Enhanced Hand Segmentation Using UNet: A Comparative Study with Classical Approaches

Remya PK[1], Rajkumar KK[2]
[1]*Research Scholar, Dept. of Computer Science, Kannur University*
[2]*Professor, Dept. of Computer Science, Kannur University*

Abstract- The segmentation of hand gestures from complex backgrounds is essential for accurately classifying classical dance movements, as it enhances feature extraction and recognition performance. This paper presents a robust segmentation framework based on the UNet architecture, a deep learning model well-suited for pixel-wise segmentation. Initially, various traditional segmentation techniques are analyzed and evaluated; among them, UNet demonstrates superior performance in handling the complexity of classical dance gestures. The proposed method consists of two main stages: developing a UNet-based model for hand segmentation and refining it to capture the intricate and expressive movements characteristic of classical dance. The model is trained on a carefully curated dataset featuring diverse hand shapes, skin tones, and poses to ensure strong generalization. Post-segmentation, feature extraction is optimized to focus on the most relevant elements for classification. Robustness and adaptability are further improved using data augmentation and transfer learning techniques. Experimental results show that the proposed system surpasses existing methods in accuracy, real-time performance, and resilience under varying conditions. Applications include real-time gesture recognition in classical dance training, augmented reality, and enhanced human-computer interaction.

Keywords: Hand Segmentation, Deep Learning, UNet, Gesture Recognition, Mudra Classification

I. INTRODUCTION

Classical Indian dance forms like Bharatanatyam (Tamil Nadu) and Kuchipudi (Andhra Pradesh) are rich expressions of Indian cultural heritage [1]. These styles integrate rhythm, Carnatic music, facial expressions, and intricate hand gestures (mudras) to narrate mythological stories from epics such as the *Ramayana* and *Mahabharata* [2]. Kuchipudi, typically performed in Telugu, emphasizes Abhinaya (expression), utilizing Samyukta (double-hand) and Asamyukta (single-hand) mudras, along with postures like Ardhamandala and Samamandala [3].

Hand gestures form a complex and expressive non-verbal language, making them an ideal subject for computer vision and gesture recognition research. Their accurate segmentation is vital for applications like virtual dance training, digital archiving, automated performance analysis, and semi-automated choreography, where algorithmic support can streamline creative workflows [4]. This work focuses on segmentation of hand regions from classical dance images. Unlike earlier studies that handle full-body or cluttered backgrounds, our dataset includes pre-cropped hand-only images. Despite the controlled input, challenges such as gesture complexity, lighting variation, and skin tone diversity make segmentation difficult and critical to downstream tasks. Traditional segmentation techniques like thresholding and background subtraction often underperform under real-world conditions, failing to capture the detailed finger alignments and contours essential for gesture classification [5]. To overcome these limitations, we propose a UNet-based deep learning model for high-precision hand segmentation [6]. UNet's encoder-decoder architecture with skip connections allows effective learning of both global structures and fine edge details. Our model generates accurate segmentation masks even from hand-only images with residual background, improving the input quality for feature extraction and classification [7].The key contribution is a segmentation strategy tailored for classical dance gestures, offering improvements over traditional methods across varying poses and lighting conditions. This robust preprocessing supports applications in dance pedagogy, cultural preservation, and artistic human-computer interaction.

The rest of this paper is organized as follows: Section II reviews related work; Section III outlines our methodology; Section IV presents method comparisons; Section V discusses results; Section VI concludes the study.

II. LITERATURE REVIEW

A comprehensive understanding of mudra classification was developed through an extensive review of related works.

Sriparna et al. (2013) applied Sobel edge detection to extract hand contours and used fuzzy L-membership classification, achieving 85.1% accuracy [8]. Shweta Mozarkar et al. (2013) used saliency detection for segmentation and classified gestures using KNN with 85% accuracy [9]. K.V.V. Kumar et al. (2017) processed internet-sourced images using HOG, SURF, SIFT, LBP, and HAAR features. Though the segmentation method wasn't detailed, ROI isolation was implied. HOG with SVM gave the best performance at 90% accuracy, and a GUI was developed to track gesture frequency [10]. Basavaraj S. Anami and Venkatesh A. Bhandage (2018, 2019) used Canny edge detection and binary thresholding for segmentation. Features such as Hu moments and intersections were extracted. Double-hand mudras were classified with a rule-based system (95.25%), and single-hand gestures using ANN achieved up to 98% accuracy[11].

Anuja P. Parameshwaran et al. (2019) used CNNs with transfer learning for single-hand gestures, employing region-based segmentation from YouTube video frames. The method achieved 95% accuracy [12]. Bhavana R. Maale et al. (2020) segmented Sattriya dance gestures using skin-based methods. Contour chain codes were used for feature extraction and classified via various algorithms, with SVM achieving 92.3% accuracy [13]. Ashwini Dayanand Naiket et al. (2020) used ResNet34 on a multi-class dance dataset. Although segmentation was not explicit, the CNN model learned local features effectively, achieving 78.88% accuracy [14].

Overall, segmentation remains a critical step in gesture classification, directly influencing feature extraction and classification outcomes. Studies consistently highlight its importance, with improved techniques like deep learning-based models showing promise in enhancing accuracy across diverse Indian classical dance forms.

III. METHODOLOGY

This work focuses on comparing hand segmentation techniques for Bharatanatyam gesture recognition. The process includes dataset creation and segmentation using traditional and deep learning methods. The aim is to identify the most effective approach for accurately isolating hand regions in complex dance images.

1. Dataset

Due to the lack of publicly available Bharatanatyam mudra datasets, we created a custom dataset (Dataset 1) with 1,350 images across 30 classes, captured in controlled conditions. A second dataset (Dataset 2) with 11,350 images from an online source was standardized to address class imbalance. The varied backgrounds and sources improve model generalizability.

IV. COMPARISON OF SEGMENTATION METHODS

Accurate hand segmentation is crucial in classical dance gesture recognition to isolate hands from complex backgrounds and ensure reliable feature extraction [15]. We analyze various segmentation techniques using performance metrics such as Intersection over Union (IoU), Dice Coefficient, and Segmentation Accuracy [16].

In this section, we compare segmentation methods including Otsu thresholding, watershed, region growing, binary thresholding, Canny edge detection, UNet-based segmentation, and other traditional techniques, evaluated by accuracy, robustness, efficiency, and visual output.

1. Otsu's Thresholding

Otsu's method is a global thresholding technique that separates foreground and background by maximizing inter-class variance [17]. It is fast and effective for images with clear contrast. However, Otsu's method underperforms on our dataset due to complex backgrounds, lighting variations, and subtle intensity

changes that violate the bimodal histogram assumption. As shown in Fig. 4(b), segmentation is inconsistent—parts of the hand are misclassified—highlighting Otsu's limitations in real-world dance gesture images.

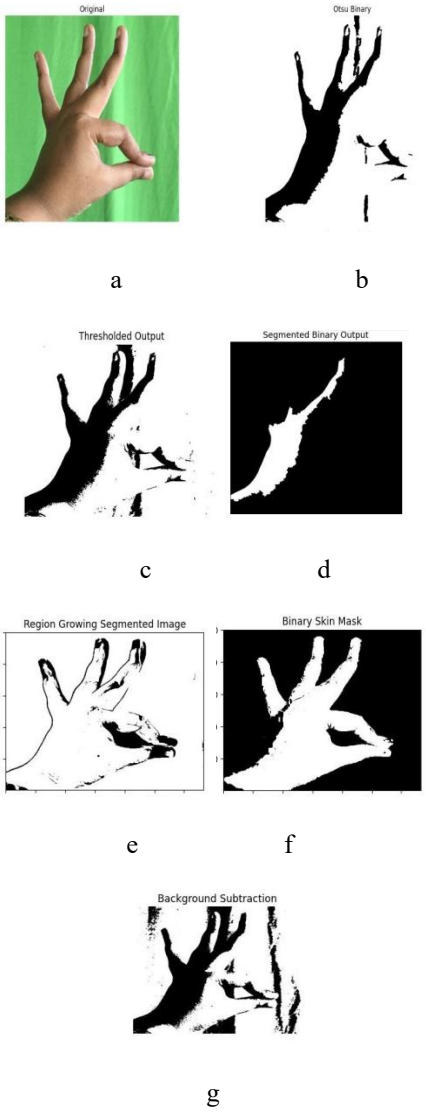

Fig 4: (a) Original color image, Segmented image after applying (b) Otsu's thresholding (c) Triangle Threshold (d) Watershed (e) Region Rrowing (f) Skin Based Segmentation (g) Background Subtraction Method.

2. Triangle Threshold Segmentation

This method identifies the histogram peak and draws a line to the tail end, selecting the point with the maximum perpendicular distance as the threshold. It is suitable for skewed histograms where the foreground is small and distinctly brighter or darker than the background [18].

Though effective in specific scenarios, triangle thresholding often fails with complex images, producing unclear edges and incomplete segmentations due to noise or low contrast. Fig. 4(c) shows how this leads to fragmented hand boundaries, making it unsuitable for tasks like gesture recognition that require precise details. Compared to Otsu's method—which assumes a bimodal distribution—triangle thresholding is better for skewed histograms with small, distinct foregrounds.

3. Watershed Algorithm

The Watershed algorithm treats a grayscale image as a topographic map, where bright regions are peaks and dark regions are valleys. Water is conceptually poured into valleys, and boundaries form where water from different catchment basins meet. Foreground and background markers, often obtained using distance transforms or morphological operations, guide segmentation [19]. Watershed segmentation partitions an image I into regions R_1, R_2, ..., R_n by minimizing the sum of the image gradients.

Watershed performs well in clean, high-contrast images but is highly sensitive to noise and marker accuracy. In our dataset with natural backgrounds and lighting variations, it caused over- or under-segmentation (Fig. 4d), failing to capture fine details like finger contours. Even on hand-only images, it produced fragmented results due to gradient sensitivity. The need for heavy preprocessing makes it unsuitable for our segmentation task.

4. Region Growing Segmentation

It is a pixel-based image segmentation technique used to group pixels or sub-regions that share similar characteristics. It works by starting from a set of seed points and growing the region by including neighboring pixels that have similar properties like intensity, color, texture, etc to the seed. The key concept behind region growing is that regions in an image tend to have similar pixel values within their boundaries, and neighboring regions tend to have different pixel values. Region growing exploits this property to segment objects from the background or to distinguish different regions within an image. The

region growing algorithm starts with a seed pixel and compares the intensity of neighboring pixels to it[20]. If the intensity difference between the seed pixel and a neighboring pixel is less than a predefined threshold T, that pixel is added to the region. Region Growing struggles to produce precise boundaries, especially in complex or cluttered images, as it relies on pixel similarity, which can lead to incomplete segmentation when hand gestures have variations in color or texture (fig 4 (e)). The threshold value, which defines pixel similarity, is sensitive to differences in hand appearance, making it challenging to choose the right value for varying gestures, orientations, and skin tones. The algorithm's reliance on the initial seed point means small changes can lead to inconsistent results, particularly when hands have overlapping fingers or complex shapes. Additionally, Region Growing is prone to over-segmentation in images with noise or varying backgrounds, causing incorrect pixel classification. Finally, the method can be computationally expensive, especially for large datasets, as it requires checking pixel similarities across the entire image, leading to slow processing times.

5. K-Mean Clustering

K-Means is an unsupervised clustering algorithm that segments images by grouping pixels into K clusters based on color or intensity similarity. It minimizes the within-cluster sum of squares by iteratively assigning pixels to the nearest cluster centroid and updating centroids accordingly.

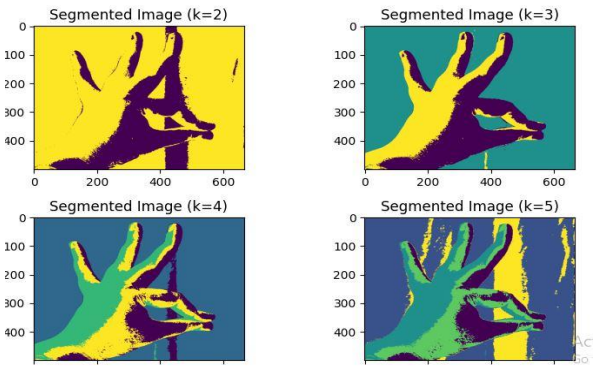

Fig 5: K-means Image Segmentation with Varying Cluster Counts (K=2 to K=5)

While K-Means clustering [21] yielded reasonably good results for gesture segmentation at K=3 (Fig. 5), its effectiveness is limited by several factors. The algorithm assumes spherical clusters of equal size, which may not reflect the variability in gesture shapes, lighting conditions, or complex backgrounds. Furthermore, the segmentation quality is highly sensitive to the choice of K, and poor selection can lead to either over- or under-segmentation. As a result, although suitable in controlled settings, K-Means is not consistently reliable for real-world gesture image segmentation.

6. Skin Based Segmentation

This method identifies skin regions by thresholding pixels in color spaces like HSV or YCrCb, which better separate skin tones than RGB. Although simple and fast [18], this technique failed on our dataset due to skin tone variations, lighting changes, and complex backgrounds [22] (Fig. 4(f)). It often results in missed hand regions or false positives, making it unreliable for accurate gesture segmentation.

7. Background Subtraction Segmentation Techniques

This technique segments the hand by subtracting a static or adaptive background model [23]. It works in controlled environments but struggles in real-world settings (Fig. 4(g)). In our dataset, hand motion, dynamic backgrounds, and lighting variation caused inaccurate segmentation, including fragmented hands and merged background regions. Thus, background subtraction is ineffective for consistent hand gesture recognition.

8. UNET-BASED SEGMENTATION

When trained on diverse hand gesture datasets, UNet effectively segments hands from complex backgrounds, outperforming traditional methods like thresholding and edge detection. It handles intricate shapes, overlapping gestures, varying lighting, skin tones, and occlusions with high accuracy and robustness. Though computationally intensive, UNet's architecture leverages GPU parallelism, enabling real-time or large-scale applications. Its precise, clean segmentation with smooth edges (Fig. 6) makes it ideal for gesture recognition and classification tasks [24].

Segmentation Method	IoU(%)		Dice Coefficient		Accuracy (%)	
	Dataset 1	Dataset 2	Dataset 1	Dataset 2	Dataset 1	Dataset 2
Otsu Method	78.23	80.69	0.881	0.893	83.48	85.31
Triangle Thredhold	79.64	81.76	0.888	0.899	84.01	85.69
Watershed Algorithm	52.89	56.02	0.519	0.543	26.94	27.88
Region Growing	20.42	18.17	0.171	0.150	26.10	24.43
Canny Edge Detection	12.33	10.05	0.041	0.207	26.92	24.85
Kmean Clustering	64.58	67.03	0.631	0.654	54.33	56.40
SkinBased	19.31	17.75	0.159	0.143	26.71	25.02
Background Subtraction	32.76	34.10	0.495	0.508	43.05	44.44
Unet Based	87.41	89.67	0.946	0.953	97.93	98.67

Table 1: Evaluation Metrics of Different Segmentation Algorithms for dataset1 and dataset2

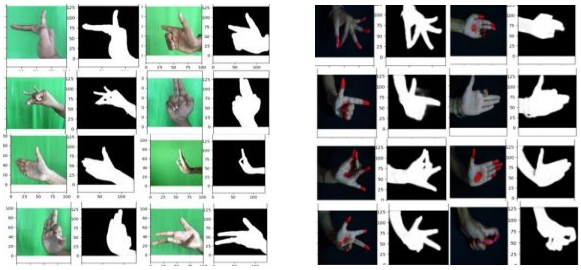

Fig 6: Original Images and Corresponding Segmented Outputs Using U-Net for Dataset1 and Dataset2

V. RESULT ANLYSIS

To further validate the segmentation performance, we present quantitative metrics such as Intersection over Union (IoU), Dice Coefficient, and Segmentation Accuracy for each method. These metrics provide an objective measure of how closely the segmented output matches the ground truth hand regions.

We evaluated segmentation techniques on all 28 mudra classes from our dataset, covering diverse hand shapes and complexities. Performance was measured as a binary task—hand vs. background—using IoU, Dice Coefficient, and Accuracy. Table 1 summarizes average results for Dataset 1 and Dataset 2. UNet-based segmentation outperformed others, achieving IoUs of 87.41% and 89.67%, Dice scores of 0.9465 and 0.9533, and accuracies of 97.93% and 98.67%, respectively.

VI. CONCLUSION

In this study, we analyzed various image segmentation techniques, including traditional methods such as region growing, skin color-based segmentation, and watershed. While these techniques showed varying levels of effectiveness, the UNet-based deep learning approach consistently outperformed them in terms of accuracy and robustness, especially in handling complex backgrounds and varying lighting conditions. The high-quality segmented outputs from UNet significantly improved the clarity and separation of hand regions. These precise segmentations play a crucial role in extracting meaningful and discriminative features, which are essential for the success of any classification algorithm. With accurate hand region isolation, the extracted features can be reliably fed into machine learning or deep learning classifiers, resulting in improved gesture recognition performance. Thus, UNet not only enhances the segmentation stage but also strengthens the entire recognition pipeline.

REFERENCE

[1] Seth, R. (2016). Annotated Bibliography on Kuchipudi Dance Sources. *Danza e ricerca. Laboratorio di studi, scritture, visioni*, 207-250.

[2] Choudhary, M. (2024). The Temporal Continuity: Temple, Theatre and Anthropology. *Indian Historical Review*, *51*(2), 182-204.

[3] Tharmenthira, S. (2024). Aesthetics and choreography of Bharatanatyam. *Journal of Research in Music*, *2*(2).

[4] Raj, R. J., Dharan, S., & Sunil, T. T. (2023). Optimal feature selection and classification of Indian classical dance hand gesture dataset. The Visual Computer, 39(9), 4049-4064.

[5] Thabet, E., Khalid, F., Sulaiman, P. S., & Yaakob, R. (2018). Fast marching method and modified features fusion in enhanced dynamic hand gesture segmentation and detection method under complicated background. *Journal of Ambient Intelligence and Humanized Computing*, *9*, 755-769.

[6] Pu, Q., Xi, Z., Yin, S., Zhao, Z., & Zhao, L. (2024). Advantages of transformer and its application for medical image segmentation: a

survey. *BioMedical engineering online*, *23*(1), 14.

[7] Shailesh, S., & Judy, M. V. (2020). Computational framework with novel features for classification of foot postures in Indian classical dance. *Intelligent Decision Technologies*, *14*(1), 119-132.

[8] Sriparna Saha, Lidia Ghosh, Amit Konar, and Ramadoss Janarthanan. Fuzzy l membership function based hand gesture recognition for bharatanatyam dance. In 2013 5th International Conference and Computational Intelligence and Communication Networks, pages 331–335. IEEE, 2013.

[9] Mozarkar, S., & Warnekar, C. S. Recognizing Bharatnatyam Mud Recognizing Bharatnatyam Mudra Using Princip ra Using Princip ra Using Principles of Gesture Recognition Gesture Recognition.

[10] KVV Kumar and PVV Kishore. Indian classical dance mudra classification using hog features and svm classifier. In Smart Computing and Informatics: Proceedings of the First International Conference on SCI 2016, Volume 1, pages 659–668. Springer, 2018.

[11] Basavaraj S Anami and Venkatesh A Bhandage. A comparative study of suitability of certain features in classification of bharatanatyam mudra images using artificial neural network. Neural Processing Letters, 50(1):741–769, 2019.

[12] Anuja P Parameshwaran, Heta P Desai, Rajshekhar Sunderraman, and Michael Weeks. Transfer learning for classifying single hand gestures on comprehensive bharatanatyam mudra dataset. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pages 0–0, 2019.

[13] Bhayana R Maale and Unnati Ukanal. Normalized chain codes and oriented distances based bharatanatvam hand gesture recognition. International Journal for Research in Applied Science and Engineering Technology IJRASET.

[14] Ashwini Dayanand Naik and M Supriya. Classification of indian classical dance images using convolution neural network. In 2020 International Conference on Communication and Signal Processing (ICCSP), pages 1245–1249. IEEE, 2020

[15] Konar, A., & Saha, S. (2017). *Gesture recognition: principles, techniques and applications* (Vol. 724). Springer.

[16] Zhao, J., Li, X. H., Cruz, J. C. D., Verdadero, M. S., Centeno, J. C., & Novelero, J. M. (2023, July). Hand gesture recognition based on deep learning. In *2023 International Conference on Digital Applications, Transformation & Economy (ICDATE)* (pp. 250-254). IEEE

[17] Tan, Z. Y., Basah, S. N., Yazid, H., & Safar, M. J. A. (2021). Performance analysis of Otsu thresholding for sign language segmentation. *Multimedia Tools and Applications*, *80*, 21499-21520.

[18] Brattland, V., Austvoll, I., Ruoff, P., & Drengstig, T. (2017). Image processing of leaf movements in Mimosa pudica. In *Image Analysis: 20th Scandinavian Conference, SCIA 2017, Tromsø, Norway, June 12–14, 2017, Proceedings, Part I 20* (pp. 77-87). Springer International Publishing.

[19] Dong, X., Xu, Y., Xu, Z., Huang, J., Lu, J., Zhang, C., & Lu, L. (2018, September). A static hand gesture recognition model based on the improved centroid watershed algorithm and a dual-channel CNN. In *2018 24th International Conference on Automation and Computing (ICAC)* (pp. 1-6). IEEE.

[20] Kaluri, R., & Reddy, P. (2016). Sign gesture recognition using modified region growing algorithm and adaptive genetic fuzzy classifier. *International Journal of Intelligent Engineering and Systems*, *9*(4), 225-233.

[21] Yuan, H., & Wang, C. (2011). A human action recognition algorithm based on semi-supervised kmeans clustering. *Transactions on edutainment VI*, 227-236.

[22] Badi, H. (2016). Recent methods in vision-based hand gesture recognition. *International Journal of Data Science and Analytics*, *1*(2), 77-87.

[23] Takhar, G., Prakash, C., Mittal, N., & Kumar, R. (2016, December). Comparative analysis of background subtraction techniques and applications. In *2016 International Conference on Recent Advances and Innovations in Engineering (ICRAIE)* (pp. 1-8). IEEE.

[24] Samal, S., Gadekellu, T. R., Rajput, P., Zhang, Y. D., & Balabantaray, B. K. (2023, May). SAS-UNet: Modified encoder-decoder network for the

segmentation of obscenity in images. In *2023 IEEE/ACM 23rd International Symposium on Cluster, Cloud and Internet Computing Workshops (CCGridW)* (pp. 45-51). IEEE.