Learning to Reconstruct High-quality 3D Shapes with Cascaded Fully Convolutional Networks

*Yan-Pei Cao[1,2] , *Zheng-Ning Liu[1] , Zheng-Fei Kuang[1] , Leif Kobbelt[3] , Shi-Min Hu[1]
* equal contribution

[1]Tsinghua University [2]Owlii Inc. [3]RWTH Aachen University
{caoyanpei,lzhengning}@gmail.com, kzf15@mails.tsinghua.edu.cn,
kobbelt@cs.rwth-aachen.de, shimin@tsinghua.edu.cn

Abstract. We present a data-driven approach to reconstructing high-resolution and detailed volumetric representations of 3D shapes. Although well studied, algorithms for volumetric fusion from multi-view depth scans are still prone to scanning noise and occlusions, making it hard to obtain high-fidelity 3D reconstructions. In this paper, inspired by recent advances in efficient 3D deep learning techniques, we introduce a novel cascaded 3D convolutional network architecture, which learns to reconstruct implicit surface representations from noisy and incomplete depth maps in a progressive, coarse-to-fine manner. To this end, we also develop an algorithm for end-to-end training of the proposed cascaded structure. Qualitative and quantitative experimental results on both simulated and real-world datasets demonstrate that the presented approach outperforms existing state-of-the-art work in terms of quality and fidelity of reconstructed models.

Keywords: high-fidelity 3D reconstruction, cascaded architecture

1 Introduction

High-quality reconstruction of 3D objects and scenes is key to 3D environment understanding, mixed reality applications, as well as the next generation of robotics, and has been one of the major frontiers of computer vision and computer graphics research for years [18, 30, 42, 39, 13]. Meanwhile, the availability of consumer-grade RGB-D sensors, such as the *Microsoft Kinect* and the *Intel RealSense*, involves more novice users to the process of scanning surrounding 3D environments, opening up the need for robust reconstruction algorithms which are resilient to errors in the input data (e.g., noise, distortion, and missing areas).

In spite of recent advances in 3D environment reconstruction, acquiring high-fidelity 3D shapes with imperfect data from casual scanning procedures and consumer-level RGB-D sensors is still a particularly challenging problem. Since the pioneering *KinectFusion* work [39], many 3D reconstruction systems, both real-time [18, 52, 32, 59, 29] and offline [13], have been proposed, which often use a volumetric representation of the scene geometry, i.e., the truncated signed

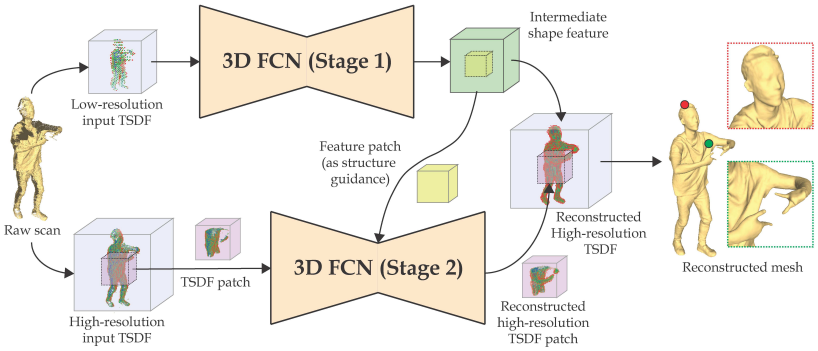


Fig. 1: Illustration of a two-stage 3D-CFCN architecture. Given partial and noisy raw depth scans as input, a fused low-resolution TSDF volume is fed to the stage-1 3D fully convolutional network (3D-FCN), producing an intermediate representation. Exploiting this intermediate feature, the network then 1) regresses a low-resolution but complete TSDF and 2) predicts which TSDF patches should be further refined. For each patch that needs further refinements, the corresponding block is cropped from a fused high-resolution input TSDF, and the stage-2 3D-FCN uses it to infer a detailed high-resolution local TSDF volume, which substitutes the corresponding region in the aforementioned regressed TSDF and thus improves the output's resolution. Note a patch of the global intermediate representation also flows into stage 2 to provide structure guidance. The rightmost column shows the high-quality reconstruction. Close-ups show accurately reconstructed details, e.g., facial details, fingers, and wrinkles on clothes. Note the input scan is fused from 4 viewpoints.

distance function (TSDF) [17]. However, depth measurement acquired by consumer depth cameras contains a significant amount of noise, plus limited scanning angles lead to missing areas, making vanilla depth fusion suffer from blurring surface details and incomplete geometry. Another line of research [30, 40, 46] focuses on reconstructing complete geometry from noisy and sparsely-sampled point clouds, but cannot process point clouds with a large percentage of missing data and may produce bulging artifacts.

The wider availability of large-scale 3D model repositories [61, 6] stimulates the development of data-driven approaches for shape reconstruction and completion. Assembly-based methods, such as [49, 10], require carefully *segmented* 3D databases as input, operate on a few specific classes of objects, and can only generate shapes with limited variety. On the other hand, recent deep learning-based approaches [14, 54, 63, 60, 55, 50, 28, 22, 51, 64, 48] mostly focus on inferring 3D geometry from single-view images [54, 63, 55, 50, 22, 51, 64] or high-level information [60, 48] and often get stuck at low resolutions (typically 32^3 voxel resolution) due to high memory consumption, which is far too low for recovering geometric details.

In this work, we present a coarse-to-fine approach to high-fidelity volumetric reconstruction of 3D shapes from noisy and incomplete inputs using a 3D cascaded fully convolutional network (3D-CFCN) architecture, which outperforms state-of-the-art alternatives regarding the resolution and accuracy of reconstructed models. Our approach chooses recently introduced octree-based efficient 3D deep learning data structures [43, 53, 56] as the basic building block, however, instead of employing a standard single-stage convolutional neural network (CNN), we propose to use multi-stage network cascades for detailed shape information reconstruction, where the object geometry is predicted and refined progressively via a sequence of sub-networks. The rationale for choosing the cascaded structure is two-fold. First, to predict high-resolution (e.g., 512^3, 1024^3, or even higher) geometry information, one may have to deploy a deeper 3D neural network, which could significantly increase memory requirements even using memory-efficient data representations. Second, by splitting the geometry inference into multiple stages, we also simplify the learning tasks, since each sub-network now only needs to learn to reconstruct 3D shapes at a certain resolution.

Training a cascaded architecture is a nontrivial task, particularly when octree-based data representations are employed, where both the *structure* and the *value* of the output octree need to be predicted. We thus design the sub-networks to learn where to refine the 3D space partitioning of the input volume, and the same information is used to guide the data propagation between consecutive stages as well, which makes end-to-end training feasible by avoiding exhaustively propagating every volume block.

The primary contribution of our work is a novel learning-based, progressive approach for high-fidelity 3D shape reconstruction from imperfect data. To train and quantitatively evaluate our model on real-world 3D shapes, we also contribute a dataset containing both detailed full body reconstructions and raw depth scans of 10 subjects. We then conduct careful experiments on both simulated and real-world datasets, comparing the proposed framework to a variety of state-of-the-art alternatives. These experiments show that, when dealing with noisy and incomplete inputs, our approach produces 3D shapes with significantly higher accuracy and quality than other existing methods.[1]

2 Related Work

There has been a large body of work focused on 3D reconstruction over the past a few decades. We refer the reader to [2] and [9] for detailed surveys of methods for reconstructing 3D objects from point clouds and RGB-D streams, respectively. Here we only summarize the most relevant previous approaches and categorize them as geometric, assembly-based, and learning-based approaches.

Geometric Approaches. In the presence of sample noise and missing data, many choose to exploit the smoothness assumption, which constrains the reconstructed geometry to satisfy a certain level of smoothness. Gradient-domain

[1] We will make our 3D-CFCN implementation publicly available.

methods [30, 1, 4] require that the input point clouds be equipped with (oriented) normals and utilize them to estimate an implicit soft indicator function which discriminates the interior region from the exterior of a 3D shape. Similarly, [5, 36] use globally supported radial basis functions (RBFs) to interpolate the surface. On the other hand, a series of moving least squares (MLS) -based methods [25, 41] attack 3D reconstruction by fitting the input point clouds to a spatially varying low-degree polynomial. By assuming local or global surface smoothness, these approaches, to a certain extent, are robust to noise, outliers, and missing data.

Sensor visibility is another widely used prior in scan integration for object and scene reconstruction [17, 23], which acts as an effective regularizer for structured noise [65] and can be used to infer empty spaces. For large-scale indoor scene reconstruction, since the prominent KinectFusion, plenty of systems [18, 13, 29] have been proposed. However, they are mostly focused on improving the accuracy and robustness of camera tracking in order to obtain a globally consistent model.

Compared to these methods, we propose to learn natural 3D shape priors from massive training samples for shape completion and reconstruction, which better explores the 3D shape space and avoids undesired reconstructed geometries resulted from hand-crafted priors.

Assembly-based Approaches. Another line of work assumes that a target object can be described as a composition of primitive shapes (e.g., planes, cuboids, spheres, etc.) or known object parts. [45, 8] detect primitives in input point clouds of CAD models and optimize their placement as well as the spatial relationship between them via graph cuts. The method introduced in [47] first interactively segments the input point cloud and then retrieves a complete and similar 3D model to replace each segment, while [10] extends this idea by exploiting the contextual knowledge learned from a scene database to automate the segmentation as well as improve the accuracy of shape retrieval. To increase the granularity of the reconstruction to the object component level, [49] proposes to reassemble parts from different models, aiming to find the combination of candidates which conforms the input RGB-D scan best. Although these approaches can deal with partial input data and bring in semantic information, 3D models obtained by them still suffer from the lack of geometric diversity.

Learning-based Approaches. 3D deep neural networks have achieved impressive results on various tasks [61, 7, 15], such as 3D shape classification, retrieval, and segmentation. As for generative tasks, previous research mostly focuses on inferring 3D shapes from (single-view) 2D images, either with only RGB channels [14, 54, 63, 60, 55, 50, 28], or with depth information [22, 51, 64]. While showing promising advances, these techniques are only capable of generating rough 3D shapes at low resolutions. Similarly, in [48, 57], shape completion is also performed on low-resolution voxel grids due to the high demand of computational resources.

Aiming to complete and reconstruct 3D shapes at higher resolutions, [19] proposes a 3D Encoder-Predictor Network (3D-EPN) to firstly predict a coarse but complete shape volume and then refine it via an iterative volumetric patch

synthesis process, which copy-pastes voxels from k-nearest-neighbors to improve the resolution of each predicted patch. [26] extends 3D-EPN by introducing a local 3D CNN to perform patch-level surface refinement. However, these methods both need separate and time-consuming steps before local inference, either nearest neighbor queries [19], or 3D boundary detection [26]. By contrast, our approach only requires a single forward pass for 3D shape reconstruction and produces higher-resolution results (e.g., 512^3 vs. 128^3 or 256^3). On the other hand, [53, 27] propose efficient 3D convolutional architectures by using octree representations, which are designed to decode high-resolution geometry information from dense intermediate features; nevertheless, no volumetric convolutional encoders and corresponding shape reconstruction architectures are provided. While [42] presents an OctNet-based [43] end-to-end deep learning framework for depth fusion, it refines the intermediate volumetric output globally, which makes it infeasible for producing reconstruction results at higher resolutions even with memory-efficient data structures. Instead, our 3D-CFCN learns to refine output volumes at the level of local patches, and thus significantly reduces the memory and computational cost.

3 Method

This section introduces our 3D-CFCN model. We first give a condensed review of relevant concepts and techniques in Sec. 3.1. Then we present the proposed architecture and its corresponding training pipeline in Sec. 3.2 and Sec. 3.3. Sec. 3.4 summaries the procedure of collecting and generating the data which we used for training our model.

3.1 Preliminaries

Volumetric Representation and Integration. The choice of underlying data representation for fusing depth measurements is key to high-quality 3D reconstruction. Approaches varies from point-based representations [31, 58], 2.5D fields [24, 38], to volumetric methods based on occupancy maps [62] or implicit surfaces [17, 18]. Among them, TSDF-based volumetric representations have become the preferred method due to their ability to model continuous surfaces, efficiency for incremental updates in parallel, and simplicity for extracting surface interfaces. In this work, we adopt the definition of TSDF from [39]:

$$\mathrm{V}(\mathbf{p}) = \Psi(\mathrm{S}(\mathbf{p})), \tag{1}$$

$$\mathrm{S}(\mathbf{p}) = \begin{cases} \|\mathbf{p} - \partial\Omega\|_2, & if\ \mathbf{p} \in \Omega \\ -\|\mathbf{p} - \partial\Omega\|_2, & if\ \mathbf{p} \in \Omega^{\mathrm{c}} \end{cases}, \tag{2}$$

$$\Psi(\eta) = \begin{cases} \min(1, \frac{\eta}{\mu})\,\mathrm{sgn}(\eta), & if\ \eta \geq -\mu \\ invalid, & otherwise \end{cases}, \tag{3}$$

where S is the standard signed distance function (SDF) with Ω being the object volume, and Ψ denotes the truncation function with μ being the corresponding

truncation threshold. The truncation is performed to avoid surface interference, since in practice during scan fusion, the depth measurement is only locally reliable due to surface occlusions. In essence, a TSDF obliviously encodes free space, uncertain measurements, and unknown areas.

Given a set of depth scans at hand, we follow the approach in [17] to integrate them into a TSDF volume:

$$\mathrm{V}(\mathbf{p}) = \frac{\sum w_i(\mathbf{p})\, \mathrm{V}_i(\mathbf{p})}{\sum w_i(\mathbf{p})}, \tag{4}$$

where $\mathrm{V}_i(\mathbf{p})$ and $w_i(\mathbf{p})$ are the TSDFs and weight functions from the i-th depth scan, respectively.

OctNet. 3D CNNs are a natural choice for operating TSDF volumes under the end-to-end learning framework. However, the cubic growth of computational and memory requirements becomes a fundamental obstacle for training and deploying 3D neural networks at high resolution. Recently, there emerges several work [43, 53, 56] that propose to exploit the sparsity in 3D data and employ octree-based data structures to reduce the memory consumption, among which we take OctNet [43] as our basic building block.

In OctNet, features and data are organized in the *grid-octree* data structure, which consists of a grid of shallow octrees with maximum depth 3. The structure of shallow octrees are encoded as bit strings so that the features and data of sparse octants can be packed into continuous arrays. Common operations in convolutional networks (e.g., convolution, pooling and unpooling) are defined on the grid-octree structure correspondingly. Therefore, the computational and memory cost are significantly reduced, while the OctNet itself, as a processing module, can be plugged into most existing 3D CNN architectures transparently. However, one major limitation of OctNet is that the structure of grid-octrees is determined by the input data and keeps fixed during training and inference, which is undesirable for reconstruction tasks where hole filling and detail refinement need to be performed. We thus propose an approach to eliminate this drawback in Sec. 3.2.

3.2 Architecture

Our 3D-CFCN is a cascade of volumetric reconstruction modules, which are OctNet-based fully convolutional sub-networks aiming to infer missing surface areas and refine geometric details. Each module $\mathcal{M}^i$ operates at a given voxel resolution and spatial extent. We find *512^3 voxel resolution* and a corresponding *two-stage* architecture suffice to common daily 3D scanning tasks in our experiments, and thus will concentrate on this architecture in the rest of the paper; nevertheless, the proposed 3D-CFCN framework can be easily extended to support arbitrary resolutions and number of stages.

In our implementation, for both sub-networks, we adopt the U-net architecture [44] while substituting convolution and pooling layers with the corresponding operations from OctNet. Skip connections are also employed between

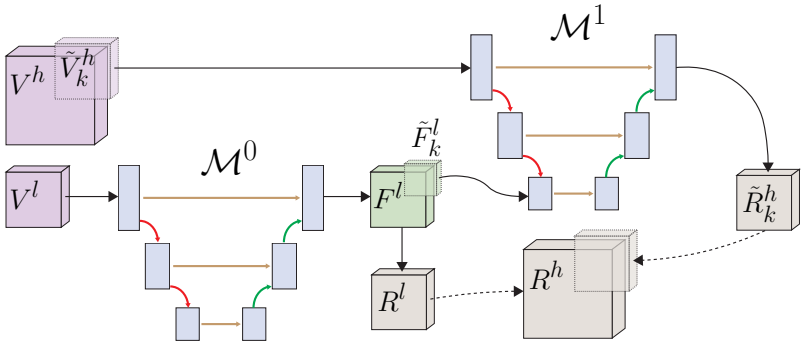


Fig. 2: Architecture of a two-stage 3D-CFCN. In this case, the network takes a pair of low- and high-resolution (i.e., 128^3 and 512^3) noisy TSDF volume $\{V_l, V_h\}$ as input, and produces a refined TSDF at 512^3 voxel resolution.

corresponding encoder and decoder layers to make sure the structures of input volumes are preserved in the inferred output predictions. To complete the partial input data and refine its grid-octree structure, we refrain from using OctNet's unpooling operation and propose a *structure refinement module*, which learns to predict whether an octant needs to be split for recovering finer geometric details.

The first sub-network, $\mathcal{M}^0$, receives the encoded low-resolution (i.e., 128^3) TSDF volume V^l (see Sec. 3.4), which is fused from raw depth scans $\{\mathcal{D}_i\}$ of an 3D object $\mathcal{S}$, as input and produces a feature map F^l as well as a reconstructed TSDF volume R^l at the same resolution. Then for each 16^3 patch $\tilde{F}_k^l$ of F^l, we use a modified structure refinement module to predict if its corresponding block in R^l needs further improvement.

If a TSDF patch $\tilde{R}_k^l$ is predicted to be further refined, we then crop its corresponding 64^3 patch $\tilde{V}_k^h$ from V^h, which is an encoded TSDF volume fused from the same depth scans $\{\mathcal{D}_i\}$, but at a higher voxel resolution, i.e., 512^3. $\tilde{V}_k^h$ is next fed to the second stage $\mathcal{M}^1$ to produce a local feature map $\tilde{F}_k^h$ with increased *spatial* resolution and reconstruct a more detailed local 3D patch $\tilde{R}_k^h$ of $\mathcal{S}$. Meanwhile, since input local TSDF patches $\{\tilde{V}_k^h\}$ may suffer from a large portion of missing data, we also propagate $\{\tilde{F}_k^l\}$ to incorporate global guidance. More specifically, a propagated $\tilde{F}_k^l$ is concatenated with the high-level 3D feature map after the second pooling layer in $\mathcal{M}^1$ (see Fig. 2). Note this extra path, in return, also helps to refine F^l during back propagation. Finally, the regressed local TSDF patch $\{\tilde{R}_k^h\}$ is substituted back into the global TSDF, which can be further used to extract surfaces.

To avoid inconsistency across TSDF patch boundaries, we add interval overlaps when cropping feature maps and TSDF volumes. When cropping $\{\tilde{F}_k^l\}$, we expand two more voxels on each side of the 3D patch, making the actual resolution of $\{\tilde{F}_k^l\}$ grow to 20^3; similarly, for $\{\tilde{V}_k^h\}$ and $\{\tilde{F}_k^h\}$, we apply 8-voxel overlapping and increase their resolution to 80^3. However, when substituting

back $\{\tilde{R}_k^h\}$, overlapping regions are discarded. So in its essence, this cropping approach acts as a smart padding scheme. Note that all local patches are still organized in grid-octrees.

Structure Refinement Module. Since the unpooling operation of OctNet restrains the possibility of refining the octree structure on-the-fly, inspired by [42, 53], we propose to replace unpooling layers with a structure refinement module. Instead of inferring new octree structures implicitly from reconstructions as in [42], we use 3^3 convolutional filters to directly predict from feature maps whether an octant should be further split. In contrast, OGN[53] predicts three-state masks using 1^3 filters followed by three-way softmax. To determine if a 3D local patch needs to be fed to $\mathcal{M}^1$, we take the average "split score" of all the octants in this patch and compare it with a confidence threshold ρ ($= 0.5$). By employing this adaptive partitioning and propagation scheme, we achieve high-resolution volumetric reconstruction while keeping the computational and memory cost to a minimum level.

3.3 Training

The 3D-CFCN is trained in a supervised fashion on a TSDF dataset $\{\mathcal{F}_n = \{V^l, V^h, G^l, G^h\}\}$ in two phases, where V^l and V^h denote the incomplete input TSDFs at low and high voxel resolution, while G^l and G^h are low- and high-resolution ground-truth TSDFs, respectively.

In the first phase, $\mathcal{M}^0$ is trained alone with a hybrid of ℓ_1, binary cross entropy, and structure loss:

$$\mathcal{L}(\theta; V^l, G^l) = \mathcal{L}_{\ell_1} + \lambda_1 \mathcal{L}_{bce} + \lambda_2 \mathcal{L}_s. \tag{5}$$

The ℓ_1 term is designed for TSDF denoising and reconstruction, and we employ the auxiliary binary cross entropy loss $\mathcal{L}_{bce}$ to provide the network more guidance for learning shape completion; while in our experiments, we find $\mathcal{L}_{bce}$ also leads to faster convergence. Our structure refinement module is learned with $\mathcal{L}_s$, where

$$\mathcal{L}_s = \frac{1}{|\mathcal{O}|} \sum_{o \in \mathcal{O}} BCE\left(1 - f(o', T_{gt}), p(o)\right). \tag{6}$$

Here, $\mathcal{O}$ represents the set of octants in the current grid-octree, and BCE denotes the binary cross entropy. $p(o)$ is the prediction of whether the octant o should to be split, while o' is the corresponding octant of o in the ground-truth grid-octree structure T_{gt} (in this case, the structure of G_l). We define $f(o', T_{gt})$ as an indicator function that identifies whether o' exists in T_{gt}:

$$f(o', T_{gt}) = \begin{cases} 1, \exists\ \tilde{o}',\ \textit{such that}\ h(\tilde{o}') \leq h(o') \\ 0, \textit{otherwise} \end{cases}, \tag{7}$$

where h denotes the height of an octant in the octree.

Furthermore, we employ multi-scale supervision [20, 15] to alleviate potential gradient vanishing. Specifically, after each pooling operation, the feature map is

concatenated with a downsampled input TSDF volume at the corresponding resolution, and we evaluate the downscaled hybrid loss at each structure refinement layer.

In the second phase, $\mathcal{M}^1$ is trained; at the same time, $\mathcal{M}^0$ is being fine-tuned. To alleviate over-fitting and speed up the training process, among all the local patches that are predicted to be fed to $\mathcal{M}^1$, we keep only K of them randomly and discard the rest (we set $K = 2$ across our experiments). At this stage, the inferred global structure $\tilde{F}_k^l$ flows into $\mathcal{M}^1$ to guide the shape completion, while the refined local features also provide feedbacks and improves $\mathcal{M}^0$. The same strategy, i.e., hybrid loss (see Eq. 5) and multi-scale supervision, is adopted here when training $\mathcal{M}^1$ together with $\mathcal{M}^0$.

3.4 Training Data Generation

Synthetic Dataset. Our first dataset is built upon the synthetic 3D shape repository ModelNet40 [61]. We choose a subset of 10 categories, with 4051 shape instances in total (3245 for training, 806 for testing). Similar to existing approaches, we set up virtual cameras around the objects[2] and render depth maps, then simulate the volumetric fusion process [17] to generate ground-truth TSDFs. To produce noisy and partial training samples, previous methods [18, 42, 26] add random noise and holes to the depth maps to mimic sensor noise. However, synthetic noise reproduced by this approach usually does not conform real noise distributions. Thus, we instead implement a synthetic stereo depth camera [21]. Specifically, we virtually illuminate 3D shapes with a structured light pattern, which is extracted from *Asus XTion* sensors using [12, 37], and apply the PatchMatch Stereo algorithm [3] to estimate disparities (and hence depth maps) across stereo speckle images. In this way, the distribution of noise and missing area in synthesized depth images behaves much closer to real ones, thus makes the trained network generalize better on real-world data. In our experiments, we pick 2 or 4 virtual viewpoints randomly when generating training samples.

In essence, apart from shape completion, learning volumetric depth fusion is to seek a function $g(\{\mathcal{D}_1, \ldots, \mathcal{D}_n\})$ that maps raw depth scans to a noise free TSDF. Therefore, to retain information from all input depth scans, we adopt the histogram-based TSDF representation (TSDF-Hist) proposed in [42] as the encoding of our input training samples. A 10D smoothed-histogram, which uses 5 bins for negative and 5 bins for positive distances, with the first and the last bin reserved for truncated distances, is allocated for each voxel. The contribution of a depth observation is distributed linearly between the two closest bins. For outputs, we simply choose plain 1-dimensional TSDFs as the representation.

Since we employ a cascaded architecture and use multi-scale supervision during network training, we need to generate training and ground-truth sample pairs at multiple resolutions. Specifically, TSDFs at 32^3, 64^3, 128^3, 256^3, and 512^3 voxel resolutions are simultaneously generated in our experiments.

[2] We place virtual cameras at the vertices of a icosahedron.

Real-world Dataset. We construct a high-quality dynamic 3D reconstruction (or, free-viewpoint video, FVV) system similar to [16] and collect 10 4D sequences of human actions, each capturing a different subject. Then a total of 9746 frames are randomly sampled from the sequences and split into training and test set by the ratio of 4 : 1. We name this dataset as *Human10*. For each frame, we fuse 2 or 4 randomly picked *raw depth scans* and obtain the TSDF-Hist encodings of the training sample; while the ground-truth TSDFs is produced by virtually scanning (see the previous section) the corresponding output triangle mesh of our FVV system. The sophisticated pipeline of our FVV system guarantees the quality and accuracy of the output mesh, however, the design and details of the FFV system is beyond the scope of this paper.

4 Experiments

We have evaluated our 3D-CFCN architecture on both ModelNet40 and Human10 and compared different aspects of our approach with other state-of-the-art alternatives.[3]

4.1 High-resolution Shape Reconstruction

In our experiments, we train the 3D-CFCN separately on each dataset for 20 epochs (12 for stage 1, 8 for two stages jointly), using the ADAM optimizer [33] with 0.0001 learning rate, which takes ≈ 80 hours to converge. Balancing weights in Eq. 5 are set to: $\lambda_1 = 0.5$ and $\lambda_2 = 0.1$. During inference, it takes ≈ 3.5 s on average to perform a forward pass through both stages on a NVIDIA GeForce GTX 1080 Ti. The Marching Cubes algorithm [35] is used to extract surfaces from output TSDFs. Figs. 1, 3, and 4 illustrate the high-quality reconstruction results achieved with our 3D-CFCN architecture.

In Fig. 3 we show a variety of test cases from both Human10 and ModelNet40 dataset. All the input TSDF-Hists were fused using depth maps from 2 viewpoints, and the same TSDF truncation threshold were applied. Despite the presence of substantial noise and missing data, our approach was able to reduce the noise and infer the missing structures, producing clean and detailed reconstructions. Comparing the second and the third column, for Human10 models, stage 2 of our 3D-CFCN significantly improved the quality by bringing more geometric details to output meshes; on the other hand, 128^3 voxel resolution suffices to ModelNet40, thus stage 2 does not show significant improves in these cases.

Auxiliary Visual Hull Information. In practice, most depth sensors can also capture synchronized color images, which opens up the possibility of getting auxiliary segmentation masks [11]. Given the segmentation masks from each view, a corresponding visual hull [34], which is essentially an occupancy volume, can be extracted. Visual hulls provide additional information about the distribution of occupied and empty spaces, which is important for shape completion.

[3] Please find more experiment results in the supplementary material.

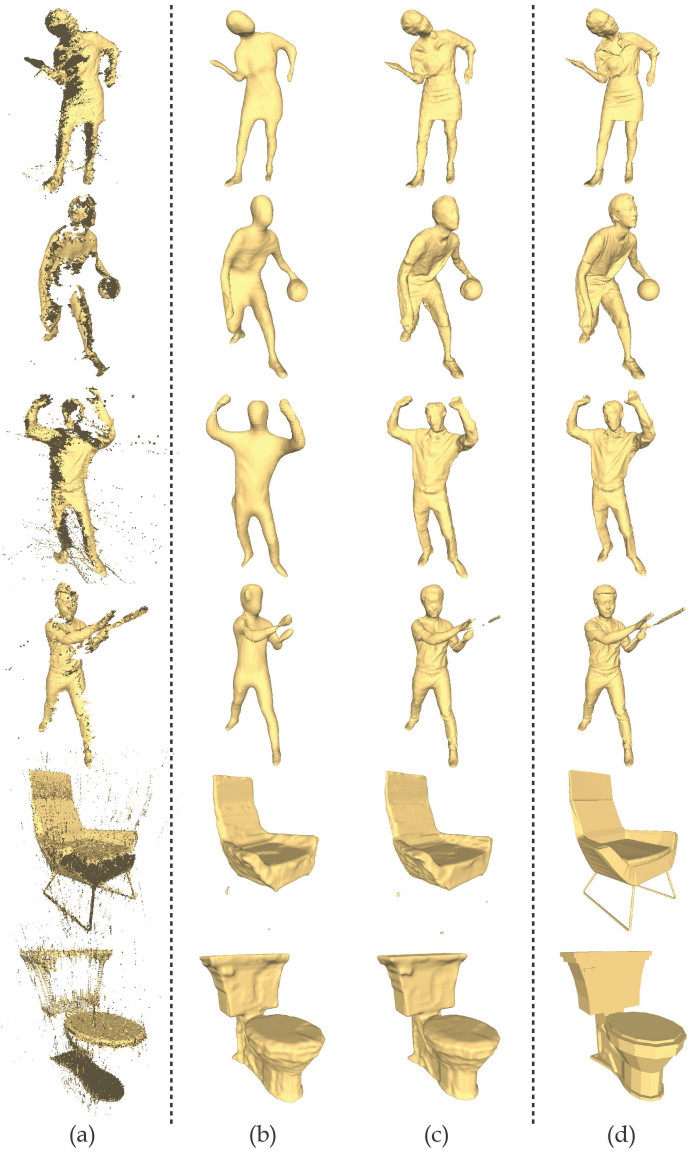


Fig. 3: Results gallery. (a): Input scans fused from 2 randomly picked viewpoints. (b): Reconstruction results of the first stage of our 3D-CFCN. (c): Full-resolution reconstruction results of the two-stage 3D-CFCN architecture. (d): Ground-truth references.

We thus evaluated the performance of our 3D-CFCN when visual hull information is available. Towards this goal, we added corresponding visual hull input

branches to both two stages, which are concatenated with intermediate features after two 3^3 convolutional layers. Table 1 reports the average Hausdorff RMS distance between predicted and ground-truth 3D meshes, showing that using additional visual hull volumes as input brought a performance gain around 11%. Both TSDF-Hists and visual hull volumes in this experiment were generated using 2 viewpoints. Note that we also scaled the models in Human10 to fit into a 3^3 bounding box.

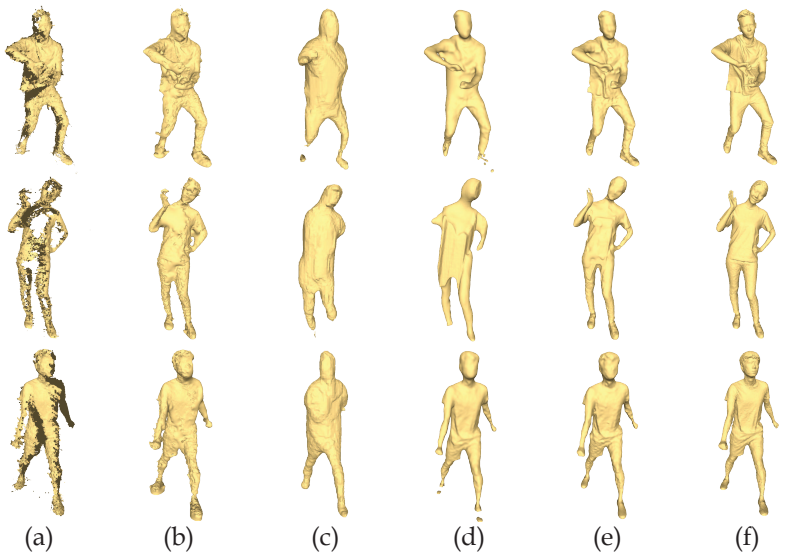


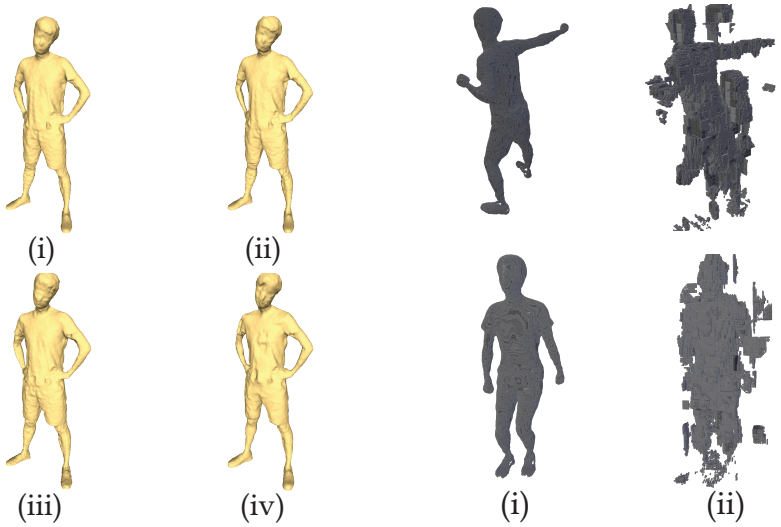
Fig. 4: Comparison of our reconstruction results with other state-of-the-art alternatives. (a): Input scans. (b): PSR[30]. (c): 3D-EPN[19]. (d): OctNetFusion[42]. (e): Ours. (f): Ground-truth references. Note the bulging artifacts on PSR's results.

Table 1: Quantitative comparisons of shape reconstruction techniques. Relative Hausdorff RMS distance with respect to the diagonals of bounding boxes are measured against the ground-truth triangle meshes. All baseline methods use input data fused from 2 views.

	PSR	3D-EPN	OctNet-Fusion	3D-CFCN (2 views)	3D-CFCN (2 views w/ visual hull)	3D-CFCN (4 views)
Human10	0.0092	0.0263	0.0040	0.0035	0.0031	0.0021
ModelNet40	0.0620	0.0178	0.0035	0.0032	0.0019	0.0010

Number of Viewpoints. Here we evaluated the impact of the completeness of input TSDF-Hists, i.e., the number of viewpoints used for fusing raw depth scans, on reconstruction quality. We trained and tested the 3D-CFCN architecture using TSDF-Hists fused from 2 and 4 viewpoints, listing the results in Table 1. As expected, using more depth scans led to increasing accuracy of output meshes, since input TSDF-Hists were less incomplete.

Robustness to Calibration and Tracking Error. Apart from sensor noise, calibration and tracking error is another major factor that can crack scanned models. To evaluate the robustness of the proposed approach to calibration and tracking error, we added random perturbations (from 2.5% to 10%) to ground-truth camera poses, generated corresponding test samples, and predicted the reconstruction results using 3D-CFCN. As shown in Fig. 5(a), although the network has not been trained on samples with calibration error, it can still infers geometric structures reasonably.



(a) Reconstruction results of the proposed 3D-CFCN under different levels of calibration error. (i): No error. (ii): 2.5%. (iii): 5%. (iv): 10%.

(b) Comparison with OGN. (i): Occupancy maps reconstructed by 3D-CFCN. (ii): Occupancy maps decoded by OGN, using features learned by 3D-CFCN.

Fig. 5: Evaluation and comparisons.

4.2 Comparison with Existing Approaches

Fig. 4 and Table 1 compare our 3D-CFCN architecture with three learning-based state-of-the-art alternatives for 3D shape reconstruction, i.e., OctNetFusion [42],

3D-EPN [19], and OGN [53], as well as the widely used geometric method Poisson surface reconstruction (PSR) [30].

OctNetFusion. Similar to our approach, OctNetFusion adopts OctNet as the building block and learns to denoise and complete input TSDFs in a multi-stage manner. However, each stage in OctNetFusion is designed to take an up-sampled TSDF and refine it globally (i.e., each stage needs to process *all* the octants in the grid-octree at the current resolution), making it infeasible to reconstruct 3D shape at higher resolutions, as learning at higher resolutions (e.g., 512^3) not only increases the memory cost at input and output layers, but also requires deeper network structures, which further challenges the limited computational resource. Fig. 4 and Table 1 summarize the comparison of our reconstruction results at 512^3 voxel resolution with OctNetFusion's results at 256^3.

3D-EPN. Without using octree-based data structures, 3D-EPN employs a hybrid approach, which first completes the input model at a low resolution (32^3) via a 3D CNN and then uses voxels from similar high-resolution models in the database to produce output distance volumes at 128^3 voxel resolution. However, as shown in Fig. 4, while being able to infer the overall shape of input models, this approach fails to recover fine geometric details due to the limited resolution.

OGN. As another relevant work to our 3D-CFCN architecture, OGN is a octree-based convolutional decoder. Although scales well to high resolution outputs, it remains challenging to recover accurate and detailed geometry information from encoded shape features via only deconvolution operations. To compare our approach with OGN, we trained the proposed 3D-CFCN on Human10 dataset to predict occupancy volumes, extracted 32^3 intermediate feature from the stage-1 3D FCN of our architecture, and used these feature maps to train an OGN. Fig. 5(b) compares the occupancy maps decoded by OGN with the corresponding occupancy volumes predicted by the proposed 3D-CFCN (both at 512^3 resolution), showing that our method performs significantly better than OGN with respect to fidelity and accuracy.

5 Conclusions

We have presented a cascaded 3D convolutional network architecture for efficient and high-fidelity shape reconstruction at high resolutions. Our approach refines the volumetric representations of partial and noisy input models in a progressive and adaptive manner, which substantially simplifies the learning task and reduces computational cost. Experimental results demonstrate that the proposed method can produce high-quality reconstructions with accurate geometric details. We also believe that extending the proposed approach to reconstructing sequences is a promising direction.

Acknowledgement

This work was supported by the Joint NSFC-DFG Research Program (project number 61761136018), and the Natural Science Foundation of China (Project Number 61521002).

References

1. Alliez, P., Cohen-Steiner, D., Tong, Y., Desbrun, M.: Voronoi-based variational reconstruction of unoriented point sets. In: Symposium on Geometry processing. vol. 7, pp. 39–48 (2007)
2. Berger, M., Tagliasacchi, A., Seversky, L.M., Alliez, P., Guennebaud, G., Levine, J.A., Sharf, A., Silva, C.T.: A survey of surface reconstruction from point clouds. In: Computer Graphics Forum. vol. 36, pp. 301–329. Wiley Online Library (2017)
3. Bleyer, M., Rhemann, C., Rother, C.: Patchmatch stereo - stereo matching with slanted support windows. In: BMVC (January 2011), https://www.microsoft.com/en-us/research/publication/patchmatch-stereo-stereo-matching-with-slanted-support-windows/
4. Calakli, F., Taubin, G.: Ssd: Smooth signed distance surface reconstruction. In: Computer Graphics Forum. vol. 30, pp. 1993–2002. Wiley Online Library (2011)
5. Carr, J.C., Beatson, R.K., Cherrie, J.B., Mitchell, T.J., Fright, W.R., McCallum, B.C., Evans, T.R.: Reconstruction and representation of 3d objects with radial basis functions. In: Proceedings of the 28th annual conference on Computer graphics and interactive techniques. pp. 67–76. ACM (2001)
6. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012 (2015)
7. Charles, R.Q., Su, H., Kaichun, M., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on. pp. 77–85. IEEE (2017)
8. Chauve, A.L., Labatut, P., Pons, J.P.: Robust piecewise-planar 3d reconstruction and completion from large-scale unstructured point data. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. pp. 1261–1268. IEEE (2010)
9. Chen, K., Lai, Y.K., Hu, S.M.: 3d indoor scene modeling from rgb-d data: a survey. Computational Visual Media **1**(4), 267–278 (2015)
10. Chen, K., Lai, Y., Wu, Y.X., Martin, R.R., Hu, S.M.: Automatic semantic modeling of indoor scenes from low-quality rgb-d data using contextual information. ACM Transactions on Graphics **33**(6) (2014)
11. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. arXiv preprint arXiv:1606.00915 (2016)
12. Chen, Q., Koltun, V.: Fast mrf optimization with application to depth reconstruction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3914–3921 (2014)
13. Choi, S., Zhou, Q.Y., Koltun, V.: Robust reconstruction of indoor scenes. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5556–5565 (June 2015)
14. Choy, C.B., Xu, D., Gwak, J., Chen, K., Savarese, S.: 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In: European Conference on Computer Vision. pp. 628–644. Springer (2016)
15. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3d u-net: learning dense volumetric segmentation from sparse annotation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 424–432. Springer (2016)

16. Collet, A., Chuang, M., Sweeney, P., Gillett, D., Evseev, D., Calabrese, D., Hoppe, H., Kirk, A., Sullivan, S.: High-quality streamable free-viewpoint video. ACM Trans. Graph. **34**(4), 69:1–69:13 (Jul 2015). https://doi.org/10.1145/2766945, http://doi.acm.org/10.1145/2766945
17. Curless, B., Levoy, M.: A volumetric method for building complex models from range images. In: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques. pp. 303–312. SIGGRAPH '96, ACM, New York, NY, USA (1996). https://doi.org/10.1145/237170.237269, http://doi.acm.org/10.1145/237170.237269
18. Dai, A., Nießner, M., Zollhöfer, M., Izadi, S., Theobalt, C.: Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration. ACM Trans. Graph. **36**(3), 24:1–24:18 (May 2017), http://doi.acm.org/10.1145/3054739
19. Dai, A., Qi, C.R., Nießner, M.: Shape completion using 3d-encoder-predictor cnns and shape synthesis. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). vol. 3 (2017)
20. Dou, Q., Yu, L., Chen, H., Jin, Y., Yang, X., Qin, J., Heng, P.A.: 3d deeply supervised network for automated segmentation of volumetric medical images. Medical image analysis **41**, 40–54 (2017)
21. Fanello, S.R., Valentin, J., Rhemann, C., Kowdle, A., Tankovich, V., Davidson, P., Izadi, S.: Ultrastereo: Efficient learning-based matching for active stereo systems. In: Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on. pp. 6535–6544. IEEE (2017)
22. Firman, M., Mac Aodha, O., Julier, S., Brostow, G.J.: Structured prediction of unobserved voxels from a single depth image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5431–5440 (2016)
23. Fuhrmann, S., Goesele, M.: Fusion of depth maps with multiple scales. In: ACM Transactions on Graphics (TOG). vol. 30, p. 148. ACM (2011)
24. Gallup, D., Pollefeys, M., Frahm, J.M.: 3d reconstruction using an n-layer heightmap. In: Joint Pattern Recognition Symposium. pp. 1–10. Springer (2010)
25. Guennebaud, G., Gross, M.: Algebraic point set surfaces. In: ACM Transactions on Graphics (TOG). vol. 26, p. 23. ACM (2007)
26. Han, X., Li, Z., Huang, H., Kalogerakis, E., Yu, Y.: High-resolution shape completion using deep neural networks for global structure and local geometry inference. In: IEEE International Conference on Computer Vision (ICCV) (October 2017)
27. Häne, C., Tulsiani, S., Malik, J.: Hierarchical surface prediction for 3d object reconstruction. arXiv preprint arXiv:1704.00710 (2017)
28. Ji, M., Gall, J., Zheng, H., Liu, Y., Fang, L.: Surfacenet: An end-to-end 3d neural network for multiview stereopsis. arXiv preprint arXiv:1708.01749 (2017)
29. Kähler, O., Prisacariu, V.A., Murray, D.W.: Real-Time Large-Scale Dense 3D Reconstruction with Loop Closure, pp. 500–516. Springer International Publishing, Cham (2016), http://dx.doi.org/10.1007/978-3-319-46484-8-30
30. Kazhdan, M., Hoppe, H.: Screened poisson surface reconstruction. ACM Trans. Graph. **32**(3), 29:1–29:13 (Jul 2013), http://doi.acm.org/10.1145/2487228.2487237
31. Keller, M., Lefloch, D., Lambers, M., Izadi, S., Weyrich, T., Kolb, A.: Real-time 3d reconstruction in dynamic scenes using point-based fusion. In: 3D Vision-3DV 2013, 2013 International Conference on. pp. 1–8. IEEE (2013)
32. Kerl, C., Sturm, J., Cremers, D.: Robust odometry estimation for rgb-d cameras. In: 2013 IEEE International Conference on Robotics and Automation. pp. 3748–3754 (May 2013). https://doi.org/10.1109/ICRA.2013.6631104
33. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)

34. Kutulakos, K.N., Seitz, S.M.: A theory of shape by space carving. International journal of computer vision **38**(3), 199–218 (2000)
35. Lorensen, W.E., Cline, H.E.: Marching cubes: A high resolution 3d surface construction algorithm. In: ACM siggraph computer graphics. vol. 21, pp. 163–169. ACM (1987)
36. Macedo, I., Gois, J.P., Velho, L.: Hermite radial basis functions implicits. In: Computer Graphics Forum. vol. 30, pp. 27–42. Wiley Online Library (2011)
37. McIlroy, P., Izadi, S., Fitzgibbon, A.: Kinectrack: 3d pose estimation using a projected dense dot pattern. IEEE transactions on visualization and computer graphics **20**(6), 839–851 (2014)
38. Meilland, M., Comport, A.I.: On unifying key-frame and voxel-based dense visual slam at large scales. In: Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on. pp. 3677–3683. IEEE (2013)
39. Newcombe, R.A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A.J., Kohi, P., Shotton, J., Hodges, S., Fitzgibbon, A.: Kinectfusion: Real-time dense surface mapping and tracking. In: 2011 10th IEEE International Symposium on Mixed and Augmented Reality. pp. 127–136 (Oct 2011)
40. Oeztireli, A.C., Guennebaud, G., Gross, M.: Feature Preserving Point Set Surfaces based on Non-Linear Kernel Regression. Computer Graphics Forum (2009). https://doi.org/10.1111/j.1467-8659.2009.01388.x
41. Öztireli, A.C., Guennebaud, G., Gross, M.: Feature preserving point set surfaces based on non-linear kernel regression. In: Computer Graphics Forum. vol. 28, pp. 493–501. Wiley Online Library (2009)
42. Riegler, G., Ulusoy, A.O., Bischof, H., Geiger, A.: Octnetfusion: Learning depth fusion from data. In: Proceedings of the International Conference on 3D Vision (2017)
43. Riegler, G., Ulusoy, A.O., Geiger, A.: Octnet: Learning deep 3d representations at high resolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. vol. 3 (2017)
44. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015)
45. Schnabel, R., Degener, P., Klein, R.: Completion and reconstruction with primitive shapes. In: Computer Graphics Forum. vol. 28, pp. 503–512. Wiley Online Library (2009)
46. Shan, Q., Curless, B., Furukawa, Y., Hernandez, C., Seitz, S.M.: Occluding contours for multi-view stereo. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition. pp. 4002–4009 (June 2014)
47. Shao, T., Xu, W., Zhou, K., Wang, J., Li, D., Guo, B.: An interactive approach to semantic modeling of indoor scenes with an rgbd camera. ACM Transactions on Graphics (TOG) **31**(6), 136 (2012)
48. Sharma, A., Grau, O., Fritz, M.: Vconv-dae: Deep volumetric shape learning without object labels. In: European Conference on Computer Vision. pp. 236–250. Springer (2016)
49. Shen, C.H., Fu, H., Chen, K., Hu, S.M.: Structure recovery by part assembly. ACM Trans. Graph. **31**(6), 180:1–180:11 (Nov 2012). https://doi.org/10.1145/2366145.2366199, http://doi.acm.org/10.1145/2366145.2366199
50. Sinha, A., Unmesh, A., Huang, Q., Ramani, K.: Surfnet: Generating 3d shape surfaces using deep residual networks. In: Proc. CVPR (2017)

51. Song, S., Yu, F., Zeng, A., Chang, A.X., Savva, M., Funkhouser, T.: Semantic scene completion from a single depth image. In: Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on. pp. 190–198. IEEE (2017)
52. Steinbrcker, F., Sturm, J., Cremers, D.: Real-time visual odometry from dense rgb-d images. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops). pp. 719–722 (Nov 2011). https://doi.org/10.1109/ICCVW.2011.6130321
53. Tatarchenko, M., Dosovitskiy, A., Brox, T.: Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs. In: IEEE International Conference on Computer Vision (ICCV) (2017), http://lmb.informatik.uni-freiburg.de/Publications/2017/TDB17b
54. Tatarchenko, M., Dosovitskiy, A., Brox, T.: Multi-view 3d models from single images with a convolutional network. In: European Conference on Computer Vision. pp. 322–337. Springer (2016)
55. Tulsiani, S., Zhou, T., Efros, A.A., Malik, J.: Multi-view supervision for single-view reconstruction via differentiable ray consistency. In: CVPR. vol. 1, p. 3 (2017)
56. Wang, P.S., Liu, Y., Guo, Y.X., Sun, C.Y., Tong, X.: O-cnn: Octree-based convolutional neural networks for 3d shape analysis. ACM Transactions on Graphics (SIGGRAPH) **36**(4) (2017)
57. Wang, W., Huang, Q., You, S., Yang, C., Neumann, U.: Shape inpainting using 3d generative adversarial network and recurrent convolutional networks. arXiv preprint arXiv:1711.06375 (2017)
58. Whelan, T., Leutenegger, S., Salas-Moreno, R.F., Glocker, B., Davison, A.J.: Elasticfusion: Dense slam without a pose graph. Robotics: Science and Systems (2015)
59. Whelan, T., Salas-Moreno, R.F., Glocker, B., Davison, A.J., Leutenegger, S.: Elasticfusion: Real-time dense slam and light source estimation. The International Journal of Robotics Research **35**(14), 1697–1716 (2016). https://doi.org/10.1177/0278364916669237, https://doi.org/10.1177/0278364916669237
60. Wu, J., Zhang, C., Xue, T., Freeman, B., Tenenbaum, J.: Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In: Advances in Neural Information Processing Systems. pp. 82–90 (2016)
61. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1912–1920 (2015)
62. Wurm, K.M., Hornung, A., Bennewitz, M., Stachniss, C., Burgard, W.: Octomap: A probabilistic, flexible, and compact 3d map representation for robotic systems. In: Proc. of the ICRA 2010 workshop on best practice in 3D perception and modeling for mobile manipulation. vol. 2 (2010)
63. Yan, X., Yang, J., Yumer, E., Guo, Y., Lee, H.: Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In: Advances in Neural Information Processing Systems. pp. 1696–1704 (2016)
64. Yang, B., Wen, H., Wang, S., Clark, R., Markham, A., Trigoni, N.: 3d object reconstruction from a single depth view with adversarial learning. arXiv preprint arXiv:1708.07969 (2017)
65. Zach, C., Pock, T., Bischof, H.: A globally optimal algorithm for robust tv-l 1 range image integration. In: Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on. pp. 1–8. IEEE (2007)