

Identification and Classification of Eye Movements in Bharatanatyam using Deep Learning Techniques

Pavithra Manikandan¹, Aravind Vijayaraghavan², S. Karthika³ and R. Srinivasan⁴

¹⁻⁴Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam, India

Email: skarthika@ssn.edu.in, aravind2010081@ssn.edu.in, pavithra2010140@ssn.edu.in, srinivasanr@ssn.edu.in

Abstract—This paper is centred around the topic of Image Processing in Bharatnatyam, specifically based on the identification and classification of eye movements in classical Indian dance forms. These eight eye movements, also known as dhristi bedas, are extremely significant in the portrayal of the dance form and convey distinct emotions when acted out by the dancer. It would be very beneficial for Bharatnatyam students and avid classical dance enthusiasts to be able to easily identify the eye movements performed by a dancer so they can understand the story being conveyed. Hence, this paper aims to develop an algorithmic system using artificial intelligence and deep learning to process a dance video as input data and display the category of eye movements performed by the dancer in the video. The framework of the system is as follows: A convolutional neural network is developed with multiple layers and the frames from the video data undergo various stages in our methodology such as pre-processing, feature extraction and mapping the recognized patterns using the softmax activation function to each eye movement. The implementation of our proposed algorithm is done using Tensorflow and Keras. The proposed algorithm will allow the audience and students to comprehend the messages of the story through the dancer's eyes.

Index Terms— Convolutional Neural Network, Sequential Neural Network, Deep Learning, Eye Movements, Dhristi Bedas, Image Processing, ResNet Model, Tensorflow, Keras.

I. INTRODUCTION

Bharatnatyam is a classical dance form that dates back over 2000 years, combining eye movements, intricate footwork, hand gestures and facial expressions to convey emotions and tell expressive stories. It has gained international recognition as a highly respected art form and is now taught and performed all over the world as an important symbol of Indian culture and identity on the global stage.

An important feature of Bharatnatyam is the dhristi bedas, which are the different eye movements or expressions that convey specific emotions and meanings. Dhristi bedas are not just static postures, but dynamic movements that require the dancer to transition smoothly between different positions and expressions. They are a means of creating symmetry, harmony and balance in a dancer's body and enhance the grace and beauty of the dance. They help in conveying the essence of the dance form, express the dancer's emotions and create a marvellous spectacle for the audience to observe.

In Bharatanatyam, there 8 major categories of eye movements each conveying its own meaning, namely: Samam, Alokita, Anuvritta, Nimilitta, Avalokita, Sachi, Pralokita, Ullokita. The mastery of the dhristi bedas is considered a vital aspect of Indian classical dance and it requires years of training and practice to perfect [1].

It is important to possess the ability to identify the dhristi bedas being portrayed by the dancer so that their performance can be fully understood and appreciated. This paper's objective is to digitally process Bharatanatyam dance videos, scan and contour the eye movements acted out by the dancers and identify the eye movements respectively using deep learning and neural networks.

A. Problem Definition

The goal of this paper is to develop an algorithmic system that accurately detects and classifies eye movements portrayed in a Bharatanatyam performer's dance video by implementing various Image Processing techniques such as constructing a neural network, applying convolutional filters, softmax function layers and using a large dataset of videos and frames to act as training and validation data.

This proposed system can positively impact the field of classical Indian dance and transform the art of teaching Bharatanatyam with modern techniques involving deep learning. Students and Bharatanatyam observers will be able to use the system to easily identify the categories of eye movements being portrayed in the videos. This will improve the quality of their classical dance education and also allow their learning of specific dance techniques to be easier and more understandable. This can also bring about an increase in online classes being conducted for students living abroad or in areas where offline classes may not be available, who yearn to learn from specific teachers and seek beneficial knowledge on this cultural art form.

B. Problem Domain

The field that this paper aims to assist in is the domain of Indian classical dance and how the practice of teaching the art-form as well as learning various aspects of it can be made easier.

Since we will be identifying eye movements, the type of data that will be used will be in the form of videos and moving images. Hence the algorithm being synthesized should not only be able to process stationary images but also process videos first and then the individual frames of the videos later. Another challenge for this paper is the amount of training and validation data required to develop the model. A large array of videos and frames (Around 200 for each category) in the datasets must be used for such a model, keeping in mind that eight major categories of dhristi bedas must be identified and the model needs to train continuously with the datasets to have the ability to identify minute changes in the expressions caught in these as well. The structure of this system must be a CNN (Convolutional Neural Network) that applies several layers and filters to conduct feature extraction for the video frames and then classify the video based on the categories of eye movements.

C. Problem Statement

Students and observers of the cultural dance form of Bharatanatyam may find the theoretical aspects of the dance hard to understand and identify when they are used in practice. They may not grasp the intricacies of the artform and the unique meanings behind different gestures that the dancers are trying to express in their story. This can cause a steady decline in the interest and awe of the minds of younger people who are not familiar with the rich and diverse culture that exists in Indian classical dances like Bharatanatyam. A solution to this problem is our proposed system of digitally analysing videos of these dance performances and identifying the different eye movements portrayed by the dancer, so students and observers can understand what the dancer is trying to convey. This system will use various deep learning techniques and theory that can train a model to classify the frames based on the eight major dhristi bedas. The 8 categories of eye movements have distinguishable movements but also have repeated patterns that are used in more than one category which makes its identification very difficult when processing happens. Thus, it is vital that we must possess an algorithm such as our proposed system that can identify the eye movements regardless of the similarities between them.

Our proposed system can profoundly increase the interest in cultural dance forms in the minds of students and younger people as it provides a progressive and comfortable way of educating themselves on this topic. Hence, our system can bring about a positive influence in preserving interest in cultural heritage and classical dance forms and create a bright path ahead for future generations.

D. Innovative Content

The subject matter of our paper is an original and unexplored area revolving around the ability to process dance videos for the sake of efficient and easier teaching methods of the art and identification of the minute details performed by the stage performers.

To our knowledge, based on the intensive research that has been done, there is no scientific paper discussing the classification of the dhristi bedas. The idea of identifying intricate eye movements of a dancer using image processing for the benefit of students and observers is unfamiliar in the modern day. This algorithm uses the

general fundamentals of deep learning but is also unique to our paper's functionality in terms of its development and usage as we must pre-process videos into individual images or frames and then process those images by passing them through the phases of the neural network.

Deriving the eye movements is a complex process that involves 2 generally significant steps: (i) We must capture the eye movement's essential features in video form and (ii) Map these features to a pattern that can be recognized. When a pattern can be recognized for each of the eye movements, classification of the dhristi bedas from the video will become possible.

An interesting discussion further ahead in this paper is the comparison between different architectural models of neural networks (CNN and ResNet, as these models seemed to be the most optimal for research work like this), why the CNN model is extremely suitable for such a paper and how it is more efficient than the latter.

II. RELATED WORKS

There are various papers that have done research in the field of Bharatanatyam image processing which is the primary domain of our work:

The classification of certain features in Bharatanatyam was proposed by [2]. The paper discussed 3 major steps in the classifying of mudras which are: (i) the pre-processing of images to obtain contours, (ii) feature extraction and (iii) using an artificial neural network for the classification. However, this methodology was developed for single hand mudra images only.

The next paper [3] developed a feature-based method to identify double hand mudras in Bharatanatyam. It presents a 3-stage methodology to identify 24 double hand images. Pre-processing of the images are done using canny edge algorithms, feature extraction is done using vertical-horizontal intersections and the images were classified into 24 types of mudras.

Another hand mudra identification model was constructed by [4] using HOG features and support vector machines (SVM). A histogram of the required features of the hand mudras was given as input to the classifier and the SVM classifies the features into mudras as text messages. This application leaned into smart computing and informatics.

An artificial neural network was developed by [5] to perform gesture recognition using image processing and pattern recognition methods. Classification is done based on the least difference of features given by an entry data and the identified mudra's health benefits are also explained.

The only paper related to Bharatanatyam eye movements was [6] which discussed the effect of dancer's movements on the observer's eyes during a performance. It explored the effects of a dancer's expertise on a viewer's eye movements and discussed the relations between narrative and non-narrative dance.

A. Gap Analysis

The above papers are some of the only papers centred around Bharatanatyam and image processing and most of them discuss the classification of hand mudras alone. Through the extensive research that we have done, it was discovered that there was no initiative in recognizing the eye movements in Bharatanatyam using neural networks. This paper is thus a novel idea as it discusses the identification of eye movements performed by a dancer and classifies them accordingly using deep learning methodologies.

III. METHODOLOGY

This paper is structured around convolutional neural networks and its various phases. Since we are processing intricate movements and not stationary images, the data that we require for input is in video format. We can process videos by separating the data into individual frames and these frames are processed through various layers of the CNN. The number of frames that are made from the video is dependent upon the frame rate.

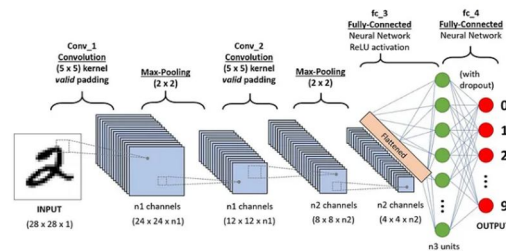


Fig 1 – Architecture of CNN. Reference [7]

The first layer is a convolutional layer, and it performs feature extraction from the frames of the video. This process is where the elements of the image matrix are multiplied to give convoluted features.

While applying the convolution filters, we additionally use the ReLu (Rectified Linear Unit) which acts as a linear function and allows deep learning algorithms to operate in a more optimal pattern if their behaviour tends to be linear. The next layer in this model is Pooling which passes a matrix filter through the convoluted features of the frame matrix and performs a mathematical function for every pool. In our algorithm, we perform maxpool to obtain the most prominent features of the eye movement. The next phase of the CNN is the Fully Connected Layer (FC) which works with a flattened input. The flattened vector is sent through additional FC layers where the mathematical operations are performed. The last phase of our CNN model which is the last FC layer's activation function is referred to as the softmax function. The neural network is configured to output N values and the softmax function's purpose is to normalize the N values by converting them from weighted sum values to probabilities. Classification can occur based on the probability of the features and the mapped category will be displayed.

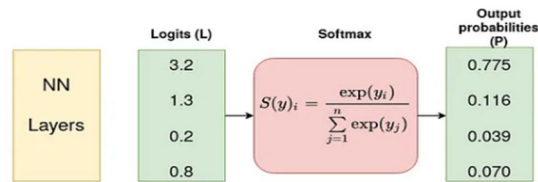


Fig 2: Softmax function operation. Reference [8]

IV. ALGORITHM EXPLANATION

Creating a neural network model:

- Import the necessary libraries such as TensorFlow and Keras
- Defining model architecture
- Sequential model object of a convolution neural network with the traditional Conv2D, MaxPooling, Flatten, Dense and a softmax layer.
- The model is compiled by specifying the optimizer, loss functions and evaluation metrics and graphs for epoch vs loss and epochs vs accuracy are plotted.

Classification of the uploaded video into one of 8 dhristi bedas:

- Once the model is compiled and loaded, get the path of the video of the eye movement that is to be identified.
- Using OpenCV, the video is split into frames of frame rate set to 5 by using cap.get().
- Once the uploaded video has been stripped into different frames, the frames then undergo pre-processing.
- During pre-processing, the video frames are resized to desired dimension, converted into a numpy array, and normalized by dividing the pixel values by 255.0.
- The processed video frames are stored in a new folder which are then used to make predictions using the neural network.

The index of the category with the highest probability is obtained and the name of the predicted category is displayed.

The table shown above contains all the methods that were used in the algorithm, with their respective parameters displayed as well. The total number of parameters is 11,169992.

V. RESULTS AND DISCUSSIONS

A. Datasets and how they were collected

The type of data that was required for the training and validation datasets is essentially in the form of videos since we are dealing with eye movements. So first we process the videos by splitting the frames of the video into separate images. The number of images is based on the length of the video that is given as input. If the video is longer, we obtain more frames per time length and hence we obtain more images. Having more images by

LAYER (type)	Output Shape	Parameters
conv2d (Conv2D)	(None,222,222,32)	896
max_pooling2d (MaxPooling2D)	(None,111,111,32)	0
conv2d_1 (Conv2D)	(None,109,109,64)	18496
max_pooling2d_1 (MaxPooling2D)	(None,54,54,64)	0
conv2d_2 (Conv2D)	(None,52,52,128)	73856
max_pooling2d_2 (MaxPooling2D)	(None,26,26,128)	0
flatten(Flatten)	(None,86528)	0
dense(Dense)	(None,128)	11075712
dropout(Dropout)	(None,128)	0
dense_1(Dense)	(None,8)	1032

Fig 3: Parameter Table

increasing the video length, allows the model to have a higher probability of success and is a crucial observation from this paper as it increases the accuracy rate.

For the training data we used 15 videos for each of the 8 categories (Fig 4 – Fig 11) that varied in time duration as different eye movements have different rates at which they are expressed. These videos were around 5 to 15 seconds depending on the category of the eye movement. We collected these videos by using professional dance videos that had clear imagery of the eyes which made it easier to train. For the validation data we used 10-15 videos depending on the category. Some of the categories required lesser videos as its identification was relatively easier than others. We obtained our validation data by filming ourselves, our family and friends and ensured that eyes were closely and clearly filmed. This gave a more realistic approach as most of us cannot perform these expressions the way a professional can. This allows the model to possess the ability to process eye movements that aren't as defined as a dancer's performance, thus increasing its intelligence, and allowing it to perform successfully for casual performers as well. Figures 4 to 11 represent each of the 8 eye movements:



Fig 4: Alokita



Fig 5: Anuvritta



Fig 6: Avalokita



Fig 7: Nimilitta

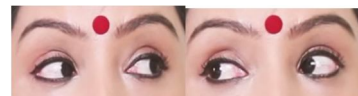


Fig 8: Pralokita



Fig 9: Samam



Fig 10: Ullokita



Fig 11: Sachi

B. Graphical Results

The following graphical representations (Fig 12 and Fig 13) showcase the variation of the accuracy rate and the rate of loss of both the training and validation datasets with respect to the number of epochs that occurred. From the accuracy rate graphs, we can observe that the training accuracy increases tremendously with every epoch and the validation accuracy also steadily increases. This tells us that the model has very low bias as well as low variance ensuring that the model is perfectly capable of generalization and does not suffer from underfitting or overfitting. From the loss rate graphs, we can observe that the training loss decreases steadily representing low error while the validation loss displayed more random changes as every epoch occurred but eventually declined steeply. This tells us that the model is continuously learning and adapting with every epoch occurring and trains itself to produce low loss rates. These observations tell us that the model has low generalization error as there isn't a large gap between the training and validation loss.

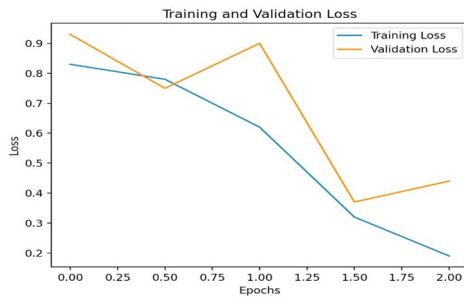


Fig 12: Graph of Loss vs Epochs

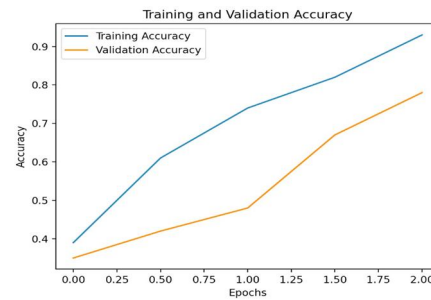


Fig 13: Graph of Accuracy vs Epochs

C. Why is the CNN Model chosen over the ResNet Model for this research work?

There are various structural formats for neural networks such as CNN, SNN, ResNet, ResuNet and the architecture we use is the Convolutional Neural Network. This layout gives us lower loss of information and has higher accuracy compared to other architectures. When a ResNet format was tried out using the same algorithm we discovered that the results had greater error and an increase in the rate of inaccuracies. There are several reasons for why the CNN format has proven to be more optimal than the ResNet model such as its relatively simpler layout and its requirement of lesser parameters. This can also explain how the CNN model may be prone to underfitting while simultaneously being less likely to overfit. On the other hand, the ResNet model is more complex and has more parameters, which can help it learn more intricate features and patterns in the data which would have been suitable for a paper such as this because identifying minute and small details in eye movements is very crucial. However, it also makes it more prone to overfitting on the training data. Thus, both overfitting and underfitting are prevalent issues when creating a model due to the lack of generalization. Overfitting is particularly more concerning as it leads to greater error when the model is used outside of its training dataset. Besides these factors, the most significant justification for why the ResNet model has proven to be less optimal is that it was originally designed to handle the [9] ImageNet dataset which contains many photographs specially taken for the trials of computer vision algorithms. This explains its incompatibility with our research work since our dataset has a relatively unique distribution of classes and images (frames of videos).

Epoch	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy
1	4.0270	0.1542	2.0804	0.083
2	2.0732	0.1819	2.0811	0.1095
3	2.0677	0.2271	2.0824	0.1095
4	2.0623	0.2271	2.0837	0.1095
5	2.0573	0.2271	2.0852	0.1095

Fig 14: Tabular representation of ResNet Model's Loss and Accuracy rates per epoch

Epoch	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy
1	0.83	0.39	0.93	0.35
2	0.78	0.61	0.75	0.42
3	0.62	0.74	0.90	0.48
4	0.32	0.82	0.37	0.67
5	0.19	0.93	0.44	0.78

Fig 15: Tabular representation of CNN Model's Loss and Accuracy rates per epoch

In conclusion, the CNN Model has an 87% accuracy rate in detecting eye movements and is proven to be a more optimal model for this research work.

D. Test Case : Pralokita

Pralokita is one of the more complex eye movements as it involves an eye movement moving from left to right and vice versa continuously making it one of the more challenging categories to accurately identify and map. The images below are the frames generated from one of the Pralokita videos in the dataset during pre-processing. When these frames are processed, the probabilities of the image matrix are calculated using softmax and mapped to the category that they belong to.



Fig 16: Frames generated from a Pralokita video for processing

In conclusion, the CNN was able to identify the eye movement as Pralokita, displaying these results:

1/1 [=====] - 0s 409ms/step

4 – index number (The 4th index number refers to Pralokita)

Predicted category: Pralokita

VI. CONCLUSION

Bharatanatyam, a traditional Indian Dance form originated in Tamil Nadu is one of the oldest and most popular forms of classical dances of India. It is a combination of beautiful eye gestures, intricate footwork, and rhythmic movements of limbs which come together to create a visual beauty. 'dhristi' in the Bharatanatyam art form refers to the eye movements of the performer that play a crucial role in the understanding and applauding of the art form. These eye movements are of 8 different types and are considered one of the most important aspects of a dancer's expression as they convey emotions and the essence of the story being told. This paper uses deep learning and image processing concepts to successfully identify and classify eye movements portrayed in a video format to one of the eight dhristi bedas. This provides a viable and improved teaching method that allows learners to effortlessly comprehend the complex and minute aspects of the art-form, particularly the eye movements or dhristi bedas that are expressed during dance. The main aspects of the methodology used to create this paper involves defining a convolutional neural network, its various phases and the datasets required to be provided as training and validation data. On execution of the algorithm, we were able to plot graphs that showed the relationship between the accuracy rates and loss rates of the training and validation data with respect to the number of epochs and showed our observations. This proves our algorithm has a high accuracy rate of 87% in detecting the eye movements. The usage of different models with the same algorithm was also thoroughly discussed and the justification for CNN being used over the ResNet model was explained with the training and validation loss and accuracies displayed in a tabular format. From the results that we have obtained after training our model with the datasets and executing our image processing algorithm, we can observe that the dhristi bedas performed by dancers are accurately recognized according to the eight types that they are categorized on, and the identified eye movement is textually displayed.

VII. FUTURE WORKS

After successfully identifying the eye movements in video format, it is certainly possible to focus on other aspects of Bharatanatyam as well. Hand gestures, foot-work and other facial expressions also have several categories that they can be classified into. In the future, it is potentially possible to develop an algorithm that can simultaneously identify all the dance movements being conveyed by the performers continuously in a time duration. The algorithm developed in this paper can also be extended to identify the eye gestures in sculptures present in the ancient temples of India. This might even help us better understand the stories and events conveyed through the sculptures present in historic temples.

REFERENCES

- [1] <https://kafqa.com/blog/eye-movements-or-drishthi-bheda-online-bharatanatyam-classes>
- [2] Anami, B.S., Bhandage, V.A. A Comparative Study of Suitability of Certain Features in Classification of Bharatanatyam Mudra Images Using Artificial Neural Network. *Neural Process Lett* **50**, 741–769 (2019). <https://doi.org/10.1007/s11063-018-9921-6>
- [3] Anami, B.S., Bhandage, V.A. A vertical-horizontal-intersections feature based method for identification of bharatanatyam double hand mudra images. *Multimed Tools Appl* **77**, 31021–31040 (2018). <https://doi.org/10.1007/s11042-018-6223-y>
- [4] Kumar, K.V.V., Kishore, P.V.V. (2018). Indian Classical Dance Mudra Classification Using HOG Features and SVM Classifier. In: Satapathy, S., Bhateja, V., Das, S. (eds) *Smart Computing and Informatics . Smart Innovation, Systems and Technologies*, vol 77. Springer, Singapore. https://doi.org/10.1007/978-981-10-5544-7_65
- [5] C. V. Soumya and M. Ahmed, "Artificial neural network based identification and classification of images of Bharatanaty gestures," 2017 International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), Bengaluru, India, 2017, pp. 162-166, doi: 10.1109/ICIMIA.2017.7975593.
- [6] Ponmanadiyil, Raganya, and Matthew H Woolhouse. "Eye movements, attention, and expert knowledge in the observation of Bharatanatyam dance." *Journal of eye movement research* vol. 11,2 10.16910/jemr.11.2.11. 28 Dec. 2018, doi:10.16910/jemr.11.2.11
- [7] <https://medium.com/@draj0718/convolutional-neural-networks-cnn-architectures-explained-716fb197b243>
- [8] <https://towardsdatascience.com/softmax-activation-function-how-it-actually-works-d292d335bd78>
- [9] <https://machinelearningmastery.com/introduction-to-the-imagenet-large-scale-visual-recognition-challenge-ilsvrc/>