HandMvNet: Real-Time 3D Hand Pose Estimation Using Multi-View Cross-Attention Fusion

Muhammad Asad Ali[1,2], Nadia Robertini[1] and Didier Stricker[1,2]
[1]*Augmented Vision Group, German Research Center for Artificial Intelligence (DFKI), Kaiserslautern, Germany*
[2]*Department of Computer Science, University of Kaiserslautern-Landau (RPTU), Kaiserslautern, Germany*
{*firstname_middlename.lastname*}*@dfki.de*

Keywords: Hand Reconstruction, Hand Pose Estimation, Multi-View Reconstruction.

Abstract: In this work, we present HandMvNet, one of the first real-time method designed to estimate 3D hand motion and shape from multi-view camera images. Unlike previous monocular approaches, which suffer from scale-depth ambiguities, our method ensures consistent and accurate absolute hand poses and shapes. This is achieved through a multi-view attention-fusion mechanism that effectively integrates features from multiple viewpoints. In contrast to previous multi-view methods, our approach eliminates the need for camera parameters as input to learn 3D geometry. HandMvNet also achieves a substantial reduction in inference time while delivering competitive results compared to the state-of-the-art methods, making it suitable for real-time applications. Evaluated on publicly available datasets, HandMvNet qualitatively and quantitatively outperforms previous methods under identical settings. Code is available at `github.com/pyxploiter/handmvnet`.

1 INTRODUCTION

3D hand pose estimation has emerged as a important research area in computer vision with applications across fields like augmented reality (AR), virtual reality (VR), and robotics. The ability to accurately capture and reconstruct hand movements holds immense potential in enhancing human-computer interaction, enabling more natural, intuitive gesture-based controls. In AR and VR, realistic and responsive hand pose estimation enriches immersive experiences, allowing users to interact seamlessly with virtual environments. Similarly, in robotics, precise hand pose estimation is important for tasks such as robotic hand retargeting, where robotic hands mimic human movements to perform intricate tasks.

Traditional approaches in 3D hand pose estimation have primarily relied on single-view images (Boukhayma et al., 2019; Chen et al., 2021a,b; Ge et al., 2019; Moon and Lee, 2020; Park et al., 2022). However, 3D hand pose estimation from monocular views presents several challenges. Depth and scale ambiguity, where the exact distance and size of the hand from the camera are difficult to determine, significantly complicates the estimation process. Consequently, many approaches only estimate root-relative hand vertices (Moon and Lee, 2020; Ge et al., 2019; Zhou et al., 2020). Occlusions, caused by the overlapping of fingers or the hand being partially obscured by other objects, further add to the complexity of accurately estimating hand poses (Park et al., 2022). Additionally, varying perspectives and unknown camera viewpoints introduce uncertainties that make the task more challenging.

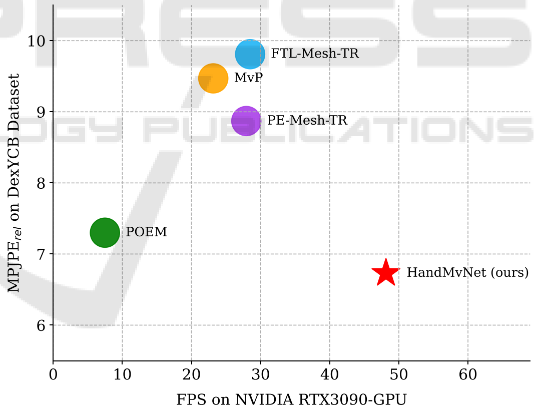


Figure 1: Comparison of error vs. inference speed across different methods. Our approach outperforms other methods in both inference speed and accuracy.

To address the limitations associated with monocular views, multi-view setups have been proposed as a solution (Yu et al., 2021; Chao et al., 2021; Yang et al., 2022; Hampali et al., 2020). A multi-view setup, consisting of multiple cameras positioned at

Ali, M. A., Robertini, N. and Stricker, D.
HandMvNet: Real-Time 3D Hand Pose Estimation Using Multi-View Cross-Attention Fusion.
DOI: 10.5220/0013107300003912
Paper published under CC license (CC BY-NC-ND 4.0)
In *Proceedings of the 20th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2025) - Volume 2: VISAPP*, pages 555-562
ISBN: 978-989-758-728-3; ISSN: 2184-4321
Proceedings Copyright © 2025 by SCITEPRESS – Science and Technology Publications, Lda.

different angles around the hand, can significantly reduce the impact of occlusions and depth ambiguities, enabling more accurate and robust estimation of hand poses and shapes at absolute 3D locations. Most multi-view approaches (Guan et al., 2006; Yang et al., 2023; Zheng et al., 2023) are computationally expensive, primarily due to the increased input space and architectural design choices that prioritize qualitative results over computational efficiency.

In this work, we propose HandMvNet, a novel neural network architecture for efficient and accurate 3D hand pose estimation from multi-view inputs. The key contributions of this work are as follows:

- We present a framework that leverages multi-view data for accurate 3D hand pose estimation.
- Our method achieves real-time performance, making it suitable for time-critical applications.
- We show that our approach performs effectively with or without camera calibration.

We conduct extensive experiments on public multi-view datasets for hand pose and shape reconstruction in challenging scenarios, including strong occlusions from object interactions. Our findings demonstrate that HandMvNet effectively and accurately estimates hand poses and shapes, outperforming existing state-of-the-art methods both qualitatively and computationally.

2 RELATED WORK

Most approaches have focused on estimating hand pose from monocular input (Ge et al., 2019; Boukhayma et al., 2019; Zhou et al., 2020; Chen et al., 2021a,b; Park et al., 2022; Moon and Lee, 2020). While various hand representations have been proposed (Chen et al., 2021a; Malik et al., 2020, 2021), the deformable hand mesh model MANO (Romero et al., 2022), which includes dense 3D hand surface representation, remains the most widely used (Chen et al., 2021a; Park et al., 2022; Zhou et al., 2020; Boukhayma et al., 2019). Similarly to (Ge et al., 2019), we uniquely estimate the hand mesh directly, bypassing the need for the MANO model parameters, thus offering a flexible, model-free solution. With the rise of transformer architectures (Vaswani et al., 2017), such frameworks have also been adopted for 3D pose estimation, showcasing their effectiveness (Park et al., 2022; Zhao et al., 2022; Lin et al., 2021). Despite recent advances, most methods focus on estimating root-relative hand poses due to limited input information and scale-depth ambiguity. In this work, we integrate contributions from multiple views using cross-attention, enabling the estimation of contextualized 3D absolute hand poses. Compared to other multi-view approaches (Ge et al., 2016; He et al., 2020; Han et al., 2022; Remelli et al., 2020; Iskakov et al., 2019), our method avoids conventional volumetric or other intermediate representations that negatively affect the inference speed. Although most approaches require multi-view camera calibration, mainly for algebraic triangulation and geometric priors to estimate 3D hand pose (Remelli et al., 2020; Bartol et al., 2022; Chen et al., 2022; Iskakov et al., 2019; Tu et al., 2020; He et al., 2020; Zhang et al., 2021b), we instead propose a more flexible, calibration-free solution that can optionally incorporate camera parameters. Recent advancements (Yang et al., 2023; Shuai et al., 2022; Ma et al., 2022) in transformer-based implicit cross-view fusion inspire our proposed method for multi-view cross-attention fusion.

3 METHOD

The aim of our HandMvNet approach is to estimate 3D hand joints and vertices from multi-view RGB images. In this section, we provide a comprehensive description of our proposed model architecture.

3.1 Architecture

The overall pipeline of HandMvNet is illustrated in Figure 2. The network processes a set of multi-view RGB images, $\mathcal{I} = \{\mathbf{I}_i\}_{i=1}^{\mathcal{C}}$, captured from $\mathcal{C}$ camera views and estimates the 3D hand joints $\mathbf{J}^{3D} \in \mathbb{R}^{\mathcal{J}\times 3}$ and vertices $\mathbf{V}^{3D} \in \mathbb{R}^{\mathcal{V}\times 3}$, where $\mathcal{J} = 21$ & $\mathcal{V} = 778$.

The architecture consists of three key stages: **(1) Pre-Fusion:** Each input image is independently processed to extract view-specific features and estimate 2D joint locations, with shared network weights across all views. **(2) Fusion:** The extracted features are then fused to aggregate multi-view information for enhanced 3D understanding (see Figure 3b). **(3) Post-Fusion:** Finally, the fused features are refined to regress the 3D hand joints and vertices, producing the complete 3D hand reconstruction. Each stage is described in detail in the sections below.

3.1.1 Pre-Fusion

Backbone. The first stage of our pipelines uses ResNet50 (He et al., 2016) as a backbone to extract the view-specific image features from input images. The backbone is pre-trained on the ImageNet dataset (Deng et al., 2009), and its weights are shared across

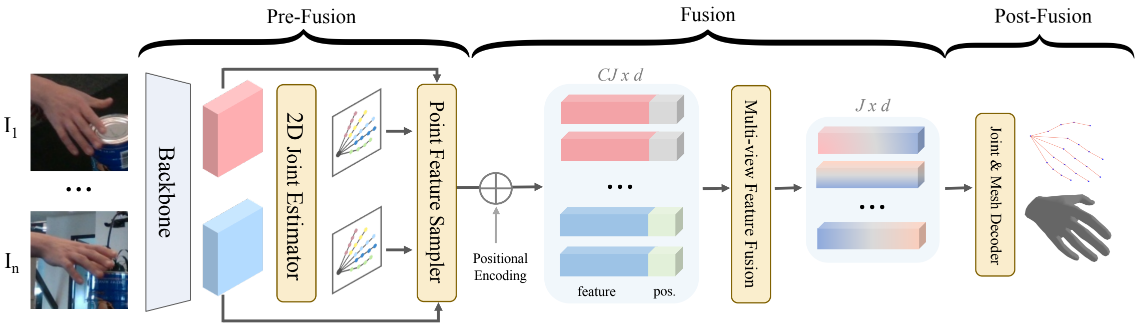
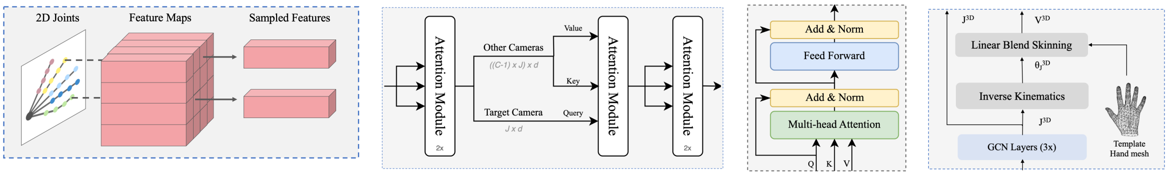


Figure 2: HandMvNet' architecture consists of three stages: (a) Sampling joint-aligned features using predicted 2D joints from each image (b) Fusing multi-view sampled features, (c) Regressing 3D hand joints and vertices.



(a) .

(b) .

(c) .

(d) .

Figure 3: Modules of HandMvNet's architecture: (a) Point Feature Sampler. (b) Multi-view Feature Fusion. (c) Attention Module. (d) Joint & Mesh Decoder.

each camera view. For each camera view i, the backbone processes the image $\mathbf{I}_i$ and outputs a corresponding view-specific feature map $\mathbf{Z}_i \in \mathbb{R}^{1024\times32\times32}$.

2D Joint Estimator. At this stage, two convolutional layers refine the features $\mathbf{Z}_i$ to produce joint-specific heatmaps $\mathbf{H}_i$. To extract the 2D joint locations from the heatmaps, we apply a differentiable soft-argmax function (Sun et al., 2018), which transforms the heatmaps into directly usable joint coordinates $\mathbf{J}_i^{2D} = \textit{soft-argmax}(f_{\text{CNN}}(\mathbf{Z}_i)) \in \mathbb{R}^{\mathcal{J}\times2}$.

Point Feature Sampler. In the final pre-fusion stage, we extract view-specific features from $\mathbf{Z}_i$ (see Figure 3a), reduced to a dimensionality of $\mathbb{R}^{512\times32\times32}$ using a convolutional layer, corresponding to 2D joint locations $\mathbf{J}_i^{2D}$, $\mathbf{S}_i = \textit{sampler}(\mathbf{Z}_i, \mathbf{J}_i^{2D})$, $\mathbf{S}_i \in \mathbb{R}^{\mathcal{J}\times512}$. The sampled joint-aligned features from all camera views are concatenated, forming the aggregated multi-view feature representation $\mathbf{S} = \textit{concat}(\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_C)$, where $\mathbf{S} \in \mathbb{R}^{C\mathcal{J}\times512}$.

3.1.2 Fusion

Positional Encoding. To preserve critical spatial and geometric information in cropped hand images, we introduce three types of positional encodings:
1) $\text{PE}_{\text{joint}} \in \mathbb{R}^{C\mathcal{J}\times2}$ embeds 2D joint positions into the feature vector to capture the hand's skeletal structure and the relative joint positions in each view.
2) $\text{PE}_{\text{crop}} \in \mathbb{R}^{C\times10}$ encodes the location of the hand crop relative to the camera (Prakash et al., 2023), with each corner and one center point (x, y) calculated as $\theta_x = \tan^{-1}((x - p_x)/f_x)$ and $\theta_y = \tan^{-1}((y - p_y)/f_y)$, where p_x, p_y are the principal point coordinates, and f_x, f_y are focal lengths. PE_{crop} is repeated $\mathcal{J}$ times for each joint in the view. This encoding is only applied if camera intrinsics are available.
3) Sinusoidal encoding $\text{PE}_{\text{sin}} \in \mathbb{R}^{C\mathcal{J}\times d}$ (Vaswani et al., 2017) captures inter-view and inter-joint relations for attention-based fusion.

The final feature vector is:

$$\mathbf{F} = \text{concat}(\mathbf{S}, \text{PE}_{\text{joint}}, \text{PE}_{\text{crop}}) + \text{PE}_{\text{sin}}. \tag{1}$$

where $\mathbf{F} \in \mathbb{R}^{C\mathcal{J}\times d}$ and $d = 512 + 2 + 10 = 524$.

Multi-View Feature Fusion. To capture the dependencies between non-local joints across C camera views, we pass the independently sampled features $\mathbf{F}$ through attention module (Figure 3c) and then, to fuse multi-view features, we employ multi-head cross-attention between the first camera view features $\mathbf{F}_1 \in \mathbb{R}^{\mathcal{J}\times d}$ acting as the **query** and the features from the remaining camera views $\mathbf{F}_{C-1} \in \mathbb{R}^{((C-1)\times\mathcal{J})\times d}$ acting as the **key** and **value** (source). The cross-attention is formulated as:

$$\mathbf{F}^* = \textit{softmax}\left(\frac{\mathbf{F}_1\mathbf{F}_{C-1}^T}{\sqrt{d}}\right)\mathbf{F}_{C-1} \tag{2}$$

The cross-attention layer outputs $\mathbf{F}^* \in \mathbb{R}^{\mathcal{J}\times d}$ where $\mathcal{J} = 21$ and $d = 524$, which aggregates the features across the camera views into the target camera feature space. Finally, self-attention is applied again to $\mathbf{F}^*$ to further refine the intra-joint relationships.

Table 1: Quantitative results (mm) on the test sets of DexYCB-MV, HO3D-MV, and MVHand. 📷 denotes the methods that require camera parameters. The best and second-best results are highlighted in bold and underlined respectively.

	#	Methods	$\text{MPJPE}_{rel}\downarrow$	$\text{PA}_J\downarrow$	$\text{AUC}_{J@20}\uparrow$	$\text{MPVPE}_{rel}\downarrow$	$\text{PA}_V\downarrow$	$\text{AUC}_{V@20}\uparrow$
DexYCB-MV	1	📷 MvP	9.47	4.26	**0.69**	12.18	8.14	0.53
	2	📷 PE-Mesh-TR	8.87	4.76	0.64	8.67	4.70	0.64
	3	📷 FTL-Mesh-TR	9.81	5.51	0.59	9.80	5.75	0.59
	4	📷 POEM	7.30	**3.93**	0.68	7.21	**4.00**	**0.70**
	5	📷 Multi-view Fit.	8.77	5.19	0.65	8.71	5.29	0.65
	6	📷 HandMvNet (ours)	**6.73**	4.08	0.67	**7.19**	4.52	0.65
	7	📷 HandMvNet-HR (ours)	6.89	4.08	0.67	7.30	4.53	0.65
	8	HandMvNet w/o cam. (ours)	7.03	4.13	0.66	7.38	4.56	0.64
	9	HandMvNet-HR w/o cam. (ours)	7.28	4.20	0.65	7.62	4.69	0.63
HO3D-MV					$\text{AUC}_{J@50}\uparrow$			$\text{AUC}_{V@50}\uparrow$
	10	📷 MvP	24.90	10.44	0.60	27.08	10.04	0.59
	11	📷 PE-Mesh-TR	30.23	11.67	0.54	29.19	11.31	0.55
	12	📷 FTL-Mesh-TR	34.74	10.72	0.52	33.53	10.56	0.53
	13	📷 POEM	21.94	**9.60**	**0.63**	21.45	9.97	**0.66**
	14	📷 HandMvNet (ours)	21.43	10.89	0.59	20.17	10.16	0.61
	15	📷 HandMvNet-HR (ours)	20.73	11.01	0.61	19.82	10.73	0.62
	16	HandMvNet w/o cam. (ours)	21.55	10.15	0.58	20.10	**9.39**	0.61
	17	HandMvNet-HR w/o cam. (ours)	**20.40**	11.98	0.61	**19.33**	11.24	0.63
MVHand					$\text{AUC}_{J@20}\uparrow$			$\text{AUC}_{V@20}\uparrow$
	18	📷 MediaPipe-DLT	17.24	9.97	0.28	18.42	7.74	0.21
	19	📷 HandMvNet (ours)	2.07	1.30	0.90	7.57	4.14	0.62
	20	📷 HandMvNet-HR (ours)	1.86	1.21	**0.91**	7.59	4.12	0.62
	21	HandMvNet w/o cam. (ours)	2.05	1.28	0.90	7.62	**4.11**	0.62
	22	HandMvNet-HR w/o cam. (ours)	**1.77**	**1.14**	**0.91**	**7.46**	4.15	**0.63**

3.1.3 Post-Fusion

Joint & Mesh Decoder. We use a three-layer graph convolutional network (GCN) to decode 3D joints from the fused feature $\mathbf{F}^* \in \mathbb{R}^{\mathcal{J}\times d}$, treating $\mathcal{J}$ joints as graph nodes with d-dimensional features, estimating the final $\mathbf{J}^{3D} \in \mathbb{R}^{\mathcal{J}\times 3}$. Inverse Kinematics (IK) is then applied to compute joint rotation angles $\theta_{\mathcal{J}^{3D}} \in \mathbb{R}^{(\mathcal{J}-5)\times 3}$, which form a hand skeleton. This skeleton deforms a hand template mesh via linear blend skinning to yield the final 3D vertices $\mathbf{V}^{3D} \in \mathbb{R}^{\mathcal{V}\times 3}$ as shown in Figure 3d.

3.2 Training

We apply mean squared error loss for the predicted 2D heatmaps (L_{H}) and L1 loss for both 2D and 3D joints (L_{2D}, L_{3D}). Additionally, if camera parameters are available, we project predicted 3D joints onto 2D camera views using the perspective function $\Pi_c(\cdot) : \mathbb{R}^3 \to \mathbb{R}^2$, and minimize the L1 loss between these projections and the ground-truth 2D joints (L_{G2D}), as well as the predicted 2D joints (L_{P2D}). The total loss is defined as:

$$\begin{aligned} L = \lambda_{\text{H}}L_{\text{H}} + \lambda_{\text{2D}}L_{\text{2D}} + \lambda_{\text{3D}}L_{\text{3D}} \\ +\lambda_{\text{G2D}}L_{\text{G2D}} + \lambda_{\text{P2D}}L_{\text{P2D}} \end{aligned} \tag{3}$$

where λ values are set as 10, 1, 1, 1, and 0.5 to balance the loss scale, respectively.

4 EXPERIMENTS AND RESULTS

In this section, we conduct experiments to validate and assess the effectiveness of our proposed architecture, along with providing implementation details. We use Pytorch (Paszke et al., 2019) to implement all our networks. The AdamW (Loshchilov, 2017) optimizer is used with a weight decay of 0.05 and an initial learning rate set to 0.0001. The model is trained on two RTXA6000 GPUs with a batch size of 32. Cropped hand images resized to 256×256, serve as input data. We also evaluate a variation of our model, denoted as HandMvNet-HR, which uses HRNet-w40 as backbone (Sun et al., 2019).

4.1 Datasets

DexYCB (Chao et al., 2021)**:** is a multi-view RGB-D dataset capturing hand-object interactions, featuring 10 subjects and 8 camera views per subject. We follow the official "S0" split, excluding left-hand samples, resulting in 25,387 training, 1,412 validation, and 4,951 test multi-view samples, same as (Yang et al., 2023). We refer to this split as DexYCB-MV.

HO3D (v3) (Hampali et al., 2020)**:** includes images of hand-object interaction from up to 5 cameras. We construct HO3D-MV by selecting 7 sequences with complete multi-view observations from all 5 cam-

Table 2: Ablation Studies.

(a) Different positional encodings.

Pos. Encoding	$\text{MPJPE}_{rel}\downarrow$	$\text{PA}_J\downarrow$	$\text{AUC}_J\downarrow$
sin	7.69	4.40	0.63
sin + joint	6.96	4.14	0.66
sin + joint + crop	**6.73**	**4.08**	**0.67**

(b) Effect of fusion layers.

Fusion Layers	$\text{MPJPE}_{rel}\downarrow$	$\text{PA}_J\downarrow$	$\text{AUC}_J\uparrow$
3	6.90	4.16	0.66
5	**6.73**	**4.08**	**0.67**
7	6.88	4.14	0.67

(c) Different number of camera views.

Camera views	$\text{MPJPE}_{rel}\downarrow$	$\text{PA}_J\downarrow$	$\text{AUC}_J\uparrow$
8	**6.73**	**4.08**	**0.67**
4	7.47	4.38	0.64
2	8.33	4.83	0.60

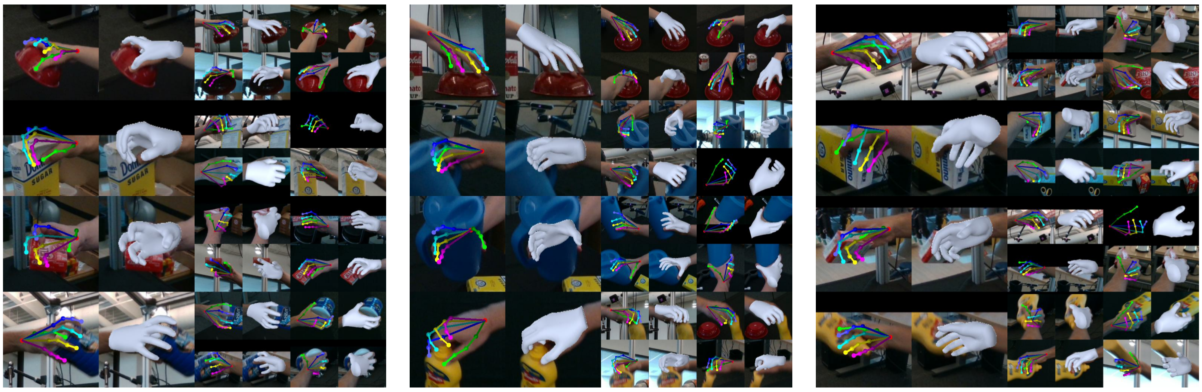


Figure 4: Qualitative results on the test set of DexYCB-MV dataset.

eras. For the training set, we use the sequences 'ABF1','BB1', 'GSF1', 'MDF1', and 'SiBF1', while the sequences 'GPMF1' and 'SB1' are reserved for testing. This results in 9,087 training and 2,706 test multi-view samples.

MVHand (Yu et al., 2021)**:** is a multi-view RGB-D hand pose dataset featuring 4 subjects and 4 camera views per subject. We split the 21,200 multi-view frames into 15,417 training, 1,927 validation, and 3,856 test multi-view samples.

4.2 Evaluation Metrics

We evaluate the performance of our method using the following standard hand pose estimation metrics. **1) MPJPE_{rel}/MPVPE_{rel}** (Mean Per Joint/Vertex Position Error) calculates the average Euclidean distance (in mm) between predicted and ground-truth joints/vertices, after aligning the root(-wrist) joint. **2) PA-MPJPE/PA-MPVPE** (Procrustes Aligned Joint/Vertex Error) measures MPJPE/MPVPE after applying procrustes analysis for scale, center and rotation alignment. We refer to these metrics as PA_J and PA_V in our experiments. **3) AUC_J/AUC_V** (Area Under Curve for Joint/Vertex Error) computes the area under the percentage of correct keypoints (PCK) curve over a range of thresholds.

4.3 Comparison with Previous Methods

We benchmark our 3D hand reconstruction approach against state-of-the-art (SOTA) multi-view methods, including **POEM** (Yang et al., 2023) and **MvP** (Zhang et al., 2021a). Although MvP is primarily designed for multi-person pose estimation, we focus on its performance in single-hand reconstruction. Given the limited availability of multi-view hand pose methods, we further evaluate simulated approaches that combine single-view hand reconstruction with advanced multi-view fusion techniques. Detailed descriptions of these simulated methods, such as **PE-Mesh-TR** (Liu et al., 2022; Lin et al., 2021), **FTL-Mesh-TR** (Remelli et al., 2020), and **Multi-view Fitting** (Hampali et al., 2020), are provided in Section 4.2 of (Yang et al., 2023). For the MVHand dataset, which lacks established multi-view benchmarks, we introduce a baseline "Mediapipe-DLT" that estimates 2D joints using Mediapipe (Zhang et al., 2020), triangulates them via Direct Linear Transform (DLT) (Hartley and Zisserman, 2003), and obtains 3D vertices through linear blend skinning.

Table 1 shows that our method consistently outperforms SOTA approaches in terms of MPJPE_{rel} and MPVPE_{rel} across all datasets, while achieving competitive performance in other metrics. In particular, our camera-independent variants, **"HandMvNet w/o cam."** and **"HandMvNet-HR w/o cam."**, also show superior performance in most cases. Our method's capacity to implicitly learn 3D geometry demands substantial data, leading to a performance decline on smaller datasets like HO3D-MV as shown in Table 1. Figure 1 shows that HandMvNet surpasses other methods in both accuracy (lower MPJPE_{rel}) and inference speed (higher FPS). We visualize qualitative results on the DexYCB-MV, HO3D-MV, and MVHand test sets in Figures 4, 5, and 6, respectively.

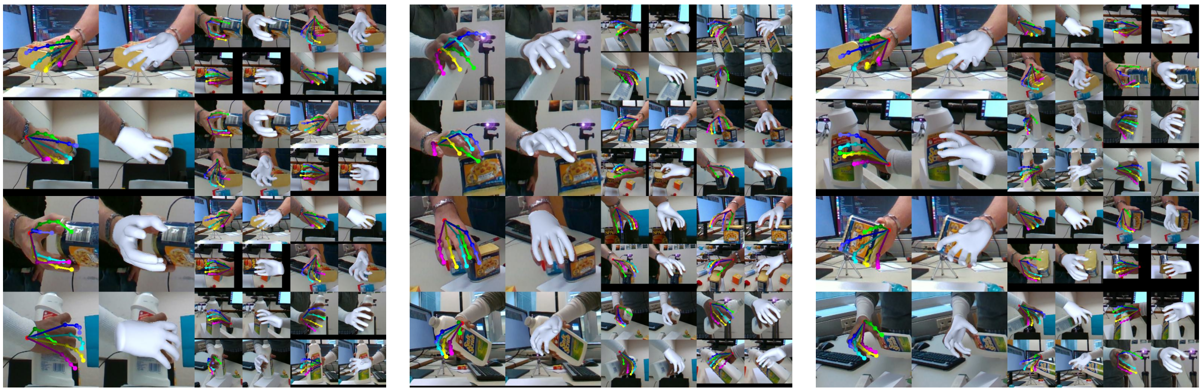


Figure 5: Qualitative results on the test set of HO3D-MV dataset.

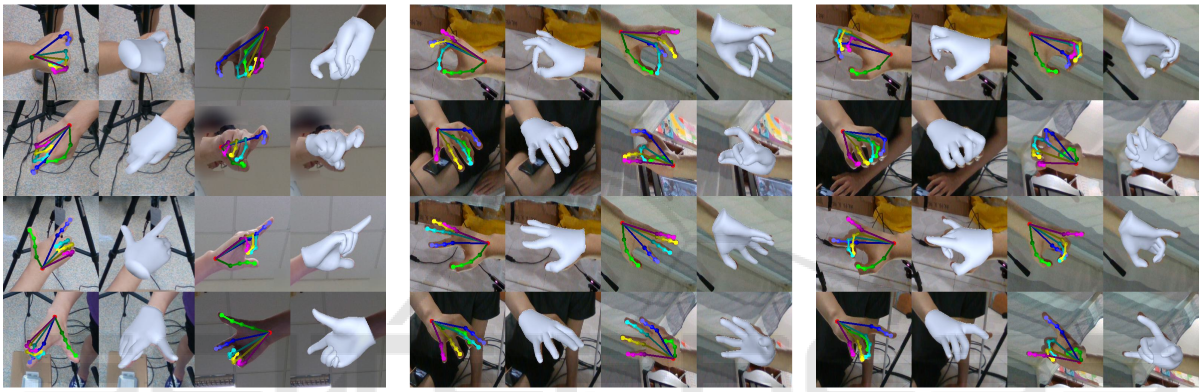


Figure 6: Qualitative results on the test set of MVHand dataset.

4.4 Ablation Study

Different Backbones. We compare the results of HandMvNet using ResNet50 as backbone and HandMvNet-HR using HRNet-w40 as backbone in the rows 6-7, 14-15, 19-20 of Table 1.

Use of Camera Parameters. In our method, camera parameters are used to add the PE_{crop} positional encoding and loss terms L_{G2D} and L_{P2D}. To evaluate the effect of removing camera dependency, we create variants "HandMvNet w/o cam." and "HandMvNet-HR w/o cam." by excluding these components. The performance of both versions, with and without camera parameters, are compared in rows 6-9, 14-17, 19-22 of Table 1.

Impact of Positional Encoding. In Table 2a, we examine the effect of different positional encodings on performance. Using the combination of sinusoidal positional encoding ($\mathrm{PE}_{\mathrm{sin}}$), joint-wise encoding ($\mathrm{PE}_{\mathrm{joint}}$) and crop encoding ($\mathrm{PE}_{\mathrm{crop}}$) results in the best performance.

Number of Fusion Layers. The impact of varying the number of fusion layers is presented in Table 2b. We observe that increasing from 3 to 5 layers improves performance, but adding more layers does not further enhance performance, suggesting that 5 layers are optimal.

Different Number of Camera Views. Table 2c shows that model performance improves gradually with increasing the number of camera views. We also compare FPS across different camera views with other approaches in Figure 7.

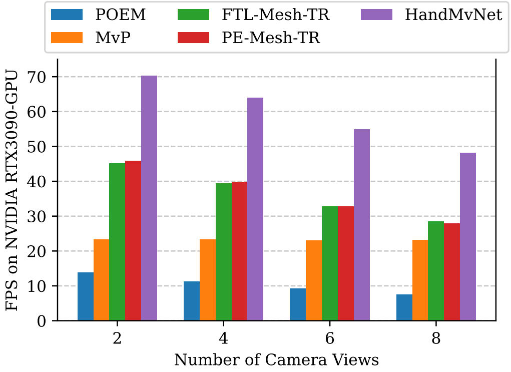


Figure 7: Inference Speed (FPS) comparison across methods with different camera views. HandMvNet achieves the highest FPS across all configurations.

5 CONCLUSION

We introduced HandMvNet, one of the first real-time methods for estimating 3D hand motion and shape from multi-view camera images. Our approach employs a multi-view attention-fusion mechanism that effectively integrates features from multiple viewpoints, delivering consistent and accurate absolute hand poses and shapes, free from the scale-depth ambiguities typically seen in monocular methods. Unlike previous multi-view approaches, HandMvNet eliminates the need for camera parameters to learn 3D geometry. We validated the architecture through extensive ablation studies and compared its performance with state-of-the-art methods. Experiments on public datasets demonstrate the effectiveness of our approach, delivering superior accuracy and inference speed compared to existing methods.

ACKNOWLEDGMENT

This research has been partially funded by the EU projects FLUENTLY (GA Nr 101058680) and Sharespace (GA Nr 10192889).

REFERENCES

Bartol, K., Bojanić, D., Petković, T., and Pribanić, T. (2022). Generalizable human pose triangulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11028–11037.

Boukhayma, A., Bem, R. d., and Torr, P. H. (2019). 3d hand shape and pose from images in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10843–10852.

Chao, Y.-W., Yang, W., Xiang, Y., Molchanov, P., Handa, A., Tremblay, J., Narang, Y. S., Van Wyk, K., Iqbal, U., Birchfield, S., et al. (2021). Dexycb: A benchmark for capturing hand grasping of objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9044–9053.

Chen, P., Chen, Y., Yang, D., Wu, F., Li, Q., Xia, Q., and Tan, Y. (2021a). I2uv-handnet: Image-to-uv prediction network for accurate and high-fidelity 3d hand mesh modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12929–12938.

Chen, X., Liu, Y., Ma, C., Chang, J., Wang, H., Chen, T., Guo, X., Wan, P., and Zheng, W. (2021b). Camera-space hand mesh recovery via semantic aggregation and adaptive 2d-1d registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13274–13283.

Chen, Z., Zhao, X., and Wan, X. (2022). Structural triangulation: A closed-form solution to constrained 3d human pose estimation. In *European Conference on Computer Vision*, pages 695–711. Springer.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

Ge, L., Liang, H., Yuan, J., and Thalmann, D. (2016). Robust 3d hand pose estimation in single depth images: from single-view cnn to multi-view cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3593–3601.

Ge, L., Ren, Z., Li, Y., Xue, Z., Wang, Y., Cai, J., and Yuan, J. (2019). 3d hand shape and pose estimation from a single rgb image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10833–10842.

Guan, H., Chang, J. S., Chen, L., Feris, R. S., and Turk, M. A. (2006). Multi-view appearance-based 3d hand pose estimation. *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)*, pages 154–154.

Hampali, S., Rad, M., Oberweger, M., and Lepetit, V. (2020). Honnotate: A method for 3d annotation of hand and object poses. In *CVPR*.

Han, S., Wu, P.-c., Zhang, Y., Liu, B., Zhang, L., Wang, Z., Si, W., Zhang, P., Cai, Y., Hodan, T., et al. (2022). Umetrack: Unified multi-view end-to-end hand tracking for vr. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9.

Hartley, R. and Zisserman, A. (2003). *Multiple view geometry in computer vision*. Cambridge university press.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

He, Y., Yan, R., Fragkiadaki, K., and Yu, S.-I. (2020). Epipolar transformers. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 7779–7788.

Iskakov, K., Burkov, E., Lempitsky, V., and Malkov, Y. (2019). Learnable triangulation of human pose. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7718–7727.

Lin, K., Wang, L., and Liu, Z. (2021). End-to-end human pose and mesh reconstruction with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1954–1963.

Liu, Y., Wang, T., Zhang, X., and Sun, J. (2022). Petr: Position embedding transformation for multi-view 3d object detection. In *European Conference on Computer Vision*, pages 531–548. Springer.

Loshchilov, I. (2017). Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Ma, H., Wang, Z., Chen, Y., Kong, D., Chen, L., Liu, X., Yan, X., Tang, H., and Xie, X. (2022). Ppt: token-pruned pose transformer for monocular and multi-view human pose estimation. In *European Conference on Computer Vision*, pages 424–442. Springer.

Malik, J., Abdelaziz, I., Elhayek, A., Shimada, S., Ali, S. A., Golyanik, V., Theobalt, C., and Stricker, D. (2020). Handvoxnet: Deep voxel-based network for 3d hand shape and pose estimation from a single depth map. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7113–7122.

Malik, J., Shimada, S., Elhayek, A., Ali, S. A., Theobalt, C., Golyanik, V., and Stricker, D. (2021). Handvoxnet++: 3d hand shape and pose estimation using voxel-based neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):8962–8974.

Moon, G. and Lee, K. M. (2020). I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single rgb image. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 752–768. Springer.

Park, J., Oh, Y., Moon, G., Choi, H., and Lee, K. M. (2022). Handoccnet: Occlusion-robust 3d hand mesh estimation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1496–1505.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Prakash, A., Gupta, A., and Gupta, S. (2023). Mitigating perspective distortion-induced shape ambiguity in image crops. *arXiv preprint arXiv:2312.06594*.

Remelli, E., Han, S., Honari, S., Fua, P., and Wang, R. (2020). Lightweight multi-view 3d pose estimation through camera-disentangled representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6040–6049.

Romero, J., Tzionas, D., and Black, M. J. (2022). Embodied hands: Modeling and capturing hands and bodies together. *arXiv preprint arXiv:2201.02610*.

Shuai, H., Wu, L., and Liu, Q. (2022). Adaptive multi-view and temporal fusing transformer for 3d human pose estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4122–4135.

Sun, K., Xiao, B., Liu, D., and Wang, J. (2019). Deep high-resolution representation learning for human pose estimation. In *CVPR*.

Sun, X., Xiao, B., Wei, F., Liang, S., and Wei, Y. (2018). Integral human pose regression. In *Proceedings of the European conference on computer vision (ECCV)*, pages 529–545.

Tu, H., Wang, C., and Zeng, W. (2020). Voxelpose: Towards multi-camera 3d human pose estimation in wild environment. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 197–212. Springer.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Yang, L., Li, K., Zhan, X., Wu, F., Xu, A., Liu, L., and Lu, C. (2022). Oakink: A large-scale knowledge repository for understanding hand-object interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20953–20962.

Yang, L., Xu, J., Zhong, L., Zhan, X., Wang, Z., Wu, K., and Lu, C. (2023). Poem: Reconstructing hand in a point embedded multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21108–21117.

Yu, Z., Yang, L., Chen, S., and Yao, A. (2021). Local and global point cloud reconstruction for 3d hand pose estimation. *arXiv preprint arXiv:2112.06389*.

Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C.-L., and Grundmann, M. (2020). Mediapipe hands: On-device real-time hand tracking. *arXiv preprint arXiv:2006.10214*.

Zhang, J., Cai, Y., Yan, S., Feng, J., et al. (2021a). Direct multi-view multi-person 3d pose estimation. *Advances in Neural Information Processing Systems*, 34:13153–13164.

Zhang, Z., Wang, C., Qiu, W., Qin, W., and Zeng, W. (2021b). Adafuse: Adaptive multiview fusion for accurate human pose estimation in the wild. *International Journal of Computer Vision*, 129:703–718.

Zhao, W., Wang, W., and Tian, Y. (2022). Graformer: Graph-oriented transformer for 3d pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20438–20447.

Zheng, X., Wen, C., Xue, Z., Ren, P., and Wang, J. (2023). Hamuco: Hand pose estimation via multiview collaborative self-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20763–20773.

Zhou, Y., Habermann, M., Xu, W., Habibie, I., Theobalt, C., and Xu, F. (2020). Monocular real-time hand shape and motion capture using multi-modal data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5346–5355.