ISSN 2616-5775 Vol. 9, Issue 8: 69-78, DOI: 10.25236/AJCIS.2026.090809

3D Hand Pose Estimation Based on Monocular RGB Image Sequences

Ruihang Li[1,a], Dongming Yan[1,b,*]

[1]*College of Electrical and Control Engineering, Jilin University of Chemical Technology, Jilin, China*
[a] *liruihang1@jluct.edu.cn*, [b] *yandongming@jluct.edu.cn*

Abstract: *Occlusion, image noise, and insufficient local joint information remain challenges in 3D hand pose estimation from monocular RGB image sequences. To address them, we propose BFHand, which extracts hand joint heatmaps from consecutive frames and estimates 3D pose from multi-frame 2D joint representations. For global feature modeling, BFHand combines ModernTCN and multi-head self-attention to capture temporal motion, while a multi-frame GraphMLP models joint relationships across frames. For the target frame, a context-aware mixture-of-experts module (CAMoE) uses context-driven routing and feature modulation to combine the outputs of multiple GCN experts. EnhGraphMLP then selectively strengthens local joint features. On the HanCo dataset, BFHand achieves an MPJPE of 12.447 mm and a PA-MPJPE of 5.504 mm. The results indicate improved estimation accuracy, with more plausible hand structures and greater temporal stability.*

Keywords: *3D hand pose estimation, GraphMLP, CAMoE, ModernTCN, temporal attention mechanism*

1. Introduction

3D hand pose estimation aims to infer the pose of a hand in 3D space from images or videos. Existing approaches mainly use monocular RGB images, multi-view images, or RGB-D data. RGB-D sensors provide depth information and have been used for hand pose estimation and robotic teleoperation[1]. Multi-view methods draw on complementary observations to reduce the effects of occlusion and complex hand poses[2]. Such observations generally require additional hardware and more demanding deployment. Monocular RGB methods rely only on a conventional camera, making them easier to deploy on mobile devices and in open human–computer interaction settings. Yet RGB images provide no direct depth measurements, so reliably inferring 3D hand pose from 2D visual information remains challenging.

A single image provides limited spatial information, while the hand has many joints, substantial freedom of movement, and closely spaced fingers. Self-occlusion, rapid motion, or low image resolution can obscure local visual cues, causing joint displacement, incorrect finger flexion, and discontinuities between predicted poses. Previous studies have explored high-resolution features, 2D pose cues, hand mesh reconstruction, and pose regression[7]. Single-frame methods, however, cannot fully use the temporal continuity of hand motion and remain sensitive to transient noise and local occlusion.

Temporal information from image sequences can help stabilize 3D hand pose estimation. Visible regions in neighboring frames may provide information missing from the current frame. Researchers have modeled hand motion in videos using motion priors, hierarchical temporal modeling, dynamic temporal modeling, and trajectory prediction[13]-[16]. The hand skeleton also has defined anatomical connections that graph neural networks can use to propagate information between joints. GraphMLP combines graph-based modeling with global feature interaction, offering another approach to 3D pose estimation [17]. How to use temporal context while retaining local joint details in the target frame remains an open question for monocular RGB 3D hand pose estimation.

To address this question, we propose BFHand (Base-Focus Hand Pose Estimator), a network for 3D hand pose estimation. It extracts heatmaps for 21 hand joints from RGB images and uses 2D joint representations from eight consecutive frames to estimate 3D pose. ModernTCN[18], multi-head self-attention, and multi-frame GraphMLP [17] combine motion information with joint relationships across frames to produce a global spatiotemporal representation. CAMoE and EnhGraphMLP then modulate and strengthen local joint features in the final, target frame before the network regresses the coordinates of 21 3D joints. BFHand is designed to improve estimation accuracy, structural plausibility, and temporal stability in monocular RGB videos.

ISSN 2616-5775 Vol. 9, Issue 8: 69-78, DOI: 10.25236/AJCIS.2026.090809

The main contributions of this work are as follows:

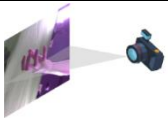
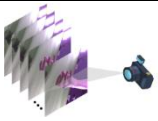
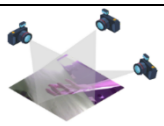
- We propose BFHand, a 3D hand pose estimation framework that combines global information from multiple frames with local features from the target frame. By jointly using temporal context, hand skeleton topology, and target-frame features, it aims to improve robustness to occlusion, image noise, and complex hand motion.
- For global feature modeling, we combine ModernTCN, multi-head self-attention, and multi-frame GraphMLP to integrate motion information and joint relationships across frames. This produces a global spatiotemporal representation for target-frame prediction.
- For local feature enhancement, we design CAMoE and EnhGraphMLP. Context-aware expert routing and feature modulation adapt joint relationship modeling to different hand poses and strengthen local joint features in the target frame.

The rest of this paper is organized as follows. Section 2 reviews related work on monocular 3D hand pose estimation, temporal modeling in videos, and graph neural networks. Section 3 describes the BFHand architecture and its components. Section 4 presents the dataset, experimental setup, evaluation metrics, comparisons, and ablation studies. Section 5 concludes the paper and discusses its limitations and directions for future work.

2. Related Work

This section reviews research on 3D hand pose estimation from monocular RGB image sequences. It compares single-frame RGB, image-sequence, multi-view, and other sensor-based methods in terms of their approaches, practical advantages, and limitations, as summarized in Table 1.

Table 1. Comparative analysis of related studies from the past five years.

Method		Advantageas	Disadvantages	Ref.
 Monocular Single Frame		Easy to deploy Low cost	Self occlusion sensitivity Data scarcity generalization ability Computational complexity	[9] [10] [11] [19] [20]
 Image Sequence		Anti occlusion Low cost Easy to deploy Temporal coherence	Data scarcity Computational complexity	[13] [14] [15] [16] [21]
 Multi Camera View		Occlusion robustness Generalization ability Anti occlusion	Difficult to deploy High cost Computational complexity	[2] [3] [4] [5] [6]
Others	 RGB-D	Illumination robustness Contains 3D information	High cost Missing detail texture	[1] [22]
	 Event Stream	Illumination robustness High time resolution Low motion blur	High cost Missing detail texture Difficulty in marking	[23] [24]
	 Lidar	Low computing costs Illumination robustness 360 ℃inspection	High cost Missing detail texture	[25]

a) Hand pose estimation from single monocular RGB images

Early monocular RGB methods typically used a single image as input and extracted local hand features with convolutional neural networks or high-resolution representations [7][8][12]. With the development of Transformer-based methods and hand mesh reconstruction, researchers have increasingly used global feature interactions to model hand structure and improve predictions for complex poses [11][19][20]. Zhang et al. [9] used 2D pose cues to assist 3D hand pose and shape prediction, while Zhou et al. [10] used multi-scale heatmaps to constrain 3D mesh regression. Single-frame methods, however, rely

ISSN 2616-5775 Vol. 9, Issue 8: 69-78, DOI: 10.25236/AJCIS.2026.090809

on visual cues in the current image. Under occlusion, motion blur, or missing local texture, limited depth information can lead to joint pose errors and discontinuities between frames.

b) Hand pose estimation from monocular RGB image sequences

Compared with a single image, RGB image sequences provide continuous motion cues that can help stabilize 3D hand pose estimation. Existing methods use temporal attention, motion priors, dynamic temporal modeling, and trajectory prediction to capture this information. Wen et al. [15] used a hierarchical structure to model short-term motion and longer-term pose changes. Duran et al. [14]introduced hand motion priors to encourage plausible predictions. Ren et al. [13]modeled fast hand motion through dynamic temporal modeling, while Ye et al. [16] studied the prediction of continuous hand trajectories. Liu et al. [21] also modeled hand–object interactions in video using hand motion and interaction regions. Although sequence information can improve stability, there is still room to better integrate global temporal features with local joint features in the target frame. Fine pose details may remain difficult to preserve during occlusion or rapid motion.

c) Hand pose estimation from multiple camera views

Multi-view methods use complementary observations from different cameras to support 3D hand pose estimation. Previous studies have modeled relationships across views through cross-view matching and pose regression [5], spatiotemporal and spatial–angular representations [6], multi-view consistency learning [2], and consistency across camera coordinate systems [3]. Other work combines geometric relationships with visual appearance for pose prediction [4]. These methods provide richer observations, but they typically require multiple cameras, synchronization, and calibration, which can increase deployment costs.

d) Hand pose estimation using other sensors

In addition to RGB images, RGB-D sensors, event cameras, and LiDAR have been used for hand pose estimation. RGB-D methods use depth observations to improve representations of local joints and hand structure [1][22]. The high temporal resolution and low latency of event cameras make them suitable for capturing hand poses during rapid motion [23][24]. LiDAR has also been used for gesture sensing under challenging lighting conditions [25]. These sensors provide complementary information, although equipment costs, data formats, and operating conditions may limit their use.

3. Method

Inspired by the “overview first, focus later” strategy of OverLoCK [26], we propose BFHand, a network for 3D hand pose estimation from monocular RGB image sequences. LSNetPose2D, built upon LSNet [27], extracts hand joint heatmaps, which are encoded across frames using 3D convolutions. BFHand then combines ModernTCN, multi-head self-attention, and multi-frame GraphMLP to integrate motion information and cross-frame joint relationships into a global spatiotemporal representation. For the final frame, a local feature enhancement branch uses CAMoE to dynamically weight the outputs of multiple GCN experts, while EnhGraphMLP modulates local joint features with global context. The overall architecture of BFHand is shown in Fig. 1.

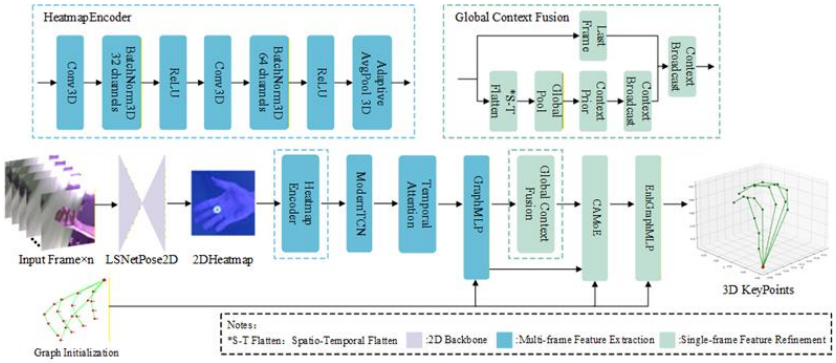


Fig.1. BFHand Architecture with Multi-Frame Fusion and Target-Frame Enhancement.

3.1. Global Spatiotemporal Feature Modeling

BFHand uses LSNetPose2D, built upon LSNet [27], to extract heatmaps containing 21 joint channels

ISSN 2616-5775 Vol. 9, Issue 8: 69-78, DOI: 10.25236/AJCIS.2026.090809

from each frame. After two 3D convolutional layers, adaptive average pooling, and linear projection, frame-level feature tensors are obtained. Because occlusion or rapid motion may shift local responses in a single-frame heatmap, the model further integrates information from adjacent frames along the temporal dimension. Temporal feature fusion adopts the large-kernel depthwise convolution and feed-forward mapping design of ModernTCN[18], with adjustments for the 8-frame input used in this study. Let the frame-level features output by the heatmap encoder be$X \in \mathbb{R}^{B\times 8\times 256}$. The model applies channel-wise 1D convolutions with kernel sizes of 9 and 3 in parallel along the temporal dimension, adds and normalizes the two responses, and then feeds them into the channel feed-forward and temporal feed-forward branches, respectively. Denoting this convolutional process as $D(.)$ and the two feed-forward mappings as $F_c(\cdot)$ and $F_t(\cdot)$, the temporal feature update can be expressed as follows:

$$\boldsymbol{Z} = \boldsymbol{X} + F_c(D(\boldsymbol{X})) + F_t(D(\boldsymbol{X})), \boldsymbol{H} = \boldsymbol{Z} + \mathrm{MHA}_4(\boldsymbol{Z}). \tag{1}$$

here, Z denotes the intermediate feature obtained by fusing the convolutional and feed-forward branches. MSA_4 represents four-head self-attention, which uses Z to construct the queries, keys, and values, yielding H with further integrated cross-frame information. The two residual additions preserve the original frame-level responses and the convolution-fused representations, respectively, enabling the attention module to establish cross-frame dependencies based on the existing temporal cues. The branch with a kernel size of 9 provides a broader temporal receptive field, whereas the branch with a kernel size of 3 focuses on changes between neighboring frames.To incorporate hand structure, the model replicates each 256-dimensional vector in H across the 21 joints and rearranges the resulting features after a shared projection, obtaining a tensor of size $B \times 168 \times 256$. The graph nodes are indexed by frame and joint. In addition to self-connections, joints within the same frame are connected according to the hand skeleton, while corresponding joints in adjacent frames are also connected.

The GraphMLP module follows the idea of parallel MLP and graph convolution [17]. Let $\boldsymbol{A}_k$ denote the k-th normalized sub-adjacency matrix and $\boldsymbol{W}_k^l$ the corresponding channel transformation. A single residual sublayer can be expressed as follows:

$$\boldsymbol{F}^{l+1} = \boldsymbol{F}^l + \mathrm{MLP}(\mathbf{U}) + \sum_{k=1}^{K} \boldsymbol{A}_k^{\top} \boldsymbol{U}\boldsymbol{W}_k^l, \boldsymbol{U} = \mathrm{LN}(\mathbf{F}^l) \tag{2}$$

the summation term aggregates information from each subgraph neighborhood, while the MLP mixes features along the node or channel dimension. The GraphMLP module consists of two sequential sublayers: the first applies node-wise normalization and a token MLP, and the second applies channel-wise normalization and a channel MLP. Each sublayer also includes a graph convolution.

The output is reshaped to $B \times 8 \times 21 \times 256$ and averaged across all frames and joints. Two linear mappings followed by a sigmoid function then produce channel weights. These weights modulate the joint features of the final frame channel by channel. CAMoE and EnhGraphMLP further refine the features, which are used to regress the 3D joint coordinates of the target frame.

3.2. Context-Aware Mixture-of-Experts Mechanism

Global spatiotemporal modeling integrates multi-frame motion information with joint topology. However, joint dependencies in the target frame vary with hand pose, occlusion, and motion state. To account for these variations, we design a context-aware mixture-of-experts mechanism (CAMoE), which adaptively combines the outputs of multiple GCN experts based on global context to refine local joint features in the target frame, as shown in Fig. 2.

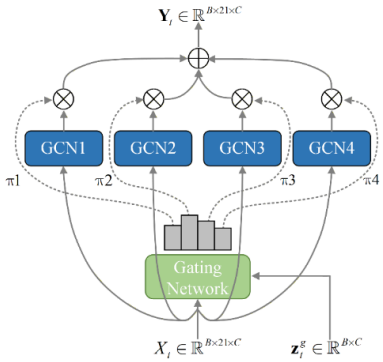


Fig. 2. Context-Aware GCN Expert Fusion Module.

ISSN 2616-5775 Vol. 9, Issue 8: 69-78, DOI: 10.25236/AJCIS.2026.090809

For the input joint features of frame *t*, $X_t \in \mathbb{R}^{B\times 21\times C}$, CAMoE contains four parallel GCN experts. The gating network takes the local joint features $X_{t,j}$ and global context $\boldsymbol{z}_t^g$ as inputs, and computes the corresponding expert weights using temperature-scaled softmax, as shown in Fig. 2.

$$\pi_{t,j} = \text{Softmax}\big(W_g[X_{t,j}; c] + b_g\big), \sum_{k=1}^{4} \pi_{t,j}^{(k)} = 1 \tag{3}$$

each expert then applies graph convolution in parallel to the same single-frame hand graph, and their outputs are fused according to the gate weights:

$$Y_{t,j} = \sum_{k=1}^{4} \pi_{t,j}^{(k)}\, GCN_k(X_t)_j, Y_t \in \mathbb{R}^{B\times 21\times C} \tag{4}$$

here, $\pi_{t,j}^{(k)}$ denotes the gating weight assigned to the *k*-th GCN expert for joint *j* in frame *t*. Each expert performs graph convolution on the same frame-level joint features X_t, producing a response for joint *j*. CAMoE computes a weighted sum of the four expert responses to obtain the fused joint feature s$Y_{t,j}$. The temperature-scaled softmax gate assigns continuous weights to all experts, combining their responses through soft assignment rather than selecting only a subset, as in sparse gating. For short-sequence hand pose estimation, this design helps maintain smooth joint-feature propagation and adapt to different poses and motion states[28][29].

After expert fusion, the joint features are passed to EnhGraphMLP, a global-local modulation enhancement module built on GraphMLP. EnhGraphMLP derives channel-wise modulation factors from the global context and applies them to the graph-convolution branch responses, allowing the target-frame joint features to adapt to the sequence's overall motion state. An MLP branch then further updates the joint features, enhancing local channel representations while preserving information propagation over the hand topology. By combining CAMoE expert fusion with EnhGraphMLP context modulation, the model refines target-frame local joint features under the guidance of global spatiotemporal information, providing more discriminative joint-level representations for subsequent 3D joint-coordinate prediction.

3.3. Loss function

To constrain the accuracy of the predicted 3D joint coordinates for the target frame and the plausibility of the hand skeleton structure, we train the network using a composite loss consisting of a 3D coordinate L1 loss, a bone-length loss, and a bone-direction consistency loss[30][31]. Let the predicted 3D joint coordinates be $\widehat{\boldsymbol{P}} \in R^{B\times T\times 21\times 3}$, and the ground-truth coordinates be *P*, where *B* denotes the batch size. The L1 loss is first used to constrain the error between the predicted and ground-truth joint coordinates:

$$L_{pose} = \frac{1}{N_v} \|\widehat{\boldsymbol{P}} - \boldsymbol{P}\|_1 \tag{5}$$

based on the topology of the 21 hand joints, we construct a set of skeletal edges *E* and use the bone-length loss to constrain the anatomical plausibility of the predictions:

$$L_{bone} = \frac{1}{|E_v|} \sum_{(i,j)\in E_v} \big| \|\widehat{\boldsymbol{P}}_i - \widehat{\boldsymbol{P}}_j\|_2 - \|\boldsymbol{P}_i - \boldsymbol{P}_j\|_2 \big| \tag{6}$$

a bone-direction consistency loss is also introduced to constrain the structural consistency of the hand pose through bone lengths and joint directions:

$$L_{angle} = \frac{1}{|E_v|} \sum_{(i,j)\in E_v} (1 - cos(\widehat{\boldsymbol{v}}_{ij}, \boldsymbol{v}_{ij})) \tag{7}$$

the overall loss function is defined as follows:

$$L_{total} = \lambda_{pose} L_{pose} + \lambda_{bone} L_{bone} + \lambda_{angle} L_{angle} \tag{8}$$

here, $\lambda_{pose} = 1.0, \lambda_{bone} = 0.2$ and $\lambda_{angle} = 0.05$are used to balance the contributions of the different loss terms.

4. Results and Analysis

4.1. Experimental Setup

Experiments were conducted on the HanCo dataset, a multi-view hand video dataset developed by extending FreiHand. It contains 1,518 short videos, each synchronously captured by eight cameras,

ISSN 2616-5775 Vol. 9, Issue 8: 69-78, DOI: 10.25236/AJCIS.2026.090809

yielding 860,304 frames in total. Each sequence contains approximately 70 time steps on average, corresponding to about 14 s of continuous motion, and 72,440 frames have manually verified hand-shape annotations [32].

In our experiments, the RGB images in HanCo were uniformly resized to 256×256. Consecutive eight frames were used as one input sample, and the data were divided into continuous sequences according to their temporal order. The dataset was split into training and validation sets at a ratio of 70% to 30%. All errors were calculated in the camera coordinate system and reported in mm.

MPJPE and PA-MPJPE [33] were used as evaluation metrics. MPJPE measures the absolute positional error between the predicted and ground-truth joints, and is defined as follows:

$$\text{MPJPE} = \frac{1}{N}\sum_{i=1}^{N}\left\|\hat{\mathbf{p}}_i - \mathbf{p}_i\right\|_2 \tag{9}$$

here, $\hat{\boldsymbol{p}}_i$ and $\hat{\boldsymbol{p}}_i$ denote the predicted and ground-truth coordinates of the *i*-th joint, respectively, and *N* is the number of joints included in the evaluation. This metric preserves errors in global position, scale, and depth, with lower values indicating more accurate localization in the camera coordinate system. Before calculating PA-MPJPE, the predicted and ground-truth poses are first aligned using Procrustes analysis to reduce the effects of global translation, rotation, and scale differences:

$$\text{PA-MPJPE} = \frac{1}{N}\sum_{i=1}^{N}\| s\mathbf{R}\hat{\mathbf{p}}_i + \mathbf{t} - \mathbf{p}_i\|_2 \tag{10}$$

here, *R*, *s*, and *t* denote the rotation matrix, scale factor, and translation vector, respectively. This metric focuses more on the local hand structure and the relative arrangement of the joints, with lower values indicating more accurate recovery of the pose shape.

4.2. Comparative Experiments

To validate the effectiveness of BFHand for 3D hand pose estimation from continuous RGB image sequences, we selected S2DHand[3], HaMeR[20], WiLoR[30], and Hamba[34] as comparison methods on the HanCo dataset. These methods represent graph-based modeling, monocular mesh recovery, hand localization in real-world scenes, and multi-view adaptation, respectively, enabling comparisons with BFHand from different perspectives. All methods used the same data split, input preprocessing, and evaluation protocol, and were evaluated using MPJPE and PA-MPJPE. The results are presented in Table 2.

Table 2. Comparison of different methods on the HanCo dataset.

MODEL	MPJPE (mm)	PA-MPJPE (mm)
S2DHand[3]	18.268	9.752
HaMeR[20]	13.913	5.834
WiLoR[30]	16.032	5.917
Hamba[34]	13.911	5.803
Ours	12.447	5.504

The experimental results are presented in Table 2. BFHand achieves an MPJPE of 12.447 mm and a PA-MPJPE of 5.504 mm, outperforming all comparison methods on both metrics. Hamba and HaMeR obtain MPJPE values of 13.911 mm and 13.913 mm, respectively, whereas WiLoR and S2DHand obtain values of 16.032 mm and 18.268 mm, respectively. Compared with Hamba, which has the closest MPJPE, BFHand reduces the error by 1.464 mm. For PA-MPJPE, Hamba, HaMeR, and WiLoR achieve values of 5.803, 5.834, and 5.917 mm, respectively. BFHand still maintains the lowest error, indicating that its predictions not only recover the global position but also preserve the local structural relationships among the hand joints.

These results suggest that BFHand jointly models motion information across consecutive frames and relationships among hand joints, providing some compensation for short-term occlusions, local heatmap shifts, and missing finger details. Compared with methods that rely primarily on single-frame appearance or local 2D cues, multi-frame global feature fusion provides the target frame with additional contextual information, while the local feature enhancement module further mitigates the influence of global information propagation on fine-grained joint localization.

ISSN 2616-5775 Vol. 9, Issue 8: 69-78, DOI: 10.25236/AJCIS.2026.090809

4.3. Ablation Study

To investigate the effects of ModernTCN, CAMoE, and EnhGraphMLP on 3D hand pose estimation performance, we conducted a module ablation study while keeping the data split, training strategy, and evaluation procedure consistent. As shown in Table 3, "✗" indicates that the corresponding module was removed.

Table 3. Ablation study results for BFHand.

ModernTCN	CAMoE	EnhGraphMLP	MPJPE(mm)	PA-MPJPE(mm)
			12.447	5.504
✗			12.906	5.862
	✗		13.379	5.957
		✗	12.776	5.752
	✗	✗	13.636	6.360
✗		✗	13.505	6.045
✗	✗		13.558	6.091
✗	✗	✗	14.146	6.787

When all three modules are used in the full model, the MPJPE and PA-MPJPE are 12.447 mm and 5.504 mm, respectively. After removing ModernTCN, the two metrics increase to 12.906 mm and 5.862 mm, corresponding to increases of 0.459 mm and 0.358 mm. This result indicates that although single-frame spatial features can support basic pose regression, they are insufficient for capturing the temporal continuity of hand motion.

After removing CAMoE, the MPJPE increases from 12.447 mm to 13.379 mm, while the PA-MPJPE increases from 5.504 mm to 5.957 mm, corresponding to increases of 0.932 mm and 0.453 mm, respectively. This finding demonstrates that coordinating local joint relationships with the overall hand pose is important for reducing localization errors at the fingertips and joint connections. After removing EnhGraphMLP, the MPJPE and PA-MPJPE are 12.776 mm and 5.752 mm, respectively. Although the increases are relatively small, both metrics remain higher than those of the full model. By introducing contextual channel modulation into graph-structured features, this module further adjusts the response strengths of different joint features.

When only ModernTCN, CAMoE, or EnhGraphMLP is retained, the MPJPE values are 13.636, 13.505, and 13.558 mm, respectively. When all three modules are removed, the MPJPE and PA-MPJPE reach 14.146 mm and 6.787 mm, representing increases of 1.699 mm and 1.283 mm, respectively, compared with the full model.

The ablation results show that the three modules contribute to pose estimation through temporal dependencies, context-adaptive fusion, and inter-joint spatial relationships, respectively. Removing any one of the modules increases the model error, with the removal of CAMoE causing the most pronounced performance degradation. This finding highlights the importance of CAMoE for integrating local joint information with global contextual information. ModernTCN and EnhGraphMLP complement this process by providing sequence dynamics and spatial feature propagation between joints, respectively.

4.4. Visualization Interface for Hand Pose Estimation Results

To present single-sample inference results for 3D hand pose estimation, we designed a model visualization interface. The top of the interface contains a sample ID input field, a detection button, and a device status indicator. The main area displays the input image, model processing information, and the 3D hand pose in sequence according to the processing workflow. Error metrics and inference time are presented together at the bottom. The functions of each region are summarized in Table 4.

Table 4. Functional description of the hand pose estimation interface.

Interface Area	Function
Sample ID Input Area	Select the HanCo sample to be evaluated
Detection Button and Device Status Area	Start inference and display the current computing device
Input Image Area	Display the original hand image of the selected sample
Model Processing Area	Display sample information and detection status
3D Pose Area	Render the predicted joint locations and hand skeleton connections
Bottom Metrics Area	Display MPJPE, PA-MPJPE, and single-sample inference time

ISSN 2616-5775 Vol. 9, Issue 8: 69-78, DOI: 10.25236/AJCIS.2026.090809

After the interface completes image loading and model inference, the original hand image is displayed on the left, while the predicted 3D pose is presented on the right using joint markers and skeleton connections. Ground-truth annotations are also provided for comparison. The pose estimation results for this sample are shown in Fig. 3.

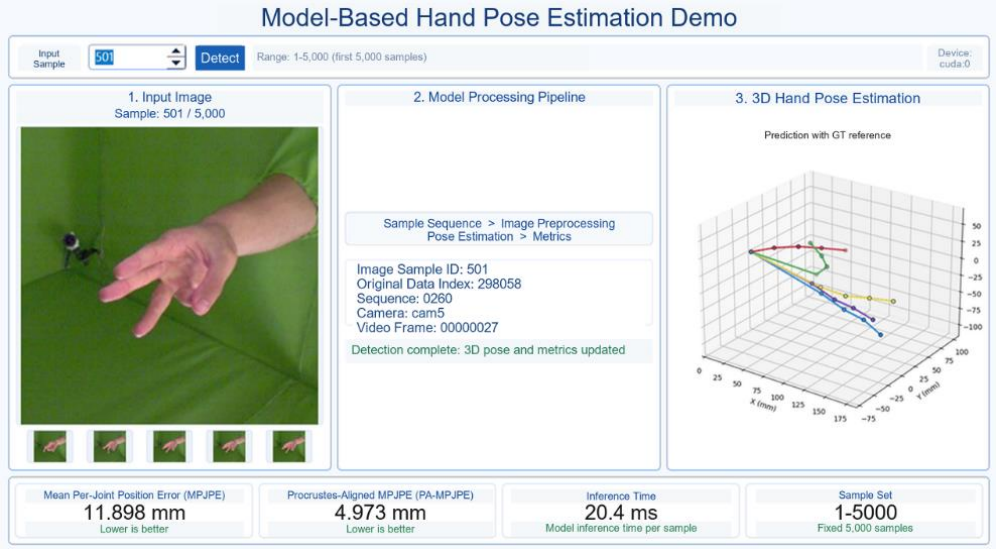


Fig. 3. Interface for displaying 3D hand pose estimation results

The figure shows the spatial distribution of the wrist and finger joints. Corresponding to the visualized pose, the bottom of the interface reports an MPJPE of 11.898 mm, a PA-MPJPE of 4.973 mm, and a single-inference time of 20.4 ms. By presenting the input image, 3D skeleton, and evaluation metrics in a unified view, the interface visualizes the hand pose estimation results and facilitates inspection of the predicted joint positions and pose structure for individual samples.

5. Conclusions

This study presents BFHand for camera-coordinate 3D hand pose estimation from monocular RGB image sequences. The method takes 2D joint heatmaps from eight consecutive frames as input and jointly models hand motion information and cross-frame joint relationships using ModernTCN, multi-head temporal attention, and multi-frame GraphMLP. We further introduce CAMoE and EnhGraphMLP to perform contextual fusion and selective enhancement of local joint features in the target frame, thereby reducing the effects of occlusions, heatmap shifts, and rapid motion on pose prediction.Experimental results show that BFHand achieves an MPJPE of 12.447 mm and a PA-MPJPE of 5.504 mm on the HanCo dataset, while maintaining accurate joint positions and structural relationships. The model remains limited by fixed-length sequence inputs, the quality of 2D heatmaps, and the scene coverage of the dataset. Errors persist under severe occlusion and when recovering fine-grained finger structures. Future work could improve the model's applicability by incorporating stronger geometric constraints, variable-length temporal modeling, and cross-scene adaptation strategies.

Acknowledgements

This work was supported by the Doctoral Startup Fund of Jilin Chemical Engineering University (Grant No.222012612033).

References

[1] Liu, X., Ren, P., Gao, Y., Wang, J., Sun, H., Qi, Q., ... & Liao, J. (2024, March). Keypoint fusion for rgb-d based 3d hand pose estimation. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 38, No. 4, pp. 3756-3764). https://doi.org/10.1609/aaai.v38i4.28166

[2] Zheng, X., Wen, C., Xue, Z., Ren, P., & Wang, J. (2023, October). Hamuco: Hand pose estimation via multiview collaborative self-supervised learning. In 2023 IEEE/CVF International Conference on Computer Vision (ICCV) (pp. 20706-20716). IEEE.https://doi.org/10.48550/arXiv.2302.00988

ISSN 2616-5775 Vol. 9, Issue 8: 69-78, DOI: 10.25236/AJCIS.2026.090809

[3] Liu, R., Ohkawa, T., Zhang, M., & Sato, Y. (2024, June). Single-to-dual-view adaptation for egocentric 3d hand pose estimation. In 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 677-686). IEEE.https://doi.org/10.48550/arXiv.2403.04381

[4] Liao, Z., Zhu, J., Wang, C., Hu, H., & Waslander, S. L. (2024, June). Multiple view geometry transformers for 3d human pose estimation. In 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 708-717). IEEE. https://doi.org/10.48550/arXiv.2311.10983

[5] Wu, S., Jin, S., Liu, W., Bai, L., Qian, C., Liu, D., & Ouyang, W. (2021). Graph-based 3d multi-person pose estimation using multi-view images. In 2021 IEEE. In CVF International Conference on Computer Vision (ICCV) (pp. 11128-11137). https://doi.org/10.48550/arXiv.2109.05885

[6] Khaleghi, L., Sepas-Moghaddam, A., Marshall, J., & Etemad, A. (2022). Multiview video-based 3-d hand pose estimation. IEEE Transactions on Artificial Intelligence, 4(4), 896-909. https://doi.org/10.1109/TAI.2022.3195968

[7] Li, S., Su, W., Luo, G., Tian, J., Han, Y., & Liu, L., et al. (2025). Monocular 3d hand pose estimation based on high-resolution network. Advances in Continuous and Discrete Models, 2025(1), 1-24. https://link.springer.com/article/10.1186/s13662-025-03948-2

[8] Chen, X., Wang, G., Guo, H., & Zhang, C.. (2017). Pose guided structured region ensemble network for cascaded hand pose estimation. Neurocomputing. https://doi.org/10.1016/j.neucom.2018.06.097

[9] Zhang, F., Zhao, L., Li, S., Su, W., Liu, L., & Tao, W. (2024). 3d hand pose and shape estimation from monocular rgb via efficient 2d cues. https://doi.org/10.1007/s41095-023-0346-4

[10] Zhou, F., Shen, P., Dai, J., Jiang, N., Hu, Y., & Lai, Y. K., et al. (2024). Ahrnet: attention and heatmap-based regressor for hand pose estimation and mesh recovery. ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 3000-3004. https://doi.org/10.1109/ICASSP48485.2024.10446600

[11] Chen, X., Liu, Y., Dong, Y., Zhang, X., Ma, C., & Xiong, Y., et al. (2021). Mobrecon: mobile-friendly hand mesh reconstruction from monocular image. https://doi.org/10.48550/arXiv.2112.02753

[12] Sharma, S., & Huang, S. (2021). An end-to-end framework for unconstrained monocular 3D hand pose estimation. Pattern Recognition, 115, 107892. https://doi.org/10.1016/j.patcog.2021.107892

[13] Ren, P., Wang, J., Sun, H., Qi, Q., Liu, X., & Zhang, M., et al. Prior-Aware Dynamic Temporal Modeling Framework for Sequential 3D Hand Pose Estimation. 2025 IEEE/CVF International Conference on Computer Vision (ICCV). IEEE. https://doi.org/10.1109/ICCV51701.2025.00611

[14] Duran, E., Kocabas, M., Choutas, V., Fan, Z., & Black, M. J. (2023). Hmp: hand motion priors for pose and shape estimation from video. 2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). https://doi.org/10.1109/WACV57701.2024.00623

[15] Wen, Y., Pan, H., Yang, L., Pan, J., Komura, T., & Wang, W. (2023). Hierarchical temporal transformer for 3d hand pose estimation and action recognition from egocentric rgb videos. 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 21243-21253. https://doi.org/10.48550/arXiv.2209.09484

[16] Ye, Y., Feng, Y., Taheri, O., Feng, H., Tulsiani, S., & Black, M. J. (2025). Predicting 4D Hand Trajectory from Monocular Videos. https://doi.org/10.48550/arXiv.2501.08329

[17] Li, W., Liu, M., Liu, H., Guo, T., Wang, T., & Tang, H., et al. (2025). Graphmlp: a graph mlp-like architecture for 3d human pose estimation. Pattern Recognition, 158(000). https://doi.org/10.1016/j.patcog.2024.110925

[18] Luo, D., & Wang, X. (2024, May). ModernTCN: A Modern Pure Convolution Structure for General Time Series Analysis. In The Twelfth International Conference on Learning Representations. https://openreview.net/forum?id=vpJMJerXHU

[19] Park, J., Jung, D. S., Moon, G., & Lee, K. M. (2023, October). Extract-and-adaptation network for 3D interacting hand mesh recovery. In 2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW) (pp. 4202-4211). IEEE. https://doi.org/10.48550/arXiv.2309.01943

[20] Pavlakos, G., Shan, D., Radosavovic, I., Kanazawa, A., Fouhey, D., & Malik, J. (2024, June). Reconstructing hands in 3d with transformers. In 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 9826-9836). IEEE. https://doi.org/10.1109/CVPR52733.2024.00938

[21] Liu, S., Tripathi, S., Majumdar, S., & Wang, X. (2022, June). Joint hand motion and interaction hotspots prediction from egocentric videos. In 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 3272-3282). IEEE. https://doi.org/10.48550/arXiv.2204.01696

[22] Gao, Q., Ju, Z., Chen, Y., Wang, Q., & Chi, C. (2022). An efficient RGB-D hand gesture detection framework for dexterous robot hand-arm teleoperation system. IEEE Transactions on Human-Machine Systems, 53(1), 13-23. https://doi.org/10.1109/THMS.2022.3206663

[23] Rudnev, V., Golyanik, V., Wang, J., Seidel, H. P., Mueller, F., Elgharib, M., & Theobalt, C. (2021, October). Eventhands: Real-time neural 3d hand pose estimation from an event stream. In 2021 IEEE/CVF International Conference on Computer Vision (ICCV) (pp. 12365-12375). IEEE.

ISSN 2616-5775 Vol. 9, Issue 8: 69-78, DOI: 10.25236/AJCIS.2026.090809

https://doi.org/10.48550/arXiv.2012.06475
[24] Jiang, J., Li, J., Zhang, B., Deng, X., & Shi, B. (2023). Evhandpose: event-based 3d hand pose estimation with sparse supervision. ArXiv, abs/2303.02862. https://doi.org/10.1109/TPAMI.2024.3380648
[25] Chamorro, S., Collier, J., & Grondin, F. (2021). Neural network based lidar gesture recognition for realtime robot teleoperation.https://doi.org/10.1109/SSRR53300.2021.9597855
[26] Meng, L., & Yizhou, Y. (2025). Overlock: An overview-first-look-closely-next convnet with context-mixing dynamic kernels. In 2025 IEEE. In CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 128-138). https://doi.org/10.48550/arXiv.2502.20087
[27] Wang, A., Chen, H., Lin, Z., Han, J., & Ding, G. (2025). LSNet: See Large, Focus Small.https://doi.org/10.1109/CVPR52734.2025.00908
[28] Cai, W., Jiang, J., Wang, F., Tang, J., Kim, S. H., & Huang, J. (2025). A survey on mixture of experts in large language models. IEEE Transactions on Knowledge and Data Engineering, 1-20. https://doi.org/10.1109/tkde.2025.3554028
[29] Puigcerver, J., Riquelme, C., Mustafa, B., & Houlsby, N. (2023). From sparse to soft mixtures of experts. https://doi.org/10.48550/arXiv.2308.00951
[30] Potamias, R. A., Zhang, J., Deng, J., & Zafeiriou, S. Wilor: End-to-end 3d hand localization and reconstruction in-the-wild. In 2025 IEEE. In CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 12242-12254). https://doi.org/10.48550/arXiv.2409.12259
[31] Lin, W., Hu, Y., Dai, L., Tang, X., & Chen, X. (2026). Geohand: unlocking prior geometry knowledge for monocular 3d hand reconstruction. https://doi.org/10.48550/arXiv.2605.17354
[32] Zimmermann, C., Argus, M., & Brox, T. (2021). Contrastive representation learning for hand shape estimation. https://doi.org/10.48550/arXiv.2106.04324
[33] Chen, X., Liu, Y., Ma, C., Chang, J., Wang, H., & Chen, T., et al. (2021). Camera-space hand mesh recovery via semantic aggregation and adaptive 2d-1d registration. https://doi.org/10.48550/arXiv.2103.02845
[34] Dong, H., Chharia, A., Gou, W., Carrasco, F. V., & De la Torre, F. (2024). Hamba: Single-view 3d hand reconstruction with graph-guided bi-scanning mamba. Advances in Neural Information Processing Systems, 37, https://doi.org/2127-2160. 10.52202/079017-0069