arXiv:2107.01205v2 [cs.CV] 5 Dec 2021

HandVoxNet++: 3D Hand Shape and Pose Estimation using Voxel-Based Neural Networks

Jameel Malik[1,2,3] Soshi Shimada[5,6] Ahmed Elhayek[4]
Sk Aziz Ali[1,2] Christian Theobalt[5,6] Vladislav Golyanik[5,6] Didier Stricker[1,2]
[1]TU Kaiserslautern [2]DFKI [3]NUST Pakistan [4]UPM Saudi Arabia
[5]MPI for Informatics [6]Saarland Informatics Campus

Abstract—3D hand shape and pose estimation from a single depth map is a new and challenging computer vision problem with many applications. Existing methods addressing it directly regress hand meshes via 2D convolutional neural networks, which leads to artefacts due to perspective distortions in the images. To address the limitations of the existing methods, we develop HandVoxNet++, *i.e.*, a voxel-based deep network with 3D and graph convolutions trained in a fully supervised manner. The input to our network is a 3D voxelized-depth-map-based on the truncated signed distance function (TSDF). HandVoxNet++ relies on two hand shape representations. The first one is the 3D voxelized grid of hand shape, which does not preserve the mesh topology and which is the most accurate representation. The second representation is the hand surface that preserves the mesh topology. We combine the advantages of both representations by aligning the hand surface to the voxelized hand shape either with a new neural *Graph-Convolutions-based Mesh Registration* (GCN-MeshReg) or classical segment-wise *Non-Rigid Gravitational Approach* (NRGA++) which does not rely on training data. In extensive evaluations on three public benchmarks, *i.e.*, SynHand5M, depth-based HANDS19 challenge and HO-3D, the proposed HandVoxNet++ achieves state-of-the-art performance. In this journal extension of our previous approach presented at CVPR 2020, we gain 41.09% and 13.7% higher shape alignment accuracy on SynHand5M and HANDS19 datasets, respectively. Our method is **ranked first** on the HANDS19 challenge dataset (*Task 1: Depth-Based 3D Hand Pose Estimation*) at the moment of the submission of our results to the portal in August 2020.

Index Terms—3D Hand Shape and Pose from a Single Depth Map, Voxelized Hand Shape, Graph Convolutions, TSDF, 3D Data Augmentation, Shape Registration, GCN-MeshReg, NRGA++.

1 Introduction

Tracking and reconstruction of human hand pose in 3D is an extensively studied computer vision problem which often arises in user authentication, augmented and virtual reality, gaming, movie production as well as human performance capture and analysis, among other fields [1], [2], [3], [4]. Accurate 3D hand tracking can facilitate gesture recognition and enable new interfaces for human-computer interaction. While there are various accurate neural methods for 3D hand pose estimation focusing on RGB and depth images [2], [5], [6], [7], simultaneous estimation of 3D hand pose and 3D shape from a single depth map is an emerging research direction.

This problem is challenging because annotating real images for 3D shapes is cumbersome, due to varying hand shapes, self-occlusions (resulting in missing data in the depth maps), high number of degrees of freedom (DOF) and self-similarity of hand parts. On the other hand, the dense 3D hand mesh is a richer representation which is, in many cases, more useful than the bare 3D joints [8], [9], [10], [11]. Benefiting from the recent advances in neural machine learning, several algorithms for simultaneous hand pose and shape estimation have been introduced [10], [12], [13], [14], [15], [16]. Malik *et al.* [16] developed a 2D Convolutional Neural Network (CNN)-based approach that estimates shapes directly from 2D depth maps. The recovered shapes suffer from artefacts due to the limited representation capacity of their hand model [10], [14]. The same problem can occur even by embedding a realistic statistical hand model (*i.e.*, MANO [9] or HTML [11]) inside a deep neural network [12], [14]. In contrast to these model-based approaches [12], [16], the direct regression-based approach proposed by Ge *et al.* [14] uses a monocular RGB image and achieves more accurate results. Another recent technique with direct regression from a single depth image is [10].

All of the approaches mentioned above treat depth maps as 2D signals and process them with 2D CNNs, even though depth maps intrinsically provide 2.5D data. Training a 2D CNN to estimate 3D hand pose or shape given 2D representation of a depth map is highly non-linear and results in perspective distortions in the estimated outputs [6]. V2V-PoseNet [6] is the first work that uses 3D voxelized grid of a depth map to regress 3D joints heatmaps and, thus, avoids perspective distortions. Extending this work for shape estimation by directly regressing 3D heatmaps of mesh vertices is not feasible in practice, due to extremely high memory requirements.

In this article, we propose the first—to the best of our knowledge—architecture with 3D CNNs and graph convolutions called HandVoxNet++, which simultaneously estimates 3D shape and 3D pose given a voxelized depth

map, see Fig. 1 for an overview. HandVoxNet++ is based on 3D and graph convolutions which regresses two different representations of hand shape, namely voxelized hand shape and hand surface (Secs. 3–5). The voxelized hand shape is estimated from a new *voxel-to-voxel* network relying on Truncated Signed Distance Field (TSDF), and establishes a one-to-one mapping between the voxelized depth map and the voxelized hand shape. Since the estimated voxelized shape does not preserve the hand mesh topology and the number of vertices, we also estimate hand surface with our *voxel-to-surface* network. The voxel-to-surface network does not establish a one-to-one mapping between the voxelized depth measurements and shapes; the accuracy of the estimated hand surface is low but the hand topology is preserved. To combine the advantages of both representations, we register the estimated hand surface to the estimated voxelized hand shape. In [17], we introduced two different variants of shape registration methods, *i.e.,* CNN-based [18] and NRGA-based [19]. In this article, we propose two additional novel and more accurate registration algorithms, namely GCN-MeshReg and NRGA++. In the GCN-MeshReg, we employ graph convolutions to explicitly utilize the topology of the hand shape. The *Non-Rigid Gravitational Approach* (NRGA) for point set alignment [19] is extended to support segmented hand surface. We call this extended version of the registration method NRGA++. Next, to increase the robustness and accuracy of the hand pose estimation, we perform 3D data augmentation on the voxelized depth maps (Sec. 4.4).

Our main contribution is the new state-of-the-art HandVoxNet++ approach for 3D hand shape and pose estimation from a single depth map. It includes a *voxel-to-voxel* and *voxel-to-surface* networks with 3D CNNs as well as two adapted and improved mesh registration components. This work is an extension of our conference paper [17]. Compared to [17], the new contributions in this article are:

- A new TSDF-based *voxel-to-voxel* network for 3D hand pose estimation (Sec. 4.1). Our TSDF-based representation of depth map achieves 19.8% improvement in accuracy compared to binary voxelized grid representation used in [6] (Sec. 6.3).
- The first work that applies graph convolutions to a registration problem between a hand mesh and a 3D voxelized hand shape (Sec. 4.3.1). Furthermore, we propose a novel iterative refinement strategy for the shape registration. This allows to significantly improve the accuracy and computational time compared to [17] (Sec. 6.2).
- A new segment-wise point set alignment method NRGA++ which is ∼150 times faster than the NRGA proposed in [17] (Sec. 4.3.2 and Sec. 6.2).
- Extensive experiments on three recent benchmarks (*i.e.,* SynHand5M, depth-based HANDS19 challenge, and HO-3D [20]) illustrate that HandVoxNet++ achieves:
    1) Superior or comparable performance than the existing approaches,
    2) More accurate hand shapes compared to the state-of-the-art HandVoxNet approach [17](Sec. 6) and
    3) The first place on the task of "depth-based 3D hand pose estimation of the HANDS19" challenge, at the moment of our submission to its web portal in August 2020.

2 Related Work

In this section, we review the most relevant existing methods for neural hand pose and shape estimation from RGB and depth images.

Neural Hand Pose and Shape Estimation. Most approaches for 3D hand reconstruction and tracking from monocular RGB and depth images and event streams estimate hand poses only, *i.e.,* a sparse set of 3D hand joints [1], [2], [3], [21], [22], [23], [24]. The approaches which estimate hand shape and pose simultaneously are still in the minority. Malik *et al.* [16] proposed the first deep neural network for hand pose and shape estimation from a single depth image. To this end, they developed a model-based hand pose and shape layer which is embedded inside their deep network. Their approach suffers from artefacts due to the difficulty in optimizing complex hand shape parameters inside the network. Ge *et al.* [14] developed a direct regression-based algorithm for hand pose and shape estimation from a single RGB image. They highlight that the representation capacity of the statistical deformable hand model (*i.e.,* MANO [9]) could be limited due to the small amount of training data and the linear bases utilized for the shape recovery. [12], [25], [26] introduced similar MANO hand-model-based neural approaches using monocular RGB image. Qian *et al.* [11] came up with the first parametric hand texture model, which can be applied in 3D hand reconstruction and personalization from a single image as well as a differential neural hand appearance layer. Next, a weakly-supervised neural approach using a single depth image was presented in [10]. [27] directly regressed the hand mesh vertices and a dense correspondence map using 2D fully convolutional neural architecture. In [28], many sources of hand training data are utilized to train a deep learning architecture which estimates 3D pose given monocular RGB image. Thereafter, the 3D pose is used to estimate the shape parameters.

All of the above-mentioned methods use 2D CNNs and treat the depth maps (the methods which support those) as 2D data. Consequently, the deep network is likely to produce perspective distortions in the shape and pose estimations [6]. In contrast, we propose the first 3D-convolution-based architecture which effectively establishes a one-to-one mapping between the voxelized depth map and the voxelized hand shape. This one-to-one mapping allows to more accurately reconstruct the hand shapes.

Shape Registration. Recently, Graph Convolutional Networks (GCNs) have received much attention in computer vision. The concept of GCNs was introduced in [29] for semi-supervised classification. Later, it was used to develop powerful solutions of many computer vision problems. For instance, Kolotouros *et al.* [30] apply graph-convolution-based architecture to regress human 3D shapes and poses given 2D RGB images and human template mesh. Tretschk *et al.* [31] demonstrate the effectiveness of graph convolutions for mesh autoencoding including hands from the SynHand5M dataset [16]. A GCNs-based model for both hand

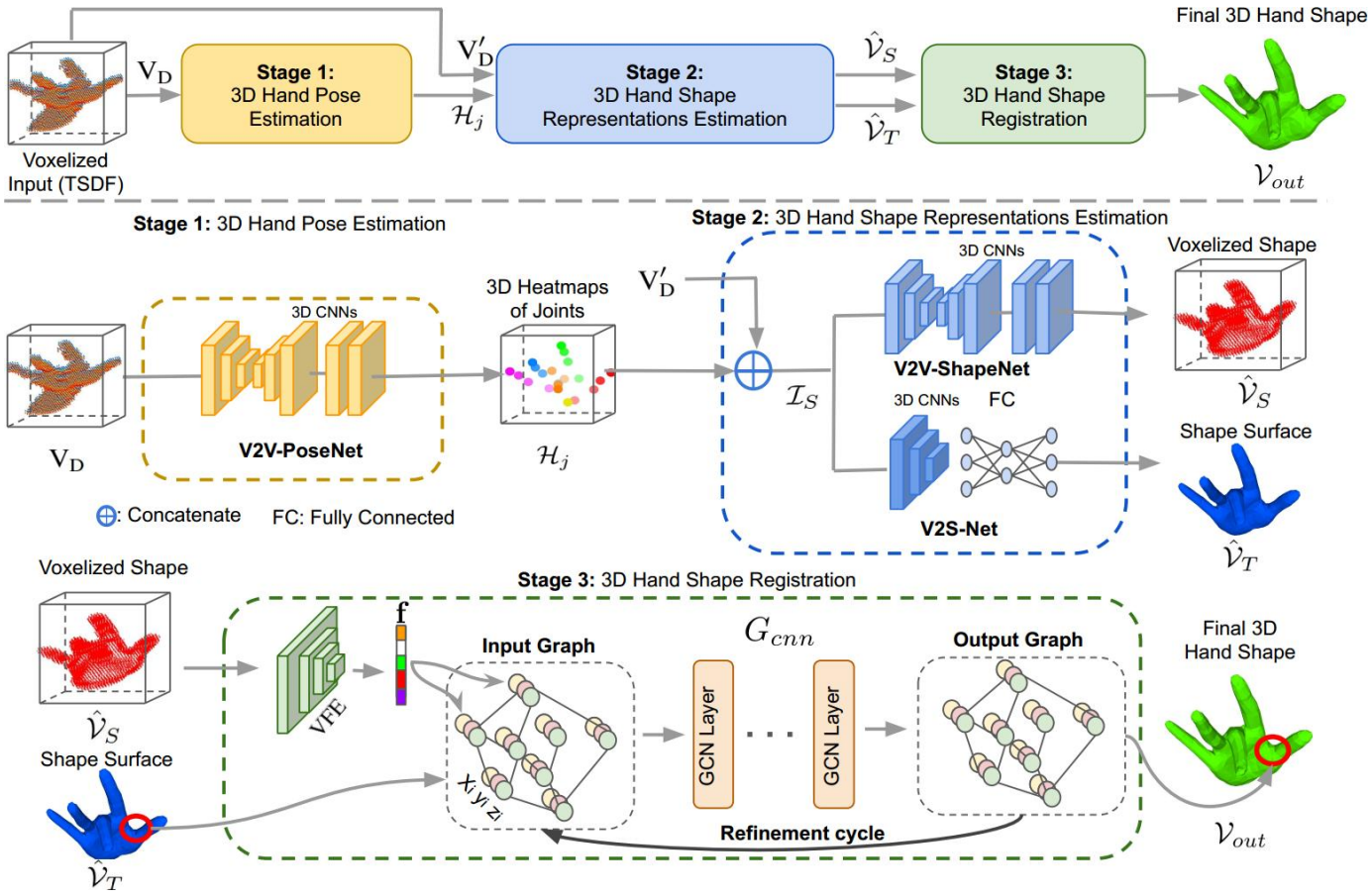


Fig. 1. **Overview of our approach for 3D hand shape and pose recovery.** The framework consists of three stages. In **Stage 1**, V2V-PoseNet accurately estimates 3D joints heatmaps $\mathcal{H}_j$ (*i.e.*, pose) from the TSDF-based 3D voxelized depth map $\mathrm{V_D}$. The final hand shape $\mathcal{V}_{out}$ is estimated by the following stages. In **Stage 2**, V2V-ShapeNet and V2S-Net estimate the voxelized shape $\hat{\mathcal{V}}_S$ and shape surface $\hat{\mathcal{V}}_T$ using 3D-convolution-based neural networks, respectively. Finally, in **Stage 3**, graph-convolution-based mesh registration (*i.e.*, GCN-MeshReg) accurately fits $\hat{\mathcal{V}}_T$ to $\hat{\mathcal{V}}_S$. In the registration phase, a voxel feature extractor (VFE) first extracts a feature vector which is connected to each node ($\mathrm{X_i}$,$\mathrm{Y_i}$,$\mathrm{Z_i}$) of the input graph. The output graph provides the deformed 3D mesh vertices. In the refinement cycle, this estimate is further improved in an iterative manner.

and object pose estimation was proposed in [32]. Unlike other works, we propose an accurate and fast GCNs-based approach for the mesh to the voxelized shape registration with an iterative refinement strategy. On the other hand, fully-convolutional networks were shown to perform well in geometry regression tasks [6], [33], [34], [35].

Neural Hand Pose Estimation from Depth. In general, deep learning-based hand pose estimation methods can be classified into two categories. The first one encompasses the discriminative methods which directly estimate hand joint locations using CNNs [5], [6], [7], [36], [37], [38], [39], [40], [41]. The second category is hybrid methods which explicitly incorporate hand structure inside deep networks [42], [43], [44], [45], [46]. The disriminative methods achieve higher accuracy compared to the hybrid methods. The *voxel-to-voxel* approach [6] is powerful and highly effective because it uses 3D convolutions to learn a one-to-one mapping between the 3D voxelized depth map and 3D heatmaps of hand joints. Notably, the voxelized representation of depth maps is best suited for 3D data augmentation to improve the robustness and accuracy of the estimations. A few methods perform data augmentation on depth maps [45], [47] or voxelized depth maps [6]. In this work, we integrate the *voxel-to-voxel* approach with our pipeline and, additionally, perform new 3D data augmentation on voxelized depth maps that further improves the 3D pose estimation. The *voxel-to-voxel* approach transformed 2D depth map to 3D binary voxelized grid representation which looses information in the 3D point cloud due to the binary quantization. Ge *et al.* [48] utilized the TSDF representation of depth map however, they directly regressed the hand joint positions using a neural architecture which does not establish a one-to-one relation between the depth map and the pose. In contrast, we propose to establish a one-to-one mapping between the TSDF-based 3D voxelized grid of 2D depth map and 3D heatmaps of joint positions. The TSDF representation better encapsulates the information of the 3D point cloud and, thereby, significantly enhances the accuracy of 3D hand pose estimation.

3 Method Overview

Given an input depth map, our goal is to estimate N 3D hand joint locations $\mathcal{J} \in \mathcal{R}^{3\times \mathrm{N}}$ (*i.e.*, 3D pose) and K 3D vertex locations $\mathcal{V} \in \mathcal{R}^{3\times \mathrm{K}}$ (*i.e.*, 3D shape). Fig. 1 shows an overview of the proposed approach. The input is transformed into a voxelized grid (*i.e.*, $\mathrm{V_D}$) of size $88 \times 88 \times 88$, by using intrinsic camera parameters, a fixed cube size and the truncated signed distance function (Section 4.1). For hand pose estimation (*i.e.*, Stage 1), $\mathrm{V_D}$ is provided as an

input to the *voxel-to-voxel* pose regression network (*i.e.,* V2V-PoseNet) that directly estimates 3D joint heatmaps $\{\mathcal{H}_j\}_{j=1}^{\mathrm{N}}$. Each 3D joint heatmap is represented as $44 \times 44 \times 44$ voxelized grid. The input of the shape estimation network (*i.e.,* stage 2) is the concatenation of $\mathcal{H}_j$ (*i.e.,* the output of Stage 1) and $\mathrm{V_D}$. We call this concatenated input as $\mathcal{I}_S$. To this end, we resize $\mathrm{V_D}$ to $44 \times 44 \times 44$ voxel grid size (*i.e.,* $\mathrm{V'_D}$). The voxelized hand shape (*i.e.,* $64 \times 64 \times 64$ binary grid) is directly regressed via 3D CNN-based *voxel-to-voxel* shape regression network (*i.e.,* V2V-ShapeNet), by using $\mathcal{I}_S$ as an input. Notably, V2V-ShapeNet establishes a one-to-one mapping between the voxelized depth map and the voxelized shape. Therefore, it produces an accurate voxelized shape representation but does not preserve the topology of hand mesh and the number of mesh vertices. To regress hand surface, $\mathcal{I}_S$ is fed to the 3D CNN-based *voxel-to-surface* regression network (*i.e.,* V2S-Net). Since the mapping between $\mathcal{I}_S$ and hand surface is not one-to-one, it is therefore less accurate. To combine the advantages of the two hand shape representations, Stage 3 registers the estimated hand surface to the estimated voxelized hand shape. The voxelized shape is provided as an input to a 3D-convolution-based network that produces a feature vector f of size = 2187. f is concatenated to each node of the hand mesh. This representation is fed to the graph convolution neural networks as input, and the networks return the registered hand shape that matches the voxelized hand shape. We repeat this procedure using the output from the graph convolution networks as a new template hand shape, which we term refinement cycle. Thanks to this refinement cycle, we obtain highly accurate and smooth hand shape as the final output. Please note that Fig. 1 is based on GCN-MeshReg as it enables the best performance. However, Stage 3 can be replaced with other registration approaches such as DispVoxNet, NRGA, or NRGA++.

4 The Proposed HandVoxNet Approach

In this section, we explain our proposed HandVoxNet++ approach by highlighting the function and effectiveness of each of its components. We develop an effective solution that produces reasonable hand shapes via voxel-based convolutional networks. To this end, our approach exploits the estimated 3D heatmaps of hand joints as a strong pose prior and voxelized depth map to accurately estimate the hand shape representations. Moreover, our 3D data augmentation on voxelized depth maps allows to further improve the accuracy and robustness of 3D hand pose estimation.

4.1 Volumetric Input Representation

In this section, we discuss two possible conversions of the depth map represented in 2D image to 3D volume; namely occupancy grid and the TSDF-based grid. The main motivation for this modification is that the depth map representation of 3D hand distort the hand during the projection from 3D space to the 2D image space. This makes the training process more challenging as the network receives as input a distorted representation of the real hand. Another drawback of the depth map representation is the highly non-linear mapping between it and the 3D output which complicates the learning process [6].

To overcome these limitations, the input depth map should be encoded into a volumetric representation. The easiest volumetric representation is the occupancy (binary) grid [6]. Although this representation allows to overcome the depth maps limitations, it cannot differentiate voxels behind and in front of the observed surface. The reason of this fact is that it is generated based on incomplete information about the 3D hand shape (*i.e.,* the depth map which only captures the observed surface from the view of camera).

We encode the depth map into 3D voxelized grid representation using the TSDF. The input depth map is first converted into a set of 3D points, and these points are further discretized in the range $[1, 88]$ (Sec. 5). Then, a 3D voxelized grid of size $88 \times 88 \times 88$ (*i.e.,* $\mathrm{V_D}$) can be created by calculating $\mathrm{V}(k)$ for each voxel k. In the TSDF representation, $\mathrm{V}(k)$ can be obtained as:

$$\mathrm{V}(k) = \min(\max(\mathrm{d}(k_c)/\mu, -1), +1), \tag{1}$$

where $\mathrm{d}(k_c)$ is the signed distance from the voxel center k_c to the nearest surface point in the set of discretized 3D points. We use the standard Euclidean distance. The sign of $\mathrm{d}(k_c)$ is positive if the depth values of k_c is less than the depth value of the nearest surface point otherwise, the sign is negative. We use $\mu = 3$ as the truncated distance value. In accurate TSDF representation, the nearest surface point is found by checking all the 3D points which is highly time consuming and therefore, not suitable for real-time applications. The projective TSDF [49] is practically more feasible because the nearest surface point is calculated on the line of sight in the camera frame. However, the projective TSDF is an approximation of the accurate TSDF and consequently, it loses some information [50]. In the projective directional TSDF reprensentation [51], the voxel value stores the 3D offset vector (*i.e.,* $[dx, dy, dz]$) to the nearest point. In the experiment section 6, we demonstrate that the projective TSDF representation yields the best performance in our case and effectively establishes a one-to-one mapping between the TSDF-based voxelized depth map and the 3D heatmaps of the joints.

4.2 3D Hand Shape Estimation

As aforementioned, estimating 3D hand shape from a 2D depth map by using 2D CNN is a highly non-linear mapping. It compels the network to perform perspective-distortion-invariant estimation which causes difficulty in learning the shapes. To address this challenge, we develop a fully voxel-based deep network that effectively utilizes the estimated 3D pose and voxelized depth map to produce reasonable 3D hand shapes. Our proposed approach for 3D shape estimation comprises of two main phases. In the first phase, we estimate the shape surface and the voxelized hand shape. In the second phase, we register the estimated shape surface to the estimated voxelized hand shape. We discuss several registration approaches and provide their comparative analysis.

Voxelized Shape Estimation. Our idea is to estimate 3D hand shape in the voxelized form via 3D CNN-based network. It allows the network to estimate the shape in such a way that minimizes the chances for perspective distortion.

Inspired by the approach proposed in the recent work [10], we consider sparse 3D joints as the latent representation of dense 3D shape. However, in this work, we combine 3D pose with the depth map which helps to represent the shape of hand more accurately. Furthermore, here we use more accurate and useful representations of 3D pose and 2D depth image which are 3D joints heatmaps and a voxelized depth map, respectively. The V2V-ShapeNet module is shown in Fig. 1. It can be considered as the 3D shape decoder:

$$\hat{\mathcal{V}}_S \sim Dec(\mathcal{H}_j \oplus \mathrm{V}'_{\mathrm{D}}) = p(\mathcal{V}_S|\mathcal{I}_S), \tag{2}$$

where $p(\mathcal{V}_S|\mathcal{I}_S)$ is the decoded distribution. The decoder $Dec(\cdot)$ learns to reconstruct the voxelized hand shape $\hat{\mathcal{V}}_S$ as close as possible to the ground truth voxelized hand shape $\mathcal{V}_S$. The V2V-ShapeNet is a 3D CNN-based architecture [17] that directly estimates the probability of each voxel in the voxelized shape indicating whether it is the background (*i.e.,* 0) or the shape voxel (*i.e.,* 1). The per-voxel binary cross entropy loss $\mathcal{L}_{\mathcal{V}_S}$ for voxelized shape reconstruction reads:

$$\mathcal{L}_{\mathcal{V}_S} = -(\mathcal{V}_S \ \log(\hat{\mathcal{V}}_S) + (1 - \mathcal{V}_S) \ \log(1 - \hat{\mathcal{V}}_S)), \tag{3}$$

where $\mathcal{V}_S$ and $\hat{\mathcal{V}}_S$ are the ground truth and the estimated voxelized hand shapes, respectively.

Shape Surface Estimation. The hand poses of the shape surfaces and voxelized shapes need to be similar for an improved shape registration. To facilitate the registration, we employ V2S-Net deep network [17] which directly regresses $\mathcal{V}$. Based on the similar concept of hand shape decoding (as mentioned before), $\mathcal{I}_S$ is provided as an input to this network while the decoded output is the reconstructed hand mesh (see Fig. 1). The hand shape surface reconstruction loss $\mathcal{L}_{\mathcal{V}_T}$ is given by the standard Euclidean loss as:

$$\mathcal{L}_{\mathcal{V}_T} = \frac{1}{2} \left\| \hat{\mathcal{V}}_T - \mathcal{V}_T \right\|^2, \tag{4}$$

where $\mathcal{V}_T$ and $\hat{\mathcal{V}}_T$ are the respective ground truth and reconstructed hand shape surfaces.

4.3 Shape Registration

As mentioned above, V2S-Net can estimate hand shapes while preserving the order and number of points. Unfortunately, as the V2S-Net uses fully connected (FC) layer for mesh vertices regression, it looses local spatial information. Although, estimating the voxelized hand shape by 3D convolutional layers guarantees a one-to-one mapping between the input and the output, it results in an inconsistent number of points and loses point order.

To preserve the hand shape topology while relying on voxelized hand shape representation, we register the shape estimated by V2S-Net to the probabilistic shape representation estimated by V2V-ShapeNet. Therefore, we propose a new neural shape registration approach GCN-MeshReg (Sec. 4.3.1) along with a classical optimization-based segmentwise non-rigid gravitational approach which we abbreviate as NRGA++ (Sec. 4.3.2). Our neural shape registration algorithm not only achieves higher accuracy compared to the classical optimization-based method but is also much faster. On the downside, compared to NRGA++, it requires a training dataset for hand shape alignment. Thus, NRGA++ can cope well with real-world datasets from depth sensors which do not provide hand shape annotations. Compared to the shape alignment counterparts from [17], GCN-MeshReg significantly improves in hand shape alignment accuracy over DispVoxNet, and NRGA++ improves both in runtime and accuracy over pointwise NRGA.

4.3.1 GCN-MeshReg: Neural Shape Registration

In this subsection, we adopt a CNN-based method and propose a new GCN-based algorithm for shape registration, *i.e.,* DispVoxNets and GCN-MeshReg, respectively. Both these components enable end-to-end deep network for reconstructing a full 3D mesh and pose of a human hand from a single depth image.

CNN-based Shape Registration. The original DispVoxNets [18] is comprised of two stages, *i.e.,* global displacement estimation and refinement. The refinement stage removes roughness on the underlying surfaces represented as point sets. In contrast to the original approach [18], we replace the refinement stage with Laplacian smoothing [52], as the hand mesh topology is known.

In DispVoxNet, the hand surface shape $\hat{\mathcal{V}}_T$ is first converted into a voxelized grid $\hat{\mathcal{V}}'_T$ of dimensions $64 \times 64 \times 64$. DispVoxNet estimates per-voxel displacements of the dimension $64^3 \times 3$ between the reference $\hat{\mathcal{V}}_S$ and voxelized hand surface $\hat{\mathcal{V}}'_T$. The displacement loss $\mathcal{L}_{\text{Disp}}$ is given by:

$$\mathcal{L}_{Disp.} = \frac{1}{Q^3} \left\| \mathbf{d} - D_{vn}(\hat{\mathcal{V}}_S, \hat{\mathcal{V}}'_T) \right\|^2, \tag{5}$$

where $D_{vn}(\cdot)$ represents DispVoxNet based registration network from [17]. Q and $\mathbf{d}$ are the voxel grid size and the ground-truth displacement, respectively. Since it is difficult to obtain $\mathbf{d}$ between the voxelized shape $\hat{\mathcal{V}}_S$ and hand surface $\hat{\mathcal{V}}_T$, the displacements are first computed between $\mathcal{V}_T$ and $\hat{\mathcal{V}}_T$, and are discretized to obtain $\mathbf{d}$. For more details on the computation of ground-truth voxelized grids, please refer to [18].

GCN-based Shape Registration. We propose in this article the first – to the best of our knowledge – mesh-to-voxel shape registration algorithm called GCN-MeshReg, which is based on GCNs. Recently, GCNs have enjoyed great attention in computer vision [30], [53], [54], and we find that they are a better alternative for such alignment problems compared to the CNN-based shape registration due to several reasons. First, GCNs have better ability to learn correct representations of graph-structured data. Second, the hand mesh registration problem can be addressed using graph-based techniques as meshes can be naturally converted to graphs. Third, graph convolutions can learn inter-vertex relationships which lead to feasible and smooth hand meshes.

Our GCN-MeshReg is inspired by [29], [30] which use graph convolutions for data classification [29] and single-image shape regression [30]. GCN-MeshReg is defined for shape alignment and differs from [29], [30] in several ways, see Table 1 for the architecture details of GCN-MeshReg. Our approach assumes that the reference shape is represented by voxel occupancy probabilities, and the template is a hand mesh. Moreover, it includes an iterative refinement cycle. GCN-MeshReg explicitly takes into account the mesh topology, unlike most of the algorithms which

ID	Layer	Output Sz	Kernel Sz	Stride/Padding
1	Input	(1) 64x64x64	-	-/-
2	3D Conv	(16) 64x64x64	7x7x7	1/3
3	LeakyReLU	(16) 64x64x64	-	-/-
4	3D MaxPooling	(16) 32x32x32	2x2x2	2/0
5	3D Conv	(8) 32x32x32	5x5x5	1/2
6	LeakyReLU	(8) 32x32x32	-	-
7	3D MaxPooling	(8) 16x16x16	2x2x2	2/0
8	3D Conv	(4) 16x16x16	3x3x3	1/1
9	LeakyReLU	(4) 16x16x16	-	-/-
10	3D MaxPooling	(4) 8x8x8	2x2x2	2/0
11	3D Deconv	(3) 9x9x9	4x4x4	1/1
12	Flatten	2187	-	-

TABLE 1
The architecture details of the voxel feature extractor (VFE) in GCN-MeshReg. The negative slope for LeakyReLU is 0.01.

operate on point clouds and directly estimate the vertex positions or displacements [18], [55], [56]. By comparing the performance of the previous CNN-based registration approach DispVoxNets [18] with the proposed GCN-based alternative, *i.e.,* GCN-MeshReg, we show that graph convolutions significantly increase the performance of the hand mesh registration, which we discuss in Sec. 6.

GCN-MeshReg consists of two stages, *i.e.,* feature extraction from the voxelized shape $\hat{\mathcal{V}}_S$ using the 3D convolution-based voxel feature extractor (VFE) and the shape registration stage G_{cnn} using GCNs. In contrast to DispVoxNets, the graph convolutions in GCN-MeshReg utilize vertex connectivity information by constructing the row-normalized adjacency matrix $\hat{\mathbf{A}} \in \mathbb{R}^{\mathrm{K}\times\mathrm{K}}$.

See Fig. 1-(Stage 3) for an overview of GCN-MeshReg. First, VFE accepts $\hat{\mathcal{V}}_S$ and obtains its feature vector $\mathbf{f} \in \mathbb{R}^{2187}$. $\mathbf{f}$ is duplicated and concatenated on each vertex $v \in \mathbb{R}^u$ of $\hat{\mathcal{V}}_T$. The vertices in our dataset contain x-, y- and z-coordinates, hence $u = 3$. G_{cnn} accepts these vertices with the concatenated features along with $\hat{\mathbf{A}}$, applies graph convolutions and returns the registered output shape. We further repeat the same procedure for registrations of higher accuracy in the refinement cycle, using the output shape of the previous registration step as a new template shape input for G_{cnn}. After the refinement cycle, GCN-MeshReg returns the final 3D hand shape $\mathcal{V}_{out}$. Significance of the refinement cycle is highlighted in Table 4 in Sec. 6, where we summarise the effect of our registration components on the final outcome.

For G_{cnn}, we follow the formulation similar to [29]:

$$\mathbf{Y} = \sigma(\hat{\mathbf{A}}\mathbf{X}\mathbf{W}), \tag{6}$$

where $\sigma(\cdot)$ denotes an activation function, $\mathbf{X} \in \mathbb{R}^{\mathrm{K}\times\mathrm{p}}$ is the input to the graph convolutional layer and $\mathbf{W} \in \mathbb{R}^{\mathrm{p}\times\mathrm{s}}$ is its weight matrix of the graph convolutional layer in G_{cnn}. p denotes the dimensionality of the feature of each hand vertex. In our case, $\mathbf{X}$ of the first graph convolution layer is the template hand vertices with the concatenated f. $\mathbf{X}$ for the subsequent graph convolution layers are the output from the previous layers. In this formulation, the row-normalized adjacency matrix $\hat{\mathbf{A}}$ explicitly disconnects the feature aggregation from non-connected vertices, which helps the network to gather relevant features associated with the topology of the mesh data. G_{cnn} operations with i refinement steps are formulated as follows:

$$\mathcal{V}_{out} = \begin{cases} G_{cnn}^{i=0}(\hat{\mathbf{A}}, \mathbf{f}, \hat{\mathcal{V}}_T) \\ G_{cnn}^{i=n}(\hat{\mathbf{A}}, \mathbf{f}, G_{cnn}^{i=n-1}) \quad (n > 0). \end{cases} \tag{7}$$

The final output $\mathcal{V}_{out}$ is further integrated in the objective function $\mathcal{L}_{reg.}$:

$$\mathcal{L}_{reg.} = \frac{1}{\mathrm{K}} \left\| \mathcal{V}_T - \mathcal{V}_{out} \right\|^2. \tag{8}$$

4.3.2 *Our Classical Shape Registration Algorithm*

Although neural registration methods achieve high accuracy in low runtime, these methods are bounded to the availability of training datasets. Recent datasets such as HANDS19 [57] and HO-3D [20] are annotated for both the hand pose and shape. However, several benchmarks (*e.g.,* MSRA [58], NYU [47], ICVL [59], MSRC [60], FHAD [61] *etc.*) do not provide hand shape annotations. Therefore, if the ground truth of hand shape is not available, we need an alternative to a neural registration method. For this reason, alternatively to DispVoxNet and GCN-MeshReg (Sec. 4.3.1), we propose NRGA++, *i.e.,* a classical physics-based algorithm for registering the template mesh $\hat{\mathcal{V}}_T$ with the reference voxelized hand $\hat{\mathcal{V}}_S$, which is a modified version of NRGA [19]. Our choice falls to NRGA as it preserves local hand mesh topology and is robust to noise in $\hat{\mathcal{V}}_S$.

NRGA splits the template point sets into overlapping regions, each of which is registered rigidly to the corresponding region of influence in the reference. The final per-point displacement is obtained as a consensus rigid transformation among the segments in which a given point is involved. NRGA is a computationally-expensive iterative method, and the updates of point positions are performed by simulating Newtonian particle dynamics. In NRGA++, we thus avoid the general-purpose automatic region allocation step and select the template regions as segments of the MANO hand model, as visualized in Fig. 2. Moreover, Fig. 2 visualises how 21 joints of the MANO model are mapped to the overlapping hand segments. For the end-effectors, *e.g.,* we select half of the vertices present in the segment nearest joint. Apart from that, all steps of NRGA++ are as in the original NRGA (*e.g.,* k-d tree building and calculating the consensus per-point transformations), see [19] for more details. The proposed segment-wise point set alignment strategy leads to a significant improvement in runtime performance of NRGA++. Moreover, NRGA++ relies on the topological information *i.e.,* vertex connectivity graph; thereby, it preserves the structure of the hand mesh.

4.4 Data Augmentation in 3D

Our method for hand shape estimation relies on the accuracy of the estimated 3D pose. Therefore, the hand pose estimation method has to be accurate and robust. Training data augmentation helps to improve the performance of a deep network [45]. Existing methods for hand pose estimation [45], [47] use data augmentation in 2D. This is mainly because these methods treat depth maps as 2D data. The representation of the depth map in voxelized form makes it convenient to perform data augmentation

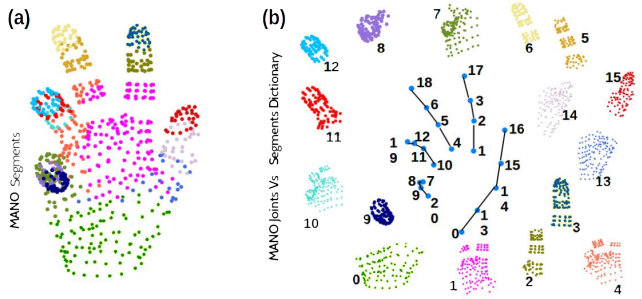


Fig. 2. **Selection of Segments in NRGA++**. (a:) Pre-defined MANO hand shape segments in different colors. (b:) MANO hand model provides a prior mapping between 21 joints (except end-effectors as fingertips) and overlapping hand segments.

in all three dimensions. In this paper, we perform 3D data augmentation which improves the accuracy and robustness of hand pose estimation (see Sec. 6.3).

During V2V-PoseNet training, we apply simultaneous rotations in all three axes (x,y,z) to each 3D coordinate (i, j, k) of $\mathrm{V_D}$ and $\mathcal{H}_j$ by using Euler transformations:

$$[\hat{i}, \hat{j}, \hat{k}]^{\mathrm{T}} = [\mathrm{Rot_x}(\theta_{\mathrm{x}})] \times [\mathrm{Rot_y}(\theta_{\mathrm{y}})] \times [\mathrm{Rot_z}(\theta_{\mathrm{z}})][i, j, k]^{\mathrm{T}}, \quad (9)$$

where $(\hat{i}, \hat{j}, \hat{k})$ is the transformed voxel coordinate. $\mathrm{Rot_x}(\theta_{\mathrm{x}})$, $\mathrm{Rot_y}(\theta_{\mathrm{y}})$ and $\mathrm{Rot_z}(\theta_{\mathrm{z}})$ are 3×3 rotation matrices around x, y and z axes. The values for θ_{x}, θ_{y} and θ_{z} are selected randomly in the ranges $[-40°, +40°]$, $[-40°, +40°]$ and $[-120°, +120°]$, respectively. In addition to rotations in 3D, following [6], we perform scaling and translation in the respective ranges $[+0.8, +1.2]$ and $[-8, +8]$.

5 The Network Training

The network input $\mathrm{V_D}$ is generated by projecting the raw depth image pixels into 3D space. Hand region points are then extracted by using a cube of size 250 that is centered on hand palm center position. 3D point coordinates of the hand region are discretized in the range $[1, 88]$. Finally, to obtain $\mathrm{V_D}$, each voxel is assigned a value using TSDF (see Eq. 1). Similarly, $\mathcal{V}_S$ is obtained by voxelizing the hand mesh except the final step where the voxel value is set to 1 for the 3D point coordinate of the hand region, and 0 otherwise. Following [6], $\mathcal{H}_j$ are generated as 3D Gaussians. $\mathcal{V}_T$ is created by normalizing the mesh vertices in the range $[-1, +1]$. We perform this normalization by subtracting the mesh vertices from the hand palm center and then dividing them by half of the cube size. We train our V2V-PoseNet on the datasets described in Sec. 6 using the 3D data augmentation technique mentioned in Sec. 4.4. For all the datasets, we use the learning rate (LR) of $2.5e^{-4}$, batch size = 8, and RMSProp as an optimization method. We train V2S-Net and V2V-ShapeNet independently on each of the datasets using $2.5e^{-4}$ and $0.5e^{-3}$ as learning rates, respectively. The value of batch size is 8 and RMSProp is used for the optimization of both the shape networks. DispVoxNet and GCN-MeshReg are trained using Adam optimizer [62] with learning rate of $3.0e^{-4}$. The training continues until the convergence of $\mathcal{L}_{\mathrm{Disp}}$ and $\mathcal{L}_{\mathrm{reg}}$ with batch size 12, respectively. We first train VFE and G_{cnn} for the registration using the ground-truth hand shapes. Then, we train another instance of G_{cnn} for the refinement step using the output from the registration step as inputs to the G_{cnn}. All models are trained until convergence on a desktop workstation equipped with Nvidia Titan X GPU.

Methods	V2V-ShapeNet 3D $\mathcal{S}$ Err.	V2S-Net 3D $\mathcal{V}$ Err. (*mm*)
w/o $\mathcal{H}_j$	0.007	8.78
w/o $\mathrm{V'_D}$	0.016	3.54
with $(\mathcal{H}_j \oplus \mathrm{V'_D})$	**0.005**	**3.36**

TABLE 2
Ablation study on inputs (*i.e.*, $\mathcal{H}_j$ and $\mathrm{V'_D}$) to V2S-Net and V2V-ShapeNet on SynHand5M [16]. We observe that combining both inputs is useful for these two networks.

Methods	3D $\mathcal{V}$ Err. (*mm*)
DeepHPS [16]	11.8
WHSP-Net [10]	5.12
HandVoxNet [17] (with DispVoxNet)	2.92
HandVoxNet [17] (with GCN-MeshReg w/o ref.)	**1.79**
HandVoxNet [17] (with GCN-MeshReg)	**1.72**

TABLE 3
Comparison of different registrations and the state of the arts on SynHand5M [16]. Notably, for a fair comparison, we replace only the registration component of [17], and keep all other components same. Our graph convolutions based registration with refinement outperforms 3D-convolution-based method (*i.e.*, DispVoxNet) by 41.09%.

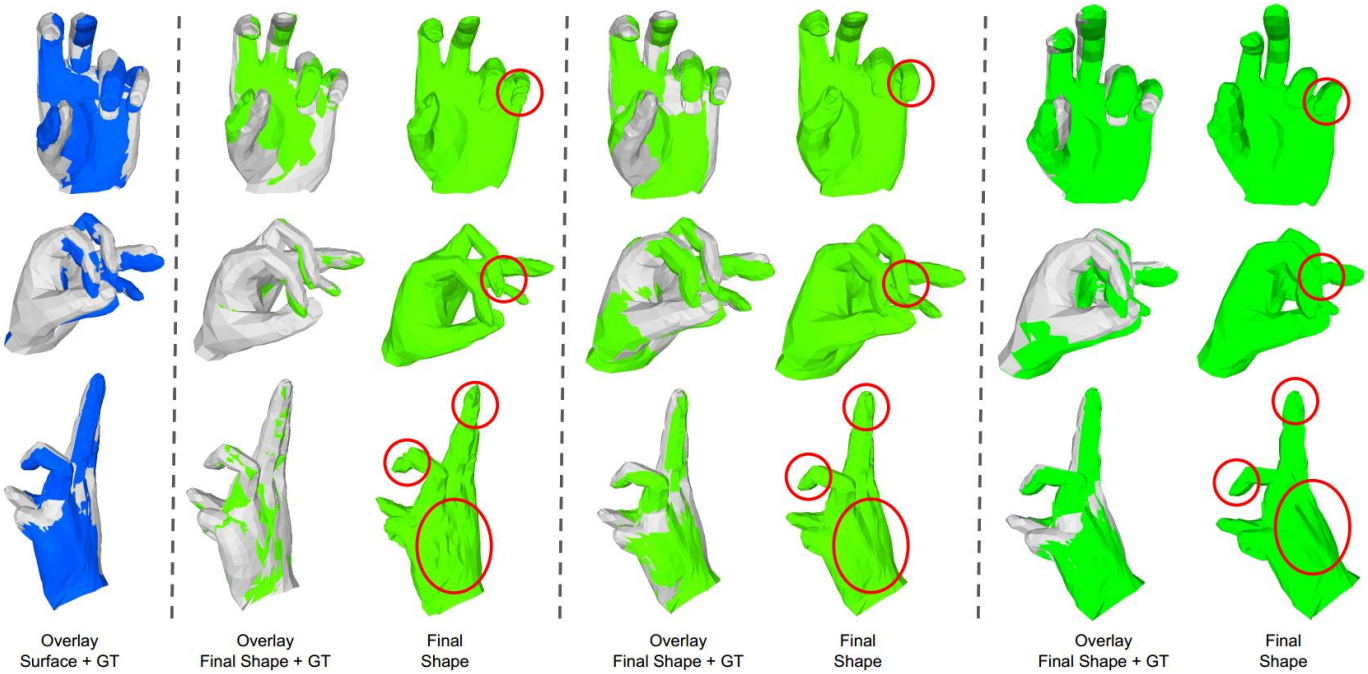
6 Experiments

We perform qualitative and quantitative evaluations of our approach on four challenging hand pose datasets, namely, HO-3D [20], HANDS19 Challenge (Task 1) [57], BigHand2.2M (HANDS17 Challenge) [63] and SynHand5M [16] datasets. An ablation study on the inputs of the proposed shape estimation network is performed on SynHand5M [16] dataset.

6.1 Datasets and Evaluation Metrics

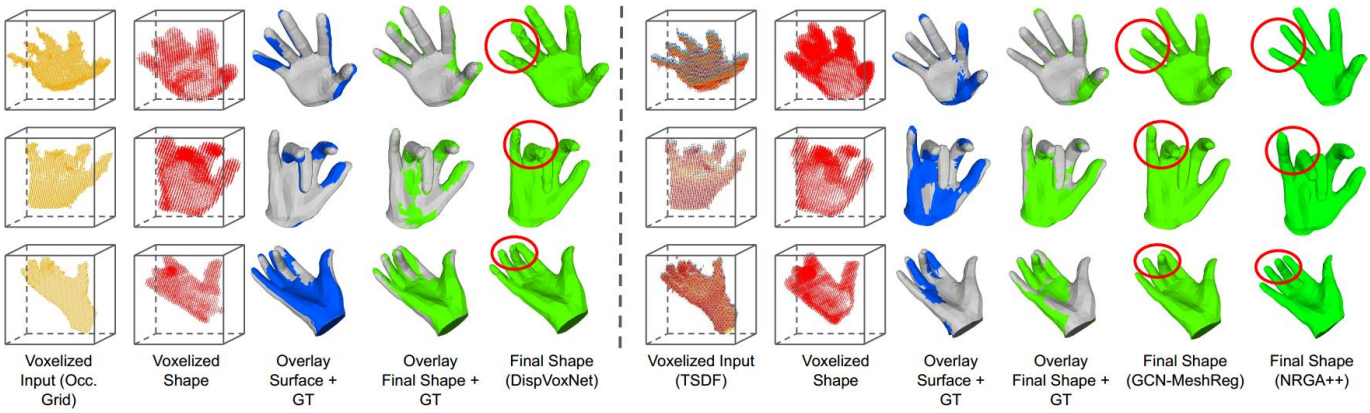
HO-3D [20] and HANDS19 Challenge [57] are the recent real benchmarks that provide annotations for 3D hand pose- and shape-based on the MANO model [9]. Task 1 of HANDS19 Challenge is depth-based hand pose estimation which builds on BigHand2.2M [63]. It contains hands in isolation from both egocentric and third-person viewpoints. The dataset provides $175,951$ training and $124,999$ testing frames. For evaluation, 3D hand pose estimations of the test set are submitted to an online portal[1] which shows the achieved accuracy on the leader-board. HO-3D dataset provides annotated RGB-D frames of hands manipulating with the objects. The training data consists of $66,034$ frames and the test set is made of $11,524$ images. The dataset contains 68 sequences which are recorded from ten different subjects manipulating with ten distinct objects. The predictions of both 3D hand pose and shape are uploaded in a specified format on web portal[2] for obtaining the accuracies, and comparisons with other submissions can be seen on the portal. SynHand5M [16] is the largest synthetic dataset that contains fully annotated five million depth images for both the 3D hand pose and shape. The sizes of its training ($\mathcal{T}_{\mathrm{S}}$) and test sets are 4.5M and 500k, respectively. The hand model of SynHand5M is created synthetically [16], which is different from the MANO model. BigHand2.2M [63] is

1. https://competitions.codalab.org/competitions/20913
2. https://competitions.codalab.org/competitions/22485?



(a) w/o regis. (b) DispVoxNet (c) **GCN-MeshReg** (d) NRGA

Fig. 3. **Qualitative comparison of different registration methods on SynHand5M [16] dataset.** For a fair comparison, we use the HandVoxNet [17] framework and replace only the registration component by: (a) no registration; see the estimated hand surface (blue) and the ground truth (grey). (b) DispVoxNet; see artefacts and roughness in the estimated shapes. (c) GCN-MeshReg produces much smoother and more accurate final shapes. (d) NRGA produces smoother hand shapes, but the registered shapes are visually less accurate and slightly blown up.



(a) HandVoxNet [17] (b) Our Method

Fig. 4. **Qualitative Comparison of 3D hand shape estimation on HANDS19 [57] dataset:** We present a one-to-one comparison of the intermediate and final predictions of the two compared configurations. The first column shows the 3D voxelized depth maps. The second column shows the estimated 3D voxelized hand shapes. The third column shows the overlay of the estimated and the ground-truth hand surfaces. The fourth column shows the overlay of the registered (using the convolutional networks) and the ground-truth hand surfaces. The fifth column shows the final shapes. The right-most column ("Our Method") shows the registered hand shapes using the NRGA++. The red circles highlight the regions where significant differences in the estimations can be observed. Our approach with graph-convolution-based registration (*i.e.,* GCN-MeshReg) shows visually smoother and accurate final shapes.

a million-scale real hand pose dataset. It does not provide hand shape annotations. For pose estimation, it provides accurate joint annotations for 956k training depth images acquired from ten subjects. The size of the BigHand2.2M's test set is 296k. The results of 3D hand pose estimation are submitted to HANDS17 Challenge[3], which provides quantitative comparisons with the other submissions.

We use three evaluation metrics: (i) the average 3D joint location error over all test frames (3D $\mathcal{J}$ Err.); (ii) mean vertex location error over all test frames (3D $\mathcal{V}$ Err.); and (iii) mean voxelized shape error (*i.e.,* per-voxel binary cross-entropy) over all test data (3D $\mathcal{S}$ Err.).

6.2 Evaluation of Hand Shape Estimation

In this subsection, we demonstrate the ability of our algorithm to reconstruct real hand shapes. Moreover, we illustrate the effectiveness of our design choice by conducting an ablation study on the inputs (*i.e.,* $\mathcal{H}_j$ and V'_{D}). To this end, we evaluate our proposed approach on SynHand5M, HANDS19 Challenge (Task 1) and HO-3D datasets.

Synthetic Hand Shape Reconstruction. We train our HandVoxNet pipeline on SynHand5M by following the training

3. https://bit.ly/3iFmlzf

Method	3D $\mathcal{V}$ Err. (*mm*) With Reg.	Err. reduction after the Reg. (%)
HandVoxNet [17] (w/o Reg.)†	4.71	-
HandVoxNet [17] (with DispVoxNet)	4.14	12.1
Ours (w/o Reg.)	4.90	-
Ours (with GCN-MeshReg w/o ref.)	4.08	16.7
Ours (with GCN-MeshReg)	**3.57**	**27.1**
Ours (with NRGA++)	4.5	8.2

TABLE 4
3D hand shape estimation results on HANDS19 (Task 1) [57] dataset. The left column shows methods that use different registrations. Middle column shows the accuracy of the final shapes after the registration. The last column shows the percentage shape error reduction after registration. We observe significant improvement with GCN-MeshReg. The proposed registration method improves the shape estimation by 13.7% compared to HandVoxNet [17].

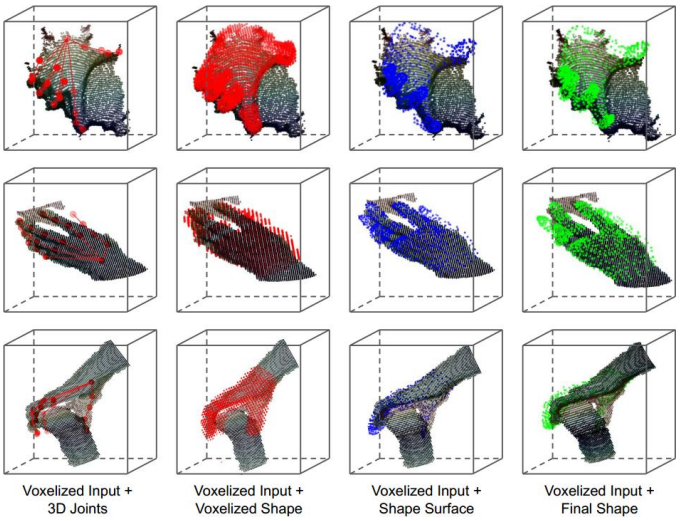


Fig. 5. **Qualitative results on HO-3D [20] dataset.** From left to right, we show overlays of the estimated 3D pose, 3D voxelized hand shape, hand surface and the final (registered) shape on the sample 3D voxelized depth inputs, respectively.

method described in Sec. 5. We conduct an ablation study on the inputs (*i.e.,* V'_{D} and $\mathcal{H}_j$) of V2V-ShapeNet and V2S-Net to show the effectiveness of our design choice. We regress $\hat{\mathcal{V}}_T$ and $\hat{\mathcal{V}}_S$ by using input V'_{D} (*i.e.,* without $\mathcal{H}_j$). Similar experiments are repeated by providing $\mathcal{H}_j$ (*i.e.,* without V'_{D}) and $\mathcal{I}_S$ (*i.e.,* with $\mathcal{H}_j \oplus \mathrm{V}'_{\mathrm{D}}$) as separate inputs to V2V-ShapeNet and V2S-Net. The results are summarized in Table 2 that clearly show the advantage of concatenating voxelized depth map with 3D heatmaps of joints.

Table 3 compares our algorithm (with different neural registration approaches) and state-of-the-art hand shape estimation algorithms. DispVoxNet [17] fits $\hat{\mathcal{V}}_T$ to the $\hat{\mathcal{V}}_S$, which results in 13.1% improvement in the surface reconstruction. Despite this quantitative improvement in the accuracy, the final shape suffers from severe artefacts, as shown in Fig. 3(b). Our proposed GCN-MeshReg not only outperforms DispVoxNet by 35.6% but also effectively removes artefacts in the shape estimation (see Table 3 and Fig. 3(c)). For a fair comparison of the above two registration methods, we use the occupancy grid representation [6], [17] of voxelized depth map as input to our approach. Furthermore, the accuracy estimated $\mathcal{V}_S$ (Table 2) is higher compared to WHSP-Net (Table 3), which clearly shows the effectiveness of employing 3D-CNN-based network for direct mesh vertex regression.

Methods	Main Err.	I.	S.	A.	V.
BT [64]	23.62	18.78	21.84	16.73	19.48
IPR [65]	19.63	8.42	14.21	7.5	14.16
NTIS [6]	15.57	4.54	12.05	4.21	8.47
AWR [66]	13.76	**3.93**	11.75	**3.65**	**7.50**
A2J [5]	13.74	6.33	11.23	6.05	8.78
V2V [6]	16.64	5.46	11.86	5.35	8.96
HandVoxNet [17]	15.57	5.98	11.48	5.73	9.12
Ours (Proj. D-TSDF)	13.94	6.18	**10.53**	5.89	8.98
Ours (Proj. TSDF)	**13.35**	6.08	10.69	5.83	8.91

TABLE 5
Comparison with the state-of-the-art methods in the task of depth-based 3D pose estimation in HANDS 2019 challenge. Our projective-TSDF-based implementation outperforms several methods in the main error and ranks first in the challenge. These error metrics are provided in mm.

Methods	Components	Runtime, *sec.*
HandVoxNet [17]	V2V-PoseNet	0.011
	V2V-ShapeNet	0.0015
	V2S-Net	0.0038
	DispVoxNet (GPU + CPU)*	0.162
	NRGA (CPU)	59 - 70
HandVoxNet++	V2V-PoseNet (Proj. TSDF)	0.022
	V2V-ShapeNet (Proj. TSDF)	0.0021
	V2S-Net (Proj. TSDF)	0.0043
	NRGA++	0.45
	GCN-MeshReg	0.0341

TABLE 6
Runtimes: forward-pass of deep networks on GPU. "*" shows that the Laplacian smoothing runs on CPU. The transformation of a depth map to TSDF-based representation is more time consuming compared to the occupancy grid. The execution times of NRGA++ and GCN-MeshReg are significantly improved over NRGA and DispVoxNet, respectively.

Real Hand Shape Reconstruction. HANDS19 Challenge dataset provides the shape annotations only for its training set. Therefore, to evaluate the performance of our complete method configuration on this dataset, we select approximately 90% of the original training data (*i.e.,* 158,355 frames) as the new train set ($\mathcal{T}_{\mathrm{H}}$) and the remaining data (*i.e.,* 17595 frames) as our test set. We train V2S-Net and V2V-ShapeNet on $\mathcal{T}_{\mathrm{H}}$ using the hyperparameters mentioned in Sec. 5. V2S-Net and V2V-ShapeNet accurately recover the hand shape representations. It is observed that the voxelized shape is more accurately estimated than the hand surface. Thereby, the alignment further refines the hand surface. Table 4 summarizes the quantitative comparison of the proposed HandVoxNet++ with HandVoxNet [17]. We train the method proposed in [17] on the HANDS19. We observe that the error reduction after the proposed GCN-MeshReg is 27.1%, while 12% error reduction is achieved for DispVoxNet-based registration [17]. A detailed qualitative comparison of the estimated shape representations is shown in Fig. 4. Notably, the artefacts can be clearly seen in the final shape estimation from DispVoxNet whereas, our proposed GCN-MeshReg does not suffer from such limitations and produces visually more accurate and smoother hand shapes thanks to the graph-convolution-based registration.

HO-3D is a new and challenging hand pose and shape dataset with hands interacting with objects. We train HandVoxNet++ on this dataset using the hyperparameters described in Sec. 5. The average 3D mesh error on the test set of HO-3D comes out to be 2.70cm. Because of high occlusion and appearance of random objects in the depth images, hand shape estimation task becomes harder. Despite

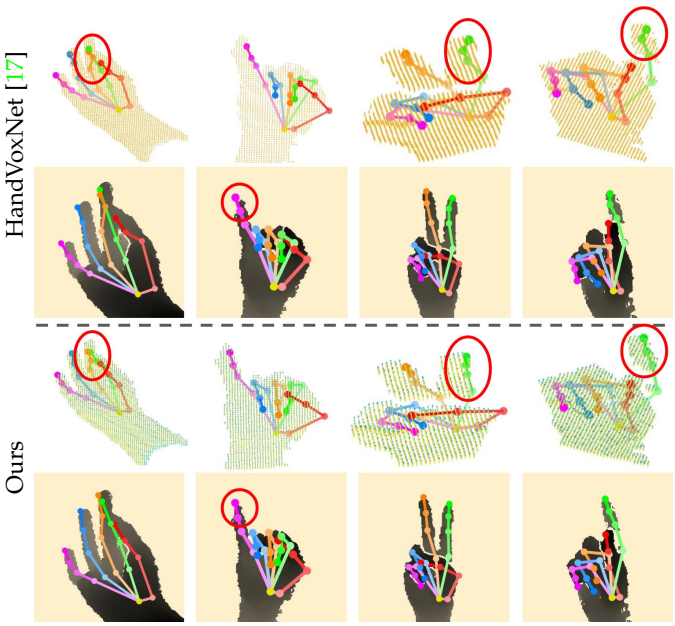


Fig. 6. **Qualitative results of depth-based 3D pose estimation on HANDS19 [57] dataset.** The top row shows the overlay of the estimated 3D pose on 3D voxelized depth input, and the bottom row shows the corresponding 2D overlay on to the depth image. The 3D view helps to better visualize the error in the estimated third dimension. Our 3D pose estimation results are more accurate compared to the occupancy-grid-based method [17], see the red circles for their differences.

we successfully recover plausible hand shapes on some challenging input voxelized depth maps (as shown qualitatively in Fig. 5).

Runtime. The runtimes of different components of HandVoxNet [17] and our proposed HandVoxNet++ are summarized in Table 6. The proposed GCN-MeshReg is 78.9% faster than DispVoxNet. Also, NRGA++ is ~150 times faster than NRGA on our hardware. With GCN-MeshReg, the runtime of each component is fast enough to develop a real-time interactive application when they are operated in parallel. Although TSDF representation of depth map allows improving the accuracy of 3D pose estimation however, generating this representation is more time consuming than producing the occupancy gird representation.

6.3 Evaluation of Hand Pose Estimation

As it can be seen in HandVoxNet++ pipeline, the shape accuracy depends on the accuracy of the estimated pose. Therefore, the hand pose estimation needs to be robust and accurate. In this work, we employ TSDF-based 3D voxelized representation which better encodes the 3D information of 2D depth map. We train V2V-PoseNet [6] using this representation on HANDS19 Challenge dataset for 3D pose estimation from single depth images (Task 1). This dataset contains hand images from egocentric and third-person viewpoints with no interaction with objects. The following error metrics are used for rigours evaluation; (i) **Main Err.**: It is the extrapolation error which uses test data containing hand shapes, viewpoints and poses that does not exist in the training set, (ii) **I.**: It is the interpolation error where test data contains hand shapes, poses and viewpoints which exist in the training set, (iii) **S.**: The test data contains hand shapes which are not present in the training data,

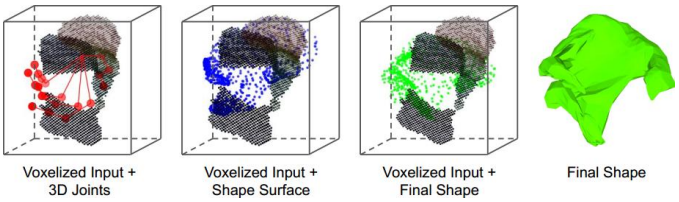


Fig. 7. **Failure case.** Our method is unable to produce plausible shapes in cases of severe occlusion of hand parts. We show our pose and shape estimations on a challenging input where most hand parts are occluded by an object.

(iv) **A.**: The test data contains hand poses which are not present in the training data, (v) **V.**: The test data contains viewpoints which are not included in the training set. Our results and comparison with the state-of-the-art methods are reported in Table 5. For a fair comparison, we also train the networks of [6] and [17] on HANDS19 Challenge dataset. Our method that uses the projective TSDF representation outperforms [6] and [17] by 19.8% and 14.2%, respectively. We observe that—in our case where a one-to-one mapping exists between the voxelized depth map and the 3D heatmaps—the projective TSDF representation converges faster and produces more accurate results compared to the projective directional TSDF representation. The qualitative comparison of our method with HandVoxNet [17] on some challenging hand poses is shown in Fig. 6. Among the state-of-the-art methods, our approach is **ranked first** during the moment of submission of our result on the web portal. Our method achieves 2.8% improvement in accuracy compared to Anchor-to-Joints (A2J) regression approach [5] which uses an ensemble predictions of 3 epochs. Also, in comparison to other existing approaches such as Adaptive Weighting Regression (AWR) [66] with an ensemble of five epochs, NTIS [6] with complex post-processing step including temporal smoothing, our method exceeds in the accuracy by 3.0% and 14.2%, respectively.

We train our best performing pose estimation network (that uses projective representation) on HO-3D dataset. By following [20] and according to the web challenge, we report the pose error in cm. The mean 3D joint location error on the test set of HO-3D is $2.46cm$. To the best of our knowledge, there is no published work so far which reports accuracy on this dataset using depth images. Qualitative results of pose estimation are shown in Fig. 5 (first column) on some challenging test samples where the hands are interacting with different objects.

Notably, our focus is to develop an effective approach for simultaneous hand pose and shape estimation. However, for completeness, we also train our method on HANDS17 Challenge[4] dataset which is created by sampling images from BigHand2.2M [63] and FHAD [61] datasets. The joint location error on the test set of HANDS17 dataset comes out to be $8.92mm$ which shows 10.3% and 3.7% improvement in accuracy compared to V2V [6] (*i.e.*, $9.95mm$) and HandVoxNet [17] (*i.e.*, $9.27mm$), respectively.

7 Discussion

Our pipeline relies on the availability of the ground truth of real hand shapes thus, we do not employ weak supervision

4. http://icvl.ee.ic.ac.uk/hands17/challenge/

or training with combined real and synthetic data. For this reason, we do not experiment with the older datasets such as NYU [47] and ICVL [59] which do not provide annotations of hand shapes. Rather we experiment with the state-of-the-art datasets (*i.e.,* HANDS19, HO-3D and SynHand5M) which provide annotations for both 3D hand pose and shape. We observe in the experiments that HandVoxNet++ significantly outperforms our previous method HandVoxNet [17] in the pose and shape accuracy, and runs faster. Our recovered hand shapes are smoother, and are less prone to artefacts. The reasons are manifold. First, it is due to the improved neural alignment component GCN-MeshReg with graph convolutional networks and iterative refinement cycle which incorporates the structure of hand mesh. Second, we found the projective TSDF representation establishes a more efficient one-to-one mapping with the 3D heatmaps of hand joints compared to the occupancy grid representation. We believe that, in the occupancy grid representation, a lot of information is lost during the binary quantization of 3D depth point clould. Whereas, the projective TSDF representation better encapsulates this information which results in an improved hand pose estimation accuracy. This claim is confirmed by the results on the HANDS19 (Task 1) challenge, where HandVoxNet++ ranks first at the moment of our submission in August 2020. The runtime performance of the proposed registration components has significantly improved over [17]. This improvement in GCN-MeshReg is mainly due to the fact that we do not need the time consuming hand shape surface voxelization step of DispVoxNet. Our generative registration approach NRGA and its extended version NRGA++ are presented as an alternative to the learning-based registration. These generative registration approaches remove the dependency of the registration task on the availability of the shape annotations. Although, such annotations are available in several recent datasets, they do not cover all possible hand shape variations. In this paper, the runtime of NRGA++ has significantly improved over NRGA specifically by using a pre-defined MANO hand model segmentation for the alignment.

All these improvements are reflected in the results on HANDS19 challenge, HO-3D and SynHand5M datasets. For HO-3D dataset, HandVoxNet++ is able to produce reasonable pose and shape estimates under the challenging scenario of hands interacting with objects, although it has not been explicitly designed for this scenario. We believe that our method brings us closer to in-the-wild AR and VR systems for hand shape estimation from depth sensors. HandVoxNet++ can regress over 30 hand shapes per second in its fully neural configuration, and has potential for improvements by considering temporal shape context.

Failure Cases. Our approach fails to estimate plausible hand shapes and poses in cases of severe occlusion of hand parts especially during hand-object interaction, see Fig. 7. The difficulty in the estimation under such a scenario might also increase due to the large variation in object shapes.

8 Conclusion and Future Work

We introduced a new HandVoxNet++ method for 3D hand shape and pose reconstruction from a single depth map, which establishes an effective inter-link between hand pose and shape estimations using 3D and graph convolutions. The experimental evaluation shows that the TSDF-based voxelized depth map representation establishes a more efficient one-to-one mapping with the 3D heatmaps of joint positions in comparison to the binary voxelized representation of the depth map. HandVoxNet++ produces more accurate hand shapes of real images compared to the previous methods, and our 3D data augmentation policy on voxelized grids further enhances the accuracy of 3D hand pose estimation. We achieve state-of-the-art results for 3D hand pose and shape estimation from depth images, which is confirmed on recent challenging benchmarks.

All these results indicate that the one-to-one mapping between voxelized depth map, voxelized shape and 3D heatmaps of joints is essential for an accurate hand shape and pose recovery. As future work, the voxelized depth map can be combined with the color image to further enrich the voxelized input representation with an additional cue. Another promising direction is an extension of our work for reconstructing shapes of interacting hands.

Acknowledgments

This work was funded by the German Federal Ministry of Education and Research as part of the project VIDETE (Grant 01IW18002), DECODE (Grant 01IW21001), and the ERC Consolidator Grant 770784.

References

[1] J. S. Supančič, G. Rogez, Y. Yang, J. Shotton, and D. Ramanan, "Depth-based hand pose estimation: methods, data, and challenges," *International Journal of Computer Vision (IJCV)*, vol. 126, no. 11, pp. 1180–1198, 2018. 1, 2

[2] S. Yuan, G. Garcia-Hernando, B. Stenger, G. Moon, J. Y. Chang, K. M. Lee, P. Molchanov, J. Kautz, S. Honari, L. Ge *et al.*, "Depth-based 3d hand pose estimation: From current achievements to future goals," in *Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 2

[3] F. Mueller, F. Bernard, O. Sotnychenko, D. Mehta, S. Sridhar, D. Casas, and C. Theobalt, "Ganerated hands for real-time 3d hand tracking from monocular rgb," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 49–59. 1, 2

[4] J. Malik, A. Elhayek, S. Guha, S. Ahmed, A. Gillani, and D. Stricker, "Deepairsig: End-to-end deep learning based in-air signature verification," *IEEE Access*, vol. 8, pp. 195 832–195 843, 2020. 1

[5] F. Xiong, B. Zhang, Y. Xiao, Z. Cao, T. Yu, J. T. Zhou, and J. Yuan, "A2j: Anchor-to-joint regression network for 3d articulated pose estimation from a single depth image," in *International Conference on Computer Vision (ICCV)*, 2019. 1, 3, 9, 10

[6] G. Moon, J. Yong Chang, and K. Mu Lee, "V2v-posenet: Voxel-to-voxel prediction network for accurate 3d hand and human pose estimation from a single depth map," in *Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 2, 3, 4, 7, 9, 10

[7] M. Rad, M. Oberweger, and V. Lepetit, "Feature mapping for learning fast and accurate 3d pose inference from synthetic images," in *Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 3

[8] J. Taylor, L. Bordeaux, T. Cashman, B. Corish, C. Keskin, T. Sharp, E. Soto, D. Sweeney, J. Valentin, B. Luff *et al.*, "Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, p. 143, 2016. 1

[9] J. Romero, D. Tzionas, and M. J. Black, "Embodied hands: modeling and capturing hands and bodies together," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 6, p. 245, 2017. 1, 2, 7

[10] J. Malik, A. Elhayek, and D. Stricker, "Whsp-net: A weakly-supervised approach for 3d hand shape and pose recovery from a single depth image," *Sensors*, vol. 19, no. 17, p. 3784, 2019. 1, 2, 5, 7

[11] N. Qian, J. Wang, F. Mueller, F. Bernard, V. Golyanik, and C. Theobalt, "HTML: A Parametric Hand Texture Model for 3D Hand Reconstruction and Personalization," in *European Conference on Computer Vision (ECCV)*, 2020. 1, 2

[12] X. Zhang, Q. Li, H. Mo, W. Zhang, and W. Zheng, "End-to-end hand mesh recovery from a monocular rgb image," in *International Conference on Computer Vision (ICCV)*, 2019. 1, 2

[13] F. Mueller, M. Davis, F. Bernard, O. Sotnychenko, M. Verschoor, M. A. Otaduy, D. Casas, and C. Theobalt, "Real-time Pose and Shape Reconstruction of Two Interacting Hands With a Single Depth Camera," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 4, 2019. 1

[14] L. Ge, Z. Ren, Y. Li, Z. Xue, Y. Wang, J. Cai, and J. Yuan, "3d hand shape and pose estimation from a single rgb image," in *Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2

[15] J. Malik, A. Elhayek, F. Nunnari, and D. Stricker, "Simple and effective deep hand shape and pose regression from a single depth image," *Computers & Graphics*, vol. 85, pp. 85–91, 2019. 1

[16] J. Malik, A. Elhayek, F. Nunnari, K. Varanasi, K. Tamaddon, A. Heloir, and D. Stricker, "Deephps: End-to-end estimation of 3d hand pose and shape by learning from synthetic depth," in *International Conference on 3D Vision (3DV)*, 2018. 1, 2, 7, 8

[17] J. Malik, I. Abdelaziz, A. Elhayek, S. Shimada, S. A. Ali, V. Golyanik, C. Theobalt, and D. Stricker, "Handvoxnet: Deep voxel-based network for 3d hand shape and pose estimation from a single depth map," in *Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 5, 7, 8, 9, 10, 11

[18] S. Shimada, V. Golyanik, E. Tretschk, D. Stricker, and C. Theobalt, "Dispvoxnets: Non-rigid point set alignment with supervised learning proxies," in *International Conference on 3D Vision (3DV)*, 2019. 2, 5, 6

[19] S. A. Ali, V. Golyanik, and D. Stricker, "Nrga: Gravitational approach for non-rigid point set registration," in *International Conference on 3D Vision (3DV)*, 2018. 2, 6

[20] S. Hampali, M. Rad, M. Oberweger, and V. Lepetit, "Honnotate: A method for 3d annotation of hand and object poses," in *Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 6, 7, 9, 10

[21] Y. Cai, L. Ge, J. Cai, N. Magnenat-Thalmann, and J. Yuan, "3d hand pose estimation using synthetic data and weakly labeled rgb images," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020. 2

[22] G. Park, A. Argyros, J. Lee, and W. Woo, "3d hand tracking in the presence of excessive motion blur," *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, vol. 26, no. 5, pp. 1891–1901, 2020. 2

[23] J. Nehvi, V. Golyanik, F. Mueller, H.-P. Seidel, M. Elgharib, and C. Theobalt, "Differentiable event stream simulator for non-rigid 3d tracking," in *CVPR Workshop on Event-based Vision*, 2021. 2

[24] V. Rudnev, V. Golyanik, J. Wang, H.-P. Seidel, F. Mueller, M. Elgharib, and C. Theobalt, "Eventhands: Real-time neural 3d hand pose estimation from an event stream," in *International Conference on Computer Vision (ICCV)*, 2021. 2

[25] Y. Hasson, B. Tekin, F. Bogo, I. Laptev, M. Pollefeys, and C. Schmid, "Leveraging photometric consistency over time for sparsely supervised hand-object reconstruction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2

[26] J. Yang, H. J. Chang, S. Lee, and N. Kwak, "Seqhand: Rgb-sequence-based 3d hand pose and shape estimation," in *European Conference on Computer Vision*, 2020. 2

[27] S. Baek, K. I. Kim, and T.-K. Kim, "Weakly-supervised domain adaptation via gan and mesh model for estimating 3d hand poses interacting objects," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2

[28] Y. Zhou, M. Habermann, W. Xu, I. Habibie, C. Theobalt, and F. Xu, "Monocular real-time hand shape and motion capture using multi-modal data," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[29] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *International Conference on Learning Representations (ICLR)*, 2017. 2, 5, 6

[30] N. Kolotouros, G. Pavlakos, and K. Daniilidis, "Convolutional mesh regression for single-image human shape reconstruction," in *CVPR*, 2019. 2, 5

[31] E. Tretschk, A. Tewari, M. Zollhöfer, V. Golyanik, and C. Theobalt, "DEMEA: Deep Mesh Autoencoders for Non-Rigidly Deforming Objects," *European Conference on Computer Vision (ECCV)*, 2020. 2

[32] B. Doosti, S. Naha, M. Mirbagheri, and D. Crandall, "Hope-net: A graph-based model for hand-object pose estimation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 3

[33] S. Shimada, V. Golyanik, C. Theobalt, and D. Stricker, "Ismo-gan: Adversarial learning for monocular non-rigid 3d reconstruction," in *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2019. 3

[34] V. Golyanik, S. Shimada, K. Varanasi, and D. Stricker, "Hdm-net: Monocular non-rigid 3d reconstruction with learned deformation model," in *International Conference on Virtual Reality and Augmented Reality (EuroVR)*, 2018. 3

[35] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum, "Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling," in *Neural Information Processing Systems (NeurIPS)*, 2016. 3

[36] Y. Chen, Z. Tu, L. Ge, D. Zhang, R. Chen, and J. Yuan, "So-handnet: Self-organizing network for 3d hand pose estimation with semi-supervised learning," in *International Conference on Computer Vision (ICCV)*, 2019. 3

[37] Y. Cai, L. Ge, J. Liu, J. Cai, T.-J. Cham, J. Yuan, and N. M. Thalmann, "Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks," in *International Conference on Computer Vision (ICCV)*, 2019. 3

[38] G. Poier, M. Opitz, D. Schinagl, and H. Bischof, "Murauer: Mapping unlabeled real data for label austerity," in *Winter Conference on Applications of Computer Vision (WACV)*, 2019. 3

[39] H. Guo, G. Wang, X. Chen, C. Zhang, F. Qiao, and H. Yang, "Region ensemble network: Improving convolutional network for hand pose estimation," in *International Conference on Image Processing (ICIP)*, 2017. 3

[40] L. Ge, H. Liang, J. Yuan, and D. Thalmann, "3d convolutional neural networks for efficient and robust hand pose estimation from single depth images," in *Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[41] J. Malik, A. Elhayek, S. Ahmed, F. Shafait, M. Malik, and D. Stricker, "3dairsig: A framework for enabling in-air signatures using a multi-modal depth sensor," *Sensors*, vol. 18, no. 11, p. 3872, 2018. 3

[42] J. Malik, A. Elhayek, and D. Stricker, "Structure-aware 3d hand pose regression from a single depth image," in *International Conference on Virtual Reality and Augmented Reality (EuroVR)*, 2018. 3

[43] C. Wan, T. Probst, L. Van Gool, and A. Yao, "Dense 3d regression for hand pose estimation," in *Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[44] L. Ge, Z. Ren, and J. Yuan, "Point-to-point regression pointnet for 3d hand pose estimation," in *European Conference on Computer Vision (ECCV)*, 2018. 3

[45] M. Oberweger and V. Lepetit, "Deepprior++: Improving fast and accurate 3d hand pose estimation," in *International Conference on Computer Vision Workshops (ICCVW)*, 2017. 3, 6

[46] J. Malik, A. Elhayek, and D. Stricker, "Simultaneous hand pose and skeleton bone-lengths estimation from a single depth image," in *International Conference on 3D Vision (3DV)*, 2017. 3

[47] J. Tompson, M. Stein, Y. Lecun, and K. Perlin, "Real-time continuous pose recovery of human hands using convolutional networks," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 5, p. 169, 2014. 3, 6, 11

[48] L. Ge, Z. Ren, Y. Li, Z. Xue, Y. Wang, J. Cai, and J. Yuan, "3d hand shape and pose estimation from a single rgb image," in *CVPR*, 2019. 3

[49] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2011. 4

[50] L. Ge, H. Liang, J. Yuan, and D. Thalmann, "Real-time 3d hand pose estimation with 3d convolutional neural networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 4, pp. 956–970, 2018. 4

[51] S. Song and J. Xiao, "Deep sliding shapes for amodal 3d object detection in rgb-d images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 4

[52] J. Vollmer, R. Mencl, and H. Mueller, "Improved laplacian smoothing of noisy surface meshes," in *Computer Graphics Forum*, 1999, pp. 131–138. 5

[53] A. Ranjan, T. Bolkart, S. Sanyal, and M. J. Black, "Generating 3d faces using convolutional mesh autoencoders," in *European Conference on Computer Vision (ECCV)*, 2018. 5

[54] S. Cheng, G. Tzimiropoulos, J. Shen, and M. Pantic, "Faster, better and more detailed: 3d face reconstruction with graph convolutional networks," in *Asian Conference on Computer Vision (ACCV)*, 2020. 5

[55] H. Chui and A. Rangarajan, "A new point matching algorithm for non-rigid registration," *Comput. Vis. Image Underst.*, vol. 89, no. 2–3, p. 114–141, 2003. 6

[56] B. Jian and B. C. Vemuri, "A robust algorithm for point set registration using mixture of gaussians," in *International Conference on Computer Vision (ICCV)*, 2005. 6

[57] A. Armagan, G. Garcia-Hernando, S. Baek, S. Hampali, M. Rad, Z. Zhang, S. Xie, M. Chen, B. Zhang, F. Xiong *et al.*, "Measuring generalisation to unseen viewpoints, articulations, shapes and objects for 3d hand pose estimation under hand-object interaction," in *European Conference on Computer Vision (ECCV)*, 2020. 6, 7, 8, 9, 10

[58] X. Sun, Y. Wei, S. Liang, X. Tang, and J. Sun, "Cascaded hand pose regression," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 824–832. 6

[59] D. Tang, H. Jin Chang, A. Tejani, and T.-K. Kim, "Latent regression forest: Structured estimation of 3d articulated hand posture," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3786–3793. 6, 11

[60] T. Sharp, C. Keskin, D. Robertson, J. Taylor, J. Shotton, D. Kim, C. Rhemann, I. Leichter, A. Vinnikov, Y. Wei *et al.*, "Accurate, robust, and flexible real-time hand tracking," in *Conference on Human Factors in Computing Systems (CHI)*, 2015. 6

[61] G. Garcia-Hernando, S. Yuan, S. Baek, and T.-K. Kim, "First-person hand action benchmark with rgb-d videos and 3d hand pose annotations," in *Computer Vision and Pattern Recognition (CVPR)*, 2018. 6, 10

[62] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, 2015. 7

[63] S. Yuan, Q. Ye, B. Stenger, S. Jain, and T.-K. Kim, "Bighand2.2m benchmark: Hand pose dataset and state of the art analysis," in *Computer Vision and Pattern Recognition (CVPR)*, 2017. 7, 10

[64] L. Yang, S. Li, D. Lee, and A. Yao, "Aligning latent spaces for 3d hand pose estimation," in *International Conference on Computer Vision (ICCV)*, 2019. 9

[65] X. Sun, B. Xiao, F. Wei, S. Liang, and Y. Wei, "Integral human pose regression," in *European Conference on Computer Vision (ECCV)*, 2018. 9

[66] W. Huang, P. Ren, J. Wang, Q. Qi, and H. Sun, "Awr: Adaptive weighting regression for 3d hand pose estimation." in *Conference on Artificial Intelligence (AAAI)*, 2020. 9, 10

Jameel Malik is a postdoctoral researcher in the Augmented Vision group at the German Research Center for Artificial Intelligence (DFKI GmbH), Kaiserslautern. He received the PhD degree in computer science from Technische Universität Kaiserslautern in 2020 for his work on depth-based 3D hand pose and shape estimation, master's degree in electrical engineering from the School of Electrical Engineering and Computer Science (SEECS), National University of Sciences and Technology (NUST), Pakistan. His current research interests are in the areas of computer vision, deep learning and their applications.

Soshi Shimada is a Ph.D. candidate in the Visual Computing and Artificial Intelligence Department at the Max Planck Institute for Informatics in Saarbrucken, Germany. Before, he received his M.Sc. in Computer Science from Technische Universität Kaiserslautern, and B.Eng. in Computer Science and Engineering from Waseda University in Japan. His research interests are in computer vision, computer graphics, and machine learning, with a focus on 3D pose estimation. He currently works on monocular-based 3D human pose estimation.

Ahmed Elhayek received the master's degree from Saarland University (Germany) in 2010. In 2015, he acquired his Ph.D. degree from both the Max-Planck-Institute and Saarland University. Thereafter, he worked as a PostDoc researcher in the Augmented Vision group at DFKI (German Research Centre for Artificial Intelligence). In 2018, he joined the faculty of Computer and Cyber Sciences in UPM (the University of Prince Mugrin) as an Assistant Professor.

Sk Aziz Ali is a researcher at the Augmented Vision group of German Research Center for Artificial Intelligence, Kaiserslautern. He is doing the doctoral research at Technische Universität Kaiserslautern, in the areas of rigid and non-rigid motion fields estimation, physics-based optimization methods and 3D reconstruction. He received the B.Tech degree from West Bengal University of Technology and the M.Sc degree in computer science from Technische Universität Kaiserslautern in 2011 and 2017, respectively.

Christian Theobalt is a Professor of Computer Science and the Director of the Visual Computing and Artificial Intelligence Department at the Max Planck Institute for Informatics, Saarbruecken, Germany. He is also a professor at Saarland University. His research lies at the Intersection of Computer Graphics, Computer Vision and Machine Learning. For instance, he works on virtual humans, 3D and 4D scene reconstruction, neural rendering and neural scene representations, marker-less motion and performance capture, machine learning for graphics and vision, and new sensors for 3D acquisition. Christian received several awards, for instance the Otto Hahn Medal of the Max-Planck Society (2007), the EUROGRAPHICS Young Researcher Award (2009), the German Pattern Recognition Award (2012), the EURIGRAPHICS Outstanding Technical Contributions Award (2020), an ERC Starting Grant (2013) and an ERC Consolidator Grant (2017). He is a co-founder of theCaptury (www.thecaptury.com).

Vladislav Golyanik is leading the "4D and Quantum Vision" research group (4dqv.mpi-inf.mpg.de/) at the Visual Computing and Artificial Intelligence Department of the Max Planck Institute for Informatics (MPII), Saarbrücken, Germany. The primary research interests of his team include 3D reconstruction and analysis of deformable scenes, matching problems on point sets and graphs, neural rendering, quantum computer vision and event-based vision. He received a doctoral degree in informatics from the University of Kaiserslautern in 2019, advised by Didier Stricker. Prior to joining MPII as a post-doctoral researcher, Vladislav was a visiting fellow at NVIDIA (San José, USA), and Institute of Robotics and Industrial Informatics (Barcelona, Spain). He is the recipient of the WACV'16 best paper award and the 2020's yearly dissertation award of the German Association for Pattern Recognition (DAGM).

Didier Stricker is Professor in Computer Science at Technische Universität Kaiserslautern and Scientific Director at the German Research Center for Artificial Intelligence (DFKI GmbH) in Kaiserslautern, where he leads the research department 'Augmented Vision'. He received the Innovation Prize of the German Society of Computer Science in 2006. He got several awards for best papers or demonstrations at different conferences. He serves as reviewer for different European or National research organizations. He is a reviewer of different journals and conferences in the area of VR/AR and Computer Vision. His research interests are cognitive interfaces, user monitoring and on-body-sensor-networks, computer vision, video/image analytics, and human computer interaction.