1	0.857142857	0.923076923
TRIPATAKA	5	2	1	142	0.714285714	0.833333333	0.769230769
TRISHULA	6	0	0	144	1	1	1
AVERAGE					0.842761905	0.810493284	0.807567577

In our work, we have calculated average recall that is used in multi-class classification to measure the overall performance of the classifier. The average recall is calculated by summing up the TP instance for all the classes and dividing it by the sum of TP and FN for all the classes combined.

$$\textbf{Average Recall} = \frac{sum_{i=1}^{N} TP_i}{sum_{i=1}^{N}(TP_i + FN_i)} = 0.810493284$$

Here, N is the number of classes TP_i is the true positive value for i classes and FN_i is the false negative for i classes.

4.4 F1 Score

The F1 score is a statistical measure that is used to evaluate the performance of a classification model. F1 score is defined as the harmonic mean of the Precision score and Recall score. The F1 score is useful in classification problems. The formula to calculate the f1 score is given below:

$$\textbf{F1 Score} = 2 * \frac{Precision * Recall}{Precision + Recall}$$

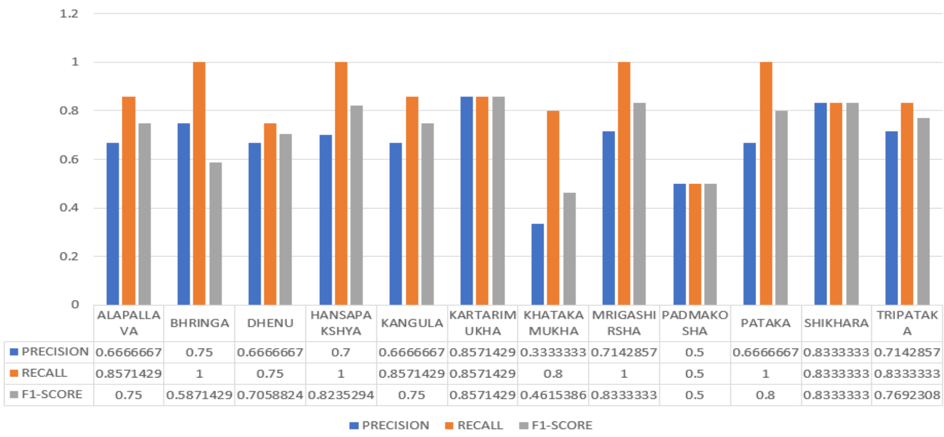

Fig. 4: Comparison graph of Precision, Recall and F1-score values for high precision hand gestures

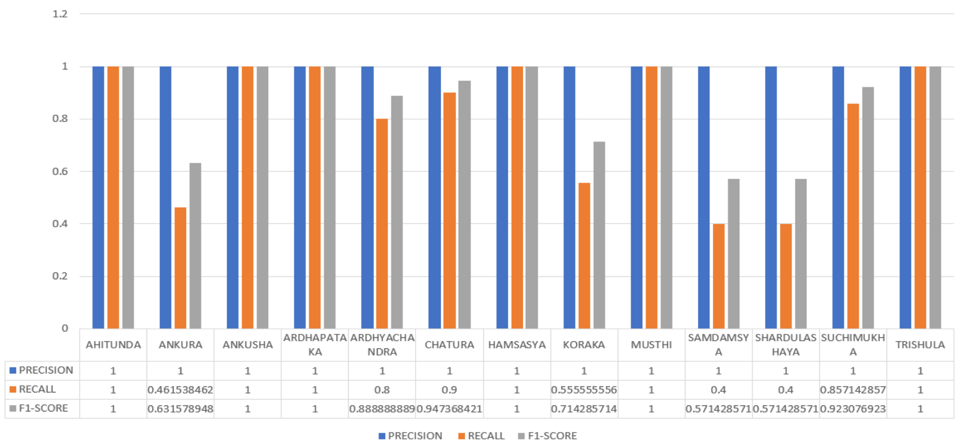

Fig. 5: Comparison graph of Precision, Recall and F1-score values for low precision hand gestures

In this paper, the average F1 score is calculated by using the formula:

$$\textbf{Average F1 score} = 2 * \frac{AveragePrecision * AverageRecall}{AveragePrecison + AverageRecall} = 0.807567577$$

A higher value of the F1 Score denotes a good quality of the classifier. Here, in this paper, we got an average F1 score of 0.807567577 (i.e. 80.76% of the classification rate) which means a good value of classification.

16

Table 3 shows various values obtained for precision, recall and F1-Score for 25 different classes of single-hand gestures of Manipuri classical.

Figure 4 and Figure 5 represents the comparison graph of precision, recall, and F1-score for the high precision single-hand gestures and low precision single-hand gestures respectively.

4.5 Classwise Accuracy Graph

The classwise accuracy graph as shown in Figure 6 describes the model's performance across the various classes of single-hand gestures of the Manipuri classical dance by displaying their accuracy. The X-axis represents the 25 classes of the single-hand gestures of the Manipuri classical dance and Y-axis represents the accuracy value which ranges from 0.0 to 1.0. Using this graph one can easily identify the strengths of the SVM classifier model.Here is a detailed analysis of the classwise accuracy graph:

- 100% Accuracy: The proposed approach achieves accuracy of 1 in recognizing single-hand gestures such as Ahtunda, Alapallava, Ardhyachandra, Chatura, Karturimukha, Khatakamukha, Suchimukha, and Trishula. This indicates that the SVM model is highly effective in distinguishing these specific gestures.
- Above 80% Accuracy: Gestures like Dhenu, Hamsasya, Hansapaksha, and Pataka also show high accuracy, ranging from 83% to 90%, suggesting that the model performs reliably but may have minor inconsistencies.
- 70% - 80% Accuracy: The Ankusha gesture has an accuracy of 75%, indicating moderate recognition performance. This suggests that while the model generally performs well, it might struggle in some cases.
- 50% - 60% Accuracy: Gestures such as Musthi and Mrigashirsha show moderate accuracy (56% and 50%, respectively). This suggests that the model has difficulty correctly identifying these gestures in a significant number of cases.
- Below 50% Accuracy: Gestures like Ankura, Koraka, Samdamsya, Shardulashaya have very low accuracy, ranging from 40% to 46%. This is a clear indication that the model struggles significantly with these gestures, potentially due to similarities with other gestures or insufficient training data for these classes.

The SVM model performs exceptionally well on some gestures but struggles with others, particularly those with lower accuracy.

4.6 Classwise Loss Graph

The graph as shown in Figure 7 describes the classwise loss for different single-hand gestures of the Manipuri classical dance recognition using the SVM classifier. The X-axis represents the 25 classes of the single-hand gestures of the Manipuri classical dance and Y-axis represents the loss value which ranges from 0.0 to 1.0. Loss is an indicator of the error between the pblackicted and actual labels for each gesture class. Here is a detailed analysis of the classwise loss graph:

- Samdamsya and Shardulashaya gestures exhibit the highest loss values at 0.60 which indicates that the SVM classifier has significant difficulty in accurately recognizing

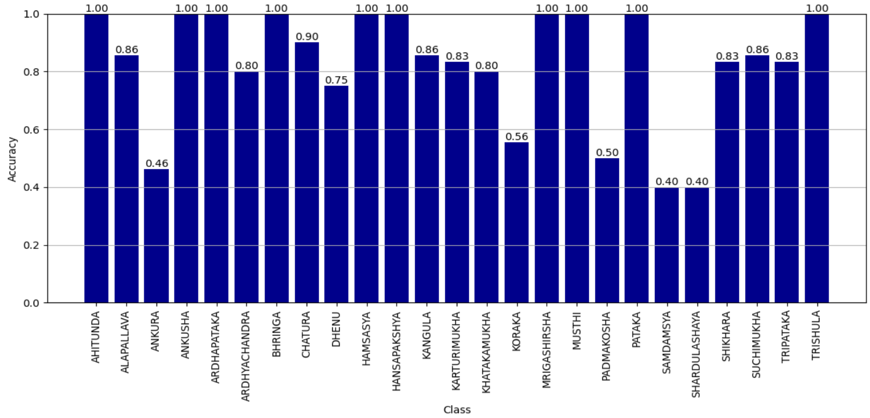

Fig. 6: Classwise accuracy graph

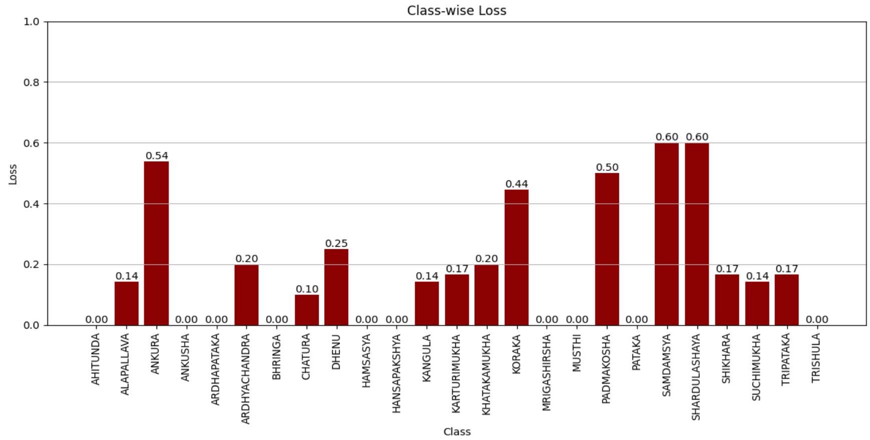

Fig. 7: Classwise loss graph

these gestures, possibly due to insufficient training data, high similarity with other gestures, or complex features that the SVM is struggling to model.

- Ankura and Padmakosha gestures also show relatively high loss values at 0.54 and 0.50, respectively. The model's pblackictions for these classes are less reliable, suggesting a need for targeted improvements.
- Koraka gesture has a loss of 0.44, indicating moderate performance. The model has some difficulty in pblackicting this class accurately, which might require further attention in training.

18

- Dhenu gesture has a loss of 0.25 which depicts that the model shows moderate accuracy in recognizing this gesture. While not as problematic as the high-loss classes, there is room for improvement.
- Karturimukha and Mrigashirsha gestures have loss values of 0.20, which suggests that while the model is generally performing well, there are still some misclassifications that need to be addressed.
- Alapallava, Suchimukha, Shikhara gestures have relatively low loss values ranging from 0.14 to 0.17, indicating that the model is performing well for these classes with only minor errors.
- Ahtunda, Ankusha, Ardhyachandra, Bhringa, Chatura, Hansapaksha, Kangula, Khatakamukha, Musthi, Pataka, Tripataka, Trishula gestures have a loss value of 0.00, showing perfect or near-perfect pblackictions by the model. The SVM effectively recognizes these gestures with no significant errors.

The high-loss classes (Samdamsya, Shardulashaya, Ankura, and Padmakosha) indicate areas where the model is underperforming and requires more data, feature refinement, or hyperparameter tuning. The moderate-loss classes (Koraka, Dhenu, Karturimukha and Mrigashirsha) suggest that while the model is performing decently, there is still potential for improvement to achieve more reliable recognition. The low-loss classes (Alapallava, Suchimukha, Shikhara, Ahtunda, Ankusha, Ardhyachandra, Bhringa, Chatura, Hansapaksha, Kangula, Khatakamukha, Musthi, Pataka, Tripataka, Trishula) reflect the model's strong performance in accurately identifying these gestures, demonstrating that the current SVM configuration is effective for these specific classes.

The following recommendations are suggested for high accuracy of single-hand gestures of Manipuri classical dance in the future viz., i) to increase the dataset size for poorly recognized gestures, ii) to refine or add features for better distinguish the challenging gestures, iii) to do experiment with different kernels or hyperparameters in the SVM to improve overall performance, especially for the less accurately recognized gestures, and iv) to implement cross-validation to ensure the model's generalization across different subsets of the data.

4.7 Receiver Operating Characteristic (ROC) Curve

The ROC curve is a visual representation of the performance of binary classifiers across various threshold values. The ROC represents the true positive (TP) rates against the false positive (FP) rates at different threshold values. The Area Under the Curve (AUC) represents the overall performance of the classifier within all possible threshold values. In total, the ROC curve is a very useful tool for calculating the performance of binary classifiers, allowing the user to visualize the trade-off between true positive rate and false positive rate, and to determine the threshold value that provides the best balance between the costs of FP and FN.

As shown in the Figure 8, the ROC curve is shown for the multi-class SVM classifier where the plotting shows the ROC curve for each 25 classes of the dataset. From the plotting, it can be easily seen that most of the classes have higher AUC, which clearly means that the SVM classifier performs very well in distinguishing between the classes.

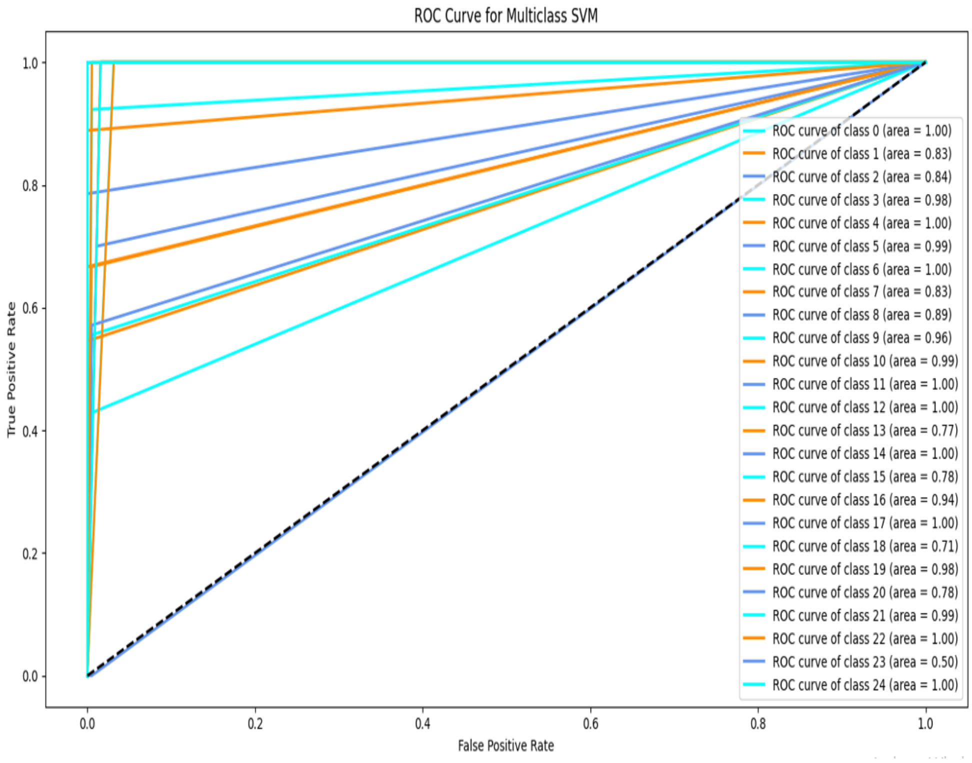

Fig. 8: ROC Curve of the SVM Classifier

However, class 13, class 15, class 18, class 20, class 23 have lower AUC, which implies it is more difficult for the classifier to discriminate between these and other classes.

4.8 Real-time Gesture Recognition

This section presents the real-time recognized single-hand gesture of Manipuri classical dance using SVM classifier. A set of unknown single-hand gestures of Manipuri classical dance is given for recognition to the trained SVM classifier model. Figure 9 depicts the correctly recognized single-hand gestures as ”PATAKA”, ”HANSAPAKSHYA”, ”DHENU” and ”AHITUNDA”.

5 Conclusion and Future Research Direction

This paper discusses how dance gesture recognition can play an important role in the cultural heritage preservation of Manipuri classical dance. Moreover, the paper contributes a dataset of 750 images for the 25 classes of single-hand gestures of Manipuri classical dance. This paper also presents a supervised machine-learning approach using SVM classifier to recognize the single-hand gestures of Manipuri classical dance. The accuracy of the proposed approach is tested with real-time single-hand gestures which

Fig. 9: Recognized single-hand gesture of Manipuri classical dance.

obtain 80.69%. Also, in this paper, the accuracy of the individual class is shown by the ROC curve.

In the future, the dataset will be extended with more images to be collected from various volunteers and will processed to make it publically available for the research community. The dataset will be tested for recognition of single-hand gestures of Manipuri classical dance using other available machine learning algorithms such as 3DCNN, RF, KNN, and LR. Moreover, double-hand gestures of Manipuri classical dance are also left as future research works.

6 Declarations

Conflicts of Interests The authors declare that they have no conflict of interest.
Data Availability Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.
Funding The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.
Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.
Author Contributions All authors contributed to the study conception and design.
Authors Consent The manuscript is submitted with the consent of all authors.

References

[1] J. Qi, L. Ma, Z. Cui and Y. Yu, "Computer vision-based hand gesture recognition for human-robot interaction: a review", Complex and Intelligent Systems, vol. 10, no. 1, pp. 1581–1606, 2024

[2] D. Noh, H. Yoon and D. Lee, "A decade of progress in human motion recognition: A comprehensive survey from 2010 to 2020", IEEE Access, vol. 12, pp.5684-5707, 2024

[3] D. Biswas, "Akademi: Hasta in manipuri-part 2", Available at https://www.akademi.co.uk/hasta-in-manipuri-part2 (2023/03/15)

[4] M. Devi and S. Saharia, "A two-level classification scheme for single-hand gestures of sattriya dance", in 2016 International Conference on Accessibility to Digital World (ICADW) IEEE, pp. 193–196, 2016

[5] M. Al-Hammadi, G. Muhammad, W. Abdul, M. Alsulaiman, M.A. Bencherif and M.A. Mekhtiche, "Hand gesture recognition for sign language using 3DCNN", IEEE access, vol. 8, pp. 79491–79509, 2020

[6] L.T. Bhavanam and G.N. Iyer, "On the classification of kathakali hand gestures using support vector machines and convolutional neural networks", in 2020 International Conference on Artificial Intelligence and Signal Processing (AISP) IEEE, pp. 1–6, 2020

[7] V.A. Shanthakumar, C. Peng, J. Hansberger, L. Cao, S. Meacham and V. Blakely, "Design and evaluation of a hand gesture recognition approach for real-time interactions", Multimedia Tools and Applications, vol. 79, pp. 17707–17730, 2020

[8] W. Zhang, J. Wang and F. Lan, "Dynamic hand gesture recognition based on short-term sampling neural networks", IEEE/CAA Journal of Automatica Sinica, vol. 8, no. 1, pp. 110–120, 2020

[9] P. Parvathy, K. Subramaniam, G. V. Prasanna, P. Karthikaikumar, J. Varghese and T. Jayasankar", "Development of hand gesture recognition system using machine learning", Journal of Ambient Intelligence and Humanized Computing, vol. 12, pp. 6793–6800, 2021

[10] V. Chang, R.O. Eniola, L. Golightly and Q.A. Xu, "An exploration into human–computer interaction: Hand gesture recognition management in a challenging environment", SN Computer Science, vol. 4, no. 5, pp. 441, 2023

[11] S. Gupta and S. Singh, "Indian dance classification using machine learning techniques: A survey", Entertainment Computing, pp. 100639, 2024

[12] G.N. Iyer, S. Bhardwaj, "Kathakali character identification–using deep learning techniques and web technologies for indian cultural heritage", Digital Applications in Archaeology and Cultural Heritage, vol. 32, pp. 00300, 2024

[13] M.A. Hearst, S.T. Dumais, E. Osuna, J. Platt and B. Scholkopf, "Support vector machines", IEEE Intelligent Systems and their applications, vol. 13, no. 4, pp. 18–28, 1998