Spectral Graphormer: Spectral Graph-based Transformer for Egocentric Two-Hand Reconstruction using Multi-View Color Images

Tze Ho Elden Tse[1,2*] Franziska Mueller[1] Zhengyang Shen[1] Danhang Tang[1] Thabo Beeler[1]
Mingsong Dou[1] Yinda Zhang[1] Sasa Petrovic[1] Hyung Jin Chang[2] Jonathan Taylor[1] Bardia Doosti[1]
[1]Google   [2]University of Birmingham

Abstract

We propose a novel transformer-based framework that reconstructs two high fidelity hands from multi-view RGB images. Unlike existing hand pose estimation methods, where one typically trains a deep network to regress hand model parameters from single RGB image, we consider a more challenging problem setting where we directly regress the absolute root poses of two-hands with extended forearm at high resolution from egocentric view. As existing datasets are either infeasible for egocentric viewpoints or lack background variations, we create a large-scale synthetic dataset with diverse scenarios and collect a real dataset from multi-calibrated camera setup to verify our proposed multi-view image feature fusion strategy. To make the reconstruction physically plausible, we propose two strategies: (i) a coarse-to-fine spectral graph convolution decoder to smoothen the meshes during upsampling and (ii) an optimisation-based refinement stage at inference to prevent self-penetrations. Through extensive quantitative and qualitative evaluations, we show that our framework is able to produce realistic two-hand reconstructions and demonstrate the generalisation of synthetic-trained models to real data, as well as real-time AR/VR applications.

1. Introduction

3D hand pose estimation is a fundamental problem in various downstream applications including augmented and virtual reality (AR/VR) [19, 20, 25, 26, 47, 58, 63]. Research efforts in designing 3D hand or hand-object reconstruction networks have shown great effectiveness for single-hand pose estimation [15, 22, 27, 59, 60, 66]. However, two-hand pose estimation has received relatively little attention. In this paper, as illustrated in Fig. 1, we focus on high fidelity two-hand reconstruction from multi-view RGB images.

With the availability of the InterHand2.6M dataset [46], various methods have been recently proposed. More recent
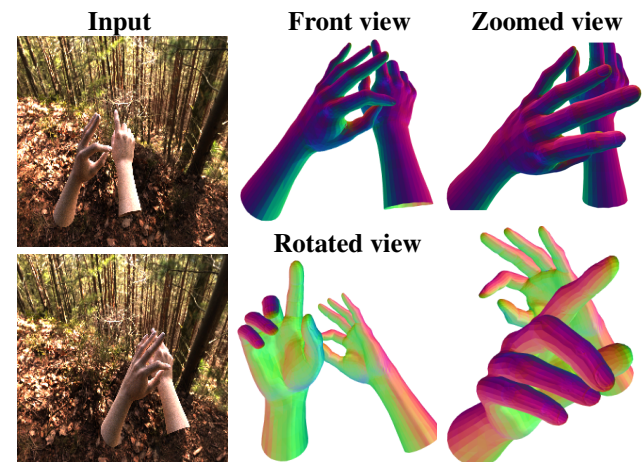

Figure 1: Our method jointly reconstructs high fidelity two-hand meshes from multi-view RGB image. The input images shown here are from our synthetic dataset, which contains challenging video sequences of two hands rendered into egocentric views.

methods focused on alleviating the self-similarity problem between interacting hands by exploiting hand part segmentation probability [18], joint visibility [33], cascaded refinement modules [65] or keypoints using Transformer [24]. Although they perform well on complex configurations, their dependence on root joint alignment does not provide absolute root pose recovery when applied to multi-view scenarios, which is especially crucial for interactions in VR.

Moreover, many immersive AR/VR applications require accurate estimation of two hands including extended forearms as it allows for a more realistic and accurate representation of hand movements and gestures. By including the forearm, the orientation and movement of the hand in relation to the arm can provide important contextual information for the user's actions in the virtual environment. Additionally, it can reduce errors and improve the stability of the tracking system, which is important for maintaining immersion and avoiding disorientation in AR/VR applications. However, as in many other areas of computer vision, there

---

is currently no suitable dataset for a typically supervised deep learning setup. Specifically, existing datasets are either infeasible for egocentric views [21] or lack variations in background and lighting conditions [46].

In this paper, we propose a spectral graph-based transformer architecture that can reconstruct high resolution two-hand meshes with extended forearms from multi-view RGB images. However, directly extending state-of-the-art methods [39, 40] from single-view to multi-view setting is nontrivial as they consume a substantial number of parameters and are computationally expensive. As pointed out by Cho *et al.* [8], the current encoder-based transformers [39, 40] have overlooked the importance of efficient token design. In order to retain spatial information in image features and avoid passing redundant global image features to the transformer encoder, we propose a soft-attention-based multi-view image feature fusion strategy to obtain region-specific features. Further, as a hand mesh in essence is graph-structured data, we incorporate properties of graph Laplacian from spectral graph theory [10] into the design of our transformer-based network architecture. To obtain realistic hand reconstructions, we leverage a hierarchical graph decoder and propose an optimisation-based refinement stage during inference to prevent self-penetrations. We show that each technical component above contributes meaningfully in our ablation study.

Besides the expensive computational cost, an additional challenge is the lack of high fidelity 3D mesh ground truth for training such a model. As manually annotating 3D hand meshes on real data is extremely laborious and expensive, we create a large-scale synthetic multi-view dataset containing realistic hand motions from egocentric camera viewpoints rendered under a large variation of background and illumination. To generalise and evaluate on real-world data, we further collected real data from a precisely calibrated multi-view studio with 18 number of high-resolution cameras. We use an automatic approach for registering meshes and obtain ground truth hand poses. Our proposed method trained on both datasets shows robust performance on challenging scenarios and can serve as a strong baseline.

Our contributions are the following:

1. We propose a novel end-to-end trainable spectral graph-based transformer for high fidelity two-hand reconstruction from multi-view RGB image.

2. We design an efficient soft attention-based multi-view image feature fusion in which the resulting image features are region-specific to segmented hand mesh. We further demonstrate a minimal reduction of 35% in the model size with this approach.

3. We introduce an optimisation-based method to refine physically-implausible meshes at inference.

4. We create a large-scale synthetic multi-view dataset

with high resolution 3D hand meshes and collect real dataset to verify our proposed method.

2. Related Work

Our work tackles the problem of two-hand reconstruction from multi-view images. We first review the literature on *Hand reconstruction* from RGB images. Then, we focus on the line that leverages *Transformer* on human body/hand reconstruction. Finally, we provide a brief review on existing *Hand datasets*.

Single-Hand Reconstruction. Many approaches have already been proposed for hand pose estimation from either RGB images or depth maps. Here we focus mainly on works that reconstruct full hand meshes. As most earlier methods do not have access to suitable datasets which contain dense 3D ground truths, the parametric hand model MANO [52] provides an alternative as its input parameters can be directly regressed and it contains prior 3D information about hand geometry. Hence, discriminative approaches [2, 5, 27] which directly predict input hand pose and shape parameters from single images have been a popular approach. [45] learns the pose and shape corrective parameters to reconstruct high-fidelity hand meshes with multi-view depth images. Concurrently, non-parametric methods [22, 36] leveraging graph convolutions demonstrate the ability to directly regress 3D hand meshes. Recently, there is an increasing interest in representing object shapes by learning an implicit function [1, 7, 42]. By extending NASA [13], [31] is a keypoint-driven implicit hand representation and [11] disentangles shape, color and pose representations.

In contrast to the above mentioned approaches, we propose in this work a non-parametric method that can be extended to multi-view settings and reconstruct high-fidelity hand meshes with extended forearms. Further, our approach is not restricted to a single subject, which is the primary constraint of [45]. Recent work on multi-view fusion leverage epipolar geometry [29, 50] to aggregate estimated 2D heatmaps from different views or Feature Transform Layers (FTL) [26, 51] to learn camera geometry-aware latent features. However, these methods require known camera parameters and the feature matching process based on feature similarity is sensitive to occlusion. Our work introduces a soft-attention mechanism coupled with transformer tokenisation which is robust to occlusion and aggregates multi-view image features efficiently.

Two-Hand Pose Estimation. Pose estimation of interacting hands can be broadly classified as discriminative and generative (or hybrid) approaches. Early methods typically follow hybrid approaches by leveraging visual cues detected by discriminative methods, followed by model fitting. [3, 61] apply collision optimisation terms and physical modelling on detected fingertips. [25, 26, 47, 63] ex-
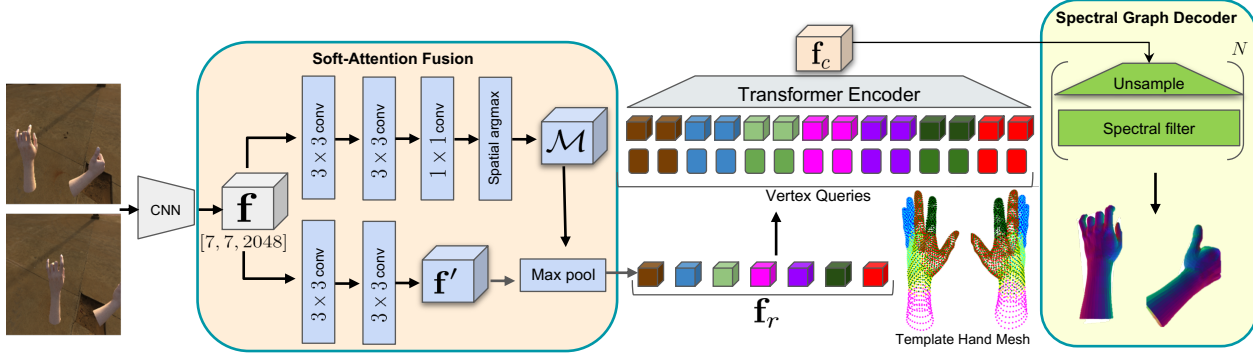
Figure 2: A schematic illustration of our framework. Given multi-view RGB images, we extract volumetric features $\mathbf{f}$ with a shared CNN backbone. The soft-attention fusion block generates the attention mask $\mathcal{M}$ and finer image features $\mathbf{f}'$ through multiple upsampling and convolution blocks. Region-specific features $\mathbf{f}_r$ are computed by first aggregating $\mathbf{f}'$ along the feature channel dimension via the attention mask $\mathcal{M}$, followed by a max-pooling operation across multi-view images to focus on useful features. Then, we apply mesh segmentation via spectral clustering on template hand meshes and uniformly subsample them to obtain coarse meshes. We perform position encoding by concatenating coarse template meshes to the corresponding region-specific features $\mathbf{f}_r$, *i.e.* matching colored features to mesh segments. Finally, our multi-layer transformer encoder takes the resulting features as input and outputs a coarse mesh representation $\mathbf{f}_c$ which is then decoded by a spectral graph decoder to produce the final two-hand meshes at target resolution. Here, each hand contains 4023 vertices.

tract image features or keypoints from RGB or depth images and fit hand models with physical constraints. Recent large-scale datasets of interacting hands [46] have led to the emergence of fully discriminative methods [24,33,38,53,65] that jointly estimate the 3D joint locations or hand model parameters from a single RGB image. In this work, we target to reconstruct two hands at high resolution while keeping the model complexity low.

Transformer in 3D Vision. Transformer-based architectures have been gaining increasing popularity in the vision community. Here we focus on methods that reconstruct the human body or hands from RGB images and refer readers to [32] for a detailed survey. In a closely related work, [39] uses cascaded Transformer encoders to reconstruct the human body and hands from a single RGB image and achieves state-of-the-art performance. [40] extends [39] with graph convolutions along the Transformer encoder. [8] improves the efficiency of [39] by disentangling image encoding and mesh estimation via an encoder-decoder architecture. [24] extends "Detection Transformer" [6] with hand-object pose estimations. While these works are targeted at a single image and their extension to the multi-view setting is non-trivial due to large number of learnable parameters, our architecture is designed to efficiently reconstruct two hands at high resolution.

Hand Pose Datasets. The success of discriminative methods depend on the availability and variability of hands interaction datasets. [46] introduces the large-scale dataset InterHand2.6M with closely interacting hand motions using a semi-automatic annotation process with multiple high-resolution RGB cameras. However, it contains minimal

background and lighting variation. FreiHAND [67] includes wider background variation but the dataset is limited to a single hand and third-person views. Therefore, the above datasets are not suitable for the egocentric two-hand reconstruction task. The available egocentric datasets are limited by either visible markers [21] or constrained lab environments [37]. This motivates us to create a large-scale egocentric synthetic dataset with improved environment and lighting variations. To validate our proposed multi-view fusion strategy, we further collect a real dataset with more challenging camera viewing angles. Altogether, both synthetic and real datasets contain diverse egocentric multi-view and -frame data points with multiple subjects. Our architecture demonstrates promising performance on both datasets and constitutes a strong baseline.

3. Method

As shown in Fig. 2 , our architecture first passes N number of multi-view RGB input images $\mathbf{x} \in \mathbb{R}^{N \times 224 \times 224 \times 3}$ to a shared CNN backbone to extract volumetric features $\mathbf{f} \in \mathbb{R}^{N \times 7 \times 7 \times 2048}$, *i.e.* features before the global average pooling layer for ResNet [28]. The volumetric features are then fed into a soft attention-based multi-view feature encoder and output K region-specific features $\mathbf{f}_r \in \mathbb{R}^{K \times C}$ where C refers to the feature channel size. The Transformer encoder takes $\mathbf{f}_r$ together with template hand meshes $\mathbf{m}' \in \mathbb{R}^{V' \times 3}$ and outputs a coarse mesh representation $\mathbf{f}_c \in \mathbb{R}^{V' \times F}$. Finally, the spectral graph decoder generates hand meshes $\mathbf{m} \in \mathbb{R}^{V \times 3}$ by upsampling on $\mathbf{f}_c$ where $V \gg V'$. With slight abuse of notation, $\mathbf{m}$ can either be two-hand or single-hand depending on the application.

In the following, we detail the multi-view image feature encoder in Section 3.2, the spectral graph convolution decoder in Section 3.3 and the loss used for training in Section 3.4. We also present an optimisation-based refinement procedure at inference time in Section 3.5.

3.1. Notations

A 3D mesh can be represented as an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{A})$, where $\mathcal{V}$ is a node set and $\mathcal{E}$ is an edge set. An adjacency matrix $\mathbf{A} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$ encodes information of pairwise relations between nodes. A degree matrix $\mathbf{D} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$ is a diagonal matrix whose diagonal element $\mathbf{D}_{ii} = \sum_j \mathbf{A}_{ij}$ refers to the degree value of each node. An essential operator in spectral graph theory [10] is the graph Laplacian $\mathbf{L}$, whose definition is $\mathbf{L} = \mathbf{D} - \mathbf{A}$, and $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$ where the graph Laplacian can be diagonalised by the Fourier basis $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_{|\mathcal{V}|}] \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$ and $\mathbf{\Lambda} = diag([\lambda_1, \dots, \lambda_{|\mathcal{V}|}]) \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$ where $\{\mathbf{u}_i\}_{i=1}^{|\mathcal{V}|}$ are the eigenvectors and $\{\lambda_i\}_{i=1}^{|\mathcal{V}|}$ are the non-negative eigenvalues of graph Laplacian ($0 = \lambda_1 \leq \dots \leq \lambda_{|\mathcal{V}|}$).

3.2. Multi-View Image Feature Encoder

To extend transformer-based architecture to multi-view settings and prevent concatenating global image features in an overly-duplicated way as in [39], we propose a simple *soft-attention fusion* strategy to better aggregate features across multiple views and attend to different hand parts via *mesh segmentation*. The resulting K region-specific features $\mathbf{f}_r$ defined by *spectral clustering* are fed to our *Transformer encoder* to obtain a coarse mesh representation $\mathbf{f}_c$.

Soft-Attention Fusion. Given volumetric features $\mathbf{f}$, we do not apply any pooling operations to avoid losing spatial information as shown by our experiments (see Table 3). Instead, we propose to aggregate multi-view features with a soft-attention mask. We first obtain a finer representation of $\mathbf{f}$, denoted as $\mathbf{f}' \in \mathbb{R}^{N \times (H \times W) \times C}$, by feeding it through two blocks each comprised of 2D upsampling with bilinear interpolation, 3×3 convolution layers, batch-normalisation [30] and ReLU. The soft-attention mask $\mathcal{M} \in \mathbb{R}^{N \times (H \times W) \times K}$ is obtained by applying 1) K 1×1 convolutional filters to reduce the feature channel of $\mathbf{f}'$ to K and 2) spatial soft arg-max which determines the image-space point of maximal activation in each C. At this stage, we can compute per-frame features $\mathbf{f}'' \in \mathbb{R}^{N \times K \times C}$ by $\mathbf{f}'' = \mathcal{M}^\mathrm{T}\mathbf{f}'$. We finally obtain region-specific features $\mathbf{f}_r \in \mathbb{R}^{K \times C}$ by max-pooling along the second dimension of $\mathbf{f}''$. There are two intuitions to this design: 1) the corresponding feature with higher attention weight contributes more to the final feature representation spatially and 2) max-pooling allows efficient feature selection across multiple views and it does not overfit to any multi-camera configurations as CNN filters are shared across all views.

Mesh Segmentation via Spectral Clustering. Inspired by [41], we perform spectral clustering to obtain a 3D mesh segmentation. Different from [41], we apply the eigen-decomposition of the graph Laplacian $\mathbf{L}$ instead of the affinity matrix (which encodes pairwise point affinities with exponential kernel), followed by k-means clustering into K clusters. With this approach, one can efficiently segment any template model mesh without manual effort. In addition, as shown in Fig. 3, the segmented mesh does not exhibit clear-cut boundaries, and certain clusters are scattered throughout the entire surface. This is in spirit similar to mask vertex modeling occlusions in [39] by encouraging the transformer to consider other relevant vertex queries.

Transformer Encoder. We now concatenate the C-dimensional region-specific image features to the corresponding K clusters of the segmented template hand mesh. The resulting features $\mathbf{f}_t \in \mathbb{R}^{V' \times (C+3)}$ [1] are fed into a multi-layer transformer encoder with progressive dimensionality reduction as described in [39]. The output is a coarse mesh representation $\mathbf{f}_c \in \mathbb{R}^{V' \times F}$, where $F = (C + 3)/2^n$ with n layers of transformer encoder.

Theoretical Motivations. Since there do not exist explicit coordinate systems as in grid graphs, aligning nodes of highly irregular graphs in nature is a non-trivial problem. Recent studies [16, 17, 35] attempted to encode positional information by leveraging the spectral domain in a way that nearby nodes have similar values and distant nodes have different values. This can be achieved as the eigenvectors of the graph Laplacian can be interpreted as the generalised concepts of sinusoidal functions of positional encoding in Transformers [62]. For instance with any given graph, nodes close to each other can be assigned values of similar positional features as the smaller the eigenvalue of the graph Laplacian λ_i (closer to 0) the smoother the coordinates of the corresponding eigenvector $\mathbf{u}_i$.

3.3. Spectral Graph Decoder

We find that simply relying on fully-connected layers to upsample meshes to target resolution is insufficient as this process introduces instability and disruption to the mesh, even with mesh regularisation loss terms [27] (see Fig. 4). Therefore, we propose to couple the fully-connected layers for upsampling with *spectral filtering* as meshes can be treated as graph signals $\mathbf{f}_c = (f_1, \cdots, f_{V'}) \in \mathbb{R}^{V' \times F}$, *i.e.* V' vertices with F-dimensional features for batch size of 1.

Spectral Filtering. As spectral graph convolution can be defined via point-wise products in the transformed Fourier space, graph signals $\mathbf{f}_c$ filtered by g_θ can then be expressed as $\mathbf{U}g_\theta(\mathbf{\Lambda})\mathbf{U}^T\mathbf{f}_c$ where $g_\theta(\mathbf{\Lambda})$ is a diagonal matrix with Fourier coefficients. To ensure that the spectral filter corre-

---

[1] +3 refers to the 3D positions of mesh template.
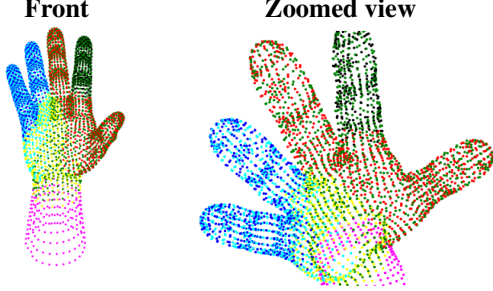
Front **Zoomed view**

Figure 3: Illustration of mesh segmentation via spectral clustering. Here, we have chosen $K = 7$ clusters and each has its own color for a single right-handed template mesh. As shown, there are no rigid boundaries across the mesh. To prepare inputs $\mathbf{f}_t$ to transformer, we decrease the size of this mesh (similarly for left hand) uniformly by a factor of 10 through subsampling, *i.e.* $V' = V/10$ and concatenate with the corresponding region-specific features $\mathbf{f}_r$.

sponds to a meaningful convolution on the graph, a natural solution is to parameterise based on the eigenvalues of the Laplacian. Here we have used the Chebyshev polynomial parametrisation of $g_\theta(\mathbf{L})$ for fast computation and readers are referred to [12] for more details.

Architecture. Similar to [9, 22], our decoder is based on a hierarchical architecture where the mesh is recovered by using fully-connected and graph convolution layers for upsampling. We followed [12, 22] to pre-computed coarse graphs using [14] and used the third-order polynomial in the Laplacian. Given the coarse mesh representation $\mathbf{f}_c \in \mathbb{R}^{V' \times F}$, the key idea here is to smooth out the F-dimensional values after upsampling on V'. Note that the fundamental difference between this approach and the one in [9, 22] is that they have a hierarchical architecture on both V' and F dimensions, we apply only spectral filtering and keep $F = 3$ constant which massively reduces the model size while maintaining performance (see Table 1).

Discussions. Recall that in Section 3.2, we leverage the properties of Laplacian eigenvectors to perform spectral clustering. Here we can interpret from a signal processing perspective where the Laplacian eigenvectors define signals that vary smoothly across the graph, with the smoothest signals indicating the coarse community structure of the mesh.

3.4. Training

Following [39, 40], our model can be trained end-to-end with L1 losses on 3D mesh vertices and 2D re-projection using the predicted camera parameters to improve image-mesh alignment. In addition, we apply an edge length regularisation $\mathcal{L}_{edge}$ [27] to encourage smoothness of the mesh:

$$\mathcal{L}_{edge}(\mathbf{m}) = \frac{1}{|\mathcal{E}_L|} \sum_{l \in \mathcal{E}_L} |l^2 - \mu(\mathcal{E}_L^2)|, \qquad (1)$$

where $\mathcal{E}_L$ refers to the set of edge lengths, defined as the L2 norms of all edges and $\mu(\mathcal{E}_L^2)$ is the average of the squared edge lengths.

3.5. Mesh Refinement at Inference

There are two main approaches in the literature for producing realistic mesh reconstruction: learning-based and optimisation-based methods. Learning-based methods [27, 45] approach the problem by proposing various repulsive losses that penalise penetration during training. However, their generalisation ability to other meshes is limited due to the need for mesh-specific pre-computation. In particular, they require manual selection of areas of interest (*i.e.* fingertips and palm) which does not prevent other forms of self-penetrations, such as finger-finger. More importantly, the presence of interpenetration at test time shows that the model is not able to learn the physical rule implicitly [27, 59]. On the other hand, recent optimisation-based methods [23, 60] leverage contact maps to refine meshes at inference. However, they rely heavily on accurate contact map estimations and are sensitive to initialisation as contact optimisation is local.

Proposed Method. To avoid penetrations, we extend the repulsion loss from [27] into an optimisation-based strategy which does not require any form of pre-computation and is more generalisable to other meshes. To identify hand vertices that contribute to collision, we follow [27, 43] and cast rays from each vertex and count the number of surface intersections. If the number is odd, it indicates penetration. After obtaining the collision mask $\mathcal{M}_C$, we compute the nearest point in the source mesh $\mathbf{m}$ with respect to the set of collision points. If the point and its nearest corresponding point have a different normal, we compute their distance as loss. We minimise this collision loss $\mathcal{L}_{collision}$ with as-rigid-as-possible (ARAP) [56] regularisation on the mesh shape:

$$\mathcal{L}_{collision}(\mathbf{m}, \mathcal{M}_C) = \sum_{v \in \mathcal{V}} \mathcal{M}_C \cdot d(v, \mathcal{V}), \qquad (2)$$

where $\mathcal{M}_C = \mathbb{1}_{v \in \text{Int}(\mathcal{V})}$ indicates which vertices belong to the interior of the mesh and $d(v, \mathcal{V}) = \min_{v' \in \mathcal{V}} \|v - v'\|_2$ denotes distances from point v' to set $\mathcal{V}$. We show that our method is able to remove self-penetration in Fig. 5 and 9.

4. Experiments

Implementation Details. We implement our method in TensorFlow. We train all parts of the network simultaneously with the Adam optimiser [34] using a learning rate of 10^{-4} when training on the synthetic dataset and 10^{-5} when fine-tuning on the real dataset. We use ResNet [28] pre-trained on ImageNet [54] for our CNN backbone. For computing region-specific features $\mathbf{f}_r$, we set $K = 7$ and $C = 256$.
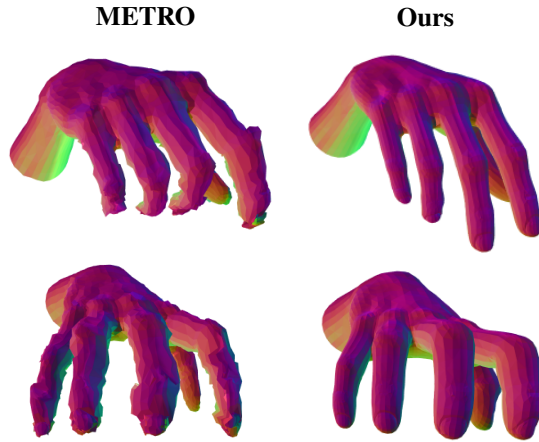
METRO   **Ours**

Figure 4: Qualitative comparison with METRO [39]. We show that relying on fully-connected layers to upsample meshes is inadequate for high-resolution mesh reconstruction. In contrast, our spectral graph decoder can accurately capture intricate surface features like nails.
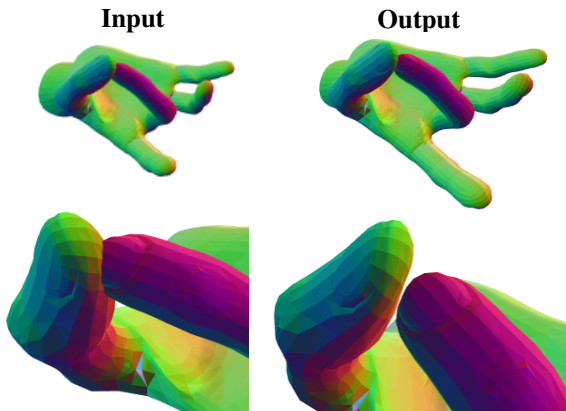

Input   **Output**

Figure 5: Qualitative example of mesh refinement at inference. More examples are available in Fig. 9.

Synthetic Dataset Creation. A large-scale multi-view ego-centric dataset with challenging interacting hand motion is required to train our pipeline. However, existing datasets are either infeasible for egocentric views or lack variations in terms of background and lighting conditions, which are crucial for AR/VR applications. In addition, the ground-truth mesh in [46] contains $5mm$ fitting error which is sufficient to cause inaccuracy in hand tracking or misalignment when interacting with virtual objects. Therefore, it is not suitable to validate our approach. To this end, we create a new large-scale synthetic egocentric dataset with two high-fidelity 3D hand meshes. In particular, we purchase commercial 3D hand models for both left and right hands. Each of them contains 4023 vertices and 4008 quad faces. By dividing each quad into two triangles, they can be decomposed into 8016 triangular faces. The size and textures of the hand
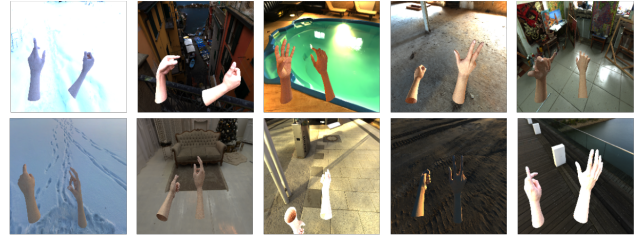

Figure 6: Qualitative examples of our synthetic dataset. The scenes are rendered with large variations in lighting, texture and background.

models can vary, and it can also be rigged with 3D joint locations. We apply photorealistic textures as well as natural lighting using High-Dynamic-Range (HDR) images. To create realistic hand motions, we generate 100 poses and interpolate between pairs of these poses randomly over 1000 frame sequences, *i.e.* pose pair is swapped every 10 frames. Then, we render hands onto 480 4K backgrounds using the Cycles renderer [49]. Our synthetic dataset comprises 1M data points. Each data point consists of 3D annotations for two hands, rendered in two egocentric views to simulate a customised camera headset scenario. We divide the entire dataset into 4 groups and use 25% randomly for testing and provide additional image examples in Fig. 6.

Real Dataset Collection. Our real hand data is captured from a multi-view stereo system with 18 synchronised Z-Cams. We build a NeRF-based [4, 48] reconstruction pipeline that simultaneously reconstructs and disentangles the foreground and the background, from which a meshing module extracts hand meshes as ground-truth target mesh. We then apply a mesh and tetrahedral registration approach [55] that registers a template hand model [64] to get a well-registered mesh for each reconstruction.

Baselines. We compare with the state-of-the-art transformer-based architecture METRO [39] that is explicitly designed for human body mesh reconstruction. We follow their official implementation and use 3 transformer encoder layers. We attempted to include other strong baselines to compare with. However, given our challenging multi-view settings on reconstructing high resolution two-hand meshes, existing methods are constrained by either 1) relying on intermediate 3D pose supervision [9, 44] or 2) directly regressing a low resolution parametric hand model [2, 5, 27, 59, 67]. In addition, the direct extension of METRO [39], Mesh Graphormer [40], is infeasible to extend to multi-view settings as they tokenised volumetric features which increases the computational complexity of each transformer layer quadratically [8]. Therefore, by considering that the performance gain is minor, we pick METRO [39] as a strong baseline and study in-depth. We provide more details about the baselines in the supplementary materials.

Table 1: Error rates on our synthetic dataset. Parametric baseline is trained and tested on right hand only. Our proposed method achieves strong performance with less than half the METRO model size.

	Hand error	# Params
Parametric baseline	24.5	40.9M
METRO [39]	7.09	116.1M
Ours (w/o graph decoder)	3.72	75.2M
Ours	**1.38**	58.3M

Table 2: Error rates (in mm) on FreiHAND.

	METRO [39]	Graphormer [40]	Ours
PA-MPVPE ↓	6.7	5.9	**5.5**
PA-MPJPE ↓	6.8	6.0	**5.6**

Evaluation Metric. We report the Mean-Per-Vertex-Error (MPVE) in mm to evaluate the *hand reconstruction error*. MPVE measures the mean Euclidean distances between the ground-truth vertices and the predicted vertices.

Quantitative Comparison. We perform a quantitative comparison for the *two-hand reconstruction* task on our synthetic dataset. As previously mentioned, comparing with other methods is not straightforward since the majority of existing methods focus on single-view settings. We extend METRO [39] to the multi-view setting by applying max-pool to image features and concatenating them with vertex queries. For fair comparisons, we share the same hyperparameter setting for the multi-layer transformer encoder and report the results in Table 1. In this experiment, we downsample the template hand mesh by 10 times. To provide more context regarding the difficulty of our synthetic dataset, we also experiment on a parametric baseline. We create a parametric hand model which uses the 200-dimensional PCA (principal component analysis) subspace from 25% of the training data. We use ResNet-50 as the backbone for all models in Table 1 and the parametric baseline has an MLP head to predict the input parameters to recover the hand mesh. Our method significantly outperforms both baseline methods with less than half the model size of METRO. We show that our soft-attention feature fusion strategy coupled with mesh segmentation using spectral clustering can effectively reduce the feature channel size from 2048 to 256 without performance drop. In addition, the performance of the parametric baseline is in-line with existing single-hand benchmarks. Lastly, we report results on single-view benchmark FreiHAND [67] in Table 2.

Ablation Study. To motivate our design choices, we present a quantitative evaluation of our method with various components disabled. We validate that each of our proposed technical component contributes meaningfully.

Table 3: Performance of different multi-view fusion strategies. We report hand error for both settings. K refers to the number of clusters for template hand mesh. Note that we do not include spectral filtering in the graph decoder here.

	Single-view	Multi-view
METRO [39]	10.87	-
METRO [39] + avg. pool	-	8.71
METRO [39] + max pool	-	7.09
Ours ($K = 1$)	-	6.59
Ours ($K = 4$)	-	4.79
Ours ($K = 7$)	-	**3.72**

Table 4: Ablations on multi-view feature fusion. We report hand error on 3 experimental settings ((a)-(c)) with different feature fusion strategies. We keep the same transformer and spectral graph decoder for all fusion strategies.

	(a)	(b)	(c)
Direct concatenation	11.3	13.9	14.7
Max pool	9.3	11.2	12.5
Soft-attention + max pool	**6.7**	**7.2**	**7.4**

Table 5: Ablations of different spectral filters.

	g_{gau}	g_{lap}	g_{cheb_3}	g_{cheb_4}	g_{cheb_5}	g_{cheb_6}
Hand error	1.56	1.45	1.38	1.47	1.38	1.39

Effects of Multi-View Feature Fusion. Table 3 shows the results of varying number of spectral clusters K (full results in supp.). The combination of soft-attention fusion and mesh segmentation consistently improves the performance. As our synthetic dataset contains only two views, we further verify our multi-view fusion strategy on more challenging viewing angles captured in our real dataset. We also demonstrate that our method does not overfit to camera setup when tested on unseen camera views in Table 4. In these experiments, we divide the 18 camera views into 2 groups, *i.e.* the first group contains the first 15 camera views and the second group contains the last 3 unseen camera views for evaluation. Note that the evaluation group contains the only egocentric view. We consider 3 experimental settings by varying number of camera views present in the training data: (a) 15 views, (b) 6 views and (c) 3 views in Table 4.

Effects of Spectral Filters. We experiment with three commonly-used spectral filters: Gaussian g_{gau}, Laplacian g_{lap} and Chebyshev filters g_{cheb} with varying order of polynomials in Table 5. We provide implementation details and additional qualitative examples in the supplementary. We choose g_{cheb} for efficiency and do not find increasing order of polynomials improves performance further.
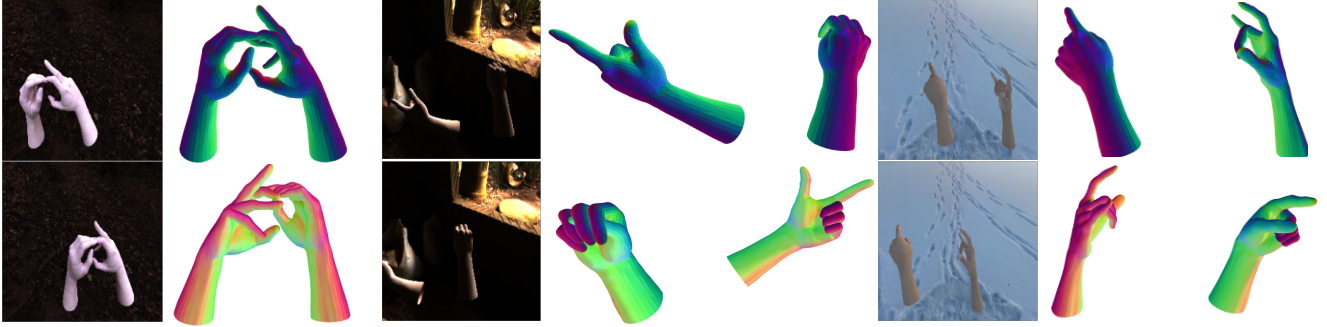
Figure 7: Qualitative examples on our synthetic dataset. Our model shows robustness to various extreme self-occlusions (left), lighting conditions (middle) and scenes (right). Fine mesh reconstruction details are well-preserved in all scenarios.

Table 6: Ablations of different backbones and hyperparameters. We denote P_{cnn} and P_{total} to be the number of parameters for CNN backbone and total model, respectively.

Backbone	P_{cnn}	V'	C	Error	P_{total}
ResNet-50	23.5M	804	256	1.38	58.3M
EfficientNet-B0	5.3M	160	256	4.12	42.8M
EfficientNet-B0	5.3M	160	128	4.96	37.8M
EfficientNet-B0	5.3M	80	64	**6.89**	**34.2M**

Figure 8: Training examples of our real dataset (*top*) and 3D hand mesh estimation results on in-the-wild image (*bottom*). The resulting mesh contains 3745 vertices.

Synthetic to Real Transfer. Large-scale synthetic dataset can be used to pre-train models in the absence of suitable real datasets. In the following, we demonstrate a use case of deploying our pre-trained model which was trained on synthetic data to general real images. First, we train on a subset of our real dataset and freeze all part of the pre-trained model except CNN backbone. As the registered meshes for our real dataset have a different mesh topology, we drop the pink cluster of our original template mesh (shown in Fig. 3) before training. This approach is analogous to

MANO-based 3D hand mesh estimation methods [2, 5] and has been demonstrated in [45]. We show that our model can generalise to in-the-wild-images in Fig. 8.

Model Compression. We are interested in finding the minimal size to which model can be compressed while maintaining the METRO baseline performance in Table 1. First, we perform computational complexity analysis on the self-attention layer in the Transformer encoder. There are two key steps to compute self-attention: 1) linear projection to C-dimensional query, key and value matrices from V' vertex queries requires $O(V'C^2)$ and softmax operation for layer output requires another $O(V'^2C)$. The total computational complexity of each transformer layer is therefore quadratic no matter whether V' or C dominates. Recall that as we uniformly subsample template hand meshes by 10 times to obtain $V' = 804$ and C is empirically set to 256 in earlier sections, V' contributes more to overall complexity as it is dominantly larger. In the following experiments, we gradually decrease V' before C while using a variant of EfficientNet [57]. We present main results in Table 6, while full results are provided in the supplementary.

5. Conclusion

In this paper, we have proposed a novel spectral graph-based transformer framework which reconstructs high-fidelity two-hand meshes from egocentric views. The main idea behind this study was to demonstrate that the fundamental properties of the graph Laplacian from the spectral graph theory can be applied to a Transformer. We have shown that our multi-view feature fusion strategy is most effective when coupled with mesh segmentation using spectral clustering. We further present spectral filtering and an optimisation-based refinement to reconstruct more physically plausible meshes. Our framework is general and can be extended to other multi-view reconstruction tasks.

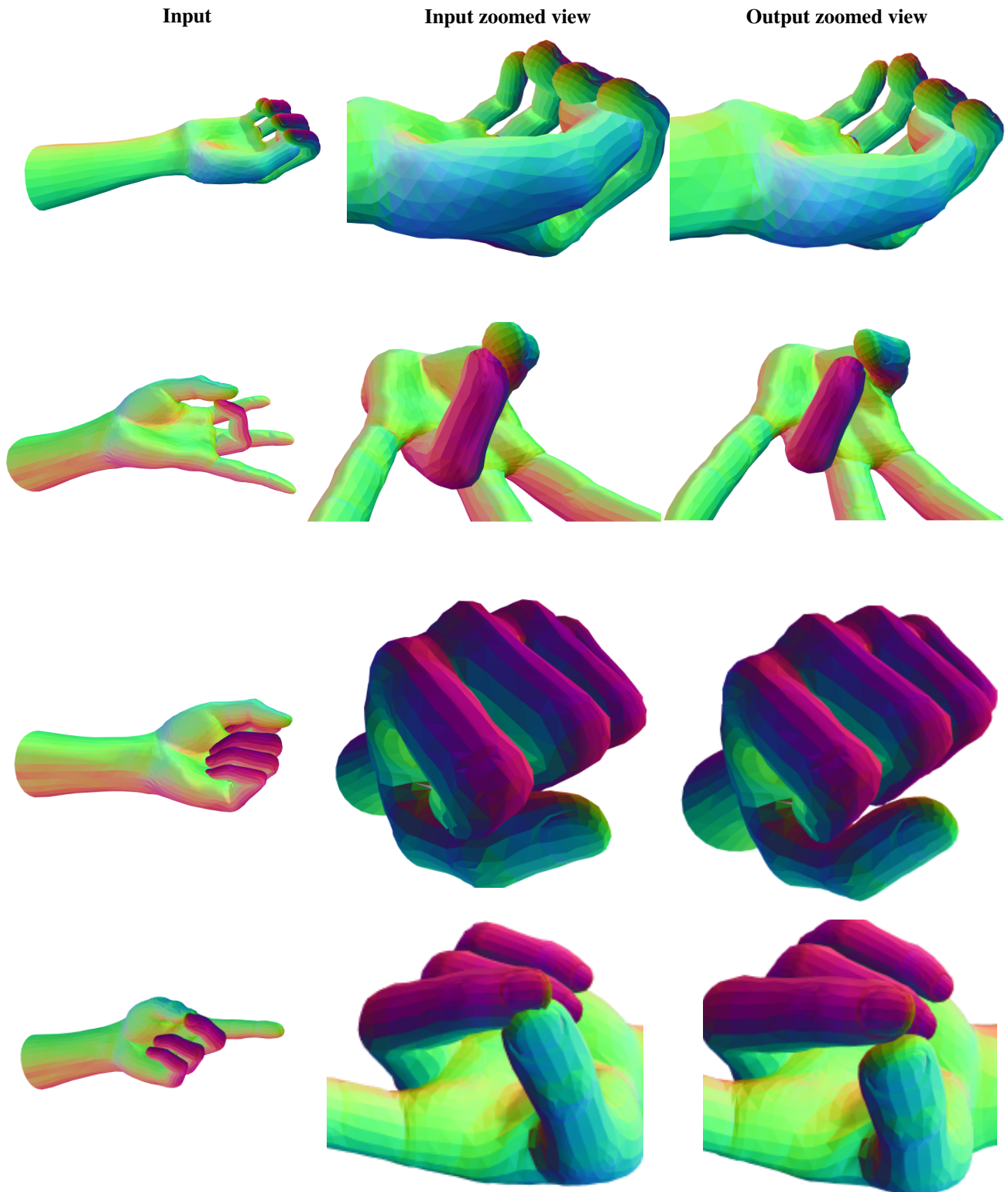| | Input | Input zoomed view | Output zoomed view |

Figure 9: Additional qualitative examples of mesh refinement at inference. Our optimisation-based strategy shows robustness to various hand poses. We present failure cases when self-penetration is highly complex in the supplementary materials.

References

[1] Matan Atzmon and Yaron Lipman. SAL: Sign agnostic learning of shapes from raw data. In *CVPR*, 2020. 2

[2] Seungryul Baek, Kwang In Kim, and Tae-Kyun Kim. Pushing the envelope for RGB-based dense 3D hand pose estimation via neural rendering. In *CVPR*, 2019. 2, 6, 8

[3] Luca Ballan, Aparna Taneja, Jürgen Gall, Luc Van Gool, and Marc Pollefeys. Motion capture of hands in action using discriminative salient points. In *ECCV*, 2012. 2

[4] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-NeRF 360: Unbounded anti-aliased neural radiance fields. In *CVPR*, 2022. 6

[5] Adnane Boukhayma, Rodrigo de Bem, and Philip HS Torr. 3D hand shape and pose from images in the wild. In *CVPR*, 2019. 2, 6, 8

[6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 3

[7] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *CVPR*, 2019. 2

[8] Junhyeong Cho, Kim Youwang, and Tae-Hyun Oh. Cross-attention of disentangled modalities for 3D human mesh recovery with transformers. In *ECCV*, 2022. 2, 3, 6

[9] Hongsuk Choi, Gyeongsik Moon, and Kyoung Mu Lee. Pose2Mesh: Graph convolutional network for 3D human pose and mesh recovery from a 2D human pose. In *ECCV*, 2020. 5, 6

[10] Fan RK Chung. *Spectral graph theory*. American Mathematical Soc., 1997. 2, 4

[11] Enric Corona, Tomas Hodan, Minh Vo, Francesc Moreno-Noguer, Chris Sweeney, Richard Newcombe, and Lingni Ma. LISA: Learning implicit shape and appearance of hands. In *CVPR*, 2022. 2

[12] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *NeurIPS*, 2016. 5

[13] Boyang Deng, John P Lewis, Timothy Jeruzalski, Gerard Pons-Moll, Geoffrey Hinton, Mohammad Norouzi, and Andrea Tagliasacchi. NASA neural articulated shape approximation. In *ECCV*, 2020. 2

[14] Inderjit S Dhillon, Yuqiang Guan, and Brian Kulis. Weighted graph cuts without eigenvectors a multilevel approach. *TPAMI*, 2007. 5

[15] Bardia Doosti, Shujon Naha, Majid Mirbagheri, and David J Crandall. HOPE-Net: A graph-based model for hand-object pose estimation. In *CVPR*, 2020. 1

[16] Vijay Prakash Dwivedi and Xavier Bresson. A generalization of transformer networks to graphs. *AAAI Workshop on Deep Learning on Graphs: Methods and Applications*, 2021. 4

[17] Vijay Prakash Dwivedi, Chaitanya K Joshi, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. Benchmarking graph neural networks. *JMLR*, 2022. 4

[18] Zicong Fan, Adrian Spurr, Muhammed Kocabas, Siyu Tang, Michael J Black, and Otmar Hilliges. Learning to disambiguate strongly interacting hands via probabilistic per-pixel part segmentation. In *3DV*, 2021. 1

[19] Runyang Feng, Yixing Gao, Xueqing Ma, Tze Ho Elden Tse, and Hyung Jin Chang. Mutual information-based temporal difference learning for human pose estimation in video. In *CVPR*, 2023. 1

[20] Runyang Feng, Yixing Gao, Tze Ho Elden Tse, Xueqing Ma, and Hyung Jin Chang. DiffPose: SpatioTemporal diffusion model for video-based human pose estimation. *arXiv preprint arXiv:2307.16687*, 2023. 1

[21] Guillermo Garcia-Hernando, Shanxin Yuan, Seungryul Baek, and Tae-Kyun Kim. First-person hand action benchmark with RGB-D videos and 3D hand pose annotations. In *CVPR*, 2018. 2, 3

[22] Liuhao Ge, Zhou Ren, Yuncheng Li, Zehao Xue, Yingying Wang, Jianfei Cai, and Junsong Yuan. 3D hand shape and pose estimation from a single RGB image. In *CVPR*, 2019. 1, 2, 5

[23] Patrick Grady, Chengcheng Tang, Christopher D Twigg, Minh Vo, Samarth Brahmbhatt, and Charles C Kemp. ContactOpt: Optimizing contact to improve grasps. In *CVPR*, 2021. 5

[24] Shreyas Hampali, Sayan Deb Sarkar, Mahdi Rad, and Vincent Lepetit. Keypoint Transformer: Solving joint identification in challenging hands and object interactions for accurate 3d pose estimation. In *CVPR*, 2022. 1, 3

[25] Shangchen Han, Beibei Liu, Randi Cabezas, Christopher D Twigg, Peizhao Zhang, Jeff Petkau, Tsz-Ho Yu, Chun-Jung Tai, Muzaffer Akbay, Zheng Wang, et al. MEgATrack: Monochrome egocentric articulated hand-tracking for virtual reality. *ACM Transactions on Graphics (ToG)*, 2020. 1, 2

[26] Shangchen Han, Po-chen Wu, Yubo Zhang, Beibei Liu, Linguang Zhang, Zheng Wang, Weiguang Si, Peizhao Zhang, Yujun Cai, Tomas Hodan, et al. UmeTrack: Unified multi-view end-to-end hand tracking for vr. In *SIGGRAPH*, 2022. 1, 2

[27] Yana Hasson, Gul Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *CVPR*, 2019. 1, 2, 4, 5, 6

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3, 5

[29] Yihui He, Rui Yan, Katerina Fragkiadaki, and Shoou-I Yu. Epipolar transformers. In *CVPR*, 2020. 2

[30] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 4

[31] Korrawe Karunratanakul, Adrian Spurr, Zicong Fan, Otmar Hilliges, and Siyu Tang. A skeleton-driven neural occupancy representation for articulated hands. In *3DV*, 2021. 2

[32] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *ACM computing surveys (CSUR)*, 2022. 3

[33] Dong Uk Kim, Kwang In Kim, and Seungryul Baek. End-to-end detection and pose estimation of two interacting hands. In *ICCV*, 2021. 1, 3

[34] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 5

[35] Devin Kreuzer, Dominique Beaini, Will Hamilton, Vincent Létourneau, and Prudencio Tossou. Rethinking graph transformers with spectral attention. *NeurIPS*, 2021. 4

[36] Dominik Kulon, Riza Alp Guler, Iasonas Kokkinos, Michael M Bronstein, and Stefanos Zafeiriou. Weakly-supervised mesh-convolutional hand reconstruction in the wild. In *CVPR*, 2020. 2

[37] Taein Kwon, Bugra Tekin, Jan Stühmer, Federica Bogo, and Marc Pollefeys. H2O: Two hands manipulating objects for first person interaction recognition. In *ICCV*, 2021. 3

[38] Mengcheng Li, Liang An, Hongwen Zhang, Lianpeng Wu, Feng Chen, Tao Yu, and Yebin Liu. Interacting attention graph for single image two-hand reconstruction. In *CVPR*, 2022. 3

[39] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *CVPR*, 2021. 2, 3, 4, 5, 6, 7

[40] Kevin Lin, Lijuan Wang, and Zicheng Liu. Mesh graphormer. In *ICCV*, 2021. 2, 3, 5, 6, 7

[41] Rong Liu and Hao Zhang. Segmentation of 3D meshes through spectral clustering. In *Pacific Graphics*, 2004. 4

[42] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3D reconstruction in function space. In *CVPR*, 2019. 2

[43] Tomas Möller and Ben Trumbore. Fast, minimum storage ray-triangle intersection. *Journal of graphics tools*, 1997. 5

[44] Gyeongsik Moon and Kyoung Mu Lee. I2L-MeshNet: Image-to-lixel prediction network for accurate 3D human pose and mesh estimation from a single RGB image. In *ECCV*, 2020. 6

[45] Gyeongsik Moon, Takaaki Shiratori, and Kyoung Mu Lee. DeepHandMesh: A weakly-supervised deep encoder-decoder framework for high-fidelity hand mesh modeling. In *ECCV*, 2020. 2, 5, 8

[46] Gyeongsik Moon, Shoou-I Yu, He Wen, Takaaki Shiratori, and Kyoung Mu Lee. InterHand2.6M: A dataset and baseline for 3D interacting hand pose estimation from a single RGB image. In *ECCV*, 2020. 1, 2, 3, 6

[47] Franziska Mueller, Micah Davis, Florian Bernard, Oleksandr Sotnychenko, Mickeal Verschoor, Miguel A Otaduy, Dan Casas, and Christian Theobalt. Real-time pose and shape reconstruction of two interacting hands with a single depth camera. In *SIGGRAPH*, 2019. 1, 2

[48] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 2022. 6

[49] Blender project. Cycles. https://www.cycles-renderer.org/. 6

[50] Haibo Qiu, Chunyu Wang, Jingdong Wang, Naiyan Wang, and Wenjun Zeng. Cross view fusion for 3D human pose estimation. In *ICCV*, 2019. 2

[51] Edoardo Remelli, Shangchen Han, Sina Honari, Pascal Fua, and Robert Wang. Lightweight multi-view 3D pose estimation through camera-disentangled representation. In *CVPR*, 2020. 2

[52] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics (ToG)*, 36(6):245, 2017. 2

[53] Yu Rong, Jingbo Wang, Ziwei Liu, and Chen Change Loy. Monocular 3D reconstruction of interacting hands via collision-aware factorized refinements. In *3DV*, 2021. 3

[54] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. ImageNet large scale visual recognition challenge. *IJCV*, 2015. 5

[55] Breannan Smith, Chenglei Wu, He Wen, Patrick Peluse, Yaser Sheikh, Jessica K Hodgins, and Takaaki Shiratori. Constraining dense hand surface tracking with elasticity. *ACM Transactions on Graphics (ToG)*, 2020. 6

[56] Olga Sorkine and Marc Alexa. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, 2007. 5

[57] Mingxing Tan and Quoc Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *ICML*, 2019. 8

[58] Tze Ho Elden Tse, Daniele De Martini, and Letizia Marchegiani. No need to scream: Robust sound-based speaker localisation in challenging scenarios. In *ICSR 2019*, 2019. 1

[59] Tze Ho Elden Tse, Kwang In Kim, Aleš Leonardis, and Hyung Jin Chang. Collaborative learning for hand and object reconstruction with attention-guided graph convolution. In *CVPR*, 2022. 1, 5, 6

[60] Tze Ho Elden Tse, Zhongqun Zhang, Kwang In Kim, Aleš Leonardis, Feng Zheng, and Hyung Jin Chang. S^{2}Contact: Graph-based network for 3D hand-object contact estimation with semi-supervised learning. In *ECCV*, 2022. 1, 5

[61] Dimitrios Tzionas, Luca Ballan, Abhilash Srikantha, Pablo Aponte, Marc Pollefeys, and Juergen Gall. Capturing hands in action using discriminative salient points and physics simulation. *IJCV*, 2016. 2

[62] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 4

[63] Jiayi Wang, Franziska Mueller, Florian Bernard, Suzanne Sorli, Oleksandr Sotnychenko, Neng Qian, Miguel A Otaduy, Dan Casas, and Christian Theobalt. RGB2Hands: real-time tracking of 3D hand interactions from monocular rgb video. In *SIGGRAPH*, 2020. 1, 2

[64] Hongyi Xu, Eduard Gabriel Bazavan, Andrei Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Ghum & ghuml: Generative 3D human shape and articulated pose models. In *CVPR*, 2020. 6

[65] Baowen Zhang, Yangang Wang, Xiaoming Deng, Yinda Zhang, Ping Tan, Cuixia Ma, and Hongan Wang. Interacting two-hand 3D pose and shape reconstruction from single color image. In *ICCV*, 2021. 1, 3

[66] Linfang Zheng, Aleš Leonardis, Tze Ho Elden Tse, Nora Horanyi, Hua Chen, Wei Zhang, and Hyung Jin Chang. TP-AE: Temporally primed 6D object pose tracking with autoencoders. In *ICRA*, 2022. 1

[67] Christian Zimmermann, Duygu Ceylan, Jimei Yang, Bryan Russell, Max Argus, and Thomas Brox. FreiHAND: A

dataset for markerless capture of hand pose and shape from single RGB images. In *ICCV*, 2019. 3, 6, 7