iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** XI **Month of publication:** November 2023

DOI: https://doi.org/10.22214/ijraset.2023.56708

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue XI Nov 2023- Available at www.ijraset.com

Automatic Dance Pose Estimation from the Hand Signs using Deep Learning

Mrs. Rashmi H[1], Bommineni Tejaswini[2], Monica CS[3], Neha Reddy S[4], S Poornachandra[5]
K S Institute of Technology, India

Abstract: Human activity is the sequential change of the human body. The collection and assessment of multimedia content connected to dance will be beneficial for the preservation of cultural heritage, the creation of video recommendation systems, and the assistance of students through tutoring systems. Indian classical dance (ICD) classification is still a fascinating subject of research because of its intricate hand gestures. Changes in learning habits make automated teaching solutions unavoidable in many fields, from traditional to internet forums. ICD also becomes an essential part of a vibrant legacy and culture that needs to be updated and maintained at all costs. Complex poses including full-body rotation and self-hands-occlusion are part of the dance. The primary goal of this research is to create a framework for Bharatanatyam dancing.

I. PRESENTATION

Indian classical dance has been practiced and performed worldwide for more than 5000 years. Only an expert can fully comprehend the hand gestures and dancing movements, as well as the complex routines accompanied by recitations of poetry and music. These traditional dance forms are referred to by "Natyam Rasa" in "Natyam Shastra – Indian Classical Dance Form". The hands, feet, and torso movements are described by 108 Karanas (gestures) in Natyam Shastra. These hand motions reflect a variety of physical meanings with deity, nature, and behavior. The stances from the well-known classical dance form Bharatnatyam have been preserved at this temple in Tamil Nadu, named Chidambaram.

II. RELATED WORKS

A. Dance Tracking

To extract the link between sound and movement features, they have created a deep learning dance generating method in this study. Coordinates of the major points of human bones taken from dancing movies are employed for training as movement characteristics during the feature extraction phase, while rhythmic features taken from music and audio beat features are used as musical features. The model's generator module is used to create soft dancing gestures and to accomplish a rudimentary mapping of music and dance movements throughout the model-building process. For dancing and music to be consistent, the identification module is utilized. To improve the representativeness of the audio function, the self-encoder module is employed. The Deep Fashion dataset's experimental findings demonstrate that the

III. PURPOSES

Understanding AI Integration in Dance Pose Recognition: Review recent studies and publications on the use of artificial intelligence (AI) to pose recognition in the creation of visual dance signals. Impact on Professional and Educational Domains: Analyze the established advantages, drawbacks, and potential ramifications of AI-powered dance posture recognition, with a focus on the ways in which it could impact professional and educational environments. Efficiency and efficacy: Compare the efficiency and efficacy of AI-driven processes to traditional methods, highlighting the advantages of time savings and the caliber of the content generated. Dance Sign Recognition Techniques: Look at ways to seamlessly integrate deep learning and open-CV components, and think about how AI fits into this process. Prospective Patterns and Obstacles: Analyze the likely developments,.

IV. METHODOLOGY

The suggested Convolutional Neural Network-Long-Short Term Memory Network (CNN-LSTM) for hand gesture assessment classification. YouTube videos of a teacher and student performing Bharatanatyam dance are used to illustrate the shloka (poem). After learning the input dataset, the image should first be pre-processed using the traditional scalar methods. Prior to processing the image, it must be rescaled, the noise must be extracted, and the image must be cleared. This noise-extracted image is provided to the hand mudra pose evaluation mode with a feature extraction segment by building the Tensor flow Efficient Net-U Net framework. On the other hand, a similarity index approach is used to extract important points. Lastly, we trained the CNNLSTM framework with the training dataset of the classification system.

For every epoch, the training accuracy and loss of the suggested system are recorded. The execution is assessed using a number of parameters, such as accuracy, confusion matrix, recall, precision, f1-score, AUC curve, and the recommended system.

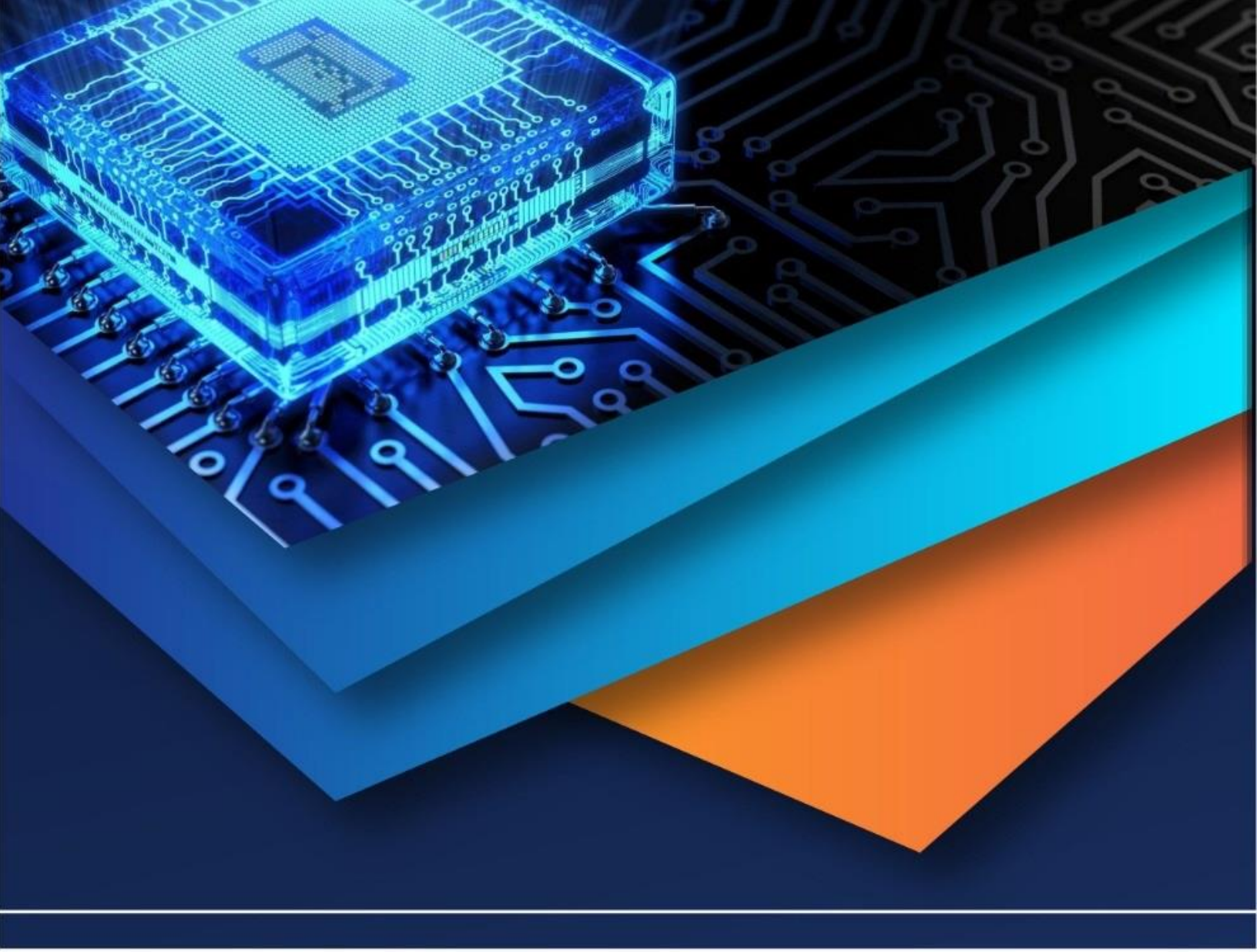
A. Explanation of Dataset

To achieve robustness, the mudra estimation strategy employs a variety of preprocessing techniques, including feature extraction, estimating, classification modules, data augmentation, and data preprocessing. The raw data from the acquired dataset is first pre-processed. The ground truth image is used to extract the precise location of the Bharatanatyam images, which are then fed into the pose estimate network. After preprocessing each input image, it is then improved.

By rotating the original image at different angles, the data for each input image is augmented. The dataset used in this study consists of 252 YouTube videos featuring the Bharatanatyam dancing style.

1) *Data Preprocessing:* Preprocessing data with standard scalar methods Consider that an input image of a certain type {wi, yi, Xi} is not-ed for each input image 1<i<="" span="" style="box-sizing: border-box; outline: 0px;"></iAdditionally, a picture of the inactive binary indicator—which classifies image positions as neither predictor nor predictee—was provided. l and l are the image positions of the images l and l, respectively, while l and l are the images l and l with the l position extracted, when seen in terms of notation. Think of l as the location of all image positions that have no corners but share a face with position l. On a typical lattice, l can have up to 4 components. Now that the redundant and noisy data has been eliminated, the information can be used to extract features.
2) *Feature Selection:* To assess how similar or dissimilar data points or samples within a dataset are, the similarity index methodology for feature extraction is utilized. By measuring the correlations between individual samples, it seeks to identify the essential traits or qualities of the data. Many similarity index measures can be used for feature extraction, depending on the properties of the data and the particular issue at hand. This suggested method makes use of the Cosine Similarity method. Think of l as the location of all image positions that have no corners but share a face with position l. On a typical lattice, l can have up to 4 components. Now that the redundant and noisy data has been eliminated, the information can be used to extract features. Feature Selection: To assess how similar or dissimilar data points or samples within a dataset are, the similarity index methodology for feature extraction is utilized. By measuring the correlations between individual samples, it seeks to identify the essential traits or qualities of the data. Many similarity index measures can be used for feature extraction, depending on the properties of the data and the particular issue at hand. This suggested method makes use of the Cosine Similarity method. One common method is to extract important image features by using deep learning models such as convolutional neural networks (CNNs) or image embedding. CNNs are highly proficient in obtaining hierarchical representations of image content, generating vector embeddings that are comparable through the application of cosine similarity. The general process for using cosine similarity on images is as follows:

a) *Preprocessing images:* Start by adjusting the image sizes to a standard size and performing any necessary preprocessing operations, like cropping or normalization.
b) *Feature Extraction from Images:* To extract features from the images, use a CNN (Efficient Net-U Net in this study) model that has already been trained. Every image will be transformed into a high-dimensional vector representation by the CNN.
c) *Compute Cosine Similarity:* Using the previously stated formula, determine the cosine similarity between the vector representations of the images. Image retrieval, image clustering, and content-based image recommendation are among the tasks that can be accomplished by comparing the cosine similarities among different image vectors. Cosine similarity facilitates efficient image analysis and retrieval by helping to identify similar image content based on their vector representations. Hand mudras classified by CNN-LSTM CNN LSTM, also referred to as CNN Long Short-Term Memory Network, is a type of LSTM architecture designed specifically for problems involving the sequence prediction of inputs with a spatial component, like images or videos. In the CNN LSTM architecture, sequence prediction is aided by LSTMs while features are extracted from input data using Convolutional Neural Network (CNN) layers. The retrieved features are used to train the CNN-LSTM classifier. The CNNLSTM incorporates both the CNN and LSTM structures. The CNN-LSTM deep learning structure has taught the feature data, and the classifier has learned the necessary details. First, a Re LU rectifier and convolutional level C-1 with 3x3kernal are used to convolve the input representation after the crucial points have been removed. Each feature vector generated by the layer is always 32 x 32 in size. C-2 and C-3 are the next two levels, and they are arranged in that order. Following layer C-3 is a pooling level called P-1 with a 2x2 kernel dimension. When the pooling layer employs a 2x2 kernel, P-1's output yields a 16 x16 kernel. The dense layer, which generates 512 neurons, is shown after the P-1 layer. The LSTM uses the layer's output as its input layer.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY