CurveFusion: Reconstructing Thin Structures from RGBD Sequences

LINGJIE LIU*, University of Hong Kong and University College London
NENGLUN CHEN*, University of Hong Kong
DUYGU CEYLAN, Adobe Research
CHRISTIAN THEOBALT, Max Planck Institute for Informatics
WENPING WANG, University of Hong Kong
NILOY J. MITRA, University College London

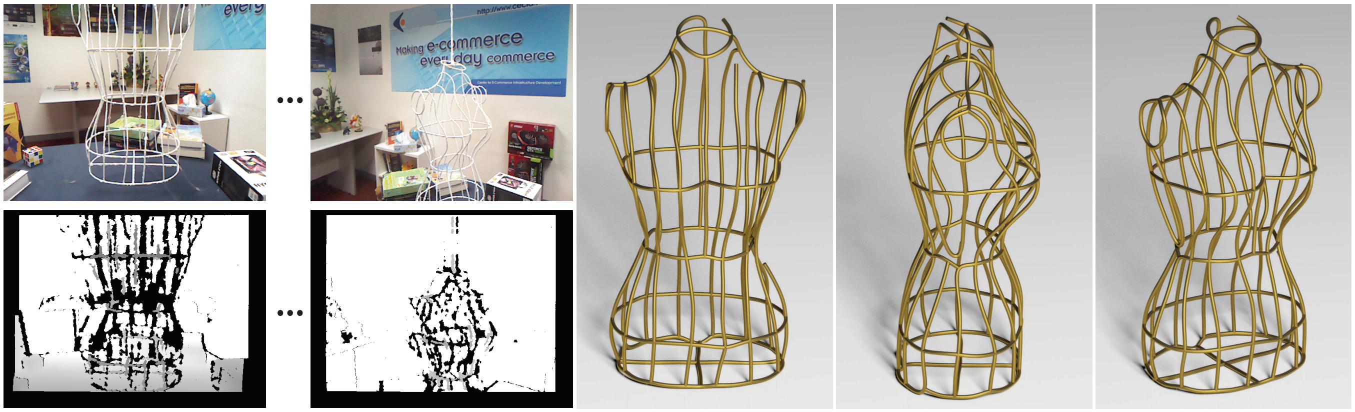

Fig. 1. We introduce CurveFusion, a method to reconstruct objects made of thin filament-like structures from an RGBD sequence by fusing information from noisy depth scans and loosely using RGB information for verification only.

We introduce CurveFusion, the first approach for high quality scanning of thin structures at interactive rates using a handheld RGBD camera. Thin filament-like structures are mathematically just 1D curves embedded in $\mathbb{R}^3$, and integration-based reconstruction works best when depth sequences (from the thin structure parts) are fused using the object's (unknown) curve skeleton. Thus, using the complementary but noisy color and depth channels, CurveFusion first automatically identifies point samples on potential thin structures and groups them into *bundles*, each being a group of a fixed number of aligned consecutive frames. Then, the algorithm extracts per-bundle skeleton curves using L_1 axes, and aligns and iteratively merges the L_1 segments from all the bundles to form the final complete curve skeleton. Thus, unlike previous methods, reconstruction happens via integration along a *data-dependent fusion primitive*, i.e., the extracted curve skeleton. We extensively evaluate CurveFusion on a range of challenging examples, different scanner and calibration settings, and present high fidelity thin structure reconstructions previously just not possible from raw RGBD sequences.

CCS Concepts: • **Computing methodologies → Parametric curve and surface models**;

Additional Key Words and Phrases: curve reconstruction, L_1 axis, data fusion, RGBD scans

ACM Reference Format:
Lingjie Liu, Nenglun Chen, Duygu Ceylan, Christian Theobalt, Wenping Wang, and Niloy J. Mitra. 2018. CurveFusion: Reconstructing Thin Structures from RGBD Sequences. *ACM Trans. Graph.* 37, 6, Article 218 (November 2018), 12 pages. https://doi.org/10.1145/3272127.3275097

1 INTRODUCTION

The past few years have seen significant research progress in the context of scanning of large-scale 3D environments. With easy access to commodity depth cameras producing real-time feeds of RGBD sequences, the most successful approach aligns and integrates many such low-quality input frames and extracts a *fused* surface. Well-known examples include KinectFusion [Newcombe et al. 2011], BundleFusion [Dai et al. 2017], etc.

The above family of methods implicitly assumes that the scanned environments consist of volumetric objects with closed or extended boundary surfaces. Hence, multiple depth scans, once aligned, can be effectively integrated using a (truncated) signed distance field (TSDF) representation [Curless and Levoy 1996] discretized on a pre-defined

*These two authors contributed equally

Authors' addresses: Lingjie Liu, University of Hong Kong , University College London, liulingjie0206@gmail.com; Nenglun Chen, University of Hong Kong, chennenglun@gmail.com; Duygu Ceylan, Adobe Research, ceylan@adobe.com; Christian Theobalt, Max Planck Institute for Informatics, theobalt@mpi-inf.mpg.de; Wenping Wang, University of Hong Kong, wenping@cs.hku.hk; Niloy J. Mitra, University College London, n.mitra@cs.ucl.ac.uk.

0730-0301/2018/11-ART218 $15.00
https://doi.org/10.1145/3272127.3275097

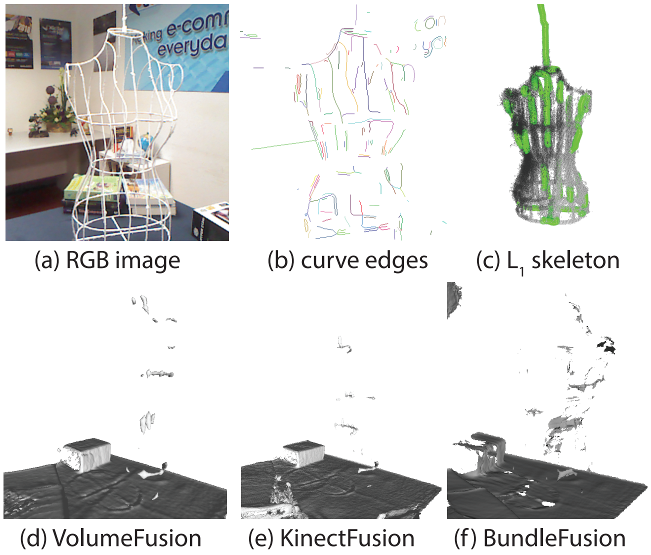

Fig. 2. For the model in Figure 1, voxel based fusion methods fail to recover the thin structure: results shown using VolumeFusion [Curless and Levoy 1996], KinectFusion [Newcombe et al. 2011], and BundleFusion [Dai et al. 2017]. We show an RGB frame for reference, curves extracted from RGB, and the effect of extracting the L_1 axis from the aligned depth data of the whole RGBD sequence, which yields poor reconstruction.

voxel grid, and a surface is extracted using Marching Cubes or its variants.

Such volumetric fusion approaches simply fail to capture objects or object parts primarily consisting of thin or filamentary structures (e.g., features less than 4 mm in diameter). Figure 2 shows typical results of a scanning session using different fusion approaches: volumetric TSDF [Curless and Levoy 1996], KinectFusion [Newcombe et al. 2011], and BundleFusion [Dai et al. 2017]. The thin parts are either noisy or completely missed in the fused output.

Specifically, capturing thin structures by 'fusing' several RGBD frames is challenging for multiple reasons: (i) An RGBD camera's depth data is noisy and of limited spatial resolution – parts of thin structures are regularly missed, or captured at incorrect depths in the raw depth images [Teichman et al. 2013], and there is no connectivity information among the few detected points. Combining aligned depth images from multiple viewpoints can only partially recover thin structure data already missed in individual input frames. Further, the inevitable camera drift is particularly detrimental when trying to align thin structure data. (ii) A depth camera's RGB channel often provides complementary information at higher pixel resolution. Unfortunately, RGB images of thin structures are equally challenging to detect and segment since it is very difficult to distinguish curve edges from texture/background edges (see Figure 2b). (iii) Moreover, despite the considerable success of existing works on image segmentation [Cao et al. 2017; Fu et al. 2017; Pan et al. 2017], it remains a challenge to segment noisy and partially missing depth images containing thin structures. Therefore, attempts to recover 3D thin structures, e.g., by lifting 2D curves from RGB images to 3D thin structures using depth images, fail on two counts: first, many incorrect edges are lifted, and unreliable or missing depth measurements prohibit correct lifting. Second, existing methods that use the color channel to upsample the pixel resolution of the depth channel, e.g., through joint filtering [Kopf et al. 2007; Richardt et al. 2012] or shading-based refinement [Wu et al. 2014], cannot recover raw thin structure depth measurements that are missing or inaccurate to start with. (iii) Previously used predetermined fusion data structures (e.g., a regular TSDF voxel grid), being oblivious of the object being scanned, are unsuited for thin scene structures. In this work, we show that central skeletons can be used as *adaptive* fusion primitives to encode connectivity and geometry, and therefore can represent thin structures more accurately and more compactly, which greatly simplifies structural completion of missing parts.

We therefore propose CurveFusion, a reconstruction approach for thin structures with a handheld commodity RGBD camera. Instead of using a pre-authored fusion primitive, such as a voxel grid, CurveFusion uses the curve skeleton as a new, data-dependent fusion primitive. CurveFusion first identifies and segments depth samples that come from thin-structures in each frame using both depth and color information. Algorithmically, to address the issue that each single raw depth frame is noisy and incomplete, we partition the input frame sequence into *bundles*, each being a group of a fixed number of consecutive frames. We further consolidate the segmented point samples from the mutually-aligned frames of the same bundle into a point set, called the *bundle difference set*. For each bundle we extract the central skeleton curve of this merged point set by computing its L_1 axis [Huang et al. 2013] and explicitly detect junctions, i.e. where three or more curve segments meet.

Given many such L_1 curve fragments along with identified junctions extracted from different bundles, we propose a novel curve alignment algorithm to fuse them into the final 3D skeleton curves with minimal drift. We retain the junctions through this fusion stage. The extracted 3D skeleton is then used to recover the final 3D surface of the scanned thin structure (see Figure 1).

We evaluated CurveFusion on a range of RGBD scans of real thin structured objects under various scanning setups and demonstrate high-quality reconstructions. We compare our method to competing image-based methods, which either require significantly more controlled imaging setups or start to degrade in complex examples even when assuming access to clean background subtracted inputs. Please note that state-of-the-art voxel-based fusion methods simply fail to produce any relevant output for most of the examples shown in this paper. In summary, we (i) develop the first method for performing high-quality 3D curve reconstruction using commodity RGBD cameras; (ii) propose a data-adaptive fusion approach that first discovers a suitable skeleton directly from the data and then use it for reconstruction; and (iii) present high-quality reconstructions of thin structures previously not possible from raw RGBD sequences.

2 RELATED WORK

Our work is related to depth-based reconstruction techniques as well as methods of reconstructing thin structures.

Depth based reconstruction. The many recent advances in 3D acquisition technology (e.g., structured light, LiDAR, and more recently commodity depth sensors) spawned an increasing number of related works on digitization of the physical world from depth measurements. Since many depth sensors output 3D points, earlier approaches propose to merge these measurements to produce an aligned set of points [Rusinkiewicz et al. 2002; Weise et al. 2009]. Such approaches, however, are limited to scanning objects and scenes of small spatial extent. To extend the scope of point-based methods, Henry et al. [2012] use surfel primitives, each of which stores a location, a surface orientation, a patch size, and color. More recently, Keller et al. [2013] fuse depth information at the level of points. In recent years, volumetric data structures holding a (truncated) signed distance field (TSDF) representation of scene surfaces [Curless and Levoy 1996] have become a common choice for scanning objects and indoor environments.

KinectFusion [Newcombe et al. 2011] is the first real-time dense volumetric scanning system that uses a regular TSDF voxel grid of a fixed size to fuse depth measurements. Follow-up works have used octrees [Zeng et al. 2013] or voxel hashing [Nießner et al. 2013] to efficiently scale TSDF-based scanning and fusion to larger scenes. In order to minimize camera drift, several methods fuse the incoming data in fragments first and then perform a global optimization [Choi et al. 2015; Zhou and Koltun 2013; Zhou et al. 2013]. Recently, the BundleFusion system [Dai et al. 2017] first uses sparse RGB features to achieve a coarse global alignment, and then refines it utilizing geometric and photometric measures. At the core of such approaches, however, is the volumetric data fusion step which assumes that the scanned environment consists of relatively large objects with *closed surfaces*, i.e., objects that have a well-defined inside and outside (at least locally). Thus, such methods are unsuitable for reconstructing thin structures. In contrast, we propose to fuse the sensor measurements at the level of underlying curve structures or skeletons, which are concurrently discovered from the data itself.

Some RGBD methods use the commonly higher pixel resolution of the RGB channel to upsample the depth channel resolution, e.g., by joint filtering [Kopf et al. 2007; Richardt et al. 2012], or via shading-based refinement on depth images [Wu et al. 2014] or TSDF volumes [Zollhöfer et al. 2015]. Unfortunately, these methods suffer from the ambiguity between geometry and color edges, and none of them recovers geometry information of thin structures that is already missing in raw depth images or TSDF volumes.

Reconstructing thin structures. Several methods [Aroudj et al. 2017; Savinov et al. 2016; Ummenhofer and Brox 2013] have recently been proposed to relax the *closed surface* assumption of the volumetric fusion techniques to enable reconstruction of *thin surfaces*. While these methods show impressive results, we are addressing a different class of scenes featuring structures that are fundamentally one dimensional and lack sufficient surface detail.

In the specific case of reconstructing objects with similar delicate structures, Li et al. [2010] introduce the deformable model *arterial snakes*. Their method, however, works on high-quality dense 3D scans whereas the input to our method comes from commodity depth sensors and thus has more noise and missing data. Huang et al. [2013] extract skeletons from point cloud data which can be used to fit generalized cylinders for reconstruction purposes [Yin et al. 2014]. However, as we show in our evaluations, due to alignment inaccuracies and drift, directly extracting skeletons from merged depth data of all the input frames does not yield compelling results.

An alternative approach to reconstructing thin structures is to use image input. Tabb et al. [2013] reconstruct thin structures from multiple image silhouettes by fusing information in a volumetric grid using a probabilistic approach (see Section 5 for a comparison). Li et al. [2018] propose a method to solve this problem by leveraging spatial curves generated from image edges, however the quality of the result declines under complex occlusions. Yücer et al. [2016b] present a method that explores local gradient information in captured dense light fields to segment out thin structures. In a follow-up work [Yücer et al. 2016a], they combine gradient information with photo-consistency measurements to compute a per-view depth map that can represent thin structures. These depth maps are aggregated using a voxel grid of fixed size, before voxel carving and Poisson surface reconstruction are applied [Kazhdan et al. 2006]. In contrast, we propose to directly use extracted 3D skeletons as a fusion primitive to successfully aggregate information from noisy depth and color images captured by commodity depth sensors.

Instead of matching standard point features, several multi-view stereo methods match higher order primitives such as lines [Hofer et al. 2014; Jain et al. 2010] or curves [Fabbri and Kimia 2010; Nurutdinova and Fitzgibbon 2015; Rao et al. 2012; Usumezbas et al. 2016; Xiao and Li 2005]. These approaches, however, produce a reconstruction in the form of individual line or curve segments which possibly suffer from noise and gaps. Several methods address this limitation by reconstructing continuous curve paths using different priors. Martin et al. [2014] reconstruct thin tubular structures such as cables from a dense set of images using physics-based simulation of rods to improve accuracy. This method assumes that 2D cable crossings can easily be recovered from the the images and disambiguated in 3D with an occupancy grid. In contrast, the typical objects we reconstruct often lack a surface, which would make such a grid very noisy. Delmas et al. [2015] reconstruct curvilinear navigating devices (e.g., guide wires, catheters) from two fluoroscopic views as a single continuous path. The more recent work of Liu et al. [2017] reconstructs objects that are composed of wires utilizing smoothness and simplicity priors. They assume images with relatively clean background (i.e., do not deal with unknown background subtraction) and cannot handle objects with many complex junctions as shown in Section 5.

3 OVERVIEW

Our goal is to reconstruct a coherent thin structure, denoted $\mathcal{T}$, using a handheld RGBD camera producing a streaming sequence of RGBD frames. Please refer to Figure 3 for illustration. Note that thin structures smaller than 2 mm in diameter are usually too thin to be scanned by a consumer-grade RGBD camera, such as Kinect V1, while tubular objects of 10 mm or larger in diameter can usually be scanned using the same camera and reconstructed in reasonable quality using a conventional pipeline, such as KinectFusion.

We precompute relative camera poses between the consecutive frames using the ORB-SLAM system [MurArtal and Tardós 2017] (in Section 5, we evaluate performance using other calibration methods). The output of our method is a *skeleton* defined as a network of curves

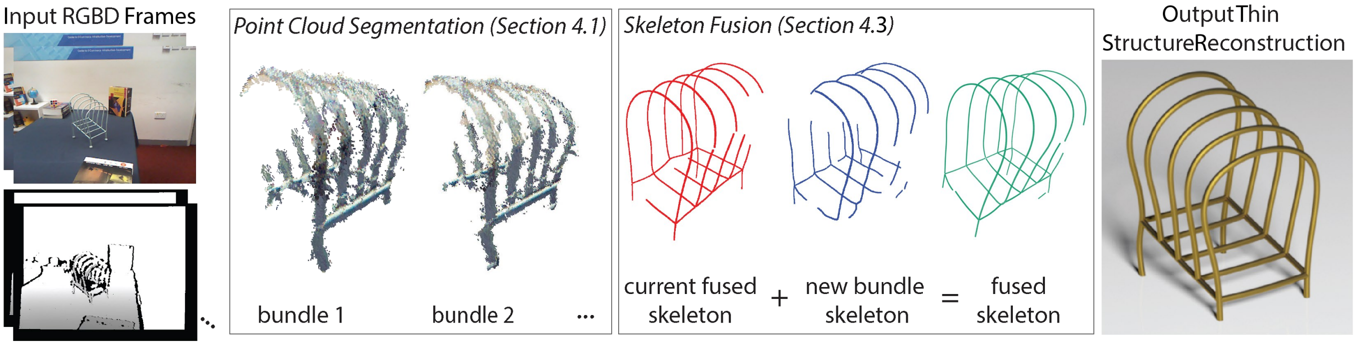
Input RGBD Frames
...
Point Cloud Segmentation (Section 4.1)
bundle 1
bundle 2
...
Skeleton Fusion (Section 4.3)
current fused skeleton
+
new bundle skeleton
=
fused skeleton
OutputThin StructureReconstruction

Fig. 3. Given a sequence of RGBD images of a thin structure, we organize the input into *bundles*, each consisting of 30 frames. For each bundle, our method first identifies the 3D point samples that belong to the thin structure (Section 4.1) and extracts the L_1 skeleton for each bundle. We then perform a novel skeleton fusion step to iteratively merge the new incoming bundle skeleton to the current partial skeleton (Section 4.3). The final fused skeleton provides a faithful reconstruction of the thin structure.

in 3D that are the central curves of the thin structure $\mathcal{T}$ along with a radius function.

There are two essential challenges in reconstructing accurate skeletons of thin structures from noisy RGBD measurements. First, the **data segmentation** issue: it is difficult to distinguish point samples belonging to the thin structure $\mathcal{T}$ from those belonging to the other scene parts (i.e., background). Specifically, the depth channel is noisy and unreliable, and it is non-trivial to differentiate thin structures from texture edges in the RGB image alone (see Figure 2b). Second, the **data consolidation** issue: in a single depth image, the part of the point cloud that represents the thin structure $\mathcal{T}$ is typically very noisy, sparse and incomplete, thus preventing reliable skeleton extraction. On the other hand, the attempt to extract the curve skeleton from the aligned depth scans of all the frames also fails (see Figure 2c) as the thin-structure gets smeared out in the accumulated point cloud due to error accumulation from camera drift.

To address the **data segmentation** issue, we separate the point samples belonging to the thin structures $\mathcal{T}$ from those belonging to large objects in the background by first running the traditional volumetric fusion with a regular TSDF voxel grid [Curless and Levoy 1996], henceforth VolumeFusion. (Note that VolumeFusion is given access to registration information obtained using ORB SLAM during camera localization.) The rationale is that the volumetric fusion methods perform quite well in reconstructing objects with relatively large extended surfaces, thus providing a useful reference for distinguishing the thin structure from the background objects. Then, we propose a simple and effective segmentation method based on topological operators that combines observations in the RGB images, raw depth maps, and depth renderings of the volumetric fusion result (Section 4.1). Specifically, a comparison of depth renderings of the fusion result with raw depth maps provides strong cues to remove the depth samples that belong to the background scene (i.e., extended surfaces). Verifying this cue with 2D image space curves in the RGB channel leads to reliable segmentation.

To address the **data consolidation** issue, we partition the entire input image frame sequence into *bundles*, each being a group of a fixed number of consecutive frames, and merge the segmented point samples from all the depth frames of each bundle into a point set, called a *bundle difference set*. Each bundle typically contains k merged frames (in our tests, $k = 30$) and provides a good balance between density and drift. The bundle difference sets are dense enough for skeleton extraction, and yet do not suffer from camera drift as severely as merging all the raw depth frames.

Once all the bundle difference sets are available, we perform L_1 skeleton extraction [Huang et al. 2013] in each bundle to obtain the skeleton for the part of $\mathcal{T}$ that is observable in the bundle, which we call the *bundle skeleton*. Note that such skeleton curves offer a natural and novel geometric representation as a fusion primitive for reconstructing thin structures. Finally, we propose a novel fusion procedure to topologically aggregate the bundle skeletons into a final 3D skeleton structure (Section 4.3) by aligning and merging skeleton segments.

4 CURVEFUSION METHOD

Given a sequence of RGBD images providing depth samples from the entire scene, first we segment the measurements belonging to the (unknown) thin structure $\mathcal{T}$.

Assumptions. We make the following assumptions about the thin structures being scanned: (i) Thin structures are sufficiently separated (~2-5cm depending on depth) from other large surfaces in the background such that the limited depth resolution of commodity sensors is sufficient to capture samples at different depths; (ii) Thin structures neither (fully) absorb RGBD camera's IR light (e.g., black surface color) nor are highly reflective, which would altogether preclude measuring their depth; (iii) Thin structures can be distinguished from the background in at least a few RGB frames in each bundle to help utilize image cues; and (iv) thin structures are in the range 2 mm to 10 mm in diameter. Here, the lower bound of 2 mm is imposed by the scanning limit of commodity hand-held RGBD scanners, while thin structures of diameter larger than 10 mm can be reconstructed with reasonable quality using the conventional KinectFusion pipeline.

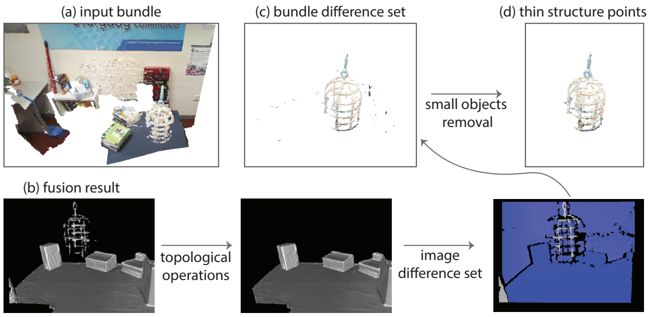

Fig. 4. To detect a thin structure $\mathcal{T}$, we first perform topological operations on the depth renderings of the VolumeFusion result and compare them with the input depth images. This produces a set of points in each depth image that potentially belong to $\mathcal{T}$, called the *image difference set*. By combining the image difference sets in a bundle, we obtain the *bundle difference set*. Once isolated ghost point clusters are removed, we obtain point samples that belong to $\mathcal{T}$ only.

4.1 Point Cloud Segmentation

Under the above assumptions, we observe that 3D points captured in each depth image can be classified as one of the following types: (i) belonging to thin structures to be detected and extracted; (ii) belonging to large background surfaces to be removed; and (iii) small and isolated erroneous 'ghost' points to be removed. Hence, we propose an automatic segmentation method that detects and removes points of the second and third types (see Figure 4).

Removing large surfaces. The result of applying VolumeFusion on the raw depth maps, henceforth called *fusion result* (see Figure 4b), is a volumetric TSDF model of the entire scene that represents large objects and extended surfaces well, but misses the thin structure $\mathcal{T}$ partially or entirely. This suggests that comparing each depth image against the aligned fusion result helps to identify and remove points belonging to large objects if they are present in both. However, depth points actually belonging to $\mathcal{T}$ may be erroneously removed here, if, against all odds, the thin structure was partially captured in the fusion result.

To address this, we first identify parts of $\mathcal{T}$ reconstructed in the fusion result before comparing it to raw depth maps. Specifically, we use ray-casting to produce a depth rendering of the fusion result from the associated camera pose of each input frame. Then, we detect edges in each of these depth renderings by analyzing depth discontinuity, and apply topological operations, namely erosion and dilation, to conservatively remove the corresponding thin parts that were captured in the VolumeFusion result. We set the distance used for both erosion and dilation to 2 cm. Difference of these resulting depth renderings from the corresponding original depth maps yields a point cloud called *image difference set* for each input frame as shown in Figure 4.

The image difference sets of all frames in the same bundle are aligned to the first frame using the camera poses provided by ORB SLAM. Thus, each bundle is associated with the camera parameters of its first frame. This aligned and merged point set is called the *bundle difference set*, from which large objects have been removed (see Figure 4c). We resample this merged point set, which is typically non-uniform and dense, using a 3D regular grid for more efficient skeleton extraction at later stages.

Removing 'ghost' points. As explained before, the bundle difference set may potentially contain unwanted small and isolated 'ghost' objects, i.e., thin and short point cloud segments that do not correspond to any physical 3D scene element. Such ghost objects often arise due to measurement noise from commodity RGBD cameras.

Our goal is to disambiguate these ghost objects from the actual thin structures in each bundle different set. Specifically, we first run the DBSCAN algorithm [Ester et al. 1996] to segment each bundle difference set into distinct point clusters. We then remove clusters smaller than an empirically determined size threshold (fewer than 10 points) and isolated floating clusters (>10 cm away from any other cluster). The remaining possible thin and long clusters are further removed if they are not *verified* by the RGB channel. Specifically, we first represent all the clusters in a voxel grid. Then we detect curve edges in each RGB frame of the bundle and lift points sampled on these edges to the voxel grid of the clusters. Here, the corresponding raw depth measurements is used for this lifting operation. Then any cluster for which less than 20% of its voxels receive a lifted curve sample is removed. These operations leave us with a set of points, for each bundle, that only belong to the thin structure, as illustrated in Figure 4.

We note that due to noise in image space curve edges and raw depth measurement, simply lifting all the detected RGB edges to 3D would produce an extremely noisy point set not suitable for extracting thin structures (see Figure 2). Nonetheless, the RGB image provides sufficient cues to validate candidate thin structure clusters produced by our proposed segmentation method.

4.2 Skeleton Extraction

Having identified the point clusters in each bundle that belong to the thin structure $\mathcal{T}$, we next extract the central skeleton of each cluster of the bundle by computing the L_1 medial axis of the cluster [Huang et al. 2013], which is called the *bundle skeleton*.

Since the L_1 axis method [Huang et al. 2013] was originally designed for objects (such as aircraft and human hands) that are quite different from our thin tubular structures, modifications were needed to make it suitable for our setting. In our implementation, we follow the basic framework of L_1 contraction with the initial radius set to 15 mm and the maximum radius set to 25 mm based on the typical width of the point cloud of the wire objects we expect to reconstruct. Figure 5(b) shows an L_1 contraction result of a point set (Figure 5(a)). The original L_1 method uses a rather sophisticated strategy to obtain skeleton branches. However, their method often fails in our setting, especially for wire objects with closely adjacent joints, as shown in Figure 5(c). We address this problem by identifying the junctions explicitly. After the convergence of contraction, each cluster of the remaining non-branch points is treated as a junction, as shown in Figure 5(d). Finally, the skeleton branches are connected by junction points. Empirically, this leads to better quality of L_1 skeleton curves for thin objects, as shown in Figure 5(e).

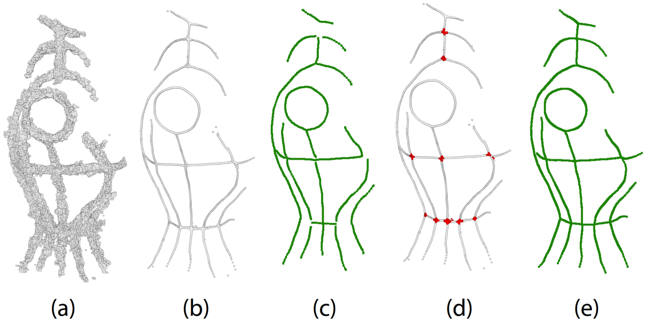

Fig. 5. Starting from an input point cloud (a), the original method of Huang et al. [2013] leads to spurious junctions (b, c). We explicitly detect junctions (d), which leads to better quality L$_1$ skeletal curves (e).

4.3 Skeleton Fusion

Our goal is to obtain the complete skeleton $\mathcal{K}$ of the thin structure $\mathcal{T}$ by fusing the bundle skeletons $\mathcal{S}_i$ of the bundles $\mathcal{B}_i$, $i = 1, 2, \ldots, n$. We achieve this goal by performing iterative skeleton fusion. We maintain a partial skeleton $\mathcal{K}_i$ obtained by fusing the bundle skeletons $\mathcal{S}_j$ of all the bundles $\mathcal{B}_j$, $j = 1, 2, \ldots, i$, processed thus far. Prior to this fusion process, we first bring each bundle skeleton to the reference frame of the first bundle using the camera parameters associated to each bundle. We represent each individual bundle skeleton as well as the partial skeleton $\mathcal{K}_i$ as a graph of connected curves in 3D – each edge is a curve segment and each node is a junction where a number of curve segments are joined. Initially, $\mathcal{K}_1$ is set to be $\mathcal{S}_1$.

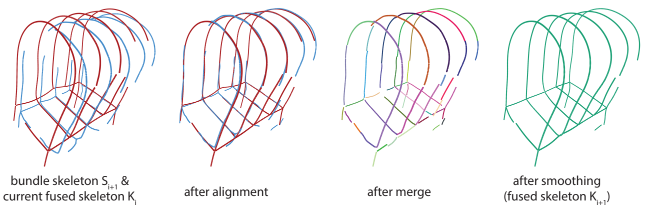
bundle skeleton S_{i+1} &
current fused skeleton K_i
after alignment
after merge
after smoothing
(fused skeleton K_{i+1})

Fig. 6. Given the current partial skeleton K_i (red) and the new bundle skeleton S_{i+1} (blue), we show the different steps of the fusion algorithm to generate K_{i+1} (green).

Given a partial skeleton $\mathcal{K}_i$, $i \geq 1$, we now describe the details of a single iteration of the skeleton fusion process where the bundle skeleton $\mathcal{S}_{i+1}$ of the next bundle B_{i+1} is fused with the current partial skeleton $\mathcal{K}_i$ to generate $\mathcal{K}_{i+1}$. As shown in Figure 6, each skeleton fusion step comprises three subsequent sub-tasks, namely (i) *alignment*; (ii) *merging*; and (iii) *smoothing*.

(i) Alignment. While registering all bundle skeletons to the reference frame of the first one brings them sufficiently closer, there still remain significant misalignments due to camera drift. To rectify this, we further bring $\mathcal{K}_i$ closer to $\mathcal{S}_{i+1}$ by using iterative closest point (ICP) [Besl and McKay 1992] to perform rigid alignment between densely and uniformly distributed sample points on the curves of $\mathcal{S}_{i+1}$ and $\mathcal{K}_i$. We do not consider the alignment of the junctions in this step, since the closest point pairs sampled on the curves provide enough constraints to determine the rigid motion.

(ii) Merging. Once $\mathcal{S}_{i+1}$ and $\mathcal{K}_i$ are rigidly aligned, we first detect overlapping curve segments, i.e., curves which are close to each other within a specified distance threshold (set to be 1 cm in our experiments). We merge the overlapping regions of such curve segments by weighted averaging, with the weight $i/(i+1)$ for $\mathcal{K}_i$ and $1/(i+1)$ for $\mathcal{S}_{i+1}$, while keeping the original curve segments for the non-overlapping areas. If a junction point exists in the overlapping region, we split the curve segment that crosses the junction and merge the resulting segments separately (see Figure 7).

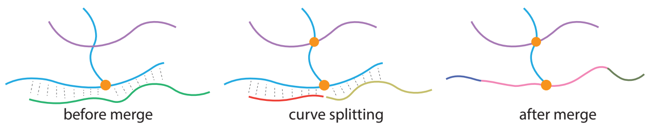

Fig. 7. During merging a segment is split when it crosses a junction in the merged region.

Once the curve segments are merged, we identify pairs of junctions of $\mathcal{S}_{i+1}$ and $\mathcal{K}_i$ that are close to each other within a specified distance threshold (set as 1 cm) and merge them with a similar weighted averaging. Any curve segment incident to one of the merged junctions now becomes incident to the new junction.

After merging, if any two curve segments intersect (within a proximity threshold) at an internal point, we create a new junction at this intersection and split each of the segments accordingly. Finally, we remove any curve segment shorter than a threshold (2 mm) to trim off possible spurious parts in reconstruction. The outcome of this merging step is a new network $\mathcal{K}_{i+1}$ of connected curves that inherits the junctions and curves from the skeletons $\mathcal{S}_{i+1}$ and $\mathcal{K}_i$, as well as the new junctions created by curve segment intersections.

(iii) Smoothing. The previous step can introduce artifacts in the form of jagged junctions produced by the weighted averaging of overlapping curve segments. To remove these, we perform skeleton smoothing via optimization-based curve fitting. Specifically, we represent each curve of $\mathcal{K}_{i+1}$ as a polyline whose vertex positions $\{v_0, \ldots, v_m\}$ are uniformly sampled along the curve to ensure a distance of 1 mm between consecutive vertices. The curve fitting optimization computes the new vertex positions by minimizing the following objective function:

$$\sum_{k=0}^{m} \|v_k - v_k^0\|^2 + \lambda \sum_{k=1}^{m-1} \|v_{k-1} - 2v_k + v_{k+1}\|^2, \qquad (1)$$

where v_k^0 denotes the position of vertex k before smoothing. The first term is a data fitting term which advocates for maintaining the original vertex positions, the second term enforces smoothness in the direction of consecutive polyline segments, and λ denotes the relative weighting between these terms ($\lambda = 60$ in our tests).

In order to smooth all the curves in the skeleton simultaneously, we treat each junction as a shared vertex for all of its incident curve

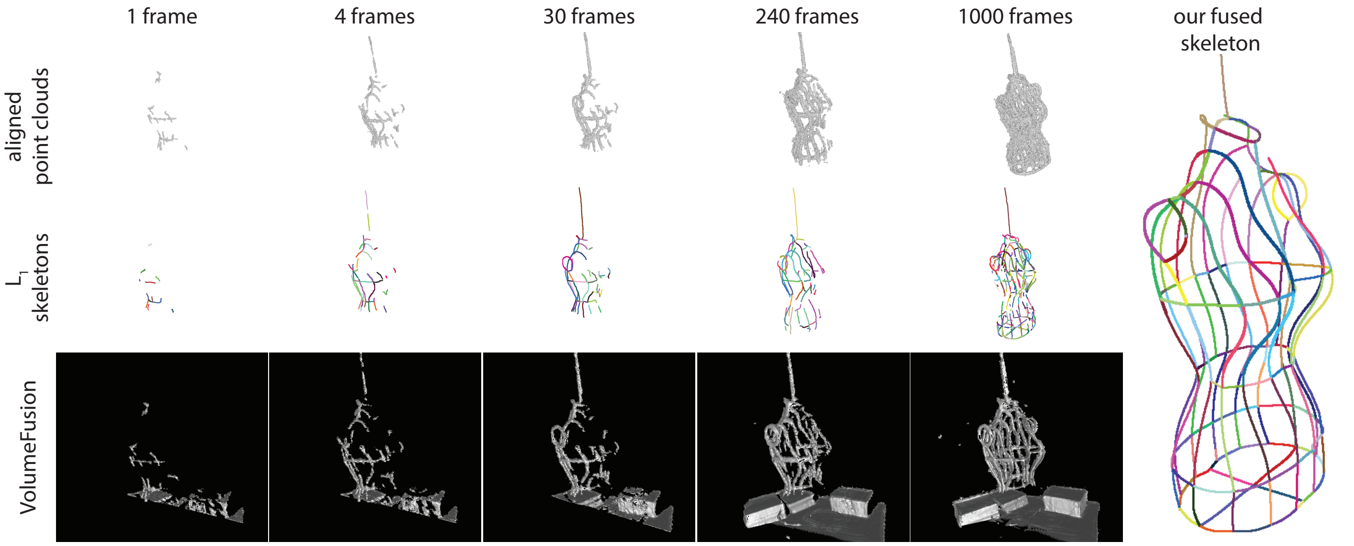

Fig. 8. For different choices of bundle size, we show the aligned point clouds and extracted L_1 skeletons. We observe that a bundle of 30 frames provides a good balance between model completeness and structural clarity. Our skeleton fusion algorithm merges per-bundle L_1 skeletons to yield a high quality reconstruction. VolumeFusion fails to produce satisfactory reconstruction from any accumulated point cloud, regardless of the number of frames used.

segments. Hence, we formulate and minimize a global objective function that sums up the energy given in Equation 1 for each of the individual curves in the network $\mathcal{K}_{i+1}$. This is a linear least squares problem that is solved efficiently by solving a linear system of equations.

Wire radius estimation. We assume that the thin structure $\mathcal{T}$ is made of tubular surfaces represented by skeleton curves with some constant radius. After the skeleton curve $\mathcal{K}$ of $\mathcal{T}$ is extracted as described in the previous step, we estimate the radius of $\mathcal{T}$ using its image in the RGB frames. Specifically, we sample a set of points from the skeleton $\mathcal{K}_i$ of each bundle and then project these points from 3D to the RGB images associated with the bundle. Note that edge extraction has been applied to each RGB frame to label the images of the wire object as a strip region. We then inspect a 16 × 16 box (in pixel) centered at the projection of each sample point. The box is accepted for further processing only if there is exactly one strip region in the box, and discarded otherwise. This ensures that boxes with spurious edges due to cluttered background, where radius estimation would be unreliable, are discarded. The width of the strip region is measured for each pixel on the central axis of the strip contained in the box, and then all these width estimates are averaged to produce the radius estimate from this box.

All accepted boxes (from all sample points on all bundle skeletons) produce a radius estimate in pixels, and these estimates are filtered and combined to produce the final radius measurement as follows. When a wire object is composed of multiple parts of different radii, as it is the case for some of the objects we processed, our method still produces a single radius value. In this case, the radius measurements from different boxes are clustered based on a histogram of the measurements and the most prominent value is chosen as the final radius measurement. The resulting measurement is a reasonable approximation for the majority of the parts having the same radius, even if radii differ across the object. Finally, the radius in pixels is converted into a metric 3D measurement.

5 RESULTS

We tested CurveFusion with a range of real world structures of varying complexity. All our real scenes were recorded with a Kinect V1 sensor. Since the accuracy of the factory calibration between RGB channel and depth channel (captured by two separate sensors) is insufficient for our purpose, we calibrate the two channels ourselves using a checkerboard visible in both IR and color. We show a wide range of examples in Figure 9. For each of these, we show an example RGB image from the original scanning sequence, as well as additional representative pictures to better convey the expected geometry. We refer to the accompanying video for close-ups.

Most of the models shown in this paper were reconstructed from an RGBD sequence of several hundred frames with a bundle size of 30 frames. Some complex modes took a longer sequence – for example, the cloth hanger shown in the teaser was scanned in 1,136 frames. Our examples have skeleton structures that contain from 14 to 336 junctions. Our algorithm successfully detects and classifies 95% of these junctions and provides a faithful 3D reconstruction.

Effect of bundle size. In Figure 8, we demonstrate how we determine the appropriate number of frames in a bundle. The first row shows the accumulated point clouds (only points identified to belong to the thin structure are merged) of different numbers of depth frames for a wire model. Clearly, when too few frames are grouped together, e.g. $k = 1$ or 4, the merged point cloud is too sparse and leads to a broken partial skeleton (see the second row of Figure 8). On the other hand, when too many frames are grouped together, e.g. $k = 240$ or 1000, the merged point cloud becomes more dense

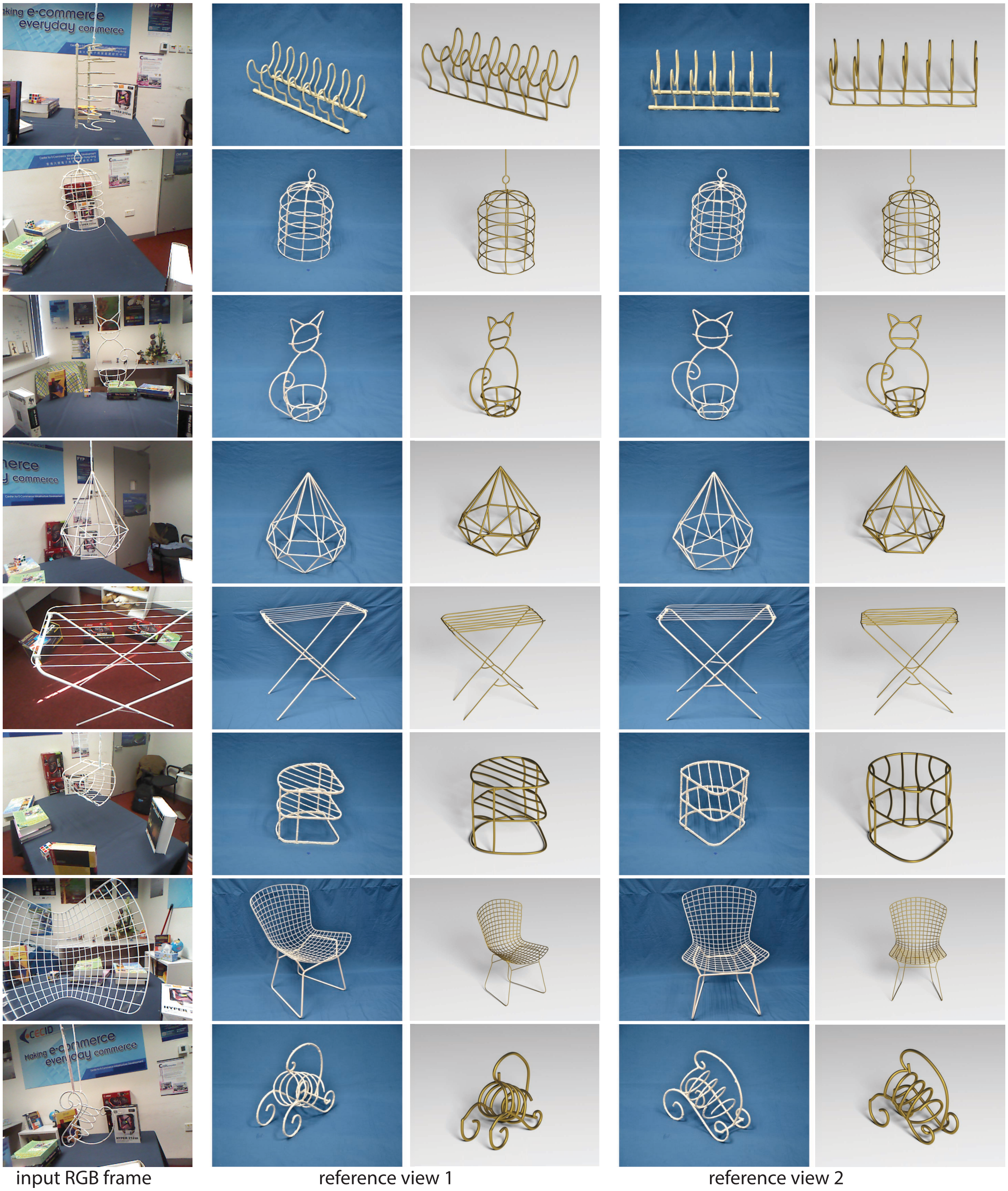

Fig. 9. For each example, we show one of the input RGB frames and our reconstruction from two reference views. We show images of the 3D models from the same reference view on a blue background to better convey their geometry (see also video).

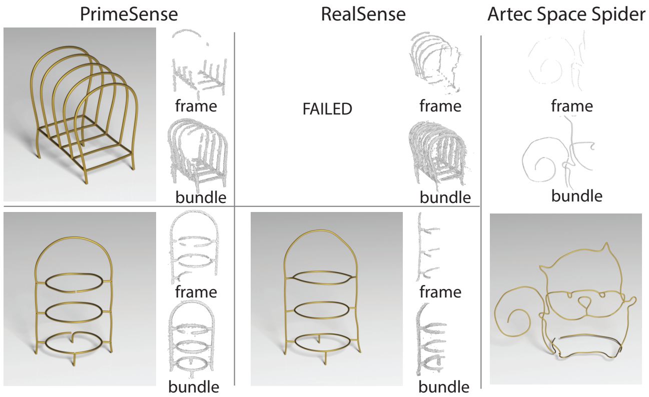

Fig. 10. While we can reconstruct both the breadholder (5 mm in diameter) and the rack (8 mm in diameter) models with PrimeSense, RealSense data is much more noisy and we can only use it to reconstruct the thicker rack model. Artec Spider Scan is a high end scanner and thus enables scanning model with wire diameter 1.5 mm. In addition to final reconstructions, we also provide a sample of a single depth frame and a bundle.

but too blurry (due to camera drift) for extracting reliable curve skeletons. Hence, we propose to group the input sequence of frames into bundles of k frames each in such a way that the merged point cloud of each bundle attains a balance between model completeness and structure clarity. The appropriate number of frames k in each bundle typically lies in a range from 10 to 80, depending on model variety and variations of scanning operation. Empirically, we find that $k = 30$ works well for most of the models we have tested. As a comparison, the third row of Figure 8 shows that VolumeFusion fails to produce satisfactory reconstruction from any accumulated point cloud, regardless of the number of frames used.

Effect of different sensors. In addition to Kinect V1 sensor used for the results presented, we also evaluated four other scanners: PrimeSense, RealSense SR300, Kinect V2, and Artec Space Spider. Since PrimeSense's working mechanism is similar to that of Kinect V1, it produces depth scanning of comparable quality. RealSense SR300 produces more noisy depth frames and hence are not suitable for reconstructing very thin structures with our method. However, it can be used for reconstructing thicker structures. Figure 10 shows the results of using these two sensors to scan two models, the breadholder (diameter=5 mm) and the rack (diameter=8 mm). Note that the breadholder cannot be reconstructed with RealSense. Finally, being a time-of-flight depth sensor, Kinect V2 yields very noisy measurements for thin structures and cannot be used for producing any reasonable reconstruction results. We have also tested Artec Space Spider, a high-end professional grade depth scanner, which is capable of scanning wires as thin as 1.5 mm in diameter. CurveFusion successfully reconstructs using such data as shown in Figure 10. In summary, as the accuracy and the depth resolution of the scanners increase, the thickness of the wire structures that can be successfully reconstructed from the corresponding depth scans decreases.

Effect of different camera calibration. In our pipeline, we used ORB-SLAM, which is an open-source SLAM system using both visual features in RGB frames and geometric features in depth frames. We also tested several other SLAM systems, namely BundleFusion, KinFu [PCL 2018], and PTAM [PTAM 2018]. Camera calibration function embedded in BundleFusion performed as well as ORB-SLAM (see Figure 11(a)). KinFu is an open source equivalent of KinectFusion that uses geometric features for camera calibration. Given enough background geometric objects, it is able to provide reasonable camera calibration (see Figure 11(b)). PTAM, on the other hand, uses only visual features and failed to provide sufficient quality camera calibration to be used by our method. In summary, we conclude it is desirable to employ both visual and geometric features for camera registration, as implemented in ORB-SLAM.

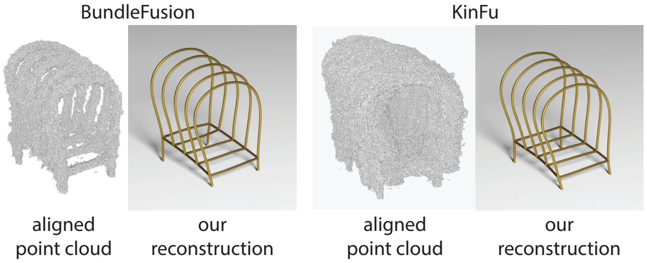

Fig. 11. We show two reconstructions of the breadholder with two different camera calibrations provided by BundleFusion and KinFu, respectively. We also show the point cloud in each case obtained by merging all the 399 input depth frames aligned with the given calibration.

Effect of wire thickness and color. Figure 12 shows the point clouds of four triangle wires of different diameters of 1 mm, 2 mm, 3 mm and 4 mm, respectively. We can see that thin structures of diameter smaller than 2 mm cannot be sufficiently scanned by the Kinect V1 for proper reconstruction. Figure 12 also shows the point clouds of another set of similar models in different colors and surface shininess, scanned with Kinect V1. We observe that the leftmost model in black color, though of the same diameter as the others, cannot be scanned using Kinect V1 due to its light-absorbing surface. Note that, the two models on the right have considerable surface shininess but can still be scanned well with Kinect V1.

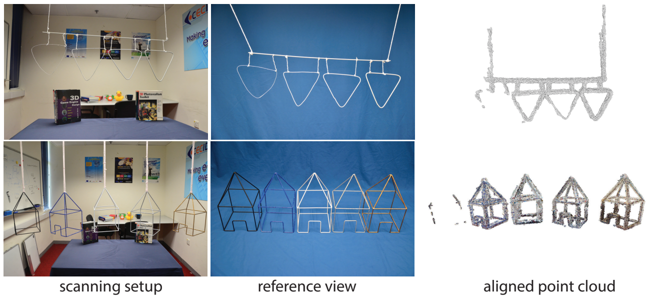

Fig. 12. We show the point clouds provided by a depth sensor for models of varying diameter (top row) and color/shininess (bottom row).

Effect of different RGB resolutions. When segmenting the point samples belonging to the thin structure, our method uses RGB frames to perform validation based on image edges. We evaluated the effect of the resolution of RGB frames on reconstruction quality. We considered the two settings of RGB resolution provided in Kinect V1: 480 × 640 and 1024 × 1280, while fixing the depth image resolution at 480 × 640, and observed no discernible difference in the output quality. This is mainly due to the fact that RGB information is used only for verifying the point cluster segmentation suggested by the depth data.

Effect of inter-wire distance. We tested CurveFusion using a set of parallel straight wires vertically arranged with decreasing spacings. This test served the purpose of determining the lower limit for the intra-wire spacing above which our method can still resolve different tubular structures correctly. Our tests show that the skeleton extraction step works robustly for spacings larger than 17 mm.

Comparisons. We compare our approach to Volumetric TSDF [Curless and Levoy 1996], KinectFusion [Newcombe et al. 2011], and BundleFusion [Dai et al. 2017] in Figure 2. As we discussed earlier, such approaches that use a fixed size volumetric grid as a fusion primitive either fail to reconstruct the thin structures entirely or provide only a noisy and partial reconstruction. In contrast, our approach detects and accumulates fusion primitives in the form of thin structure skeletons.

We also provide comparisons with the state-of-the-art visual-silhouette based method of Tabb et al. [2013]. The heavy studio setup with 40 calibrated cameras used in the original experiment has since been dismantled and has been replaced by a camera installed on a robot arm capable of taking photos of a thin structure from arbitrary viewpoints. With the assistance of the original authors, we tested this new setup together with the original visual silhouette based method for reconstructing the cloth hanger model and show the result in Figure 13. Our result, as can be seen in Figure 1, provides a smoother and more complete reconstruction while using a much more light-weight setup.

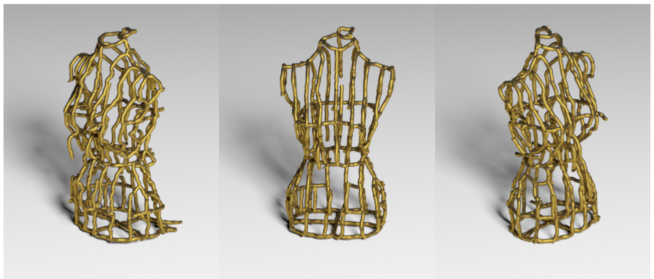

Fig. 13. We provide a comparison to the visual-silhouette based method of Tabb et al. [2013] (compare with Figure 1).

Finally, we perform a comparison to the recent image-based wire reconstruction method of Liu et al. [2017] (see Figure 14). Given 3 input images of a thin structure with known camera parameters (VisualSFM [Wu 2011] used to obtain the camera information), this method first detects a set of 2D curves in each image. Then, a large set of 3D candidate curves are generated using epipolar constraints between a pair of images. Candidates which do not receive sufficient support from the third view are then discarded. We observe that in presence of cluttered background and moderately dense wire structures, this image-based reconstruction and verification approach results in many spurious 3D curves which are hard to disambiguate in later stages of the algorithm. Our method, on the other hand, uses the image curves only as a verification cue. We identify potential thin structure point clusters by using the depth information and retain those that also receive image support. Furthermore, Liu et al. [2017] utilize smoothness and simplicity priors whereas the models we reconstruct are typically composed of many distinct wires that come together at sharp junction points.

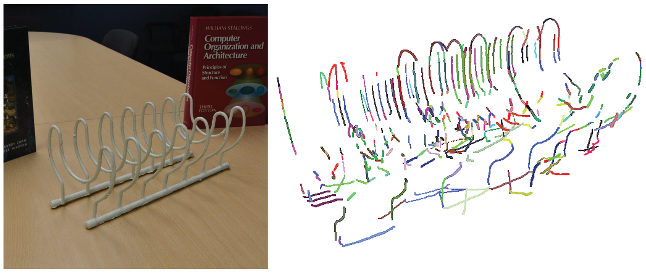

Fig. 14. For dense wire structures, assuming access to precise camera calibration and clean background, the image-based reconstruction method of Liu et al. [2017] results in many spurious 3D curves leading to significant time complexity as it requires to approximate the mTSP problem.

Performance. We measure the execution time of different stages of our method on a machine with Intel(R) Core(TM) i7-6820HK CPU @ 2.70GHz, 16GB memory. While the point cloud segmentation takes about 8 seconds for each bundle, L_1 skeleton extraction takes about 12 seconds and remains as a bottleneck. Finally, each iteration of skeleton fusion takes about 1-3 seconds. Please refer to Table 1 for more information.

Limitations. Our method may fail when the assumptions stated in Section 4 are violated in the input. If a thin structure, or a part of it, is close to a wall or lies on a table (i.e. a resting surface), then in general it cannot be scanned as an object distinct from the resting surface due to the limited depth resolution of the depth sensor. Hence, it will be missing from the reconstruction. As another limitation, our method implicitly assumes the RGBD images capture traces of points arising from thin structures, even if partial and noisy. In case of very thin structures below 2 mm in diameter (e.g., threads or very thin wires), commodity RGBD scanners are unlikely to record any evidence in the depth channel, and hence, CurveFusion will fail to reconstruct such thin structures. We note that this minimum diameter value can vary depending on the accuracy and the resolution of the sensor as discussed in Section 5. For objects with junctions having high valence, our junction merging step can produce imperfect results. For example, a junction of valence 6 may instead be interpreted as two separate junctions of valence 3 each and joined together. The challenge here is that both the depth and color channels are too unreliable to sufficiently disambiguate such

Table 1. Statistics and timings corresponding to the results presented in the paper.

	dish rack	bird cage	cat	decoration	large hanger	rack	chair	wine rack
number of frames	684	227	165	330	261	180	428	656
number of junctions	14	50	22	14	30	32	336	12
thickness of wire	5 mm; 9 mm	4.5 mm	4.5 mm; 6 mm	5.5 mm	4 mm; 12 mm	5.5 mm; 11 mm	4.5 mm; 12 mm	6 mm
estimated thickness of wire	5.4 mm	4.8 mm	5.6 mm	5.1 mm	4.5 mm	5.8 mm	4.5 mm	5.4 mm
reconstruction time (s) (without L1 extraction)	178	89	37	64	86	55	111	142

situations. A scanner with input of higher accuracy may partially address this issue. Finally, our radius estimation assumes that a thin structure has a constant radius for all its parts. So a better method needs to be devised for radius estimation when reconstructing a thin structure of varying radius.

6 CONCLUSION

We presented CurveFusion as the first algorithm to produce high quality 3D reconstruction of thin filament-like structures from commodity RGBD sequences. We demonstrated that existing state-of-the-art fusion approaches that align and integrate noisy measurements over fixed primitives (e.g., voxels) are unsuited to this task. Instead, as our key contribution, we described how to first reliably extract underlying thin-structured object skeletons from raw RGBD sequences, and then perform surface reconstruction using a data-dependent fusion primitive. We presented several high-quality reconstructions of complex thin-structure objects using our method.

Several future research directions remain to be explored:

Towards a real-time system. We would like to speed up our method to work in real-time. Given the current breakdown of timings, the main bottleneck is in detecting L_1 axes. One possible direction is to use data-driven learning approaches that infer local properties directly from point sets.

Hybrid structures. In the current formulation, CurveFusion handles thin structures only. However, many real world objects have a mixture of thin and extended surfaces. In such cases, it would be interesting to design a hybrid approach where voxels are used to recover the surfaces, and extracted skeletons to recover the thin structures (see Figure 15). The challenge is to automatically assign a fusion method to each part.

Different thickness. In this work we have assumed that a thin structure has the same radius in all of its parts. We will study how to extend our method to reconstructing thin structures that consist of parts of different radii or have a varying radius, possibly by performing more accurate segmentation of thin structures in the RGB images.

Dynamic structures. Finally, many thin structures, being lightweight, are easily affected by surrounding movements, e.g., plants or hanging lights swaying under wind effects. We would like to extend our method to also capture such dynamic structures borrowing ideas from DynamicFusion [Newcombe et al. 2015] and variations designed for a comparable surface setting.

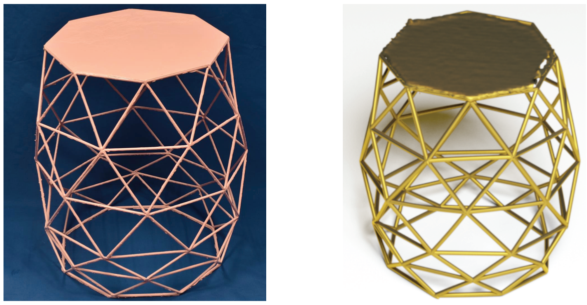

Fig. 15. We would like to consider a hybrid fusion algorithm that handles both thin structures and extended surfaces. Right shows a first result where the we manually combined KinectFusion output and CurveFusion results.

ACKNOWLEDGMENTS

We thank our reviewers for their invaluable comments. We thank Hui Huang, Amy Tabb and Zheng Wang for their great help with the testing and validation of our work. We also thank Jiatao Gu, Cheng Lin, Zhiming Cui, Runnan Chen, Maria Lam, Paul Guerrero, Elizabeth Schildge for their help. This work was partially funded by the ERC Starting Grant SmartGeometry (StG-2013-335373), the Research Grant Council of Hong Kong (GRF 17210718), a Google Faculty award, a UCL visiting student program, and gifts from Adobe.

REFERENCES

Samir Aroudj, Patrick Seemann, Fabian Langguth, Stefan Guthe, and Michael Goesele. 2017. Visibility-Consistent Thin Surface Reconstruction Using Multi-Scale Kernels. *ACM SIGGRAPH Asia* 36, 6 (2017), 187:1–187:13.

Paul J. Besl and Neil D. McKay. 1992. A Method for Registration of 3-D Shapes. *IEEE Trans. Pattern Anal. Mach. Intell.* 14, 2 (Feb. 1992), 239–256. https://doi.org/10.1109/34.121791

Y-P Cao, T Ju, J Xu, and S-M Hu. 2017. Extracting Sharp Features from RGB-D Images. In *Computer Graphics Forum*, Vol. 36. Wiley Online Library, 138–152.

Sungjoon Choi, Q. Y. Zhou, and V. Koltun. 2015. Robust reconstruction of indoor scenes. In *IEEE CVPR*. 5556–5565. https://doi.org/10.1109/CVPR.2015.7299195

Brian Curless and Marc Levoy. 1996. A Volumetric Method for Building Complex Models from Range Images. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '96)*. ACM, New York, NY, USA, 303–312. https://doi.org/10.1145/237170.237269

Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. 2017. BundleFusion: Real-Time Globally Consistent 3D Reconstruction Using On-the-Fly Surface Reintegration. *ACM TOG* 36, 3, Article 24 (May 2017), 18 pages. https://doi.org/10.1145/3054739

Charlotte Delmas, Marie-Odile Berger, Erwan Kerrien, Cyril Riddell, Yves Trousset, René Anxionnat, and Serge Bracard. 2015. Three-dimensional curvilinear device reconstruction from two fluoroscopic views. In *SPIE, Medical Imaging 2015: Image-Guided Procedures, Robotic Interventions, and Modeling*, Vol. 9415. San Diego, CA, France, 94150F. https://doi.org/10.1117/12.2081885

Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A Density-based Algorithm for Discovering Clusters a Density-based Algorithm for Discovering

Clusters in Large Spatial Databases with Noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96)*. AAAI Press, 226–231. http://dl.acm.org/citation.cfm?id=3001460.3001507

Ricardo Fabbri and Benjamin Kimia. 2010. 3D curve sketch: Flexible curve-based stereo reconstruction and calibration. In *IEEE CVPR*. 1538–1545. https://doi.org/10.1109/CVPR.2010.5539787

Huazhu Fu, Dong Xu, and Stephen Lin. 2017. Object-based multiple foreground segmentation in RGBD video. *IEEE Transactions on Image Processing* 26, 3 (2017), 1418–1427.

Peter Henry, Michael Krainin, Evan Herbst, Xiaofeng Ren, and Dieter Fox. 2012. RGB-D Mapping: Using Kinect-style Depth Cameras for Dense 3D Modeling of Indoor Environments. *Int. J. Rob. Res.* 31, 5 (April 2012), 647–663. https://doi.org/10.1177/0278364911434148

Manuel Hofer, Michael Maurer, and Horst Bischof. 2014. Improving Sparse 3D Models for Man-Made Environments Using Line-Based 3D Reconstruction. In *International Conference on 3D Vision (3DV)*.

Hui Huang, Shihao Wu, Daniel Cohen-Or, Minglun Gong, Hao Zhang, Guiqing Li, and Baoquan Chen. 2013. L1-medial Skeleton of Point Cloud. In *ACM SIGGRAPH*. ACM, New York, NY, USA, Article 65, 8 pages. https://doi.org/10.1145/2461912.2461913

Arjun Jain, Christian Kurz, Thorsten Thormählen, and Hans-Peter Seidel. 2010. Exploiting Global Connectivity Constraints for Reconstruction of 3D Line Segment from Images. In *IEEE CVPR*. San Francisco, CA.

Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. 2006. Poisson Surface Reconstruction. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing (SGP '06)*. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 61–70. http://dl.acm.org/citation.cfm?id=1281957.1281965

M. Keller, D. Lefloch, M. Lambers, S. Izadi, T. Weyrich, and A. Kolb. 2013. Real-Time 3D Reconstruction in Dynamic Scenes Using Point-Based Fusion. In *2013 International Conference on 3D Vision - 3DV 2013*. 1–8. https://doi.org/10.1109/3DV.2013.9

Johannes Kopf, Michael F. Cohen, Dani Lischinski, and Matt Uyttendaele. 2007. Joint Bilateral Upsampling. *ACM Trans. Graph.* 26, 3, Article 96 (July 2007).

Guo Li, Ligang Liu, Hanlin Zheng, and Niloy J. Mitra. 2010. Analysis, Reconstruction and Manipulation using Arterial Snakes. In *ACM SIGGRAPH Asia*, Vol. 29. Article 152, 10 pages.

Shiwei Li, Yao Yao, Tian Fang, and Long Quan. 2018. Reconstructing Thin Structures of Manifold Surfaces by Integrating Spatial Curves. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2887–2896.

Lingjie Liu, Duygu Ceylan, Cheng Lin, Wenping Wang, and Niloy J. Mitra. 2017. Image-based Reconstruction of Wire Art. *ACM SIGGRAPH* 36, 4, Article 63 (July 2017), 11 pages. https://doi.org/10.1145/3072959.3073682

Tobias Martin, Juan Montes, Jean-Charles Bazin, and Tiberiu Popa. 2014. Topology-aware Reconstruction of Thin Tubular Structures. In *SIGGRAPH Asia 2014 Technical Briefs (SA '14)*. ACM, New York, NY, USA, Article 12, 4 pages. https://doi.org/10.1145/2669024.2669035

R. MurArtal and J. D. Tardós. 2017. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras. *IEEE Transactions on Robotics* 33, 5 (Oct 2017), 1255–1262. https://doi.org/10.1109/TRO.2017.2705103

R. A. Newcombe, D. Fox, and S. M. Seitz. 2015. DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time. In *IEEE CVPR*. 343–352.

Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J. Davison, Pushmeet Kohli, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. 2011. KinectFusion: Real-time Dense Surface Mapping and Tracking. In *IEEE ISMAR (ISMAR '11)*. IEEE Computer Society, Washington, DC, USA, 127–136. https://doi.org/10.1109/ISMAR.2011.6092378

Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Marc Stamminger. 2013. Real-time 3D Reconstruction at Scale Using Voxel Hashing. *ACM SIGGRAPH Asia* 32, 6, Article 169 (Nov. 2013), 11 pages. https://doi.org/10.1145/2508363.2508374

Irina Nurutdinova and Andrew Fitzgibbon. 2015. Towards Pointless Structure from Motion: 3D Reconstruction and Camera Parameters from General 3D Curves. In *IEEE ICCV*. IEEE Computer Society, Washington, DC, USA, 2363–2371. https://doi.org/10.1109/ICCV.2015.272

Xiao Pan, Yuanfeng Zhou, Feng Li, and Caiming Zhang. 2017. Superpixels of RGB-D images for indoor scenes based on weighted geodesic driven metric. *IEEE Transactions on Visualization and Computer Graphics* 23, 10 (2017), 2342–2356.

Point Cloud Library PCL. 2018. Kinfu. https://github.com/PointCloudLibrary/pcl/tree/master/gpu/kinfu. (2018).

Oxford PTAM. 2018. PTAM-GPL. https://github.com/Oxford-PTAM/PTAM-GPL. (2018).

Dushyant Rao, Soon-Jo Chung, and Seth Hutchinson. 2012. CurveSLAM: An approach for vision-based navigation without point features. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 4198–4204. https://doi.org/10.1109/IROS.2012.6385764

Christian Richardt, Carsten Stoll, Neil A. Dodgson, Hans-Peter Seidel, and Christian Theobalt. 2012. Coherent Spatiotemporal Filtering, Upsampling and Rendering of RGBZ Videos. *CGF (Proc. of EUROGRAPHICS)* 31, 2 (May 2012). https://doi.org/10.1111/j.1467-8659.2012.03003.x

Szymon Rusinkiewicz, Olaf Hall-Holt, and Marc Levoy. 2002. Real-time 3D Model Acquisition. *ACM SIGGRAPH* 21, 3 (July 2002), 438–446. https://doi.org/10.1145/566654.566600

Nikolay Savinov, Christian Hane, Lubor Ladicky, and Marc Pollefeys. 2016. Semantic 3D Reconstruction With Continuous Regularization and Ray Potentials Using a Visibility Consistency Constraint. In *IEEE CVPR*.

A. Tabb. 2013. Shape from Silhouette Probability Maps: Reconstruction of Thin Objects in the Presence of Silhouette Extraction and Calibration Error. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*. 161–168. https://doi.org/10.1109/CVPR.2013.28

Alex Teichman, Stephen Miller, and Sebastian Thrun. 2013. Unsupervised Intrinsic Calibration of Depth Sensors via SLAM.. In *Robotics: Science and Systems*, Vol. 248. 3.

B. Ummenhofer and T. Brox. 2013. Point-Based 3D Reconstruction of Thin Objects. In *IEEE ICCV*. 969–976. https://doi.org/10.1109/ICCV.2013.124

Anil Usumezbas, Ricardo Fabbri, and Benjamin B. Kimia. 2016. From Multi-view Image Curves to 3D Drawings. In *ECCV*. 70–87. https://doi.org/10.1007/978-3-319-46493-0_5

T. Weise, T. Wismer, B. Leibe, and L. Van Gool. 2009. In-hand scanning with online loop closure. In *IEEE ICCV Workshops*. 1630–1637.

Changchang Wu. 2011. VisualSFM: A Visual Structure from Motion System. (2011). http://ccwu.me/vsfm/

Chenglei Wu, Michael Zollhöfer, Matthias Nießner, Marc Stamminger, Shahram Izadi, and Christian Theobalt. 2014. Real-time Shading-based Refinement for Consumer Depth Cameras. *ACM Trans. Graph.* 33, 6, Article 200 (Nov. 2014), 10 pages. https://doi.org/10.1145/2661229.2661232

Yi Jun Xiao and Youfu Li. 2005. Optimized stereo reconstruction of free-form space curves based on a nonuniform rational B-spline model. *J. Opt. Soc. Am. A* 22, 9 (Sep 2005), 1746–1762. https://doi.org/10.1364/JOSAA.22.001746

Kangxue Yin, Hui Huang, Hao Zhang, Minglun Gong, Daniel Cohen-Or, and Baoquan Chen. 2014. Morfit: Interactive Surface Reconstruction from Incomplete Point Clouds with Curve-driven Topology and Geometry Control. In *ACM SIGGRAPH Asia*. ACM, New York, NY, USA, Article 202, 12 pages. https://doi.org/10.1145/2661229.2661241

K. Yücer, C. Kim, A. Sorkine-Hornung, and O. Sorkine-Hornung. 2016a. Depth from Gradients in Dense Light Fields for Object Reconstruction. In *2016 Fourth International Conference on 3D Vision (3DV)*. 249–257. https://doi.org/10.1109/3DV.2016.33

Kaan Yücer, Alexander Sorkine-Hornung, Oliver Wang, and Olga Sorkine-Hornung. 2016b. Efficient 3D Object Segmentation from Densely Sampled Light Fields with Applications to 3D Reconstruction. *ACM TOG* 35, 3, Article 22 (March 2016), 15 pages. https://doi.org/10.1145/2876504

Ming Zeng, Fukai Zhao, Jiaxiang Zheng, and Xinguo Liu. 2013. Octree-based Fusion for Realtime 3D Reconstruction. *Graph. Models* 75, 3 (May 2013), 126–136.

Qian-Yi Zhou and Vladlen Koltun. 2013. Dense Scene Reconstruction with Points of Interest. *ACM SIGGRAPH* 32, 4, Article 112 (July 2013), 8 pages. https://doi.org/10.1145/2461912.2461919

Q. Y. Zhou, S. Miller, and V. Koltun. 2013. Elastic Fragments for Dense Scene Reconstruction. In *IEEE ICCV*. 473–480. https://doi.org/10.1109/ICCV.2013.65

Michael Zollhöfer, Angela Dai, Matthias Innmann, Chenglei Wu, Marc Stamminger, Christian Theobalt, and Matthias Nießner. 2015. Shading-based Refinement on Volumetric Signed Distance Functions. *ACM Trans. Graph (Proc. SIGGRAPH)* 34, 4, Article 96 (July 2015), 14 pages. https://doi.org/10.1145/2766887