On the Classification of Kathakali Hand Gestures Using Support Vector Machines and Convolutional Neural Networks

Lakshmi Tulasi Bhavanam
Department of Computer Science and Engineering
Amrita School of Engineering, Coimbatore
Amrita Vishwa Vidyapeetham, India
tulasi6543lakshmi@gmail.com

Ganesh Neelakanta Iyer
Department of Computer Science and Engineering
Amrita School of Engineering, Coimbatore
Amrita Vishwa Vidyapeetham, India
ni_ganesh@cb.amrita.edu

***Abstract*—Indian classical dance such as Kathakali is composed of complex hand gestures, body moments, facial expressions and background music. Due to the complexities involved in its hand-gesture language, it is often difficult to understand kathakali mudras. In this paper, we generated a dataset for Kathakali hand gestures and explore ways to recognize Kathakali dance mudras performed by artistes with the help of machine learning and deep learning techniques. There are 24 classes of hand gestures that are used to convey the story by the performer in Kathakali. We proposed a Support Vector Machine (SVM) model and Convolutional Neural Network (CNN) model which classify the images into 24 different classes. We compared the performance of machine learning algorithms and deep learning algorithms. Our results show that deep learning algorithms gave up to 74% accuracy. To the best of our knowledge, this is the first attempt to generate a dataset of Kathakali hand gestures, explore data pre-processing techniques for machine learning techniques and applying deep learning techniques for classification of Kathakali hand gestures.**

***Index Terms*—Kathakali, mudra, Support Vector Machine, Deep Learning, Convolution Neural Network, Image Processing.**

Fig. 1. Single hand mudras

I. Introduction

Kathakali is an Indian classical dance drama from the south Indian state of Kerala originated in 17th century. The story of 'Kathakali' dance performance is communicated to audience through hand gestures, facial expressions and dances along with music support. Kathakali is traditionally performed by male dancers in courts and theatres of Hindu regions. Kathakali hand gestures are considered as a complete language by itself with necessary grammatical elements and language structures associated with it. With the help of 24 hand gestures available, one can communicate any message to another completely using hands.

Generally it is very difficult for a common man to understand the meaning of Kathakali dance drama because of its complicated hand gesture language structure and dance movements. Unless you know all the hand gestures and the words and sentences you can make using these hand gestures, it is difficult for one to appreciate the meaning conveyed by the artiste through the gesture language.

Kathakali hand gestures are based on an ancient text Hastalakshana Deepika [1]. There are a total of 24 hand gestures (also called as Mudras) as specified in this ancient text. Using single hand as well as using both hands, mudras are formed. Combination of these hand gestures convey certain meaning to them. Mudras are illustrated in Figure 1 and the mudra names are summarized in Table I. Due to different combination of mudras in certain ways represent different meaning, some even depending upon the context, unless one is well versed with these mudras, their combinations and meanings, it is difficult for one to understand and appreciate this art.

We have attempted to solve this problem and this is the first step towards the same. In this work, first we build a dataset of Kathakali hand gestures (which is not presently available at present to the best of our knowledge). As a second step, we examine and devise strategies for data preprocessing to be applied to the generated dataset and finally, we study machine learning and deep learning techniques for classification of hand

978-1-7281-4458-0/20/$31.00 ©2020 IEEE

TABLE I
LIST OF KATHAKALI MUDRAS

No	*Mudra name*	*No*	*Mudra name*
1	Pathaaka	13	Mukuram
2	Mudraakhyam	14	Bhramaram
3	Katakam	15	Soochimukham
4	Mushti	16	Pallavam
5	Kartharee Mukham	17	Thripathaaka
6	Sukathundam	18	Mrigaseersham
7	Kapidhakam	19	Sarpasirassu
8	HamsaPaksham	20	Vardhamanakam
9	Sikharam	21	Araalam
10	Hamsaasyam	22	Oornanabham
11	Anjaly	23	Mukulam
12	Ardhachandram	24	Katakaamukham

gestures.

Reminder of the paper is organized as follows. Section II briefly explains literature review, section III describes the proposed classification techniques briefly, section IV describes the results in detail including the effect of preprocessing techniques and classification results and finally section V concludes the paper.

II. LITERATURE SURVEY

As per our knowledge this is the first attempt that has been focussed on Kathakali dance mudras. All the previous works have mostly focussed on other dance forms such as kuchipudi, baratanatyam, kathak, odissi, sattriya, Manipuri, Aceh (Indonesian) and Korean pop dance (k-pop). In this section, we focus primarily on giving current advancement in human gesture recognition and how it is used in classifying dance hand gestures in above mentioned dance forms. We have also summarized our literature review in Table II.

Basavaraj S. Anami et al. [2] have proposed a 3 stage method which involves pre-processing of mudras by obtaining contours of images, extracting features by using Eigen values, Hu-moments and intersections and finally classifying the mudras using Artificial Neural Network. The Baratanatyam dataset is created with 2800 images (100 images per mudra). Mudras are divided into conflicting and non-conflicting mudras. The reported accuracy is 97.1%, 99.5% and 96.03% for whole hand mudras, conflicting and non-conflicting hand mudras respectively.

P.V.V.Kishore et al. [3] have proposed architecture for Deep Neural Network for classification of Indian classical dance actions. The dataset of videos are collected from both online (Youtube, live performances) and offline (recordings) and. The overall recognition rate of 93.3% is obtained for the CNN.

K.V.V. Kumar et al. [4] have presented an approach for classifying kuchipudi dance mudras into text messages (meaning of mudra) by using Histogram of oriented Gradients(HOG) features as an feature extraction algorithm and Support vector machine (SVM) is used as the classifier. The Graphical User Interface (GUI) is developed for calculating Mudra Recognition Frequency (MRF). The SVM classifier has acquired an MRF of 90%.

P.V.V. Kishore et al. [5] have proposed a Super pixel based Linear Iterative Clustering (SLIC) and Marker Controlled Watershed algorithms to perform segmentation out of which SLIC performs better than Watershed algorithms. The result shows that a SLIC algorithm performs better than Watershed algorithm. The experiments are performed on dataset collected with 120 kuchipudi images (both training and testing images). An average of 52.55% of the total images is segmented correctly by watershed algorithm whereas 74.25% of images are segmented well by SLIC algorithm.

Soumitra Samanta et al. [6] have created their own Indian Classical Dance dataset from the YouTube videos. Each class contains of 30 videos with different resolutions (maximum resolution. 400x350). The proposed sparse representation based Dictionary learning technique involves representing every frame of the video by a movement descriptor based on HOOF (Histogram of Oriented Optical Flow). An average accuracy of 86.67% is achieved for classifying the videos frames using SVM.

D.Anil Kumar et al. [7] have proposed a new segmentation model using Local Binary Patterrn (LBP) and wavelet transform for segmentation. The Adaboost multi class classifier is used to identify the Indian Classical Dance Action from the five different features such as Zernike moments, Hu moments, shape signature, Haar features and LBP features. The overall accuracy of 86.67% is obtained on the dataset created from online videos of Bharatanatyam and Kuchipudi and offline videos captured from lab.

Mampi Devi et al. [8] have classified the asamyukta hastas of Sattriya dance form by using a two level classification method. Based on the structural similarity the images from the collected dataset of 1015 images are classified into twenty nine classes and extracts the Medical Axis transformation (MAT) to identify the groups. The accuracy acquired by using SVM with PBF kernel is 97.24%.

Ankita Bisht et al. [9] have proposed a framework for classifying the Indian Classical Dance forms from 626 videos collected from both offline and online. The videos are of different dance forms such as Sattriya, Bharatnatyam, Kuchipudi, Kathak, Odissi, Manipuri, Mohiniyattam. The framework had achieved the accuracy of 75.83% when feeded 211 videos to a Deep Convolutional Neural Network (DCNN).

Nurfitri Anbarsanti et al. [10] have recorded several gesture instances of Aceh Traditional Dance by the XBOX Kinetic sensor. The complete recognising system was proposed by using the Simulink programming package by MatLab. The system classifies the input testing gestures into one of six different classes of predefined gestures or a single class of undefined gesture. For a single gesture the classifier system has achieved an accuracy of 94.87%.

N. Y. Tongpaeng et al. [11] have implemented a tool which compiles the mistakes and also analyses the instructions and

TABLE II
COMPREHENSIVE LITERATURE REVIEW

Ref	***Dance form***	***Objective***	***Dataset***	***Learning***	***Image processing technique***
[2]	Bharatanatyam	Classication of single hand mudras	2800 images	ANN	Contour extraction: Canny Edge Detector Feature extraction: Hu-moments, eigenvalues and intersections
[3]	Indian classical Dance form	Classification of Indian classical dance actions Offline and online data	2800 images	CNN	not required
[4]	Kuchipudi	classification mudras as text messages	Hand mudras collected from internet	SVM	Feature extraction: HOG, SURF, SIFT, LBP and HAAR
[5]	Kuchipudi	Classification of various Indian dance forms	Images collected from internet	SLIC	Segmentation: Watershed algorithm
[6]	Bharatanatyam, Kathak and Odissi	Classification of various Indian Classical Dacne	30 videos	SVM	pose descriptor: histogram of oriented optical flow (HOOF)
[7]	Bharatanatyam and Kuchipudi	Recognizing complex human movements	offline and online videos	Adaboost multi class classifier	Segmentation: discrete wavelet transform and local binary pattern (LBP) features
[8]	sattriya	Classification of hand mudras	1015 images recorded offline	SVM and Decision tree	Background removal using GMM Smoothening the images using Gaussian filter
[9]	Bharatanatyam, Kathak, Kuchipudi, Manipuri, Mohiniyattam, Odissi, Sattriya	Classifying Indian classical dance forms from videos	626 videos	SVM	SVM Motion capturing: optical flow algorithm
[10]	Aech (Indonesian)	Classifying the dance gestures	2169 images	HMM	not specified
[11]	Thai dance	Thai dance system that compares experts dance movement and real-time dance movement	Real time dance movement	Kinetic motion sensor	-
[12]	Korean pop dance	classifying Korean pop dance movements (ELMC)	800 images	Extreme Learning Machine Classiifer (ELMC)	dimensionality reduction: PCA and FLDA

gives a feedback to the dancer for improving the dancing skills. The system compares the pose from Thai dance expert and real-time dancer pose to calculate accuracy.

Dohyung Kim et al. [12] have designed an efficient Rectified Linear Unit (ReLU) based Extreme Learning Machine Classifier (ELMC) 800 dance movement data points of 200 types of dances. The proposed method performs a better classification than those of KNN and SVM.

Zhi-huaChen et al. [13] have proposed a framework for classifying and predicting the labels of hand gestures. The framework will extract the hand region of 1300 images by subtracting the background using background subtraction method and then segmenting palm and fingers to recognize the fingers and to predict the labels using a rule classifier.

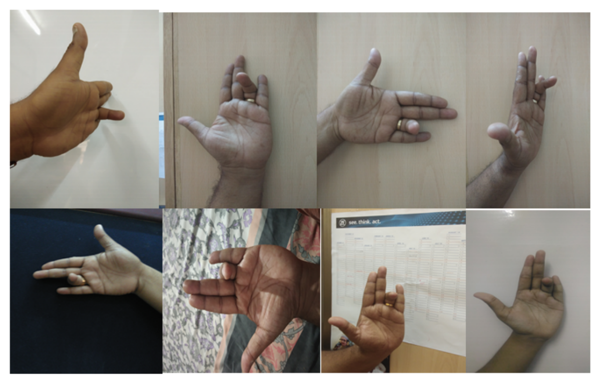

Fig. 2. Illustration of diversity in dataset

III. METHODOLOGY

In this section, we explain our model in detail. Initially, we brief about the data set that we have created from scratch and the pre-processing techniques explored. Then we explain the machine learning technique for classification using SVM and deep learning approach for image classification using CNN.

A. Dataset preparation and data pre-processing

To the best of our knowledge, there exists no dataset for Kathakali hand gestures at present. The primary challenge in this work was to create a dataset for all mudras from scratch. We build a new dataset of 654 images of Kathakali hand gestures in which each mudra has 27 images. The images are taken under natural light and room light. The mudra images are taken by both left and right hands and from different persons. Illustration of diversity in dataset is shown in the figure 2. You can understand that we have considered multiple factors including different positions, background color, hands by different people etc in generating the dataset. This is the first step towards the dataset creation. Here we have only hand images with clear background without any costumes for the

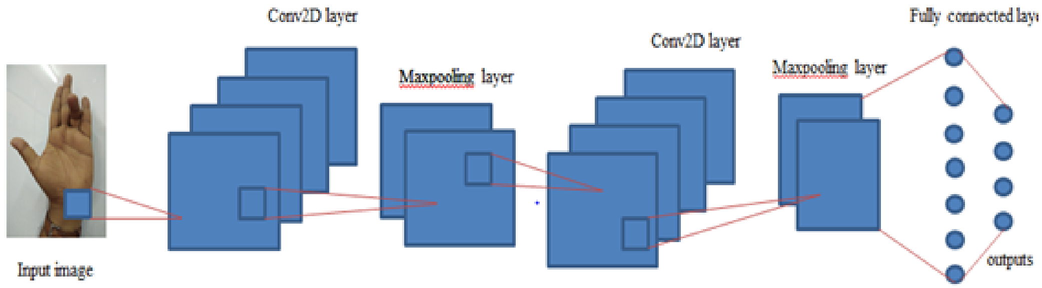

Fig. 3. CNN Architecture

persons showing the mudras. For the scope of this project, this dataset would suffice. This dataset has been made publically available and can be found at [17].
For pre-processing of Kathakali mudra classification, we explore four feature extraction methods - Haar wavelet features, Histogram of Oriented Gradients, Contour extraction and canny edge detection [4].

B. SVM Classification

To examine how machine learning classifies our data, we have used a multi class SVM model to classify Kathakali hand gesture image dataset. SVM classifier will take features as input and classify those features into different classes. SVM classification can be done in two stages. The first stage of SVM classification is pre-processing and feature extraction and the second stage is the classification stage.

In the first step, we tried four different methods such as Haar Wavelet features, Histogram of Oriented features, contour extraction and canny edge detection. The extracted features are given as the input to the SVM classifier and finally classifying the images using SVM classifier.

C. CNN Classification

CNN is one of the best methods for a higher level representation of image data [16]. CNN learns how to extract features from image pixel's data which has been given as input and tries to return the inference about pixels. CNN processes the input image and classify it in to different categories. Basically, each input image undergoes several convolutional layers of the CNN model. The convolutional layer contains filters, max or min pooling layers, Rectified linear unit (ReLU) layers and fully connected layers.

We designed a multi-layer Convolutional Neural Network model with two convolutional layers, two ReLU layers and two max pooling layers. We also used a batch normalization technique for calculating mean and standard deviation. The proposed CNN architecture is given in figure 3.

In the first stage the images are undergone through pre-processing stage. In this stage the images are resized into 56*56*3. Resizing the input images will actually increase the computational capacity. After resizing, the images are then converted to gray scale images. The output of the pre-processing stage is the images of size 56*56*1.

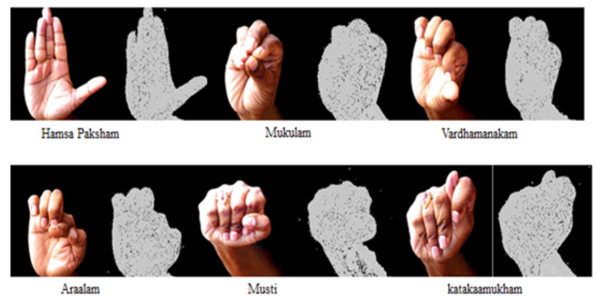

Fig. 4. Haar Wavelet features

The second step is feeding the pre-processed images to the Convolution Neural Network. The CNN model will learn from the features that have been extracted from the input images. Out of total 654 images, 480 images (20 images of each of the 24 classes) are taken as training set and 168 images (7 images of each of the 24 classes) are taken as testing set of images. The training set of images is divided into training set (388 images) and validation set (96 images).

IV. Results and Discussion

A. SVM Results

The SVM classification is performed in two steps. The first step is to perform pre-processing and extracting the features. The second step is to classify the images based on the features extracted by using SVM classifier.

The results of haar wavelet feature extraction are given in figure 4. The results show that in the Hamsa Paksham mudra for example, the shape is extracted properly and can be classified correctly from the classifier whereas for mudras Mukulam, Araalam and Vardhamanakam there might be collision happening when doing classification because of the extracted feature shape from the haar wavelet features. Another example case for simular issue is the mudras Musti and Katakaamukham.

The feature set of Histogram of Oriented Gradients are shown in figure 5. Similar to Haar Wavelet feature extraction method,

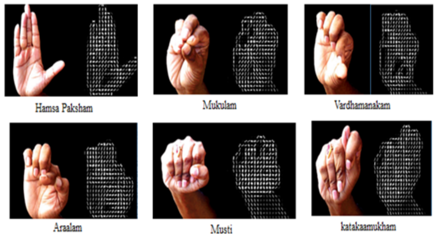

Fig. 5. Histogram of Oriented features

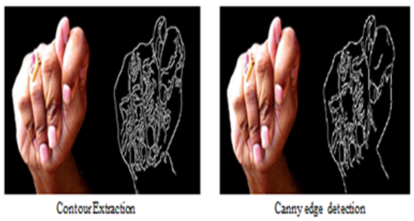

Fig. 6. Countour extraction and Canny edge detection

for Hamsa Paksham mudra the feature set is extracted properly here, where as mis-classification happens in case of Mukulam, Vardhamanakam and Araalam group as well as for the set Musti and Kaakamukham.

We have explored Canny edge detection and Contour extraction methods as edge detection methods for detecting edge features of our dataset. The results of Canny edge detection and Contour extraction methods are illustarted in figure 6. In case of Canny edge detection and contour extraction, instead of detecting edges, it is detecting many other lines and hence features are varying. Thus the classification method based on these features are not giving the best results.

There are several parameters for analysing the SVM classifier such as precision, recall, F1-score and accuracy.The Confusion matrix obtained for SVM classifier is given in the below Table III. From the table we can identify that for 6 classes or mudras [4, 5, 14, 15, 16, 23] the precision is 100% which is the highest possible precision and the results of recall and f1-score is presented in figure 7. The lowest precision is noticed for classes [3, 6, 10,17] and the results for recall and f1-score is compared in figure 8. The rest of the classes have the average precision and the overall average accuracy is 39%. Based on these metrics values, we can clearly tell that SVM classification model fails in the Kathakali mudra classification problem.

B. CNN Results

If we need to proceed with machine learning algorithms such as SVM, we need a lot more manual steps for data cleaning and data pre-processing. Thus we have chosen the Deep learning CNN architecture for classifying the Kathakali mudra dataset. The training images are divided into training and validation images. Among 20% of the training images are taken as validation set. We train the model with these

TABLE III
CONFUSION MATRIX

No	*Mudra*	*Precision*	*Recall*	*f1-score*
1	Pathaaka	0.60	0.60	0.60
2	Mudraakhyam	0.75	0.43	0.55
3	Katakam	0.25	1.00	0.40
4	Mushti	1.00	0.12	0.22
5	Kartharee Mukham	1.00	0.09	0.17
6	Sukathundam	0.17	1.00	0.29
7	Kapidhakam	0.33	0.25	0.29
8	HamsaPaksham	0.33	0.25	0.29
9	Sikharam	0.50	0.50	0.50
10	Hamsaasyam	0.17	0.20	0.18
11	Anjaly	0.75	0.50	0.60
12	Ardhachandram	0.29	0.29	0.29
13	Mukuram	0.29	0.40	0.33
14	Bhramaram	1.00	0.40	0.57
15	Soochimukham	1.00	0.17	0.29
16	Pallayam	1.00	0.25	0.40
17	Thripathaaka	0.09	0.50	0.15
18	Mrigaseersham	0.27	0.50	0.35
19	Sarpasirassu	0.50	0.25	0.33
20	Vardhamanakam	0.33	1.00	0.50
21	Araalam	0.33	1.00	0.50
22	Oornanabham	0.50	0.67	0.57
23	Mukulam	1.00	0.60	0.75
24	Katakaamukham	0.50	0.50	0.50

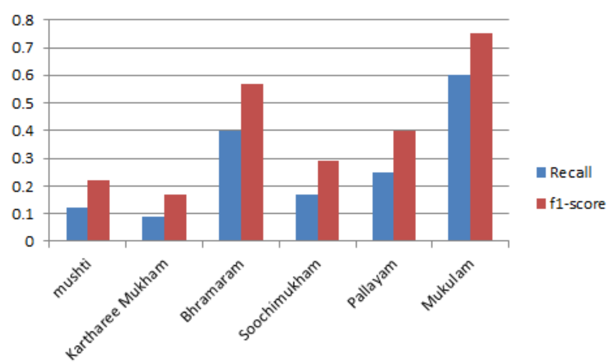

Fig. 7. Recall and f1-score values for High precision mudras

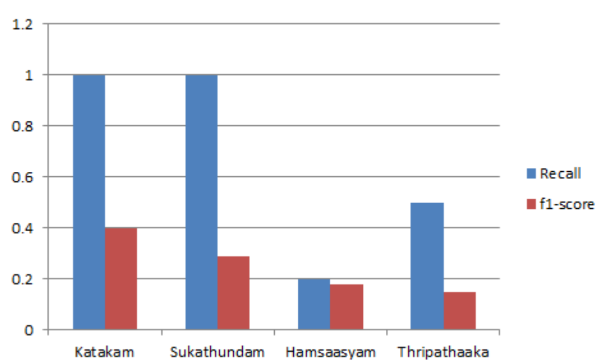

Fig. 8. Recall and f1-score values for Low precision mudras

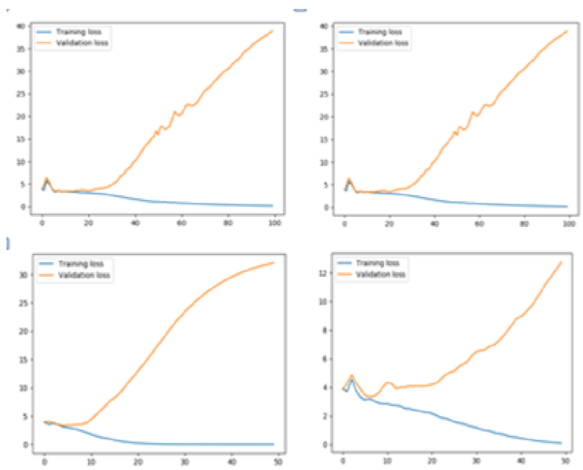

Fig. 9. Plot of training and validation loss

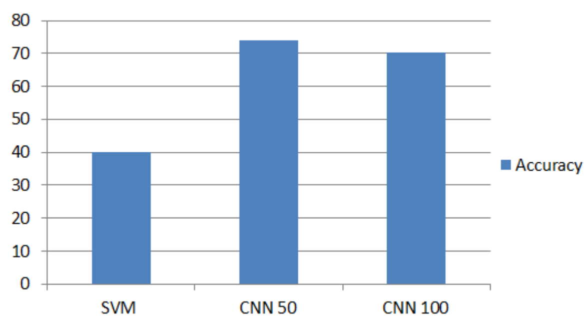

Fig. 10. Comparison of SVM and CNN in terms of testing accuracy

input images. The model is trained different times by varying number of epochs and the results are illustrated in figure 10. If the number of epochs is 50, the average accuracy is 78% and 74% for training images and testing images respectively. For 100 numbers of epochs, the average accuracy is 83% and 70.45% for training images and testing images respectively.

The training loss and validation loss is illustrated in figure 9. As the number of epochs are increasing the training loss is decreasing because for each epoch the CNN model will acquire new set of features and classify the images based on them. The plot of validation loss is raising as the number of epochs are increasing because a new set of images are trained and tested by the CNN model.

V. Conclusion and Future Remarks

In this paper, we have created dataset from scratch for Kathakali hand gestures, identified appropriate pre-processing techniques and compared SVM with CNN for mudra classification. The dataset consists of 684 images with 70% of training images and 30% of testing images. The features extracted from the Kathakali image dataset are given to the SVM classifier as input. The classifier classifies the images into different classes of mudras and we got an accuracy of 40%. The accuracy we got is very low because of the collision in the feature extraction results. The challenges include hand size, complexities in mudra shapes, hand makeup and so on. In case of CNN, we got an accuracy of up to 74%. This is the first attempt of this kind to create dataset for Kathakali images, identifying pre-processing techniques for machine learning classification and classification of mudras using deep learning techniques. Several possible future works include, generating larger dataset, identifying mudras shown in both hands, interpreting the mudra meaning from a video, identifying mudras from video streams etc.

VI. Dataset

Dataset developed for this work has been published for the benefit of public and is available in Mendeley Data [17]. We built a new dataset of 654 images of Kathakali hand gestures in which each mudra has 27 images.

References

[1] Kadathanattu Udyavarma Thampuran, Hasta Lakshana Deepika, Janranjinee Achukoodam (Printers), Nadapuram, 1892

[2] Anami, Basavaraj & Bhandage, Venkatesh, A Comparative Study of Suitability of Certain Features in Classification of Bharatanatyam Mudra Images Using ANN, Neural Processing Letters, 2018

[3] Kishore, et.al, Indian Classical Dance Action Identification and Classification with CNN. Advances in Multimedia. 2018.

[4] K.V.V. Kumar et.al, “Indian Classical Dance Mudra Classification Using HOG Features and SVM Classifier”. International Journal of Electrical and Computer Engineering (IJECE) Vol. 7, No. 5, October 2017

[5] K.V.V. Kumar, P.V.V. Kishore. “Computer Vision Based Dance Posture Extraction Using SLIC.” Journal of Theoretical and Applied Information Technology 15th May 2017. Vol.95. No 9.

[6] S. Samanta, et.al, ”Indian Classical Dance classification by learning dance pose bases,” IEEE Workshop on the Applications of Computer Vision (WACV), Breckenridge, CO, 2012, pp. 265-270.

[7] K. V. V. Kumar, et.al, ”Indian classical dance action identification using adaboost multiclass classifier on multifeature fusion,”Conference on Signal Processing And Communication Engineering Systems (SPACES), Vijayawada, 2018, pp. 167-170.

[8] M. Devi and S. Saharia, ”A two-level classification scheme for single-hand gestures of Sattriya dance,”International Conference on Accessibility to Digital World (ICADW), Guwahati, 2016, pp. 193-196.

[9] A. Bisht, et.al, ”Indian Dance Form Recognition from Videos,”13th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), Jaipur, 2017, pp. 123-128.

[10] Anbarsanti and A. S. Prihatmanto, ”Dance modelling, learning and recognition system of aceh traditional dance based on hidden Markov model,”International Conference on Information Technology Systems and Innovation (ICITSI), Bandung, 2014, pp. 86-92.

[11] N. Y. Tongpaeng, et.al, ”Evaluating real-time thai dance using Thai dance training tool,”International Conference on Digital Arts, Media and Technology (ICDAMT), Phayao, 2018, pp. 185-189.

[12] Kim, Dohyung & Kim, Dong-Hyeon Kwak, Keun-Chang. (2017). Classification of K-Pop Dance Movements Based on Skeleton Information Obtained by a Kinect Sensor. Sensors. 17. 1261

[13] Chen,”,Zhi-Hua et.al, ”Real-Time Hand Gesture Recognition Using Finger Segmentation. The Scientific World Journal. 2014

[14] De Gu (2015). Fingertip Tracking and Hand Gesture Recognition by 3D Vision. International Journal of Computer Science and Information Technologies, Vol.6(6), 2015

[15] R. M. Prakash, et. al, ”Gesture recognition and finger tip detection for human computer interaction,”International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore, 2017, pp. 1-4.

[16] R. Suresh and Prakash, P., “Deep learning based image classification on amazon web service”, Journal of Advanced Research in Dynamical and Control Systems, vol. 10, pp. 1000-1003, 2018

[17] Neelakanta Iyer, Ganesh; Bhavanam, Lakshmi Tulasi (2019), “Kathakali Mudras”, Mendeley Data, V1, doi: 10.17632/wdbm9srwn7.1