Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.084514

ARTICLE

Deep Hand Segmentation and Multi-Modal Gesture Recognition for Human-Robot Interaction via 3D Volumetric Encoding

Zarnab Kausar[1], Shaheryar Najam[2], Hadeel Alsolai[3], Bayan Alabdullah[3], Fatimah Alhayan[3], Ahmad Jalal[4,5,*] and Hui Liu[6,7,8,*]

[1]Department of Engineering Technology, Foundation University, Islamabad, Pakistan
[2]Department of Electrical Engineering, Bahira University, Islamabad, Pakistan
[3]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia
[4]Department of Computer Science, Air University, E-9, Islamabad, Pakistan
[5]Department of Computer Science and Engineering, College of Informatics, Korea University, Seoul, Republic of Korea
[6]Guodian Nanjing Automation Co., Ltd., Nanjing, China
[7]Jiangsu Key Laboratory of Intelligent Medical Image Computing, School of Artificial Intelligence (School of Future Technology), Nanjing University of Information Science and Technology, Nanjing, China
[8]Cognitive Systems Lab, University of Bremen, Bremen, Germany

*Corresponding Authors: Ahmad Jalal. Email: ahmadjalal@mail.au.edu.pk; Hui Liu. Email: hui.liu198510@gmail.com

Received: 23 April 2026; Accepted: 17 June 2026; Published: 22 October 2026

ABSTRACT: Hand gesture recognition (HGR) is essential for Human–Robot Interaction (HRI) but remains challenging due to variations in hand shape, motion, viewpoint, illumination, and background, while vision-based methods often suffer from sensitivity to skin tone, occlusions, deformations, and limited interpretability. To address these issues, we propose a unified framework integrating deep learning, geometry-driven analysis, and temporal motion modeling. We introduce Z-HandSegNet framework, involving a U-Net with a ResNet-34 encoder for robust hand segmentation, and the Ellipse-Guided Geometric Finger Segmentation and Keypoint Extraction (EG-FSKE) method, which decomposes hand silhouettes into palm and finger regions using distance transforms, Gaussian Mixture Models, ellipse fitting, and a fragment recovery strategy for occluded and deformed fingers. To capture the hierarchical structure and dense motion, adaptive octree-based volumetric representation and Ef-RAFT optical flow are used for dynamic modeling. These representations are combined with saliency and ridge-based descriptors into a multimodal feature and classified with a Transformer encoder for modelling temporal dependencies. The evaluations on three datasets (Jester, IPN Hand and EgoGesture) achieve state-of-the-art performance with accuracies of 93.22%, 92.34%, and 94.17%, respectively, and precision, recall and F1-scores above 0.87. Ablation studies confirm the contribution of each component, particularly Z-HandSegNet and EG-FSKE, highlighting the robustness, interpretability, and potential of the proposed approach for future real-time HRI applications.

KEYWORDS: Human–robot interaction; hand gesture recognition; hand segmentation; geometric finger segmentation; optical flow; 3D volumetric representation; multi-modal feature fusion; transformer-based temporal modeling

1 Introduction

Human-Robot Interaction (HRI) attempts to provide intuitive interaction in various fields, including health care, industrial automation, assistive devices, and service robotics. Hand gestures have high potential because they are natural and expressive in nature. They can be categorized into two broad categories: static

Copyright © 2026 The Authors. Published by Tech Science Press. This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

and dynamic gestures. While the latter type is closer to practical use, gesture recognition still poses many difficulties related to hand posture, motion velocity, viewpoint, illumination, and complex backgrounds [1].

Initial HGR systems employed glove sensors but were invasive, expensive, and not applicable for natural HRI. Though depth-based and infrared techniques helped in achieving accurate 3D reconstruction, they involved specific equipment and were sensitive to noise and occlusions. Later, techniques based on RGB appeared as affordable substitutes. Handcrafted feature extraction techniques were inadequate due to sensitivity to illumination changes, object presence, and variation among users, while deep learning improved spatial-temporal learning but often lacked interpretability and required large annotated datasets [2].

To address such problems, we present a novel, fusion pipeline which leverages the use of deep learning, geometric features, motion models, and volume representation in order to perform robust hand gesture recognition. Using deep learning segmentation, a reliable extraction of hand region is performed, despite the presence of background clutter, variable illumination, and occlusions. The segmented hand will be further analyzed using a novel Ellipse-Guided Geometric Finger Segmentation and Keypoint Extraction (EG-FSKE) model, octree-based voxelization, optical flow, and ridge-saliency descriptors, to extract geometric, volume, motion, and structure-based hand features. Finally, a transformer-based encoder models long-range temporal dependencies for accurate gesture classification.

The key contributions of this work are as follows:

- We introduce Z-HandSegNet, a novel deep learning-based framework for egocentric hand segmentation without relying on skin detection. Built upon U-Net with a ResNet-34 encoder, the framework integrates augmentation-aware learning, hybrid Binary Cross-Entropy and Dice optimization, and segmentation refinement to achieve robust pixel-level hand segmentation under complex backgrounds, illumination variations, occlusion, and diverse skin tones.
- We propose EG-FSKE, a geometry-driven method that decomposes hand silhouettes into anatomically consistent palm and finger regions using distance transform–based palm localization, GMM clustering, and ellipse-constrained refinement. A key contribution is a fragment recovery mechanism that reconnects detached finger segments toward the palm center, enabling robust handling of bent, twisted, and occluded fingers without training data.
- We introduce adaptive octree-based volumetric representation and Ef-RAFT optical flow for gesture recognition. The octree provides hierarchical sparse 3D encoding of hand geometry, while Ef-RAFT captures dense motion via correlation volumes and GRU-based refinement, producing magnitude–direction features robust to appearance variations.
- The proposed architecture performs early multimodal fusion of geometric, motion, appearance, and structural cues for holistic hand gesture representation. By combining volumetric, kinematic, and Transformer-based temporal features, it effectively recognizes complex gestures and outperforms existing methods on Jester, IPN Hand, and EgoGesture under occlusion and environmental variations.

2 Related Work

Intuitive HRI uses contact-free gesture recognition for non-verbal interactions. Gestures can be classified either as static or dynamic, as well as sensor- or vision-based [3]. Because of scale and intrusiveness problems, there was a trend from sensor-based to vision and deep-learning-based research. Convolutional Neural Network models demonstrated good results in gesture and sign language recognition using pre-trained light-weight CNNs under low-data circumstances [4], whereas YOLOv3 provided better gesture detection robustness [5].

In order to deal with temporal limitations, various solutions were proposed in which sequence modeling using LSTM models was used for capturing temporal information from visual features [6]; temporal CNNs were employed along with attention mechanisms to increase efficiency and discrimination capacity [7]. More recently, light-weight Transformers, pose estimation using Swin Transformer models, and efficient pipelines with MediaPipe have become available [8,9]. Simultaneously, progress on pose representation and skeletal representation enhanced gesture recognition via structured motion representation. 3D hand pose estimation from RGB images is able to reconstruct joints accurately without any depth camera [10], while semantic-guided GCNs and attention-based skeletal representation demonstrate impressive results in various benchmarks [11], which can handle even occlusions and two-handed interaction [12].

Gesture recognition in HRI is incorporated into control and teleoperation pipelines [13], where task-specific datasets provide HRI-oriented gestures [14]. The use of multimodal fusion, wearable devices, and convolution-Transformer frameworks increases robustness and accuracy [15,16], and gesture speech interfaces increase flexibility [17]. Nonetheless, such methods create higher hardware requirements, necessitating further investigations on robust vision-based solutions. Overall, even though improvements have been made in deep learning, temporal models, and pose estimation techniques, problems exist in accurate hand segmentation and motion capture, encouraging hybrid strategies for scalable and expressive HRI.

3 Materials and Methods

The proposed model is illustrated in Fig. 1, where RGB input is transformed into spatial, temporal, and geometric representations. For robust finger motion detection in HRI applications, hand segmentation, motion analysis, geometric decomposition, and octree volume encoding jointly provide complementary motion and structural information.

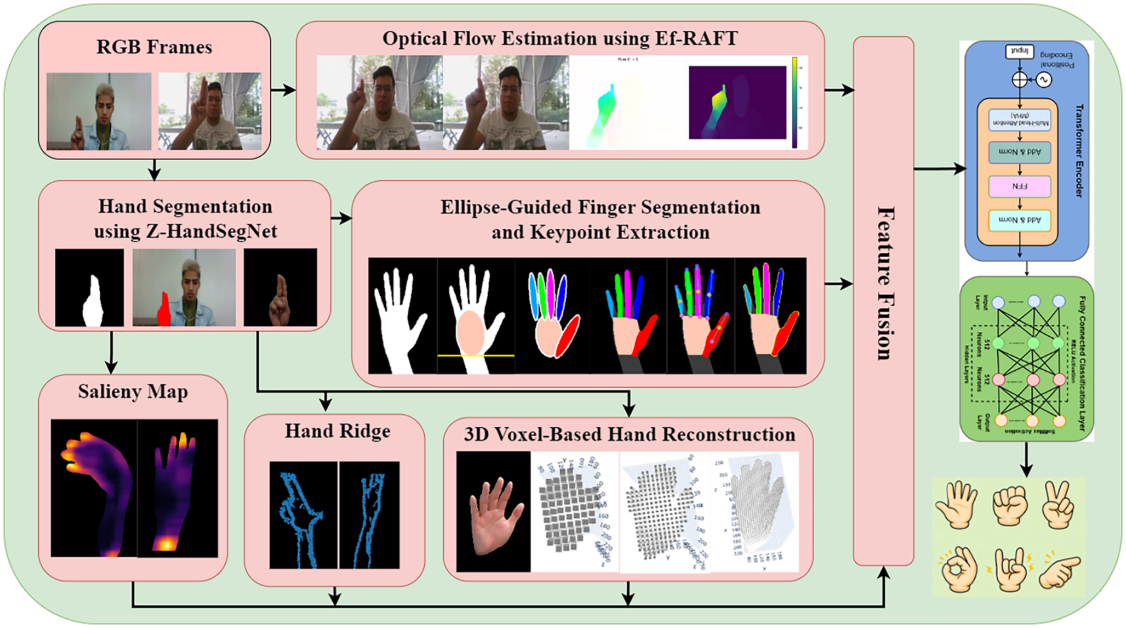


Figure 1: Archietcure diagram of multi-modal gesture recognition for human-robot interaction via geometric decomposition, 3D volumetric encoding, and motion dynamics.

3.1 Preprocessing

Hand localization is key for accurate hand gesture analysis from RGB video, as the presence of clutter, illumination variations and occlusions may introduce noise and affect feature extraction. Each frame $I \in \mathbb{R}^{H\times W\times 3}$ is resized to 256×256 for consistency and efficiency, then normalized to $[0,1]$ via $I_{norm}(x,y,c) = I(x,y,c)/255$ to reduce illumination and sensor variations. A light Gaussian smoothing with kernel G_σ is applied, yielding $I_{smooth} = I_{norm} * G_\sigma$, which suppresses noise while preserving edges.

3.2 Hand Region Segmentation Using Z-HandsegNet

Reliable hand localization is essential for accurate gesture recognition since feature extraction may be hindered by factors such as occlusions, lighting conditions, and clutter [1]. The framework based on U-Net with ResNet-34 encoder pre-trained on ImageNet, integrates augmentation-aware learning, hybrid optimization, and segmentation refinement for improved foreground extraction, as shown in Fig. 2. Trained on the EgoHands dataset (4800+ annotated frames with polygon masks converted to binary), it enables reliable hand segmentation for subsequent geometric and motion analysis. Images are resized to 256 × 256, normalized, and augmented (flipping, brightness, contrast), with an 80–20 train–validation split using PyTorch DataLoaders. The network captures multi-scale features, preserves spatial detail via skip connections, and outputs a logit map z, converted to a hand probability map using a sigmoid function (Eq. (1)).

$$\hat{y} = \sigma(z) = \frac{1}{1+e^{-z}} \tag{1}$$

here, $\hat{y}$ represents the predicted probability of each pixel belonging to the hand region. The model is trained for 40 epochs using the Adam optimizer with an initial learning rate of 1×10^{-3}, along with a ReduceLROnPlateau scheduler to adjust the learning rate based on validation loss. Training uses a hybrid loss combining Binary Cross-Entropy (BCE) and Dice loss, where the BCE loss is defined in Eq. (2).

$$\mathcal{L}_{BCE} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)\right] \tag{2}$$

where $y_i \in \{0,1\}$ represents the ground-truth label, $\hat{y}_i$ is the predicted probability, and N is the total number of pixels. To encourage spatial overlap between predicted and ground-truth masks and to mitigate class imbalance, Dice loss was employed, defined as Eq. (3).

$$\mathcal{L}_{Dice} = 1 - \frac{2\sum_{i=1}^{N}\hat{y}_i y_i + \epsilon}{\sum_{i=1}^{N}\hat{y}_i + \sum_{i=1}^{N} y_i + \epsilon} \tag{3}$$

here ϵ is a small constant added for numerical stability. Overall training objective is given by Eq. (4).

$$\mathcal{L}_{total} = \mathcal{L}_{BCE} + \mathcal{L}_{Dice} \tag{4}$$

Segmentation performance was evaluated using the Intersection over Union (IoU) and Dice coefficient metrics. IoU is computed as Eq. (5).

$$IoU = \frac{|P\cap G|}{|P\cup G|} = \frac{\sum_i \hat{y}_i y_i}{\sum_i \hat{y}_i + \sum_i y_i - \sum_i \hat{y}_i y_i} \tag{5}$$

where P and G denote the predicted and ground-truth binary masks, respectively. The Dice coefficient is defined as Eq. (6).

$$\text{Dice} = \frac{2|P \cap G|}{|P| + |G|} = \frac{2\sum_i \hat{y}_i y_i}{\sum_i \hat{y}_i + \sum_i y_i} \tag{6}$$

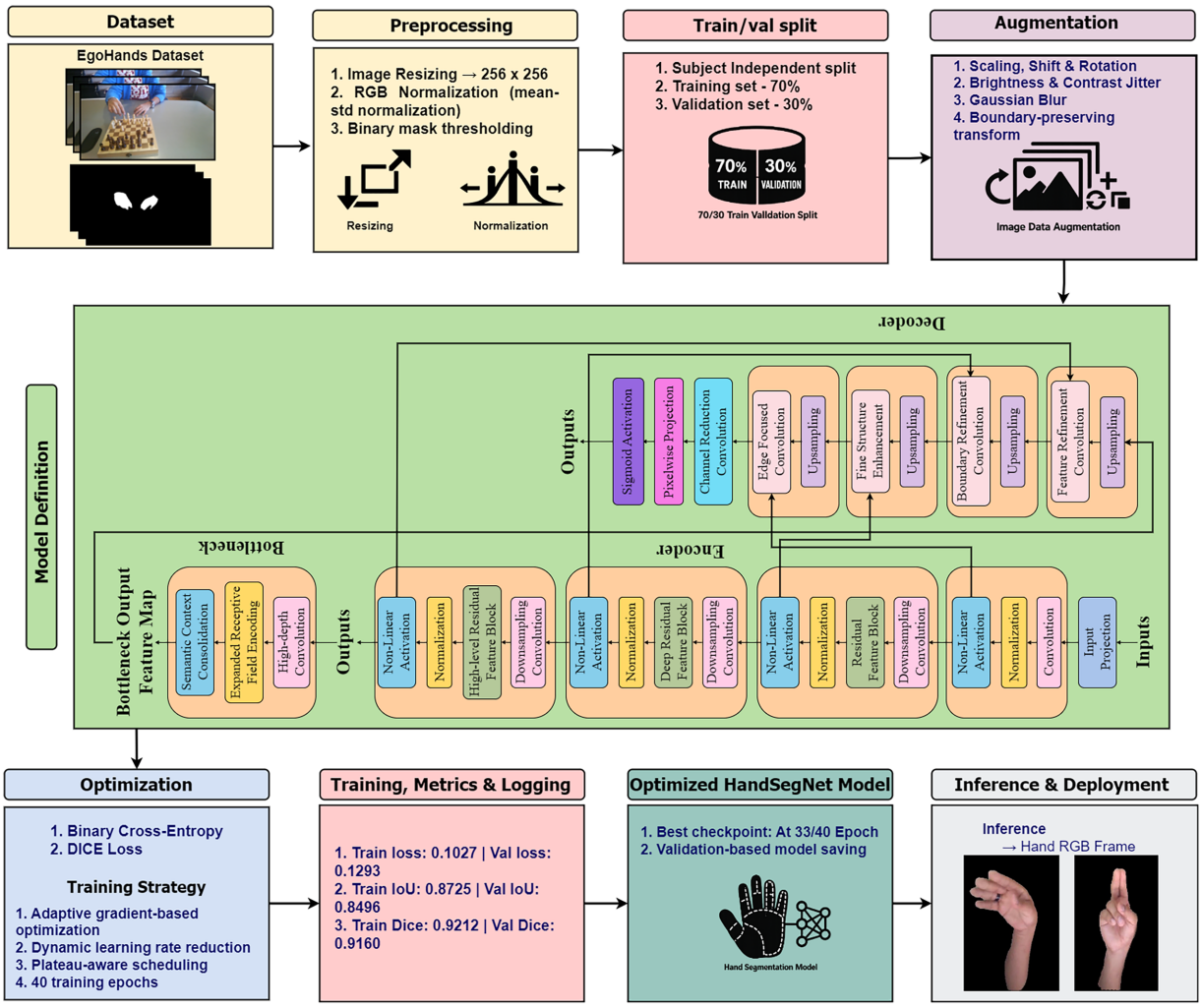


Figure 2: Visualization of layer-by-layer architectural specification of Z-HandSegNet for deep hand segmentation.

During inference, images are resized and normalized, then processed to obtain probability maps that are thresholded into binary masks. Post-processing (morphological refinement and Gaussian smoothing) is used to remove artifacts and improve boundaries. The proposed framework is hierarchical, with the hand segmentation being the foundation for the subsequent stages in the pipeline. Z-HandSegNet first localizes accurate hand regions, shown in Fig. 3, and then they are used in the adaptive octree-based volumetric encoding, saliency extraction, ridge analysis, motion feature computation, and EG-FSKE. Therefore, the quality of the segmentation is directly affecting the reliability of the downstream representations.

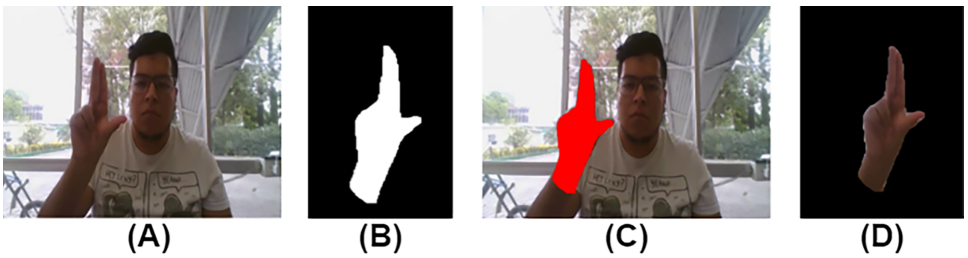


Figure 3: Hand extraction using Z-HandSegNet: (**A**) Original image (**B**) Binary mask (**C**) Mask overlay (**D**) Hand extracted.

3.3 Adaptive Octree-Based Volumetric Hand Representation

Hand geometry is represented as a 3D voxel volume and compressed using an adaptive octree to preserve structural hierarchy while reducing computational cost. The segmented hand mask is resized and converted into a voxel representation. This step produces a dense but redundant 3D representation of the hand. To efficiently encode spatial sparsity, we construct an adaptive octree in which each node represents a cubic region $C = (x, y, z, s)$ where s is the side length. A node is subdivided into eight smaller child cubes only if it contains foreground occupancy and has not reached the maximum depth d_{max}. The subdivision follows the standard spatial partitioning into eight equal octants using Eq. (7).

$$C_{pqr} = \left(x + p\frac{s}{2}, y + q\frac{s}{2}, z + r\frac{s}{2}, \frac{s}{2}\right), p, q, r \in \{0,1\} \tag{7}$$

This recursive splitting continues until either empty space is reached or the maximum resolution is achieved, producing a sparse set of leaf nodes $\mathcal{O}_{leaf}$. As a result, memory usage is significantly reduced from $\mathcal{O}(N^3)$ for dense voxel grids to $\mathcal{O}(|\mathcal{O}_{leaf}|)$, where $|\mathcal{O}_{leaf}| \ll N^3$. Three depths $d_{max} \in \{4, 5, 6\}$ are tested, corresponding to effective resolutions of 64^3, 128^3, and 256^3, respectively, as shown in Fig. 4. While higher depths preserve finer geometric details, they increase the number of nodes exponentially, making computation more expensive. The growth in representation size follows Eq. (8)

$$|\mathcal{O}| \propto \sum_{d=0}^{d_{max}} 8^d \tag{8}$$

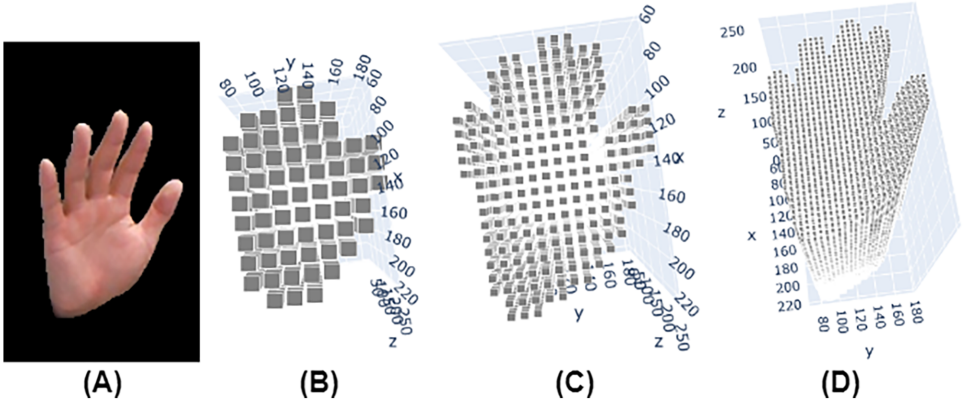


Figure 4: Voxel based Image reconstruction with different configurations (**A**) RGB hands (**B**) 64^3 (**C**) 128^3 (**D**) 256^3.

Empirically, the 128^3 configuration provides the best trade-off between geometric fidelity and computational efficiency, and is therefore used in all experiments. Each leaf node is then converted into a cubical mesh, and the final hand surface is reconstructed as the union of all leaf cube meshes defined in Eq. (9).

$$\mathcal{M} = \bigcup_{C \in \mathcal{O}_{\text{leaf}}} F(C) \tag{9}$$

yielding a multi-resolution, block-based three-dimensional representation.

3.4 A Novel Ellipse-Guided Geometric Finger Segmentation and Keypoint Extraction (EG-FSKE)

This novel, geometry-driven approach decomposes binary hand silhouettes into palm and finger regions, providing interpretable structural cues that complement learned and motion-based representations by integrating distance transforms, clustering, and ellipse-constrained refinement for robust and efficient palm–finger separation under bent, twisted, and partially occluded finger configurations. Given a binary hand silhouette $B(x, y)$, Eq. (10) computes the Euclidean distance transform.

$$D(x, y) = \min_{(u,v) \in \partial B} \sqrt{(x-u)^2 + (y-v)^2} \tag{10}$$

where ∂B denotes the hand boundary. The palm center $c_p = (x_p, y_p)$ is defined as the location of the maximum value of the distance transform, while the corresponding maximum distance value is taken as the palm radius r_p. A palm bottom boundary, is defined as Eq. (11).

$$y_b = y_p + r_p \tag{11}$$

This step separates the palm from the wrist/forearm, preventing wrist pixels from affecting clustering. Boundary noise is reduced via morphological erosion, given in Eq. (12).

$$B_e = B \ominus K \tag{12}$$

Candidate finger pixels are then extracted by selecting foreground pixels from the eroded boundary map that lie above the palm bottom boundary, ensuring that only finger regions are retained for further processing. These pixels are clustered using a GMM given in Eq. (13), with K = 5 components:

$$p(x) = \sum_{k=1}^{5} \pi_k \mathcal{N}(x \mid \mu_k, \Sigma_k) \tag{13}$$

where π_k, μ_k, and Σ_k represent the mixture weights, means, and covariance matrices. Connected components are extracted per cluster; those with centroids within palm radius r_p are labeled as palm, others as finger candidates. Candidates are refined via morphological closing and dilation, and fragmented segments are iteratively extended toward the palm center (Eq. (14)) to maintain anatomical consistency,

$$d = \frac{c_p - c_f}{\| c_p - c_f \|} \tag{14}$$

until overlap with the palm is achieved or iteration limit is reached, merging clusters of bent fingers into palm-connected regions. Each finger is then fitted with an ellipse via contour-based least squares, converted to a polygon, and overlapping regions are merged using normalized intersection area using Eq. (15).

$$\frac{A(E_i \cap E_j)}{\min(A(E_i), A(E_j))} \geq \tau \tag{15}$$

where E_i and E_j denote ellipse regions (Fig. 5A), $A(\cdot)$ is area, and $\tau = 0.7$. Finger masks with abnormal width (>1.15× second-largest) are replaced by aligned neighbors to maintain anatomical consistency. The final output (Fig. 5B) includes a unified palm and up to five consistent fingers; the palm is shown as a fitted ellipse with a bottom line, and fingers are color-coded for visualization.

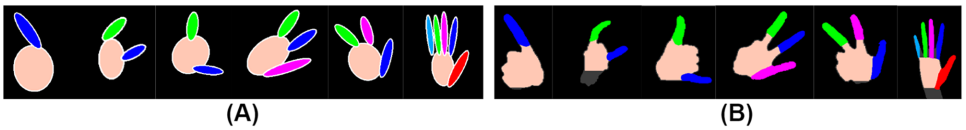


Figure 5: Ellipse-guided geometric finger segmentation: (**A**) Fitted ellipses (**B**) Segmented fingers.

After ellipse-guided finger segmentation, geometric keypoints, shown in Fig. 6A, are extracted from each finger to support pose representation and gesture analysis. Each finger instance (f_i) is represented as the set of pixels belonging to its assigned color label (c_i). The palm center is obtained as the deepest interior point of the binary hand silhouette using the Euclidean distance transform given by Eq. (16).

$$D(x, y) = EDT(B(x, y)) \tag{16}$$

where the pixel corresponding to the maximum distance value is selected as the palm center (x_p, y_p). This point serves as a stable reference for finger direction estimation. For each pixel belonging to a finger instance, the radial distance from the palm center is computed as Eq. (17).

$$d(x, y) = \sqrt{(x - x_p)^2 + (y - y_p)^2} \tag{17}$$

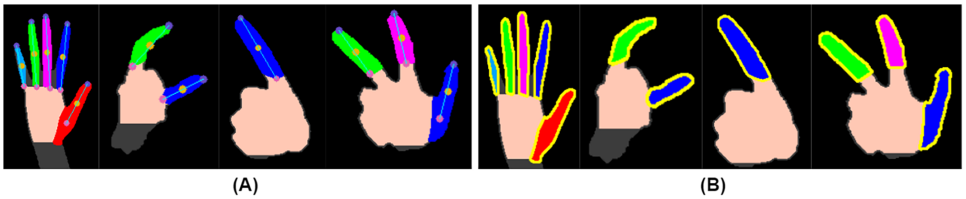


Figure 6: Keypoints extraction and contour analysis: (**A**) Fingertip, base and center of finger (**B**) Contour analysis.

The finger base point is defined as the pixel closest to the palm center while the fingertip is defined as the pixel farthest from the palm center. The middle point is computed as the midpoint between the base and tip, given by Eq. (18).

$$(x_m, y_m) = \left(\frac{x_b + x_t}{2}, \frac{y_b + y_t}{2} \right) \tag{18}$$

To capture finger boundary information, all finger regions are merged into a binary mask and external finger contours $\mathcal{C}_j = \{(x_k, y_k)\}_{k=1}^{N_j}$ are extracted from this mask, as shown in Fig. 6B. This process yields, for each finger, a set of discriminative geometric features consisting of base, center, and tip keypoints, a principal direction vector, and the full finger contour.

3.5 Optical Flow Estimation Using Ef-RAFT

Precise hand motion modeling is obtained with Ef-RAFT [18], an efficient variant of the RAFT, developed for the reliable estimation of dense hand motion during varying speed and appearances. Given

frames I_t and I_{t+1}, it computes dense optical flow $F \in \mathbb{R}^{H\times W\times 2}$, capturing pixel-wise displacement for motion features (Fig. 7). The model includes a feature encoder, context encoder, and iterative update block, where the encoder extracts pixel features and a correlation volume captures pairwise similarity (Eq. (19)).

$$C(\mathbf{x}_i, \mathbf{x}_j) = \langle f(I_t(\mathbf{x}_i)), f(I_{t+1}(\mathbf{x}_j))\rangle \tag{19}$$

where $\langle\cdot,\cdot\rangle$ denotes the dot product. The correlation volume captures pixel-wise similarity between frames, enabling efficient long-range matching. The first frame I_t initializes the hidden state h via the context encoder, and the update block iteratively refines the flow using Eq. (20).

$$F^{k+1} = F^k + \Delta F^k, \Delta F^k = g(F^k, C, h^k) \tag{20}$$

where g is the learned update function and h^k the recurrent hidden state encoding motion context. After K iterations, the model outputs the final flow F^K at full resolution. Flow vectors are represented by magnitude and direction (Eq. (21)), providing rich motion features.

$$v_{i,j} = \sqrt{u_{i,j}^2 + v_{i,j}^2}, \theta_{i,j} = \arctan\frac{v_{i,j}}{u_{i,j}} \tag{21}$$

here, $v_{i,j}$ denotes motion magnitude and $\theta_{i,j}$ its direction at pixel (i, j). These can be used directly or aggregated into temporal descriptors. By focusing on motion, Ef-RAFT remains robust to variations in hand appearance and background, making it promising for practical real-time HRI scenarios.

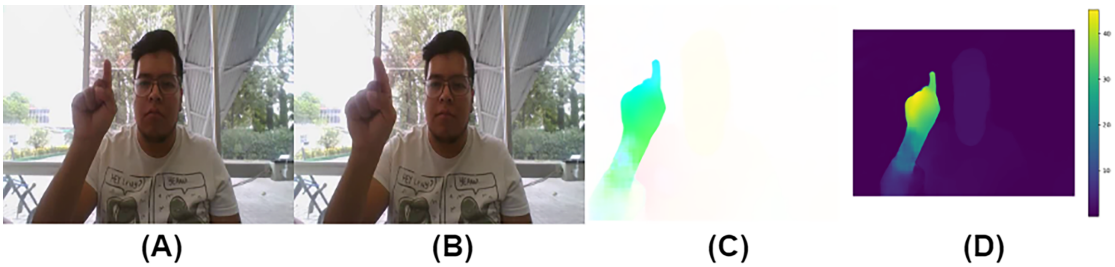


Figure 7: Ef-RAFT optical flow estimation: (**A**) Frame 1 (**B**) Frame 2 (**C**) Ef-Raft results showing the motion between two frames (**D**) Heatmap based on Ef-RAFT results.

3.6 Saliency Map

Visual saliency highlights important regions in hand images for gesture recognition complementing geometric and motion descriptors when the silhouette is degraded or incomplete. It is computed from RGB images within the hand region by converting $I(x, y)$ to 8-bit grayscale, then generating a binary mask $M(x, y)$ via thresholding. The spectral residual method is then used to estimate visually prominent regions. First, the Fourier transform of the image is computed using (Eq. (22)).

$$F(u, v) = \mathcal{F}\{I(x, y)\} \tag{22}$$

Next, the log-amplitude spectrum $L(u, v)$ is obtained, and a locally smoothed version is subtracted to compute the spectral residual using Eq. (23).

$$R(u, v) = L(u, v) - \mathcal{H}(L(u, v)) \tag{23}$$

where $\mathcal{H}(\cdot)$ denotes a local averaging filter. The saliency map, shown in Fig. 8A, is reconstructed in the spatial domain by combining the spectral residual with the original phase information and applying the inverse Fourier transform. Finally, the mask is applied to suppress background responses using Eq. (24)

$$S_f(x, y) = S(x, y) \cdot M(x, y) \tag{24}$$

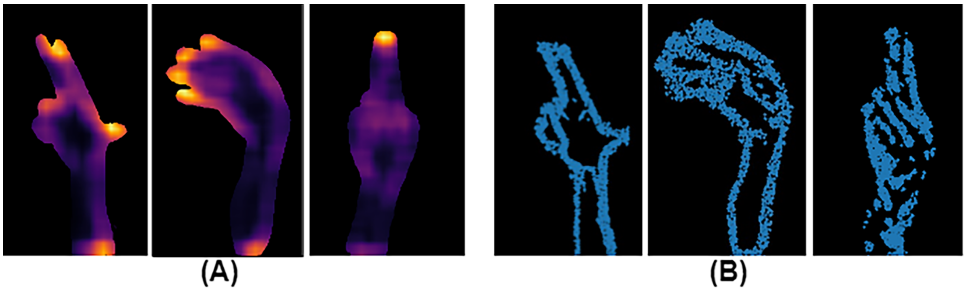


Figure 8: (**A**) Saliency map on various hand gestures (**B**) Ridge detection on different hand gestures.

For visualization, $S_f(x, y)$ is scaled to the range 0–255 and color-mapped to emphasize salient hand regions. The method is unsupervised, computationally efficient, and suitable for robust hand motion analysis.

3.7 Hand Ridge Detection

Geometric structures from ridge points can be found within hand images, such as central lines and folds of fingers, serving as strong feature points for gesture recognition and complement silhouette-based representations under shape deformation. The extraction of ridge points is done based on the use of the Hessian matrix of the hand image $I(x, y)$ that has been Gaussian smoothed. First, the Gaussian smoothing is defined through Eq. (25):

$$\tilde{I}(x, y) = G_\sigma(x, y) * I(x, y) \tag{25}$$

The Hessian matrix is computed at each pixel using Eq. (26).

$$H(x, y) = \begin{bmatrix} I_{xx}(x, y) & I_{xy}(x, y) \\ I_{xy}(x, y) & I_{yy}(x, y) \end{bmatrix} L \tag{26}$$

where I_{xx}, I_{yy}, I_{xy} are the second-order derivatives of the smoothed image. The eigenvalues of H are then calculated as Eq. (27).

$$\lambda_{1,2} = \frac{\text{trace}(H) \pm \sqrt{\text{trace}(H)^2 - 4\det(H)}}{2} \tag{27}$$

Ridge response, shown in Fig. 8B, at each pixel is defined as Eq. (28).

$$R(x, y) = |\min(\lambda_1, \lambda_2)| \tag{28}$$

The response is normalized and thresholded to obtain a binary ridge map, retaining high-response pixels. A fixed number of ridge points are sampled to reduce computation. These points provide interpretable, low-cost geometric features independent of pretrained models, suitable for dynamic gesture analysis.

3.8 Early Feature Fusion

The extracted motion, geometric, and structural features, including saliency, ridge descriptors, octree-based voxel representations, finger keypoints, and optical flow, are fused to form a holistic hand representation. These complementary features capture texture, motion, geometry, saliency, and spatial information, and are embedded into a common space through concatenation to preserve inter-modality dependencies. The resulting feature vector for frame t can be expressed as Eq. (29).

$$F_t = [W_{kp} f_t^{KP} \parallel W_{contour} f_t^{Contour} \parallel W_{ridge} f_t^{Ridge} \parallel W_{motion} f_t^{Motion} \parallel W_{saliency} f_t^{Sal} \parallel W_{voxel} f_t^{Voxel}] \quad (29)$$

where $f_t^{KP}, f_t^{Contour}, f_t^{Ridge}, f_t^{Motion}, f_t^{Sal}$, and f_t^{Voxel} denote the finger keypoints, finger contour, ridge point, motion, saliency, and voxel features, and W. are learnable projections. The fused representation is used for classification or temporal modeling, enabling robust gesture recognition under varying pose, lighting, and motion conditions.

3.9 Transformer-Based Classification

A Transformer encoder models the temporal evolution of fused features across frames, enabling adaptive integration of spatial, geometric, and motion cues for robust gesture recognition. It was selected for its ability to capture long-range temporal dependencies and complex gesture dynamics more effectively than conventional temporal models. Positional encoding is added to preserve spatial and temporal order using Eq. (30).

$$X_{PE} = X + P \quad (30)$$

where X represents the input sequence of merged keypoint features and P denotes the positional encoding; thus, the enriched input X_{PE} will be passed through L layers of stacked Transformer encoders, where every layer operates via multi-head self-attention and derives its own query, key, and value matrices through learned linear projection, as given in Eq. (31).

$$Q = XW^Q, K = XW^K, V = XW^V \quad (31)$$

The self-attention operation computes attention weights using Eq. (32).

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V \quad (32)$$

Outputs from all attention heads are concatenated and linearly projected to produce the final multi-head attention output using Eq. (33).

$$\text{MHA}(X) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O \quad (33)$$

Following the attention mechanism, a position-wise feed-forward network (FFN) processes each token independently, given as Eq. (34).

$$\text{FFN}(x) = \text{ReLU}(xW_1 + b_1)W_2 + b_2 \quad (34)$$

Residual connections and layer normalization are applied after each sub-layer to stabilize training using Eq. (35).

$$Z_1 = \text{LayerNorm}(X + \text{MHA}(X)), Z_2 = \text{LayerNorm}(Z_1 + \text{FFN}(Z_1)) \quad (35)$$

After L encoder layers, the Transformer outputs $Z \in \mathbb{R}^{(T \cdot n) \times d'}$, reshaped to $\mathbb{R}^{T \times n \times d'}$ to align frames and keypoints. Features are pooled and classified via Eq. (36).

$$\hat{y} = \text{softmax}\left(W_c \cdot \text{Pool}(Z) + b_c\right) \tag{36}$$

By modeling spatial–temporal dependencies, Transformer captures dynamic hand patterns for robust gesture recognition [19]. The detailed implementation settings for model training are provided in Table 1. Further, the complete algorithm of HRI model is given below as Algorithm 1.

Table 1: Model training and configuration.

Parameter	Value	Parameter	Value
Architecture	Encoder-Only Transformer	Input Features	Early-Fused Multimodal Feature Vector
Embedding Dimension (d_model)	128	Number of Encoder Layers	2
Number of Attention Heads	4	Feed-Forward Dimension	512
Dropout Rate	0.10	Activation Function	GELU
Positional Encoding	Sinusoidal	Optimizer	Adam
Initial Learning Rate	1×10^{-4}	Weight Decay	1×10^{-5}
Batch Size	32	Training Epochs	100
Early Stopping Patience	10 Epochs	Loss Function	Cross-Entropy
Classification Head	Fully Connected + Softmax		

Algorithm 1: Human-robot interaction model

Input: RGB frame sequence $\{\text{frame}_t\}$
Output: Gesture label
Initialize F ← [ ]
for each frame t **do**
Hand Segmentation
M ← UNET_SEGMENT (frame_t) H ← EXTRACT_HAND(M, frame_t) **if** H = ∅ then continue
Finger Decomposition
S ← BINARY_MASK (H) (c_p, r_p) ← DISTANCE_TRANSFORM(S)
F_m ← REFINE_FINGERS (SEPARATE_FINGERS (S, c_p, r_p))
F_m ← REFINE_FINGERS (SEPARATE_FINGERS (S, c_p, r_p))
K ← EXTRACT_FINGER_POINTS (F_m, c_p) C ← EXTRACT_FINGER_CONTOURS(F_m)
Motion
if t > 0 **then**
Flow ← EFRAFT (prev_frame, frame_t) M_f ← MAGNITUDE_DIRECTION(Flow)

(Continued)

Algorithm 1 (continued)

Structural Features
R ← THICKER_DOTS(EXTRACT_RIDGE_POINTS(H))
Sal ← MASK_BACKGROUND(SPECTRAL_RESIDUAL($frame_t$), M)
3D Representation V ← VOXELIZE(S) O ← BUILD_OCTREE(V, $depth_{max}$)
Feature Fusion f_t ← CONCAT(K, C, M_f, R, Sal, O)
Append F ← F ∪ {f_t} prev_frame ← $frame_t$ **end for**
Temporal Modeling & Classification
X ← STACK(F) X ← ADD_POSITIONAL_ENCODING(X) Z ← TRANSFORMER_ENCODER(X)
gesture_label ← SOFTMAX_CLASSIFIER(Z)
return gesture_label

4 Results and Discussion

4.1 Experimental Setup

The experiments were performed on NVIDIA Tesla T4 GPU running with Python 3.10.13, PyTorch 2.1.0 (CUDA 11.8, cuDNN 8.9.1) and other standard packages for data processing and visualization

4.2 Computational Cost

Table 2 gives the computational complexity (i.e., execution time and MFLOPS) for each module. Although optical flow and geometric feature detection slightly increase the cost, the inclusion of saliency and ridge adds almost no overhead, while retaining the balance of efficiency and robustness.

Table 2: Computational time and MFLOPS of each component in the proposed model.

Technique		Execution Time	MFLOPS	Technique	Execution Time	MFLOPS
DL Hand segmentation		0.0151 s	3601.64	Optical flow	0.3759 s	30,000.00
Octree-based voxelization		0.4900 s	36.026	Hand Ridge	0.0701 s	38.640
EG-FSKE	Finger segmentation	0.3237 s	4.302	Saliency Map	0.0012 s	30.912
	Keypoints Extraction	0.8822 s	4.260	Transformer	0.0385 s	286.4
	Contour Extraction	0.3607 s	0.985			

4.3 Dataset Description

The three datasets were used to evaluate the model performance which include IPN Hands [20], having 13 classes with fine motion classification, Jester [21] having 25 classes with varying performance and lighting conditions, and EgoGesture [22] that has more than 4000 classes of egocentric RGB-D sequences.

4.4 Confusion Matrix

The confusion matrices of Jester, IPN Hand and EgoGesture are shown in Fig. 9. The diagonal dominance is high, suggesting high classification accuracy. Small errors are caused by gesture similarity, occlusion, low contrast and overlap; and EgoGesture performance is the most consistent with 83 classes.

4.5 Discussion

The performances for all the datasets are presented in Table 3. With cluttered conditions, IPN Hand exhibits stable performance, whereas Jester provides better performance with reliable confidence bounds.

The highest performance is obtained on EgoGesture, which indicates the robustness and consistency of the proposed model in varied gesture scenarios.

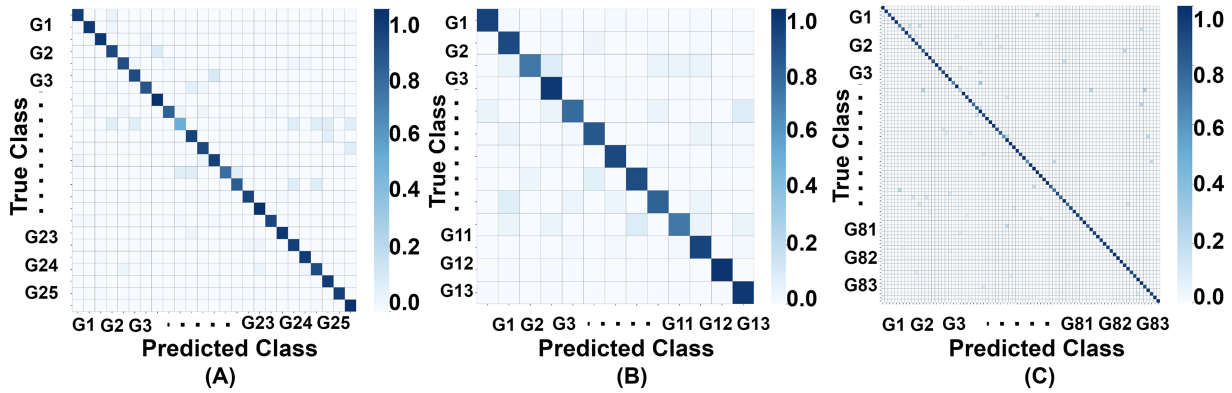


Figure 9: Results on three datasets in terms of confusion matrices: (**A**) Jester; (**B**) IPN Hand; (**C**) EgoGesture.

Table 3: Performance of proposed framework across different datasets.

Dataset	Accuracy (%)	Accuracy 95% CI (%)	Precision (Mean ± SD)	Precision 95% CI	Recall (Mean ± SD)	Recall 95% CI	F1-Score (Mean ± SD)	F1-Score 95% CI
IPN Hand	92.34 ± 0.68	[91.50–93.18]	0.880 ± 0.0120	[0.855–0.885]	0.867 ± 0.0133	[0.854–0.887]	0.871 ± 0.0140	[0.853–0.887]
Jester	93.22 ± 0.45	[92.66–93.78]	0.892 ± 0.0260	[0.858–0.922]	0.901 ± 0.0284	[0.865–0.935]	0.893 ± 0.0270	[0.857–0.924]
EgoGesture	94.17 ± 0.52	[93.52–94.82]	0.921 ± 0.0100	[0.908–0.932]	0.917 ± 0.0110	[0.906–0.934]	0.913 ± 0.0122	[0.895–0.925]

The ROC curves in Fig. 10 shows good AUC≈1 for Jester, IPN Hand and EgoGesture towards the upper left hand corner, indicating good discrimination ability. There is a slight overlap because of similarity in gestures, motion overlapping, cluttering and occlusions. In general, the system provides promising results in different environments.

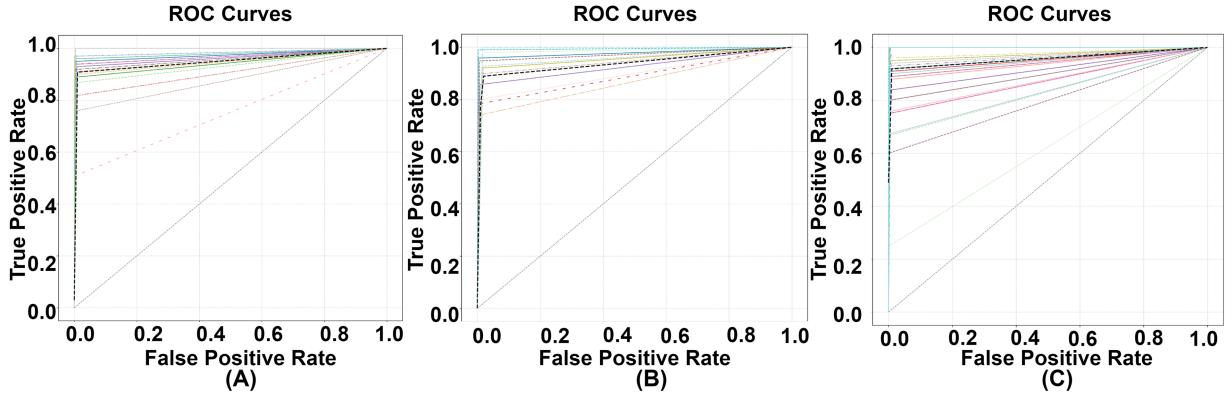


Figure 10: Results on three datasets in terms of ROC Curves: (**A**) Jester; (**B**) IPN Hand; (**C**) EgoGesture.

Beyond benchmark evaluation, the proposed framework shows strong potential for real-world human–robot interaction. Z-HandSegNet enables reliable hand localization in cluttered scenes, while EG-FSKE provides interpretable geometric representations for transparent gesture-based communication. In addition, octree-based volumetric encoding and motion-aware features capture rich spatial–temporal information for robotic interaction tasks. Future work will focus on real-time deployment and integration into closed-loop robotic systems.

4.5.1 Explainability and Interpretability of the Proposed Framework

One important benefit of the proposed approach is its interpretation capability compared to the end-to-end deep learning systems. Rather than depending only on latent representations, the proposed technique creates interpretable outputs like hand masks, palm detection, fingers segmentation, keypoints, ridge/saliency maps, and volume representations. Visualization of these elements facilitates traceable identification and anatomically meaningful predictions., improving transparency and reliability in human–robot interaction applications.

4.5.2 Performance Evaluation under Occlusion, Finger Overlapping and Dynamic Environment

To evaluate robustness under occlusion, artificial occlusions covering ~20% of the hand region were introduced during testing. Although accuracy decreased by 5%–12%, the framework remained effective due to multimodal fusion, where volumetric, saliency, ridge, and motion features compensated for degraded geometric segmentation. Overall, the integrated framework remained robust under challenging conditions as illustrated in Table 4.

Table 4: Performance evaluation under challenging environments.

Dataset	Original Accuracy (%)	20% Occlusion Accuracy (%)	Drop (%)
Jester	93.22	86.30	6.92
IPN Hand	92.34	80.10	12.24
EgoGesture	94.17	86.40	7.77

Furthermore, the proposed framework was evaluated on three publicly available datasets, namely Jester, IPN Hand, and EgoGesture, which were captured using different cameras, viewpoints, backgrounds, illumination conditions, and gesture execution styles. Despite these variations, the framework consistently achieved accuracies of 93.22%, 92.34%, and 94.17%, demonstrating strong robustness and generalization across diverse real-world capture environments.

The proposed Transformer encoder outperformed other temporal models (RNN, CNN, LSTM, GNN) as shown in Table 5, when tested on the Jester dataset. The results confirmed that the Transformer encoder has better recognition accuracy, indicating its superior performance in capturing long-term dependencies and complex dynamics of gestures.

Table 5: Comparison of transformer with other models.

Dataset	RNN	CNN	LSTM	GNN	Transformer (Proposed)
Jester (%)	77.40	81.80	70.90	86.80	93.22

4.6 Ablation Study

The ablation analysis on three datasets is presented in Table 6 with full model accuracies of 93.22%, 92.34%, and 94.17%. Removing the adaptive octree-based 3D representation causes the most significant drop (~14%–17%), followed by EG-FSKE (~11%–15%). This highlighs the importance of geometric and volumetric modeling. Ef-RAFT provides moderate gains (~5%–7%), while saliency and ridge features offer complementary improvements (~2%–4%). Overall, each module contributes towards the model performance and robustness under challenging conditions.

Table 6: Ablation study for different model configuration performance (accuracy) across three datasets.

Component	Jester	IPN	Ego Gesture	Component	Jester	IPN	Ego Gesture
Full Model	93.22%	92.34%	94.17%	Without EG-FSKE	81.38%	78.98%	77.32%
Without Preprocessing	88.22%	89.10%	90.95%	Without Optical Flow	83.59%	81.35%	80.56%
Without Voxel Based 3D Reconstruction	79.24%	75.27%	81.99%	Without Saliency Detection	88.12%	86.89%	89.68%
				Without Hessian Based Ridge	87.93%	86.12%	90.12%

4.7 Comparison with Other Methods

Comparison of the proposed work with recent lightweight CNN-LSTM, skeleton-based, dynamic image-based, and attention-based approaches for Jester [23–28], IPN Hand [29–33], and EgoGesture [23,25,34–37] datasets is shown in Table 7, based on prior studies. The results show that our model achieves the best accuracies of 93.22%, 92.34%, and 94.17%, respectively, demonstrating robust performance across diverse gesture recognition scenarios.

Table 7: Comparison of proposed model with other state-of-the-art methods.

Jester		IPN Hand		EgoGesture	
Author	**Accuracy**	**Author**	**Accuracy**	**Author**	**Accuracy**
Alonazi et al. [23]	89.33%	Leite et al. [29]	87.20%	Alonazi et al. [23]	90.73%
Truong et al. [24]	90.19%	Mustafa et al. [30]	89.92%	Alabdullah et al. [25]	91.86%
Alabdullah et al. [25]	91.57%	Benitez-Garcia et al. [31]	91.20%	Ansar and Jalal [34]	92.67%
Wu et al. [26]	92.50%	Benitez-Garcia et al. [31]	86.2%	Jain et al. [35]	92.91%
Yang et al. [27]	92.31%	Benitez-Garcia et al. [31]	89.9%	Cui et al. [36]	93.20%
Wang et al. [28]	89.25%	Hampiholi et al. [32]	91.4%	Karsh et al. [37]	91.64%
		Jena et al. [33]	85.16%		
Proposed	93.22%	Proposed	92.34%	Proposed	94.17%

Note: The three results reported from Benitez-Garcia et al. [31] correspond to different evaluation settings. Specifically, 91.2% refers to training and testing on IPN HandS annotations, 86.2% refers to training on MediaPipe Hands and testing on MediaPipe Holistic annotations, and 89.9% refers to training on MediaPipe Holistic and testing on MediaPipe Hands annotations.

5 Limitations and Future Work

While efficient in terms of computation in many stages, the overall multi-stage approach poses an inference overhead, rendering the system unsuitable for fully real-time HRI application. Moreover, EG-FSKE shows limited performance under severe hand occlusion and silhouettes of poor quality. Further, the problem of cross-dataset generalization under unseen environments has not been explored because of differences in gestures used and annotation procedures across the benchmarks Future research will focus on lightweight optimization techniques, occlusion robust feature extraction methods, and cross-domain generalization under unified evaluation settings.

6 Conclusion

In this paper, we propose a HGR system using Z-HandSegNet, EG-FSKE, adaptive octree representation, Ef-RAFT optical flow, and Transformer for temporal modeling. The model shows strong performance on Jester dataset (93.22%), IPN Hand dataset (92.34%) and Ego-Gesture dataset (94.17%), demonstrating effective recognition under diverse ego-centric HRI scenarios.

Acknowledgement: This work was supported through Princess Nourah bint Abdulrahman University Researchers Supporting Project, Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Funding Statement: This work was supported through Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2026R303), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Author Contributions: Study conception and design: Zarnab Kausar and Shaheryar Najam; data collection: Bayan Alabdullah and Hadeel Alsolai; analysis and interpretation of results: Ahmad Jalal, Shaheryar Najam and Fatimah Alhayan; draft manuscript preparation: Ahmad Jalal, Shaheryar Najam and Hui Liu. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: All publicly available datasets are used in the study.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Kausar Z, Najam S, Alshehri M, AlQahtani Y, Alshahrani A, Jalal A, et al. Two-hand static and dynamic Arabic sign language recognition using keypoints and shape descriptors with attention-driven feature fusion. PeerJ Comput Sci. 2025;11(17):e3275. doi:10.7717/peerj-cs.3275.
2. Wang S, Laforet J, Rida I, Druart B, Becheler A, Istrate D, et al. A comprehensive survey of multi-modal manipulative hand gesture recognition: sensors, datasets, and models. J Vis Commun Image Represent. 2026;119(1):104840. doi:10.1016/j.jvcir.2026.104840.
3. Oudah M, Al-Naji A, Chahl J. Hand gesture recognition based on computer vision: a review of techniques. J Imaging. 2020;6(8):73. doi:10.3390/jimaging6080073.
4. Sharma S, Singh S. Vision-based hand gesture recognition using deep learning for the interpretation of sign language. Expert Syst Appl. 2021;182(3):115657. doi:10.1016/j.eswa.2021.115657.
5. Mujahid A, Awan MJ, Yasin A, Abed Mohammed M, Damaševičius R, Maskeliūnas R, et al. Real-time hand gesture recognition based on deep learning YOLOv3 model. Appl Sci. 2021;11(9):4164. doi:10.3390/app11094164.
6. Athira KA, Divya Udayan J. Temporal fusion of time-distributed VGG-16 and LSTM for precise action recognition in video sequences. Procedia Comput Sci. 2024;233(5):892–901. doi:10.1016/j.procs.2024.03.278.
7. Rahimian E, Zabihi S, Asif A, Farina D, Atashzar SF, Mohammadi A. Hand gesture recognition using temporal convolutions and attention mechanism. In: Proceedings of the ICASSP 2022—2022 IEEE International Conference

on Acoustics, Speech and Signal Processing (ICASSP); 2022 May 23–27; Singapore, Singapore. p. 1196–200. doi:10.1109/icassp43922.2022.9746174.
8. Li HH, Hsieh CC. Dynamic hand gesture recognition using MediaPipe and transformer. Eng Proc. 2025;108(1):22. doi:10.3390/engproc2025108022.
9. Li J, Al-qaness MAA. SG-DGCN: semantic-guided dynamic graph convolutional network for skeleton-based gesture recognition. Knowl Based Syst. 2026;347(9):116254. doi:10.1016/j.knosys.2026.116254.
10. Avola D, Cinque L, Fagioli A, Foresti GL, Fragomeni A, Pannone D. 3D hand pose and shape estimation from RGB images for keypoint-based hand gesture recognition. Pattern Recognit. 2022;129(8):108762. doi:10.1016/j.patcog.2022.108762.
11. Dang R, Feng G. Hand gesture 3D pose estimation method based on swin transformer and CNN. Sci Rep. 2026;16(1):11551. doi:10.1038/s41598-026-41974-6.
12. Kim DU, In Kim K, Baek S. End-to-end detection and pose estimation of two interacting hands. In: Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV); 2021 Oct 10–17; Montreal, QC, Canada. p. 11169–78. doi:10.1109/iccv48922.2021.01100.
13. Peral M, Sanfeliu A, Garrell A. Efficient hand gesture recognition for human-robot interaction. IEEE Robot Autom Lett. 2022;7(4):10272–9. doi:10.1109/LRA.2022.3193251.
14. Kollakidou A, Haarslev F, Odabasi C, Bodenhagen L, Krüger N. HRI-gestures: gesture recognition for human-robot interaction. In: Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications; 2022 Feb 6–8; Online. p. 559–66. doi:10.5220/0010871200003124.
15. Qi W, Ovur SE, Li Z, Marzullo A, Song R. Multi-sensor guided hand gesture recognition for a teleoperated robot using a recurrent neural network. IEEE Robot Autom Lett. 2021;6(3):6039–45. doi:10.1109/LRA.2021.3089999.
16. Duan S, Wu L, Liu A, Chen X. A global–local fusion model exploring temporal–spatial dependence for multimodal hand gesture recognition. IEEE Trans Med Robot Bionics. 2025;7(2):723–33. doi:10.1109/tmrb.2025.3550646.
17. Deng Y, Li F, Xin H. Research on multimodal human-robot interaction based on speech and gesture. Comput Electr Eng. 2018;72(5–7):443–54. doi:10.1016/j.compeleceng.2018.09.014.
18. Eslami N, Arefi F, Mansourian AM, Kasaei S. Rethinking RAFT for efficient optical flow. In: Proceedings of the 13th Iranian/3rd International Machine Vision and Image Processing Conference (MVIP); 2024 Mar 6–7; Tehran, Iran. p. 1–7. doi:10.1109/mvip62238.2024.10491183.
19. Liu Z, Ning J, Cao Y, Wei Y, Zhang Z, Lin S, et al. Video swin transformer. In: Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2022 Jun 18–24; New Orleans, LA, USA. p. 3192–201. doi:10.1109/cvpr52688.2022.00320.
20. Benitez-Garcia G, Olivares-Mercado J, Sanchez-Perez G, Yanai K. IPN hand: a video dataset and benchmark for real-time continuous hand gesture recognition. In: Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR); 2021 Jan 10–15; Milan, Italy. p. 4340–7. doi:10.1109/icpr48806.2021.9412317.
21. Materzynska J, Berger G, Bax I, Memisevic R. The jester dataset: a large-scale video dataset of human gestures. In: Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW); 2019 Oct 27–28; Seoul, Republic of Korea. p. 2874–82. doi:10.1109/iccvw.2019.00349.
22. Zhang Y, Cao C, Cheng J, Lu H. EgoGesture: a new dataset and benchmark for egocentric hand gesture recognition. IEEE Trans Multimed. 2018;20(5):1038–50. doi:10.1109/TMM.2018.2808769.
23. Alonazi M, Ansar H, Al Mudawi N, Alotaibi SS, Almujally NA, Alazeb A, et al. Smart healthcare hand gesture recognition using CNN-based detector and deep belief network. IEEE Access. 2023;11:84922–33. doi:10.1109/access.2023.3289389.
24. Truong TD, Bui QH, Duong CN, Seo HS, Phung SL, Li X, et al. DirecFormer: a directed attention in transformer approach to robust action recognition. In: Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2022 Jun 18–24; New Orleans, LA, USA. p. 19998–20008. doi:10.1109/cvpr52688.2022.01940.
25. Alabdullah BI, Ansar H, Al Mudawi N, Alazeb A, Alshahrani A, Alotaibi SS, et al. Smart home automation-based hand gesture recognition using feature fusion and recurrent neural network. Sensors. 2023;23(17):7523. doi:10.3390/s23177523.

26. Wu J, Sun W, Gan T, Ding N, Jiang F, Shen J, et al. Neighbor-guided consistent and contrastive learning for semi-supervised action recognition. IEEE Trans Image Process. 2023;32:2215–27. doi:10.1109/TIP.2023.3265261.
27. Yang J, Dong X, Liu L, Zhang C, Shen J, Yu D. Recurring the transformer for video action recognition. In: Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2022 Jun 18–24; New Orleans, LA, USA. p. 14043–53. doi:10.1109/cvpr52688.2022.01367.
28. Wang Z, Wu Z, Qi R, Dou X. Lightweight visual dynamic gesture recognition system based on CNN-LSTM-DSA. Sensors. 2026;26(5):1558. doi:10.3390/s26051558.
29. Leite C, Byvshev P, Mauranen H, Xiao Y. Simulation-driven design of smart gloves for gesture recognition. Sci Rep. 2024;14(1):14873. doi:10.1038/s41598-024-65069-2.
30. Mustafa Z, Nsour H, Tahir SBUD. Hand gesture recognition via deep data optimization and 3D reconstruction. PeerJ Comput Sci. 2023;9(21):e1619. doi:10.7717/peerj-cs.1619.
31. Benitez-Garcia G, Olivares-Mercado J, Sanchez-Perez G, Takahashi H. IPN HandS: efficient annotation tool and dataset for skeleton-based hand gesture recognition. Appl Sci. 2025;15(11):6321. doi:10.3390/app15116321.
32. Hampiholi B, Jarvers C, Mader W, Neumann H. Convolutional tsssransformer fusion blocks for multi-modal gesture recognition. IEEE Access. 2023;11:34094–103. doi:10.1109/ACCESS.2023.3263812.
33. Jena A, Biswas S, Nandy A. Enhancing hand gesture recognition using entropy-based deep neural architecture. Sādhanā. 2025;50(1):37. doi:10.1007/s12046-025-02697-y.
34. Ansar H, Jalal A. Robust hand gesture tracking and recognition for healthcare via recurrent neural network. In: Proceedings of the 20th International Bhurban Conference on Applied Sciences and Technology (IBCAST); 2023 Aug 22–25; Murree, Pakistan. p. 148–53. doi:10.1109/IBCAST59916.2023.10712949.
35. Jain R, Karsh RK, Barbhuiya AA. Linked motion image-based dynamic hand gesture recognition. Comput Animat Virtual. 2023;34(6):e2137. doi:10.1002/cav.2137.
36. Cui C, Sunar MS, Eg Su G. Deep vision-based real-time hand gesture recognition: a review. PeerJ Comput Sci. 2025;11(6):e2921. doi:10.7717/peerj-cs.2921.
37. Karsh B, Laskar RH, Karsh RK. mXception and dynamic image for hand gesture recognition. Neural Comput Appl. 2024;36(15):8281–300. doi:10.1007/s00521-024-09509-0.