

Accurate 3-D Reconstruction with RGB-D Cameras using Depth Map Fusion and Pose Refinement

Markus Ylimäki and Janne Heikkilä

Center for Computer Vision and Signal Analysis
University of Oulu
Oulu, Finland
Email: firstname.lastname@oulu.fi

Juho Kannala

Department of Computer Science
Aalto University
Espoo, Finland
Email: juho.kannala@aalto.fi

Abstract—Depth map fusion is an essential part in both stereo and RGB-D based 3-D reconstruction pipelines. Whether produced with a passive stereo reconstruction or using an active depth sensor, such as Microsoft Kinect, the depth maps have noise and may have poor initial registration. In this paper, we introduce a method which is capable of handling outliers, and especially, even significant registration errors. The proposed method first fuses a sequence of depth maps into a single non-redundant point cloud so that the redundant points are merged together by giving more weight to more certain measurements. Then, the original depth maps are re-registered to the fused point cloud to refine the original camera extrinsic parameters. The fusion is then performed again with the refined extrinsic parameters. This procedure is repeated until the result is satisfying or no significant changes happen between iterations. The method is robust to outliers and erroneous depth measurements as well as even significant depth map registration errors due to inaccurate initial camera poses.

I. INTRODUCTION

The three-dimensional (3-D) reconstruction of a scene or an object is a classical problem in computer vision [1]. The reconstruction methods can be roughly categorized into passive stereo approaches (e.g. [2], [3], [4]) and RGB-D based methods using an active depth sensor (e.g. [5], [6], [7]). The stereo approaches reconstruct the scene purely from photographs while the RGB-D methods use a specific depth sensor, such as Microsoft Kinect, to provide depth measurements of the scene.

In the depth map based stereo reconstruction methods, such as [8], [9], [10], [3], and especially in the RGB-D reconstruction, the fusion of depth maps is an essential part of the modeling pipeline and may have a significant influence on the final result. The simplest way to fuse depth maps is to register them into the same coordinate system but this approach will lead to a huge number of redundant points, which makes the further processing very slow.

A better way is to aim directly at a non-redundant point cloud so that overlapping points from different depth maps do not increase the redundancy of the point cloud [11], [12]. In this paper, we propose a method which is able to reconstruct non-redundant point clouds from redundant, noisy and poorly registered depth maps. That is, at first, the method merges a sequence of depth maps into a single non-redundant point cloud so that new measurements are either added to the cloud

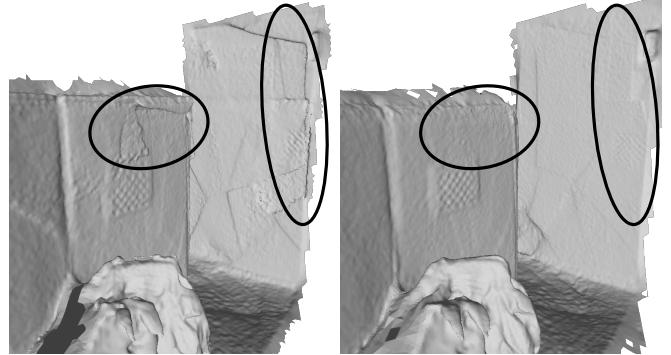


Fig. 1. Parts of the mesh reconstructions of Office2 made with Poisson Surface Reconstruction [13] from the point clouds made with the method in [12] (left) and the proposed one (right). The proposed method reduces the ambiguity of the surfaces, and thus, produces more accurate results (black ellipses).

or used to refine the nearby existing points. Then, the method re-registers the original depth maps into the fused point cloud to refine the camera poses and repeats the fusion step. The experiments show that the proposed method significantly reduces the amount of outliers and, especially, badly registered points in the final point cloud.

Figure 1 presents parts of the triangle meshes created using Poisson Surface Reconstruction (PSR) [13] from the point cloud created with the method in [12] and with the proposed method. The figure clearly shows the difference between the results. Inaccurate registration of the depth maps cause ambiguous surfaces in the point cloud as well as in the mesh. The proposed method is able to produce results where the reconstructed surfaces are smoother and less ambiguous (black ellipses).

The rest of the paper is organized as follows. Section II introduces some previous works and their relationship to our approach which is described in more detail in Section III. The experimental results are presented and discussed in Section IV and Section V concludes the paper.

II. RELATED WORK

Fusion of depth maps as a part of a 3-D reconstruction has been widely studied both in passive stereo reconstruction pipelines [14], [8], [15], [9], [10] as well as in RGB-D based

methods [5], [6], [16], [7], [17]. Some of the existing methods are shortly described below.

Goesele et al. [14] merge the depth maps using the volumetric approach in [18], which uses a directional uncertainty for the depth measurements captured with a range scanner. However, the uncertainty in [14] is based on the photometric consistency of the depth measurements among the views where they are visible, and therefore, the uncertainty is not directly depth dependent. Later, their work has led to publicly available reconstruction application called Multi-View Environment [3] where the depth map fusion is still based on [18] but enhanced to work with depth maps having multiple scales.

The depth map fusion presented in [8] is able to reconstruct a scene from a live video in real time. Their method uses simple visibility violation rules but also exploits confidence measures for the points. However, the confidence is based on image gradients, and thus, is not depth dependent either.

Zach et al. have proposed a depth map fusion which incorporates a minimization of an energy function consisting of a total variation (TV) as a regularization term and a L^1 norm as a data fidelity term [15]. The terms are based on signed distance fields computed similarly as in [18].

Li et al. [9] have utilized the bundle adjustment approach in the depth map fusion. Again, their method relies strongly on the photo consistency of matched pixels between stereo images, and thus, the method may not work that well in RGB-D based reconstruction approaches like [11] and [12].

Being relatively simple, the fusion in [10] can be applied both in passive stereo and RGB-D based approaches. Similarly to [11] and [12], their method produces a non-redundant point cloud out of a set of overlapping depth maps. Nevertheless, they simply preserve points which do not have redundancy or have the highest accuracy among redundant points and do not exploit any uncertainty based merging of the redundant points.

As a summary, the passive stereo reconstruction approaches, described above, do not exploit any depth dependent uncertainty for the depth measurements like in [11] and [12] and the confidence measures are naturally very often based on the photometric consistence. In addition, they usually assume noisy depth maps, similarly to [12], but relatively accurate camera poses.

The interest towards RGB-D based approaches has been increasing widely since Microsoft released the first generation Kinect device (Kinect V1) in 2010. The RGB-D reconstruction algorithms, such as KinectFusion presented in [5], are known as real-time approaches producing scale and resolution limited reconstructions because of the memory consuming voxel based representation of the models.

KinectFusion is able to reconstruct a scene inside a fixed volume in real-time. The registration of the depth maps as well as the camera pose estimation is based on the iterative closest point (ICP) algorithm. Whelan et al. [6] and Roth & Vona [16] have provided extensions to KinectFusion which allow larger reconstructions but are still quite scale and resolution limited.

The memory restriction has been avoided especially by Nießner et al. [7] who proposed a method where the surface

data can be efficiently streamed in or out of a specific hash table. However, as for all methods designed for live video reconstruction, their method may not work that well with depth maps having wide baselines.

Point cloud based RGB-D reconstruction approaches do not have similar scale or resolution limitation than the voxel based approaches. One of such methods was proposed by Kyöstiä et al. in [11] where the point cloud is obtained by merging a sequence of depth maps iteratively. That is, the method loops through every pixel in every registered depth map in the sequence and either adds the point into the cloud of points in the space, if there are no other points nearby, or uses the measurement to refine an existing point. This way, the fusion does not increase the redundancy of the cloud and the location uncertainty of each point guarantees that the refinement takes all redundant measurements into account [11] and not just preserve the one which seems to be the most accurate [10]. The uncertainty is based on empirically defined, depth depended variances.

Kyöstiä's method is designed for Kinect V1 and the main contribution of their work is the fusion of redundant depth maps. Thus, their method is not very robust to outliers or certain measurement or registration errors. Ylimäki et al. recently improved the method in order to make it work with the newer Kinect device (Kinect V2) and provided three extensions to boost its robustness and accuracy [12]. The extensions include depth map pre-filtering to reduce the amount of outliers, improved uncertainty covariance to compensate for the measurement variances and make the method more accurate and filtering of the final point cloud to reduce the amount of erroneous measurements.

Although the method in [12] is able to reduce the amount of outliers quite significantly, it cannot handle severe registration errors which may cause ambiguous and rough surfaces both in the point cloud but also in the meshed model build with PSR [13], for instance (see Fig. 1).

In this paper, to overcome the above limitation of [12], we further develop the method with a re-registration extension. That is, after acquiring the first fused point cloud, our method first registers the original depth maps with the point cloud and then, like in KinectFusion [5], refines the camera poses according to the new registration, and finally, reruns the fusion. In the experimental results as well as in Figure 1, we show that the proposed method produces significantly better results robustly, than presented in [11] and [12].

III. METHOD

An overview of the proposed method is presented in Figure 2. As shown in the figure, the method takes a set of depth maps and RGB images with initial camera positions as input and outputs a point cloud. The method improves the approach in [12] with the re-registration extension marked with darker boxes in the figure. Similarly to [11] and [12], the proposed method can be used as a pipeline to process one depth map at a time so that the new depth maps are re-registered with the overlapping areas in the current point cloud, similarly as in

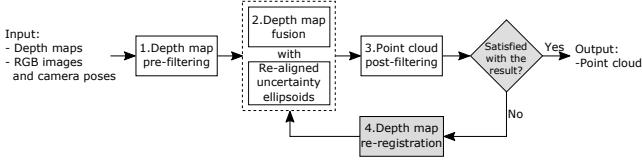


Fig. 2. An overview of the proposed fusion pipeline. In this paper, we propose the re-registration extension (parts with a gray background) to the fusion algorithm.

KinectFusion [5]. Thus, the only thing that limits the scale or size of the reconstruction is the available memory for storing the point cloud.

The pipeline has totally four steps: 1) depth map pre-filtering, 2) actual depth map fusion, 3) post-filtering of the final point cloud and 4) re-registration of the depth maps into the fused point cloud. The first three steps are briefly described in Section III-A. The fourth step is described in more detail in Section III-B.

A. Fusion pipeline

The proposed reconstruction pipeline consists of the depth map fusion and the re-registration. This section shortly introduces the fusion part of the pipeline.

As an overview, the depth map fusion provides three different ways to measure the uncertainties of the depth measurements and tries to replace, remove or refine the most uncertain ones with better measurements from the other overlapping depth maps.

The first phase in the pipeline is the depth map pre-filtering step, which tries to remove such measurements from the depth maps which seem to be outliers or too inaccurate. Inaccuracy is typically caused by the lens distortion (especially in the corners of the image) or by a phenomenon called multi-path interference (MPI) [19]. MPI errors typically occur in the depth measurements which are acquired with a time-of-flight depth sensor such as Kinect V2 and it causes positive biases to the measurements. MPI happens when the depth sensor receives multiple scattered or reflected signals from the scene for the same pixel. In addition, the depth maps usually have outliers and inaccurate points near depth edges and in dark areas which absorb the majority of the infrared light emitted by the sensor.

The filtering is based on the observation, that the density of points in a backprojected depth map near outliers and inaccurate or MPI distorted points, is usually smaller than in other, more accurately measured regions. That is, the filtering compares the distances between every backprojected depth measurement and its nearest neighbors to a corresponding reference distance, and removes the measurement if the distance is longer than a certain threshold. In this work, the measurement m is removed if

$$d_m > \gamma d_r(z_m), \quad (1)$$

where d_m is the distance from the point m to its 4th nearest neighbor in the measured backprojected depth map, $d_r(z_m)$

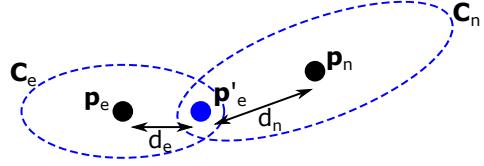


Fig. 3. Fusion of two nearby depth measurements. \mathbf{C}_e and \mathbf{p}_e are the covariance and location of the existing measurement. \mathbf{C}_n and \mathbf{p}_n are the covariance and location of the new measurement. These two measurements are merged together to a point \mathbf{p}'_e if the merged point is inside both covariance regions. The point with lower uncertainty (here \mathbf{p}_e) gets a bigger weight in the refinement ($d_e < d_n$).

is the corresponding reference distance at depth z_m and γ is constant variable ($\gamma = 1.83$ in our experiments). The reference distance is a function of the depth telling the average distance from a point to its 4th nearest neighbor in a planar point cloud obtained by backprojection of a depth map which has the same depth value in every pixel (e.g. z_m for point m).

The second phase in the depth map fusion does the actual fusion based on the depth dependent uncertainty of the measurements. The uncertainty is based on the measurement variances of the used depth sensor. That is, each measurement has a covariance matrix

$$\mathbf{C} = \begin{bmatrix} \lambda_1 \left(\frac{\beta_x z}{\sqrt{12}} \right)^2 & 0 & 0 \\ 0 & \lambda_1 \left(\frac{\beta_y z}{\sqrt{12}} \right)^2 & 0 \\ 0 & 0 & \lambda_2 (\alpha_2 z^2 + \alpha_1 z + \alpha_0)^2 \end{bmatrix},$$

which represent the location uncertainty of a point in x , y and z directions in the 3-D space as depth dependent variances. The parameters λ_1 , λ_2 , are used to scale the variances, β_x , β_y , define the width and height of a back projected pixel at one meter away from the sensor and α_2 , α_1 and α_0 describe a quadratic depth variance function [20]. The parameters were calibrated as described in [12] and [11]. The uncertainty \mathbf{C} defines an ellipsoid in the 3-D space which is aligned so that the z -axis is parallel to the line-of-sight, i.e. the line from the camera center to the point in the space.

Now, when fusing the depth maps, the uncertainties of overlapping points from different depth maps determine whether the points are merged together or not. If the points will be merged together, the new measurement is used to refine the location of the existing measurement. As shown in Figure 3, the measurement which seems to be more certain gets a bigger weight, i.e. the refined point \mathbf{p}'_e is nearer to the point with lower uncertainty ($d_e < d_n$). If there is no existing measurement near enough, the new measurement is simply added to the cloud of points.

Finally, the third phase filters out the points from the final point cloud appearing in locations which are unlikely based on the statistics collected during the fusion. The statistics consists of the number of merges and the visibility violations of every point. A visibility violation is a vote for a point to be an outlier. The bigger the number of merges the more certainly the point belongs to the reconstruction. Therefore, the points whose visibility violation count is bigger than the count of

merges are removed from the cloud at the end. That is, if two nearby points in the space, which were not merged together during the fusion, project into the same pixel in a depth map and both should be visible (i.e. their normals point toward the same half space where the camera is located) then the other point violates the visibility of the other, and thus, is more likely an outlier.

In this work, two points violate the visibility of each other if

$$\arccos(\mathbf{n}_e \bullet \mathbf{v}_e) < \frac{\pi}{2}, \quad \arccos(\mathbf{n}_n \bullet \mathbf{v}_n) < \frac{\pi}{2} \text{ and} \quad (2)$$

$$|s_e - s_n| < 0.1s_n \quad (3)$$

where \bullet is the dot product, \mathbf{n}_e and \mathbf{n}_n are the normals of the existing and new measurement, respectively, \mathbf{v}_e and \mathbf{v}_n are normalized vectors from the two points towards the camera and s_e and s_n are the distances between the camera center and the existing and new point, respectively.

B. Pose refinement via re-registration of the depth maps

In a generic reconstruction pipeline, the initial registration of depth maps or RGB images is based on solving the structure from motion (SfM) problem followed by a bundle adjustment. SfM is based on the tracking of movements of relatively sparse sets of feature points between images, and thus, its accuracy depends on the image content. Therefore, in complicated environments having repetitive textures, the initial registration usually has room for improvement.

As shown in Figure 2, our method incorporates a re-registration step into the reconstruction pipeline. The registration aligns the original depth maps with the last fused point cloud. It is based on the iterative closest point (ICP) algorithm, which iteratively refines the given extrinsic parameters of a depth camera to minimize the distance between the corresponding backprojected depth map and the fused point cloud. That is, let denote \mathbf{R}_i and \mathbf{t}_i the initial rotation and translation of the i th camera in relation to the global coordinate frame and $\mathbf{D}'_{xi} = \mathbf{R}_i^T \mathbf{D}_{xi} - \mathbf{R}_i^T \mathbf{t}_i$, the backprojected pixel x of depth map i in the global coordinate frame, where \mathbf{D}_{xi} are the coordinates of the corresponding pixel in the coordinate frame of the i th camera. Now, the method iteratively tries to find a rotation $\hat{\mathbf{R}}_i$ and a translation $\hat{\mathbf{t}}_i$ for each camera i so that the error between the points $\hat{\mathbf{R}}_i \mathbf{D}'_{xi} + \hat{\mathbf{t}}_i, \forall x$ and the points in the full fused point cloud is minimized [21]. After m iterations ($m = 10$ in our experiments), the camera poses are updated with $\mathbf{R}_i \leftarrow \hat{\mathbf{R}}_i^T \mathbf{R}_i$ and $\mathbf{t}_i \leftarrow \hat{\mathbf{R}}_i^T \mathbf{R}_i - \hat{\mathbf{R}}_i^T \hat{\mathbf{t}}_i$ and the fusion step is repeated as illustrated in Figure 2. The registration and fusion steps can be repeated until the result is satisfying or does not get significantly better between iterations. In our experiments, the pipeline was iterated six times. The re-registration was implemented with C++ using the Point Cloud Library¹ (PCL).

¹<http://pointclouds.org/>



Fig. 4. A sample image from each dataset used in the experiments. From left to right: CCORNER, OFFICE1 and OFFICE2.

IV. EXPERIMENTAL RESULTS

The experiments were made using three datasets (i.e. CCORNER, OFFICE1 and OFFICE2), captured with Kinect V2. Figure 4 illustrates a sample image from each dataset. The first dataset is a simple concave corner whereas the other two are rather complicated office environments. The results made with the proposed method were compared with the results made with the methods in [11] and [12]. The experiments include both visual and quantitative evaluations of the obtained results.

A. Visual evaluation of the results

In the first experiment, we compared the results acquired from OFFICE1 and OFFICE2 datasets. Figure 5 presents the point clouds made with [11], [12] and the proposed method. The figure clearly shows that, compared with [11], the method in [12] is able to reduce the amount of outliers but it cannot correct the registration errors. The registration errors cause ambiguous surfaces (green ellipses) and inaccurate object boundaries (red dashed ellipses), for examples, which do not appear in the results made with the proposed method, as shown on the right in Figure 5.

B. Quantitative evaluations

In the second experiment, the results obtained with the methods in [11], [12] and the proposed one were compared quantitatively in three ways. First, Table I presents the sizes of the used datasets and the sizes of the final point clouds. As shown, the proposed method decreases the number of points in the final point cloud.

Then, the reconstructions of CCORNER were evaluated in two ways by comparing them against a ground truth which consists of three orthogonal planes. In the first evaluation, the errors between the point clouds and the ground truth were obtained by calculating distances from every point to the nearest ground truth plane (i.e. floor, right wall or left wall). The errors are

TABLE I
AN OVERVIEW OF THE STATISTICS OF THE DATASETS.

Dataset	View count	Original point count	Method	Final point count	Ratio of reduction
CCORNER	59	9 307 296	[11]	1 299 555	86.0%
			[12]	939 730	89.9%
			Ours	881 994	90.5%
OFFICE1	98	16 690 662	[11]	5 930 663	64.5%
			[12]	4 352 962	73.9%
			Ours	4 252 937	74.5%
OFFICE2	114	20 400 588	[11]	6 777 222	66.7%
			[12]	5 221 117	74.4%
			Ours	4 956 266	75.7%

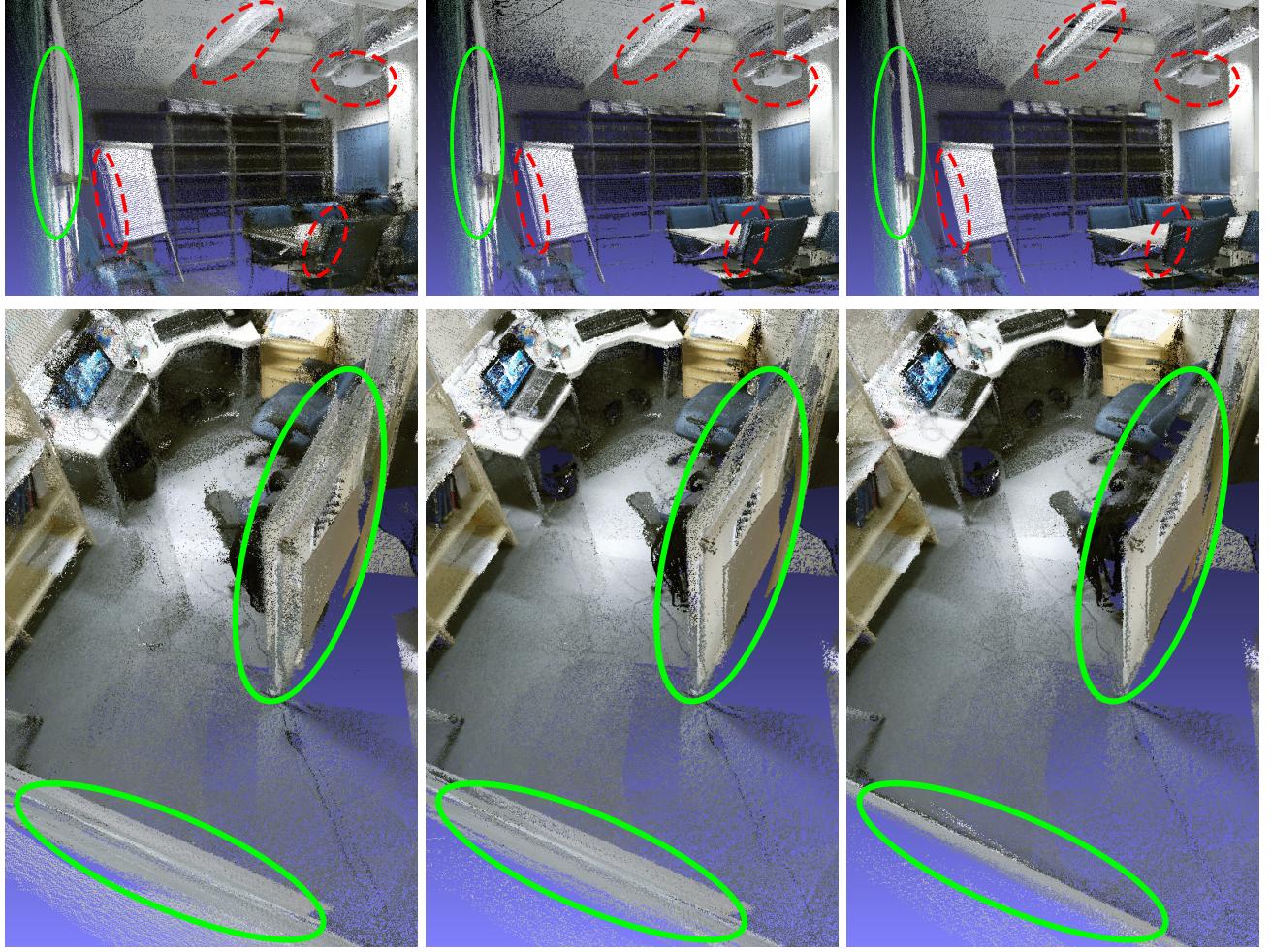


Fig. 5. Comparison between Office1 (top) and Office2 (bottom) results made with the methods in [11] (left) and [12] (middle) and with the method proposed in this paper (right). The proposed method significantly reduces the amount of registration errors which makes the surfaces less ambiguous (green ellipses) and the object boundaries more accurate (red dashed ellipses).

presented in Figure 6 as a cumulative curve where the value on the y-axis is a percentage of points whose error is smaller or equal to the corresponding value on the x-axis.

As shown in the figure, the proposed method enhances the accuracy of the model. That is, the proportions of points whose error is below or equal to 0, 0.1 or 0.2, are clearly bigger than the corresponding values of the other two methods.

Then, CCorner point clouds were evaluated using the voxel based evaluation metric proposed in [22]. The evaluation values are presented in Table II. The evaluation values indicate the completeness and compactness of the reconstructions. The completeness is defined with Jaccard index which indicates the proportion of the ground truth which is covered by the reconstruction within a certain threshold. Jaccard index is calculated by comparing the voxel representation of the reconstruction, and thus, the threshold, mentioned above, is the size of the voxel (i.e. length of an edge of a voxel). The compactness, instead, is a compression ratio calculated as a ratio of the number of the points in the ground truth and the reconstruction.

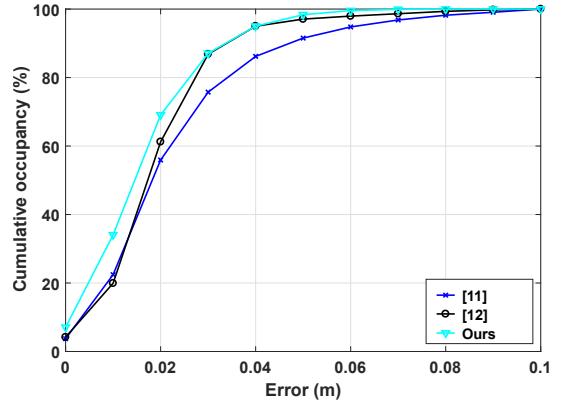


Fig. 6. Evaluation of the leftover errors in the CCorner reconstructions made with [11], [12] and the proposed method.

As the table shows, the point cloud achieved with the proposed method is both more compact (bigger compression ratio) and complete (bigger Jaccard index regardless of the size of a voxel) than the reconstructions made with [11] and [12].

TABLE II
EVALUATION OF COMPLETENESS (JACCARD INDEX) AND COMPACTNESS (COMPRESSION RATIO) OF CCORNER RECONSTRUCTIONS [22]. SEE THE TEXT FOR THE DETAILS.

		Method		
		[11]	[12]	Ours
Compression ratio		0.443	0.612	0.652
Jaccard index with voxel size	5mm	0.027	0.026	0.045
	20mm	0.162	0.174	0.190
	45mm	0.309	0.350	0.355
	85mm	0.388	0.440	0.456

C. Discussion

The results showed that the proposed method produced significantly better results than its predecessors. The visual comparisons indicated that the point clouds made with the proposed method are cleaner and less ambiguous than those made with [11] and [12]. Although the method in [12] is able to remove the majority of the incorrect depth measurements, which appear in [11], it cannot properly handle points which have been registered incorrectly. Therefore, the re-registration extension is an important improvement to the reconstruction pipeline.

The re-registration of the proposed method improves the registration of a depth map if at least part of its points overlap with points in other depth maps so that they have been fused together during the fusion phase. In the fusion phase, as described earlier, a new measurement is used to refine a nearby existing point if there is any or it is added to the cloud otherwise. If most of the measurements of a depth map are only added to the cloud, the re-registration is not able to properly refine the alignment of the depth map because the error between the added points and the original map is zero. However, such situations could be avoided by capturing the scene more carefully, and thus, ensuring that the depth maps have enough redundancy and the initial camera poses can be calculated relatively well.

The quantitative evaluations showed that the reconstructions obtained with the proposed method contained less points than the reconstructions made with [11] and [12]. Nevertheless, the completeness of the models even increased slightly. That is, the proposed method is able to decrease the redundancy without decreasing the completeness.

V. CONCLUSION

In this paper, we proposed a method for depth map fusion. The proposed method merges a sequence of depth maps into a single non-redundant point cloud. The fusion pipeline consists of the actual depth map fusion and a re-registration phase which are iterated until the result is satisfying or does not change significantly. The fusion phase gets the depth maps and corresponding camera poses as input and produces a non-redundant point cloud. The re-registration phase instead, tries to refine the original poses of the cameras by the registration of the original backprojected depth maps into the fused point cloud. Then, the depth maps and refined camera poses are fed again to the fusion phase. The experiments showed that

the proposed method is able to produce more accurate and unambiguous reconstructions than its predecessors.

REFERENCES

- [1] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-view stereo reconstruction algorithms," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006, pp. 519–528.
- [2] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multi-view stereopsis," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 32, no. 8, pp. 1362–1376, 2010.
- [3] S. Fuhrmann and M. Goesele, "Floating scale surface reconstruction," *ACM Transactions on Graphics (TOG)*, 2014.
- [4] M. Ylimäki, J. Kannala, J. Holappa, S. S. Brandt, and J. Heikkilä, "Fast and accurate multi-view reconstruction by multi-stage prioritised matching," *IET Computer Vision*, vol. 9, no. 4, pp. 576–587, 2015.
- [5] N. Richard A., I. Shahram, H. Otmar, M. David, K. David, D. Andrew J., K. Pushmeet, S. Jamie, H. Steve, and F. Andrew, "Kinectfusion: Real-time dense surface mapping and tracking," in *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, October 2011, pp. 127–136.
- [6] T. Whelan, M. Kaess, F. Maurice, H. Johannsson, J. Leonard, and J. McDonald, "Kintinuous: Spatially extended kinectfusion," Tech. Rep., 2012.
- [7] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger, "Real-time 3d reconstruction at scale using voxel hashing," *ACM Transactions on Graphics (TOG)*, 2013.
- [8] P. Merrell and et al., "Real-time visibility-based fusion of depth maps," in *IEEE International Conference on Computer Vision (ICCV)*, 2007.
- [9] J. Li, E. Li, Y. Chen, L. Xu, and Y. Zhang, "Bundled depth-map merging for multi-view stereo," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2010.
- [10] E. Tola, C. Strecha, and P. Fua, "Efficient large-scale multi-view stereo for ultra high-resolution image sets," *Machine Vision and Applications*, vol. 23, no. 5, pp. 903–920, 2012.
- [11] T. Kyöstiä, D. Herrera C., J. Kannala, and J. Heikkilä, "Merging overlapping depth maps into a nonredundant point cloud," in *Scandinavian Conference on Image Analysis (SCIA)*, June 2013, pp. 567–578.
- [12] M. Ylimäki, J. Kannala, and J. Heikkilä, "Robust and practical depth map fusion for time-of-flight cameras," in *Scandinavian Conference on Image Analysis (SCIA)*, June 2017, pp. 122–134.
- [13] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson surface reconstruction," in *Eurographics Symposium on Geometry Processing*, 2006.
- [14] M. Goesele, B. Curless, and S. M. Seitz, "Multi-view stereo revisited," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.
- [15] C. Zach, T. Pock, and H. Bischof, "A globally optimal algorithm for robust $TV-L^1$ range image integration," in *IEEE International Conference on Computer Vision (ICCV)*, 2007.
- [16] H. Roth and M. Vona, "Moving volume kinectfusion," in *British Machine Vision Conference*, 2012.
- [17] S. Choi, Q. Y. Zhou, and V. Koltun, "Robust reconstruction of indoor scenes," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 5556–5565.
- [18] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '96. New York, NY, USA: ACM, 1996, pp. 303–312.
- [19] N. Naik, A. Kadambi, C. Rhemann, S. Izadi, R. Raskar, and S. B. Kang, "A light transport model for mitigating multipath interference in time-of-flight sensors," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2015, pp. 73–81.
- [20] D. Herrera C., J. Kannala, and J. Heikkilä, "Joint depth and color camera calibration with distortion correction," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 34, no. 10, pp. 2058–2064, 2012.
- [21] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *Image Vision Comput.*, vol. 10, no. 3, pp. 145–155, 1992.
- [22] M. Ylimäki, J. Kannala, and J. Heikkilä, "Optimizing the accuracy and compactness of multi-view reconstructions," in *International Conference on Computer Analysis of Images and Patterns (CAIP)*, September 2015, pp. 171–183.