Exploiting Object Similarity in 3D Reconstruction

Chen Zhou[1,2] Fatma Güney[3] Yizhou Wang[1,2] Andreas Geiger[3]
[1]Nat'l Engineering Laboratory for Video Technology
[2]Cooperative Medianet Innovation Center, Peking University, China
[3]MPI for Intelligent Systems Tübingen
{zhouch,yizhou.wang}@pku.edu.cn {fatma.guney,andreas.geiger}@tue.mpg.de

Abstract

Despite recent progress, reconstructing outdoor scenes in 3D from movable platforms remains a highly difficult endeavour. Challenges include low frame rates, occlusions, large distortions and difficult lighting conditions. In this paper, we leverage the fact that the larger the reconstructed area, the more likely objects of similar type and shape will occur in the scene. This is particularly true for outdoor scenes where buildings and vehicles often suffer from missing texture or reflections, but share similarity in 3D shape. We take advantage of this shape similarity by localizing objects using detectors and jointly reconstructing them while learning a volumetric model of their shape. This allows us to reduce noise while completing missing surfaces as objects of similar shape benefit from all observations for the respective category. We evaluate our approach with respect to LIDAR ground truth on a novel challenging suburban dataset and show its advantages over the state-of-the-art.

1. Introduction

3D reconstruction is one of the fundamental problems in computer vision and has consequently received a lot of attention over the last decades. Today's hardware capabilities allow for robust structure-from-motion pipelines capable of reconstructing sparse 3D city models from millions of internet images [10] or dense models from video sequences in real time [42]. With the advent of the Kinect sensor, depth information has become available indoors and approaches based on volumetric fusion [31] or mesh optimization [46] are able to produce reconstructions with fine details given a large number of RGB-D observations.

However, obtaining accurate reconstructions outdoors and in less constrained environments observed during urban driving [13, 33] remains highly challenging: RGB-D sensors (e.g., Kinect) can't be employed outdoors due to their very limited range and expensive LIDAR-based mobile solutions result in relatively sparse 3D point clouds. Besides, lighting conditions are difficult, occlusions are omnipresent and many objects are only visible in a few frames as illustrated in Fig. 1. In this work, we address these difficulties by taking advantage of a fact which hitherto has been largely ignored: The larger the reconstructed area, the more likely objects of similar type and shape (e.g., buildings or vehicles) will occur in the scene. Furthermore, man-made environments are designed to be visually pleasing. Therefore, reoccuring structures like buildings often exhibit similar shapes in local neighborings.

Inspired by the impressive capability of toddlers to learn about objects from very few observations [3], we ask the following question: Can we exploit recurring objects of similar 3D shapes for jointly learning a shape model while reconstructing the scene? Such a system would be useful in many ways: First, we can expect increased completeness of the 3D reconstruction by regularizing across shapes using the principle of parsimony (i.e., by limiting the number of models and parameters). Second, we obtain a decomposition of a scene into its constituent objects in terms of 3D bounding boxes and segments. And finally, other tasks such as recognition or synthesis could benefit from the learned 3D shape models as well.

Our approach is summarized as follows: We first obtain an initial 3D reconstruction by structure-from-motion and volumetric fusion of disparity maps using a memory efficient representation based on voxel hashing [31]. Next, we train several generic 3D object detectors by extending exemplar SVMs [29] to truncated signed distance functions (TSDF) in 3D. Given the 3D box proposals from these detectors, we formulate a discrete-continuous optimization problem which we solve using block coordinate descent. More specifically, we minimize the difference between the TSDF values of the initial reconstruction and the predicted TSDF values by jointly assigning each proposal to a volumetric 3D shape model, optimizing for the pose of the 3D proposals and the latent shape variables, and finding the shape model parameters. Furthermore, we contribute

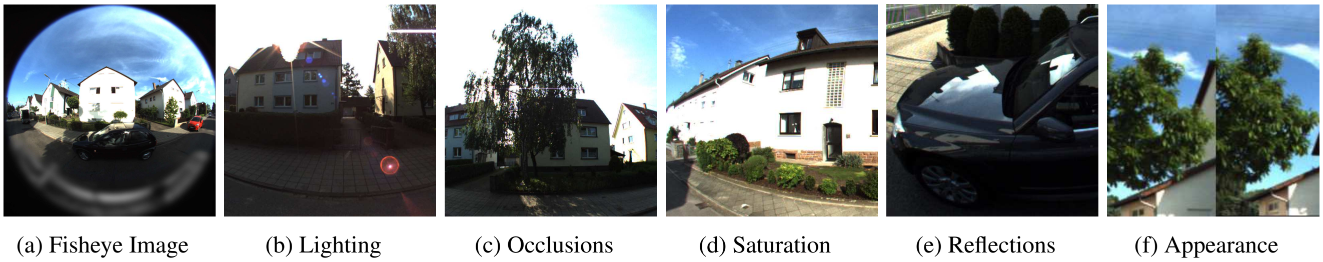

(a) Fisheye Image (b) Lighting (c) Occlusions (d) Saturation (e) Reflections (f) Appearance

Figure 1: **Challenges of 3D Reconstruction from Movable Platforms.** We have collected a novel suburban dataset for omnidirectional 3D reconstruction from fisheye images (a). Our dataset has been captured under normal daylight conditions and poses a variety of challenges to current reconstruction pipelines, including uncontrolled light conditions (b), occlusions (c), sensor saturation at bright surfaces (d), reflecting surfaces (e), and large appearance changes between successive frames when driving at regular speeds (f). We propose to overcome these challenges by regularizing across objects of similar shape.

a novel multi-view reconstruction dataset recorded from a moving platform on which we evaluate the proposed approach with respect to several state-of-the-art baselines. We make our code, dataset and supplementary material available on our project website[1].

2. Related Work

Existing works on multi-view 3D reconstruction can be roughly grouped into three categories according to the employed representation: Point-based approaches [12, 37] which produce a set of (oriented) 3D points from which a mesh is extracted in a second step using, e.g., Poisson surface reconstruction [23]. Methods, which directly optimize the vertices and faces of a textured 3D mesh [7,41] (or geometric primitives [27]) in order to minimize a photo-consistency measure. And finally, volumetric techniques [30, 39] which represent the surface implicitly as the zero-crossing of a distance function or using a probability map defined at regular voxel locations in 3D. In this paper, we follow the third line of methods as volumetric representations are flexible in terms of surface topology and allow for combining different shapes via volumetric fusion [6].

To overcome outliers in range maps obtained from stereo vision, local regularization is typically employed. This leads to smoother reconstructions by penalizing the perimeter of level sets [43] or encouraging local planarity [18]. For some scenarios, even stronger assumptions can be leveraged, such as piecewise planarity [13] or a Manhattan world [11]. The model proposed in this paper can also be seen as a prior on the reconstruction, but taking a different viewpoint: Instead of encouraging *smoothness* of surfaces, we encourage *shape similarity* across different instances of the same object category. Our objective integrates observations from several different instances which allows our model to fill-in occluded parts or textureless regions. In contrast to models assuming piecewise planarity or a Manhattan world, our approach also applies to non-planar object classes such cars. In this sense, our ideas are related to the depth super-resolution method of Hornacek et al. [21]. However, while they tackle single depth images using a patched-base representation, we model complete 3D scenes at the object level.

Lately, models integrating appearance information into the reconstruction problem have gained popularity [9, 32]. Häne et al. [17] leverage the fact that semantics and surface orientation are mutually dependent, e.g., the surface normal of the ground is more likely to face upwards than downwards. This knowledge helps in particular for object classes with a dominant orientation (e.g., ground) but has less advantages for classes with a more uniform normal distribution (e.g., building, car). Kundu et al. [24] directly constrain the range of possible depth values by conditioning ray potentials on the semantic class. To overcome shape ambiguities, object knowledge has been leveraged by Güney et al. [15] for stereo matching and by Häne et al. [16] for volumetric reconstruction. Prisacariu et al. [34,35] have demonstrated impressive results by leveraging GPLVMs for learning a non-linear TSDF embedding of the object shape for segmentation and reconstruction. While those approaches typically rely on pre-trained appearance models of known object classes, semantic classifiers or a dataset of 3D CAD models, our approach learns these models "on the fly" while reconstructing the scene. Apart from a generic object detector, no semantic annotation or 3D models are required. Furthermore, most of the approaches for object-based reconstruction focus on a single object while our method handles several objects at the same time.

Related to our approach are also a number of techniques which consider joint image alignment and segmentation or appearance estimation [5] with applications in face recognition [8] and medical imaging [40] . Our approach shares similarity with these methods in terms of estimating an instance specific transformation jointly with a low-dimensional representation of the object. However, our focus is on 3D reconstruction rather than 2D segmentation and we do not assume that objects are presented in a stereotypical pose. Instead, our method localizes objects accord-

[1] http://www.cvlibs.net/projects/similarity_reconstruction

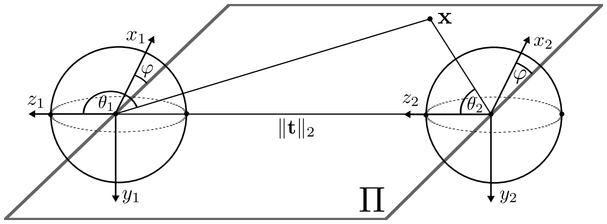


Figure 2: **Spherical Rectification.** Given two fisheye cameras (x_1, y_1, z_1) and (x_2, y_2, z_2), we rectify the images yielding an angular representation (θ, φ) where pixels in the same row (i.e., same φ) are located on an epipolar plane Π.

ing to a discriminatively trained object proposal generator. Furthermore, our model handles missing information (e.g., unobserved voxels) and is able to deal with a broad range of shapes by clustering them into different model components.

Unfortunately, existing datasets such as KITTI [14] or Google Street View [2] provide only a very limited field of view [14] or sampling rate [2]. For evaluating our approach, we therefore recorded a novel urban dataset with 3D ground truth. Our setup comprises a combination of front-facing perspective cameras, side-facing Fisheye cameras, a Velodyne 3D scanner and a SICK pushbroom scanner, capturing images and laser scans at roughly 10 fps.

3. Method

In this work, we are interested in volumetric 3D reconstruction leveraging the fact that 3D objects with similar shapes (such as cars or buildings) appear frequently in large environments. In particular, we aim at jointly clustering objects into categories according to their 3D shape, learning a 3D shape model for each of these categories and completing the 3D reconstruction by filling-in missing information via restriction to the jointly estimated 3D shape models. We start with a basic volumetric reconstruction pipeline which recovers a volumetric representation in terms of a truncated signed distance function (TSDF) of the scene. Based on this reconstruction, we apply an ensemble of exemplar-based object detectors to find instances of objects in the scene. The detected objects are then jointly optimized for their shape and after convergence fused back into the original reconstruction to smoothly blend with non-object parts of the scene such as road or sidewalk.

3.1. Volumetric Fusion

This section describes our basic volumetric reconstruction pipeline which serves as input and baseline to our method. We leverage the truncated signed distance function (TSDF) representation [6] due to its generality and as it makes no assumptions about the 3D surface topology, which is unknown in our case. Surfaces are implicitly represented as the 0-level set of the TSDF and can be easily recovered using, e.g., the marching cubes algorithm by Lorensen et al. [28].

Given a sequence of fisheye images, we first obtain the intrinsic camera parameters as well as the camera poses using the approach of Heng et al. [19]. We rectify the images of adjacent frames spherically (see Fig. 2) such that epipolar lines become horizontal and run semi-global matching [20] in order to obtain disparity maps. Taking advantage of the efficient hashmap representation of Niesner et al. [31], we perform volumetric fusion [6], i.e., we update the weights $w(\mathbf{p})$ and the truncated signed distance values $d(\mathbf{p})$ via

$$d_{i+1}(\mathbf{p}) = \frac{w_i(\mathbf{p}) d_i(\mathbf{p}) + \hat{w}(\mathbf{p}) \hat{d}(\mathbf{p})}{w_i(\mathbf{p}) + \hat{w}(\mathbf{p})} \tag{1}$$

$$w_{i+1}(\mathbf{p}) = w_i(\mathbf{p}) + \hat{w}(\mathbf{p}) \tag{2}$$

where $\mathbf{p} \in \mathbb{R}^3$ denotes the location of a cell in the volumetric grid, and $\hat{d}(\cdot)$, $\hat{w}(\cdot)$ represent the truncated signed distance value and weight of the current observation (i.e., disparity map), respectively. We use a truncation threshold of 1.2 m and a weight that slowly decays behind the surface. As we found that approximating the distances $\hat{d}(\cdot)$ by calculating them along the viewing ray [39] results in artifacts when the ray hits the surface at highly slanted angles, we use a different strategy which approximates the TSDF values more faithfully: We triangulate each disparity map and convert the resulting meshes into volumes using a 3D signed distance transform[2]. This leads to correct distance estimates also for surfaces which are not fronto-parallel.

3.2. Object Detection

Given the volumetric 3D reconstruction from the previous section, we are interested in discovering objects of similar 3D shape to form the basis for our joint optimization in Section 3.3. Establishing correspondences in 3D space has several advantages over correspondences in the image domain: Besides the fact that similar 3D shapes can be matched even if the appearance disagrees, 3D models must not be scale invariant, thereby gaining robustness. A variety of methods can be leveraged for object discovery in 3D, including methods based on convexity/symmetry criteria [22], discriminative approaches [44] or graph matching [45].

In this work, we use a discriminative approach to obtain 3D box proposals: For each object category (e.g., building or car), we train an ensemble of linear exemplar SVMs [29] directly on the TSDF volume using a small (≤ 3) number of annotated instances. Compared to learning more complex features [38], we found this simple approach sufficient for our needs. First, we extract a small subsequence for training in which we label the relevant objects using 3D bounding boxes. For efficient object labeling we developed a 3D visualization tool based on WebGL, which allows to label

[2] http://grail.cs.washington.edu/software-data/ply2vri/

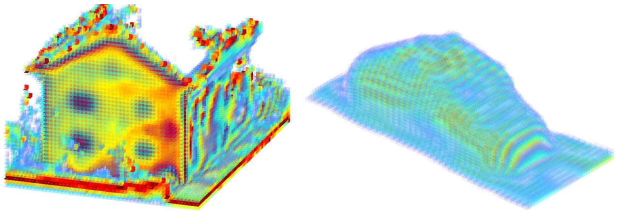


Figure 3: **Visualization of TSDF Model.** This figure shows a volumetric visualization of the learned TSDF model mean for a building (left) and a car (right). Red colors indicate voxels close to the estimated surface (i.e., 0-level set)

objects in 3D point clouds in a few seconds. Next, we define a feature vector by concatenating all voxels within the 3D box. Each voxel comprises the truncated signed distance to the surface $d(\mathbf{p})$ as well as a binary flag indicating if the voxel has been observed or not (i.e., it is within the truncation limit) as feature. We train exemplar SVMs on this representation using several rounds of hard negative mining and apply them in a sliding window fashion to the full 3D reconstruction. As 3D representations offer the advantage of scale invariance, we slide and rotate a 3D box of fixed scale over the 3D reconstruction volume. Fig. 4 (left) illustrates our detection results.

3.3. Joint Object Reconstruction

Given the initial volumetric 3D reconstruction and the box proposals, our goal is to jointly recover objects with similar shape while learning an object model for each category. More formally, let $d(\mathbf{p}) \in \mathbb{R}$ denote the truncated signed distance to the closest surface at point $\mathbf{p} \in \mathbb{R}^3$. Let further $w(\mathbf{p}) \in \mathbb{R}_{\geq 0}$ denote the weight at point $\mathbf{p}$ which takes $w = 0$ in unobserved regions or outside the truncation band, and $w = 1$ close to the estimated surface. As the TSDF representation estimated by volumetric fusion (Section 3.1) yields only values at discrete voxel locations, we use bilinear interpolation for calculating $d(\mathbf{p})$ and $w(\mathbf{p})$ at intermediate points. We formulate our objective as minimizing the distance between the TSDF values of the initial reconstruction specified by the mappings $d(\mathbf{p})$ and $w(\mathbf{p})$, and a set of jointly optimized 3D shape models. Fig. 3 illustrates two of the 3D shape models learned by our approach.

In the following, let N denote the number of observations (i.e., 3D bounding box proposals from Section 3.2) and let M denote the number of shape models we want to learn. Let further $\Omega \subset [0,1]^3$ be the discrete domain of our shape models. While many choices are possible, we simply take Ω as an axis aligned 3D grid with equidistant spacing between points. Let now $\boldsymbol{\pi} = \{\pi_1, \dots, \pi_N\}$ denote a set of affine mappings from the unit cube to $\mathbb{R}^3$, i.e., $\pi_i : [0,1]^3 \rightarrow \mathbb{R}^3$. We may think of these mappings as specifying the location of an observed object i in terms of its 3D bounding box, obtained by mapping the edges of the unit cube $[0,1]^3$ via π_i to coordinate system of the initial reconstruction. In this work, we focus our attention on a subset of affine mappings: We assume that all objects are located on a common (and known) ground plane, and thus parametrize each mapping π_i in terms of the following 5 parameters:

- translation in the 2D ground plane: $\mathbf{t}_i \in \mathbb{R}^2$
- rotation around the vertical axis: $r_i \in [0, 2\pi]$
- scaling in all three dimensions: $\mathbf{s}_i \in \mathbb{R}^3_{\geq 0}$

To model the shape of the objects, many choices are possible. For simplicity, we consider a linear embedding. Let $\boldsymbol{\phi} = \{\phi_1, \dots, \phi_M\}$ denote a set of D-dimensional linear embeddings ($D \ll |\Omega|$) which specify a real value for every $\mathbf{p} \in \Omega$ given a coefficient vector $\mathbf{x} \in \mathbb{R}^D$:

$$\phi_j(\mathbf{p}, \mathbf{x}) = \mu_j(\mathbf{p}) + \sum_{d=1}^{D} x_d \, \xi_d^{(j)}(\mathbf{p}) \tag{3}$$

Here, the parameters $\mu \in \mathbb{R}^{|\Omega|}$ and $\xi_d \in \mathbb{R}^{|\Omega|}$ specify the mean as well as an orthonormal basis, respectively. Each ϕ_j represents the shape (or TSDF) of model j by specifying a signed distance value at each point $\mathbf{p} \in \Omega$, given an observation dependent coefficient vector $\mathbf{x}$ as input. We specify one coefficient vector for each observation $\mathbf{x}_i \in \mathbb{R}^D$, $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$. The complexity of the model varies with the dimensionality D. For the simplest case ($D = 0$) we obtain the mean model. Finally, we associate each model with its average scale $\mathbf{v}_j \in \mathbb{R}^3_{\geq 0}$, $\mathbf{V} = \{\mathbf{v}_1, \dots, \mathbf{v}_M\}$ to encourage scale consistency between different observations associated to a model. The assignment of models to observations is determined by an index set $\mathbf{k} = \{k_1, \dots, k_N\}$ with $k_i \in \{1, \dots, M\}$.

We are now ready to formulate our objective in terms of a discrete-continuous optimization problem

$$\underset{\boldsymbol{\pi}, \boldsymbol{\phi}, \mathbf{X}, \mathbf{V}, \mathbf{k}}{\operatorname{argmin}} \sum_{i=1}^{N} \Psi_i(\boldsymbol{\pi}, \boldsymbol{\phi}, \mathbf{X}, \mathbf{V}, \mathbf{k}) \tag{4}$$

with energy function

$$\begin{aligned} \Psi_i(\cdot) \;=\; & \psi_{shp}(\pi_i, \phi_{k_i}, \mathbf{x}_i) + \lambda_{scale}\, \psi_{scale}(\mathbf{s}_i, \mathbf{v}_{k_i}) \\ & + \lambda_{reg}\, \psi_{reg}^{(i)}(\pi_i) \end{aligned} \tag{5}$$

where $\psi_{shp}(\cdot)$ ensures that the shape of observation i fits the associated model k_i, $\psi_{scale}(\cdot)$ encourages agreement in scale, and $\psi_{reg}(\cdot)$ is a regularizer which penalizes strong deviations from the initial detections. $\lambda_{scale}, \lambda_{reg} \in \mathbb{R}_{\geq 0}$ are parameters controlling the influence of the different terms. We define $\psi_{shp}(\cdot)$ as the *weighted squared TSDF difference* between the associated model ϕ and the observation specified via π:

$$\psi_{shp}(\pi, \phi, \mathbf{x}) = \sum_{\mathbf{p} \in \Omega} w(\pi(\mathbf{p}))\, [\phi(\mathbf{p}, \mathbf{x}) - d(\pi(\mathbf{p}))]^2 \tag{6}$$

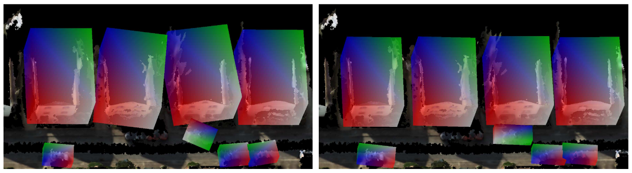


Figure 4: **Object Poses.** This figure shows the initial 3D detections (left) and the result after joint alignment and model learning (right) overlayed with the initial reconstruction.

Scale discrepancy is measured by the squared distance between the observation scale $\mathbf{s}$ and the model scale $\mathbf{v}$:

$$\psi_{scale}(\mathbf{s},\mathbf{v}) = \|\mathbf{s}-\mathbf{v}\|^2 \tag{7}$$

Finally, strong deviations from the initial detection i are penalized as

$$\psi_{reg}^{(i)}(\pi_i) = (r_i - \hat{r}_i)^2 + \|\mathbf{t}_i - \hat{\mathbf{t}}_i\|^2 + \|\mathbf{s}_i - \hat{\mathbf{s}}_i\|^2 \tag{8}$$

where $(r_i, \mathbf{t}_i, \mathbf{s}_i)$ are the transformation parameters of object i and $(\hat{r}_i, \hat{\mathbf{t}}_i, \hat{\mathbf{s}}_i)$ denote the initial transformation parameters specified by the detection.

3.4. Inference

Optimizing Eq. 4 directly is a very difficult task due to the large dimensionality of the parameter space. We therefore partition the set of variables into tractable blocks and apply block coordinate descent (BCD) to find a local minimizer. Each block lowers the value of the objective function, thus our algorithm is guaranteed to converge. More specifically, we initialize $\boldsymbol{\pi}$, $\mathbf{V}$ and $\mathbf{k}$ according to the detections, the mean of $\boldsymbol{\phi}$ to a random observation, $\mathbf{X} = \mathbf{0}$, and iterate the following blocks:

Block $\{\boldsymbol{\pi}, \mathbf{V}\}$: The first block optimizes the object poses $\boldsymbol{\pi}$ jointly with the model scales $\mathbf{V}$ while keeping the other variables fixed. Due to the small number of parameters involved, we leverage gradient descent in order to find a local minimum. We first differentiate the objective function in Eq. 4 with respect to $\{\mathbf{t}_i, r_i, \mathbf{s}_i\}$ and $\mathbf{V}$ (see supplementary material for details), and then solve the non-linear least squares problem in Eq. 4 using the Ceres solver [1]. Fig. 4 illustrates the object poses before (left) and after (right) convergence.

Block $\{\boldsymbol{\phi}, \mathbf{X}\}$: The second block optimizes the shape models $\boldsymbol{\phi}$ jointly with the coefficients $\mathbf{X}$ while keeping the other variables fixed. Note that this optimization does not depend on the scale term $\psi_{scale}(\cdot)$. Further, the object poses $\boldsymbol{\pi}$ are fixed. Thus the objective in Eq. 4 reduces to M independent weighted PCA problems which we solve using the robust approach of Torre et al. [25]. For the case $D = 0$, this becomes equivalent to computing the weighted mean.

Block $\{\mathbf{X}, \mathbf{k}\}$: The final block optimizes the coefficients $\mathbf{X}$ jointly with the model associations $\mathbf{k}$ while keeping the other variables fixed. We first note that this optimization can be performed independently for each observation $i \in \{1, \ldots, N\}$. We thus obtain $\mathbf{x}_i$ and k_i for observation i as

$$\operatorname*{argmin}_{\mathbf{x}_i, k_i} \; \psi_{shp}(\pi_i, \phi_{k_i}, \mathbf{x}_i) + \lambda_{scale}\, \psi_{scale}(\mathbf{s}_i, \mathbf{v}_{k_i}) \tag{9}$$

which can be found by minimization with respect to $\mathbf{x}_i$ for each $k_i \in \{1, \ldots, M\}$. As $\psi_{scale}(\cdot)$ does not depend on $\mathbf{x}_i$, the minimizer of $\psi_{shp}(\cdot)$ with respect to $\mathbf{x}_i$ can be identified with the solution to an ordinary linear least squares problem. For details, we refer the reader to the supplementary material.

4. Experimental Evaluation

This section presents results of the proposed approach and compares our method to several baselines both quantitatively as well as qualitatively. As existing datasets for quantitative evaluation of multi-view reconstruction (e.g., Middlebury [36]) focus on small scenes of single objects where our algorithm is not applicable, we have recorded a novel suburban dataset for our purpose where objects of similar shape such as buildings and cars occur frequently. Towards this goal, we have equipped a station wagon with fisheye cameras to the side as well as a Velodyne and a pushbroom laser scanner in order to generate ground truth. All cameras and laser scanners have been synchronized at a frame rate of roughly 10 fps, yielding about one captured frame every meter at typical driving speeds of 25 mph. For calibration we used the approach proposed by Heng et al. [19].

Using this setup, we recorded a sequence of 320 frames containing several buildings and cars. As the sequences have been recorded under regular daylight conditions, they are (unlike Middlebury [36]) highly challenging for current reconstruction pipelines. Some of the challenges are highlighted in Fig. 1. For evaluation, we subsample the laser scanner point cloud and the meshes produced by the methods equidistantly at 0.1 m, extrude incomplete planar surfaces, and clip all points in the ground truth as well as all results at 20 m distance from the closest camera center. See supplementary for an illustration of our 3D ground truth.

4.1. Methods

As baseline, we leverage the state-of-the-art reconstruction pipeline PMVS2 from Furukawa et al. [12] in combination with two meshing alternatives, namely Poisson reconstruction [23] and smooth signed distance surface reconstruction [4]. As those surface reconstruction methods tend to produce closed surfaces, we clip large triangles as suggested by Furukawa et al. [12]. In order to make our data applicable to PMVS2, we projected the fisheye images

	All			Buildings			Cars		
	Comp.	Acc.	F1 score	Comp.	Acc.	F1 score	Comp.	Acc.	F1 score
PMVS2 (default)	12.24 %	79.26 %	21.20 %	16.35 %	85.70 %	27.46 %	0 %	0 %	0 %
PMVS2 (optimized)	26.65 %	78.31 %	39.77 %	30.57 %	91.59 %	45.84 %	7.31 %	96.97 %	13.59 %
PMVS2 (default) + Poisson	20.90 %	64.12 %	31.53 %	25.49 %	77.05 %	38.30 %	1.58 %	**100.00 %**	3.12 %
PMVS2 (optimized) + Poisson	27.38 %	51.93 %	35.85 %	29.39 %	75.50 %	42.31 %	1.46 %	**100.00 %**	2.88 %
PMVS2 (default) + SSD	25.27 %	54.21 %	34.47 %	34.12 %	66.32 %	45.05 %	0 %	0 %	0 %
PMVS2 (optimized) + SSD	39.27 %	61.51 %	47.94 %	44.07 %	81.19 %	57.13 %	6.94 %	**100.00 %**	12.98 %
Ours (Initial)	60.21 %	**94.15 %**	73.45 %	52.70 %	**93.84 %**	67.50 %	85.20 %	94.35 %	89.54 %
Ours (PC 0)	**76.83 %**	91.35 %	**83.46 %**	**72.24 %**	91.26 %	**80.64 %**	**92.69 %**	93.46 %	**93.07 %**
Ours (PC 1)	76.65 %	89.53 %	82.59 %	71.55 %	89.46 %	79.51 %	91.53 %	92.90 %	92.21 %

Table 1: **Quantitative Evaluation.** This figure shows the performance of the baseline methods (PMVS2 variants and initial reconstruction) and our method (PC 0/1) with respect to completeness, accuracy and F1 score using a 0.5 m detection threshold. We separately evaluate all regions within 20 m from the camera, as well as regions representing the categories “buildings” and “cars”.

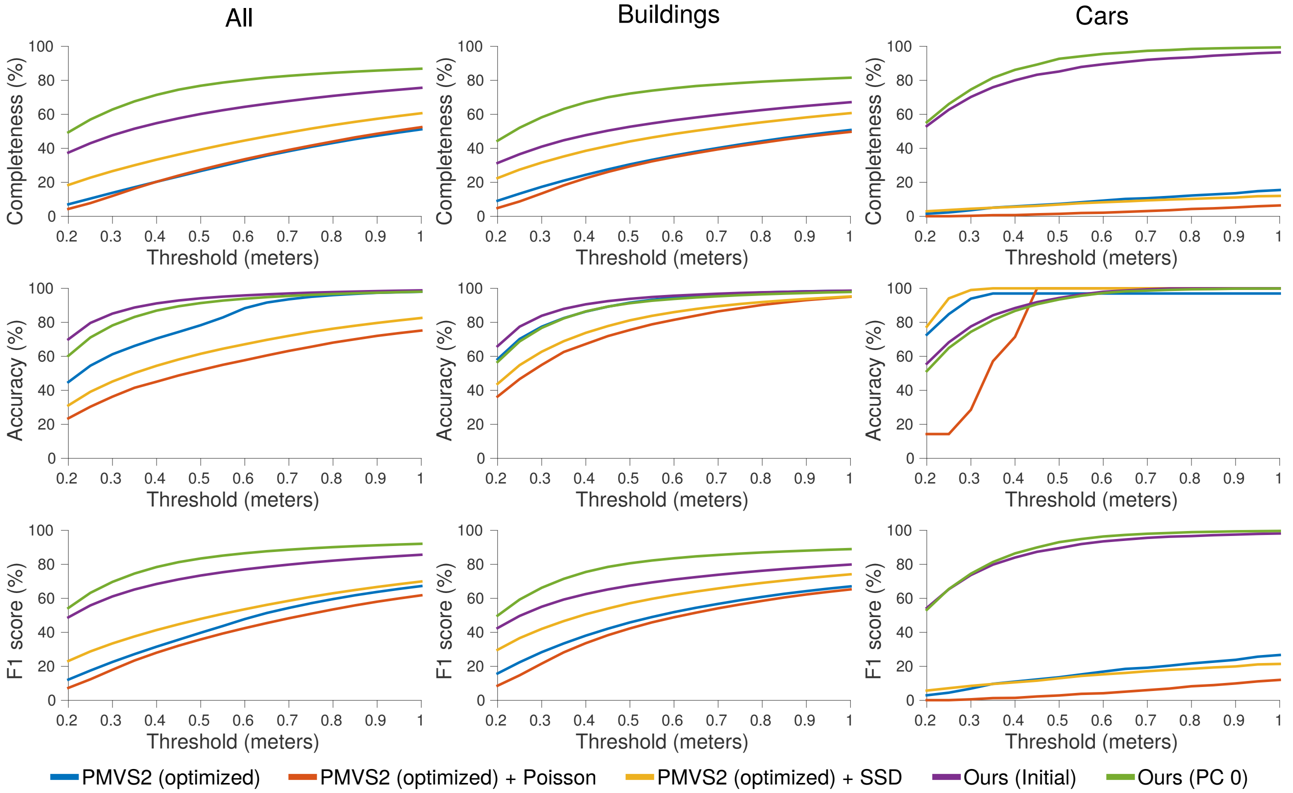


Figure 5: **Varying the Evaluation Distance.** This figure shows quantitative results in terms of completeness, accuracy and F1 score when varying the evaluation distance τ between 0.2 and 1.0 m. To avoid clutter, we only show results for a subset of the methods. We refer the reader to the supplementary material for the full plots.

onto virtual perspective image planes using an opening angle of 120°, such that the images cover all objects which appear in the ground truth. We realized that the performance of PMVS2 depends heavily on the parameter settings. We thus optimized all parameters of the baseline with respect to the reconstruction metrics using grid search on our compute cluster. For completeness we show both, PMVS2 results using the optimized parameter settings as well as PMVS2 using the default parameters. In addition to the meshed results, we also directly evaluate the point cloud returned by PMVS2. Furthermore, we compare the results of our full model with respect to the initial reconstruction. We leverage the marching cubes algorithm [28] to turn our volumetric reconstruction results into meshes. For all meshes, we remove spurious isolated vertices created by the reconstruction algorithm in a post-processing step. Throughout all experiments, we set the parameters in our model to $\lambda_{size} = 1000$ and $\lambda_{reg} = 50$ which we have been determined empirically.

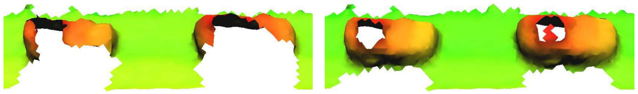


Figure 7: **Model Completion.** Our method fills in the occluded/unseen side of objects (left: input; right: our result).

Furthermore, as sky regions often lead to spurious matches, we trained a sky detector using ALE [26] and removed sky regions before processing the images for all methods.

4.2. Quantitative Experiments

For quantitative evaluation, we measure performance in terms of completeness, accuracy and F1 score. We calculate completeness as the percentage of ground truth 3D points for which at least one reconstructed 3D point (i.e., vertex of the subsampled mesh) is within a distance of $\tau = 0.5$ m. Similarly, we calculate accuracy as the percentage of reconstructed 3D points for which at least one ground truth 3D point is within a distance of $\tau = 0.5$ m. Furthermore, we provide the combined F1 score:

$$F_1 = 2 \cdot \frac{\text{completeness} \cdot \text{accuracy}}{\text{completeness} + \text{accuracy}} \tag{10}$$

Our quantitative results are shown in Table 1, evaluated at all 3D ground truth points (left column), as well as restricted to buildings and vehicles[3] (middle and right column). We evaluate our initial reconstruction as well as our joint reconstruction results for $D = 0$ (PC 0) and $D = 1$ (PC 1). As evidenced by our experiments, our initial reconstruction is able to outperform all variants of PMVS2 [12] in terms of both completeness as well as accuracy in almost all regions. Note that for cars, PMVS2 recovers less than 10% of the surfaces due to the challenges in matching textureless and specular surfaces. In contrast, our joint reconstruction (PC 0 / PC 1) transfers surface information from similar shapes in the scene, boosting completeness by more than 15% with respect to to our initial reconstruction baseline with a moderate loss in accuracy. This leads to significant improvements in F1 score for all three categories.

It may seem surprising that our weighted mean model (PC 0) slightly outperforms the more expressive model comprising one principal component (PC 1). After inspection, we found this effect to be caused by systematic errors in the initial reconstruction. This systematic noise can be partially overcome by our mean model (PC 0), which poses a stronger regularization on our joint reconstruction. An alternative for alleviating this effect would be to integrate class-specific object knowledge into the process [34, 35].

Results for a subset of the methods when varying the evaluation distance threshold τ are illustrated in Fig. 5. While completeness and accuracy decreases for all methods with smaller detection thresholds, the relative gain in performance of our method with respect to the baselines increases. This indicates that our reconstructions are not only more complete, but also *metrically* more accurate. The full plot is provided in the supplementary material.

4.3. Qualitative Experiments

Our qualitative results are shown in Fig. 6-8. Fig. 6 demonstrates the variability of our learned models and Fig. 7 illustrates the ability of our model to complete regions of objects which have never been observed. Fig. 8 shows (from-top-to-bottom): the point cloud created by PMVS2 [12] using the default parameter setting, the result with optimized parameters, meshed results of the optimized PMVS2 point cloud using Poisson [23] and SSD [4] surface reconstruction, our initial reconstruction, and our final results (PC 0). The colors denote the height, normalized with respect to the highest and the lowest point of the reconstruction. Note how our method is able to recover cars as well as missing walls of the buildings. Furthermore, our reconstructions convey much more detail than the baseline methods. As roofs in this sequence are barely observed from the viewpoint of the vehicle, none of the algorithms was able to reconstruct them. Additional results are provided in the supplementary material.

5. Conclusions

We have presented a novel method for jointly reconstructing objects with similar shapes in 3D by optimizing their pose and shape parameters using a volumetric representation. We demonstrated the value of our model with respect to PMVS2 and our initial reconstruction on a novel challenging suburban dataset. Our method improves in particular with respect to completeness as it transfers surface knowledge between objects of similar shape. One weakness of our model is that it is only little robust to outliers in the detection. In the future, we thus plan to incorporate outlier handling by optimizing a robust function. Further, we want extend our method to more object categories and combine it with external 3D shape knowledge by learning a joint model from 3D CAD models as well as a collection of automatically retrieved objects in large scenes.

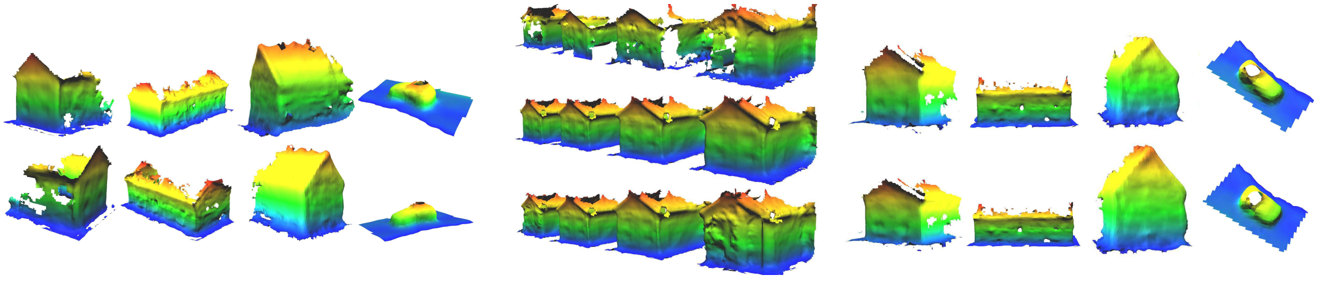
Acknowledgements. We would like to express thanks for support from the following research grants: NSFC-61231010, 61272027, 61421062, 61210005 and the Okawa Foundation Research Grant.

References

[1] S. Agarwal, K. Mierle, and Others. Ceres solver. http://ceres-solver.org. 5

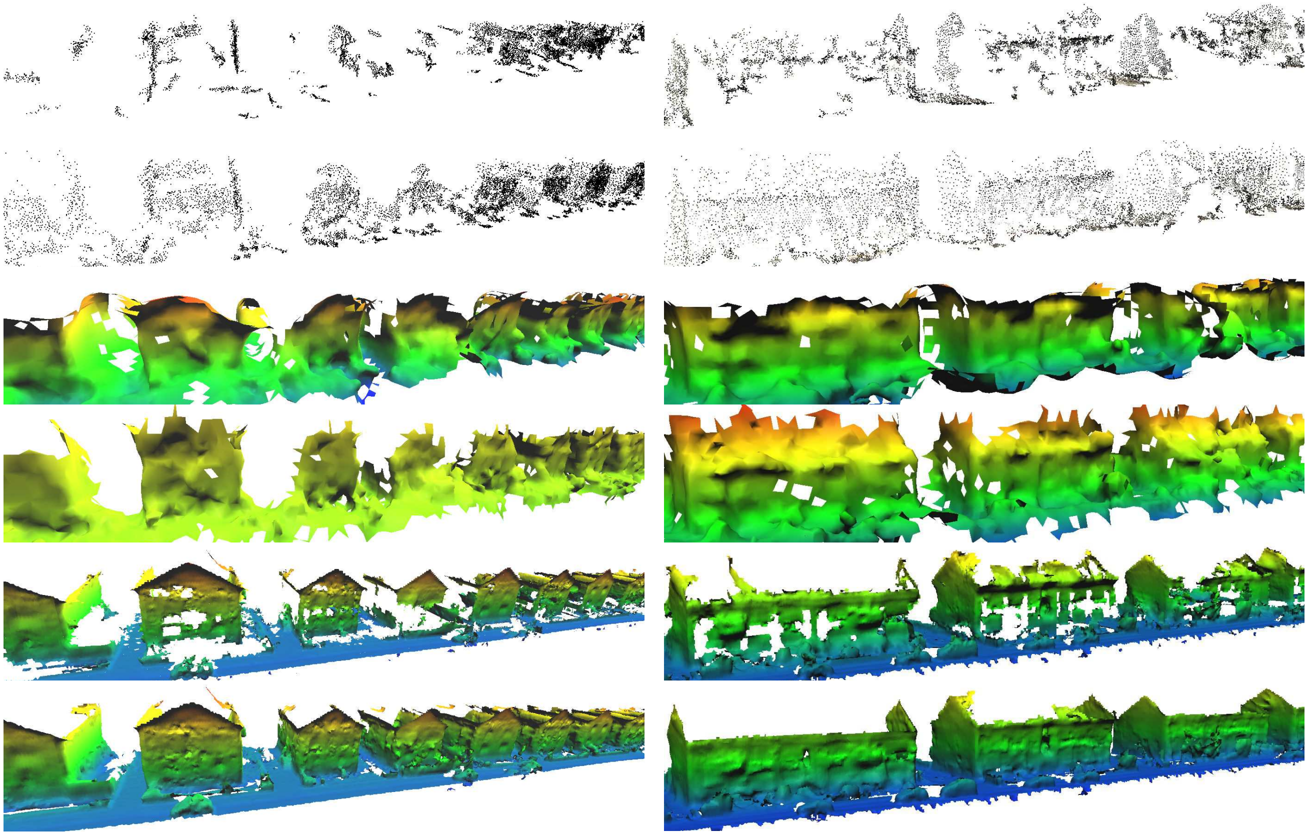
[2] D. Anguelov, C. Dulong, D. Filip, C. Frueh, S. Lafon, R. Lyon, A. S. Ogale, L. Vincent, and J. Weaver. Google

[3] In order to evaluate buildings and vehicles separately, we have annotated them with ground truth 3D bounding boxes.



(a) **Model Variety.** (b) **Intra-model Variation.** (c) **Variation in Scale.**

Figure 6: **Visualization of Learned Models.** Figure a shows different shape models learned by our approach from two viewpoints. Intra-model shape variations are illustrated in b, where the rows show (top-to-bottom): the input data, the reconstruction with 0 PCs and the reconstruction with 1 PC. Note how the PC 1 model is able to capture the step in the rightmost building while PC 0 enforces stronger regularization. Finally, we show variation in model scale ($\mathbf{s}$) after optimization in figure c. Note how the width and height of the objects differs between instances.



(a) **Left Side of the Street.** (b) **Right Side of the Street.**

Figure 8: **Qualitative Results.** This figure shows our reconstruction results from both sides of the same street, respectively. From top-to-bottom: Point cloud from PMVS2 using the default parameter setting, point cloud from PMVS2 with optimized parameters, PMVS2 (optimized) + Poisson, PMVS2 (optimized) + SSD, our initial reconstruction, and our fused result (PC 0). Note how our method is able to recover cars as well as missing building walls, and handling different types of buildings.

street view: Capturing the world at street level. *Computer*, 43(6):32–38, 2010. 3

[3] I. Biederman. Recognition-by-components: a theory of human image understanding. *Psychological Review*, 94(2):115, 1987. 1

[4] F. Calakli and G. Taubin. SSD: smooth signed distance surface reconstruction. *Computer Graphics Forum*, 30(7):1993–2002, 2011. 5, 7

[5] D. Cremers, M. Rousson, and R. Deriche. A review of statistical approaches to level set segmentation: Integrating color,

texture, motion and shape. *IJCV*, 72:215, 2007. 2

[6] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *SIGGRAPH*, 1996. 2, 3

[7] A. Delaunoy and M. Pollefeys. Photometric bundle adjustment for dense multi-view 3d modeling. In *CVPR*, 2014. 2

[8] W. Deng, J. Hu, J. Lu, and J. Guo. Transform-invariant PCA: A unified approach to fully automatic facealignment, representation, and recognition. *PAMI*, 36(6):1275–1284, 2014. 2

[9] D. F. Fouhey, A. Gupta, and M. Hebert. Data-driven 3D primitives for single image understanding. In *ICCV*, 2013. 2

[10] J.-M. Frahm, P. Fite-Georgel, D. Gallup, T. Johnson, R. Raguram, C. Wu, Y.-H. Jen, E. Dunn, B. Clipp, S. Lazebnik, and M. Pollefeys. Building rome on a cloudless day. In *ECCV*, 2010. 1

[11] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski. Manhattan-world stereo. In *CVPR*, 2009. 2

[12] Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *PAMI*, 32(8):1362–1376, 2010. 2, 5, 7

[13] D. Gallup, J.-M. Frahm, and M. Pollefeys. Piecewise planar and non-planar stereo for urban scene reconstruction. In *CVPR*, 2010. 1, 2

[14] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The KITTI dataset. *IJRR*, 32(11):1231–1237, 2013. 3

[15] F. Güney and A. Geiger. Displets: Resolving stereo ambiguities using object knowledge. In *CVPR*, 2015. 2

[16] C. Haene, N. Savinov, and M. Pollefeys. Class specific 3d object shape priors using surface normals. In *CVPR*, 2014. 2

[17] C. Haene, C. Zach, A. Cohen, R. Angst, and M. Pollefeys. Joint 3D scene reconstruction and class segmentation. In *CVPR*, 2013. 2

[18] C. Haene, C. Zach, B. Zeisl, and M. Pollefeys. A Patch Prior for Dense 3D Reconstruction in Man-Made Environments. In *3DIMPVT*, 2012. 2

[19] L. Heng, B. Li, and M. Pollefeys. Camodocal: Automatic intrinsic and extrinsic calibration of a rig with multiple generic cameras and odometry. In *IROS*, 2013. 3, 5

[20] H. Hirschmüller. Stereo processing by semiglobal matching and mutual information. *PAMI*, 30(2):328–341, 2008. 3

[21] M. Hornacek, C. Rhemann, M. Gelautz, and C. Rother. Depth super resolution by rigid body self-similarity in 3d. In *CVPR*, 2013. 2

[22] A. Karpathy, S. Miller, and L. Fei-Fei. Object discovery in 3d scenes via shape analysis. In *ICRA*, 2013. 3

[23] M. M. Kazhdan and H. Hoppe. Screened poisson surface reconstruction. *SIGGRAPH*, 32(3):29, 2013. 2, 5, 7

[24] A. Kundu, Y. Li, F. Dellaert, F. Li, and J. Rehg. Joint semantic segmentation and 3d reconstruction from monocular video. In *ECCV*, 2014. 2

[25] F. D. la Torre and M. J. Black. Robust principal component analysis for computer vision. In *ICCV*, pages 362–369, 2001. 5

[26] L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr. Associative hierarchical random fields. *PAMI*, 36(6):1056–1077, 2014. 7

[27] F. Lafarge, R. Keriven, M. Bredif, and H.-H. Vu. A hybrid multiview stereo algorithm for modeling urban scenes. *PAMI*, 35(1):5–17, 2013. 2

[28] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *SIGGRAPH*, 1987. 3, 6

[29] T. Malisiewicz, A. Gupta, and A. A. Efros. Ensemble of exemplar-svms for object detection and beyond. In *ICCV*, 2011. 1, 3

[30] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *ISMAR*, 2011. 2

[31] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger. Real-time 3d reconstruction at scale using voxel hashing. In *SIGGRAPH*, 2013. 1, 3

[32] A. Owens, J. Xiao, A. Torralba, and W. T. Freeman. Shape anchors for data-driven multi-view reconstruction. In *ICCV*, 2013. 2

[33] M. Pollefeys. Detailed real-time urban 3d reconstruction from video. *IJCV*, 78(2-3):143–167, July 2008. 1

[34] V. A. Prisacariu and I. Reid. Nonlinear shape manifolds as shape priors in level set segmentation and tracking. In *CVPR*, 2011. 2, 7

[35] V. A. Prisacariu and I. Reid. Shared shape spaces. In *ICCV*, 2011. 2, 7

[36] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR*, 2006. 5

[37] Q. Shan, B. Curless, Y. Furukawa, C. Hernandez, and S. M. Seitz. Occluding contours for multi-view stereo. In *CVPR*, 2014. 2

[38] S. Song and J. Xiao. Sliding shapes for 3D object detection in depth images. In *ECCV*, 2014. 3

[39] F. Steinbrucker, C. Kerl, and D. Cremers. Large-scale multiresolution surface reconstruction from rgb-d sequences. In *ICCV*, 2013. 2, 3

[40] A. Tsai, W. M. W. III, S. K. Warfield, and A. S. Willsky. An em algorithm for shape classification based on level sets. *Medical Image Analysis*, 9(5):491–502, 2005. 2

[41] H. Vu, P. Labatut, J. Pons, and R. Keriven. High accuracy and visibility-consistent dense multiview stereo. *PAMI*, 34(5):889–901, 2012. 2

[42] A. Wendel, M. Maurer, G. Graber, T. Pock, and H. Bischof. Dense reconstruction on-the-fly. In *CVPR*, 2012. 1

[43] C. Zach, T. Pock, and H. Bischof. A globally optimal algorithm for robust tv-l1 range image integration. In *ICCV*, 2007. 2

[44] Q. Zhang, X. Song, X. Shao, H. Zhao, and R. Shibasaki. Start from minimum labeling: Learning of 3d object models and point labeling from a large and complex environment. In *ICRA*, 2014. 3

[45] Q. Zhang, X. Song, X. Shao, H. Zhao, and R. Shibasaki. When 3d reconstruction meets ubiquitous RGB-D images. In *CVPR*, 2014. 3

[46] Q.-Y. Zhou, S. Miller, and V. Koltun. Elastic fragments for dense scene reconstruction. In *ICCV*, 2013. 1