Robust and Practical Depth Map Fusion for Time-of-Flight Cameras

Markus Ylimäki[1], Juho Kannala[2] and Janne Heikkilä[1]

[1]Center for Machine Vision Research, University of Oulu, Finland
firstname.lastname@ee.oulu.fi
[2]Department of Computer Science, Aalto University, Finland
juho.kannala@aalto.fi

Abstract. Fusion of overlapping depth maps is an important part in many 3D reconstruction pipelines. Ideally fusion produces an accurate and nonredundant point cloud robustly even from noisy and partially poorly registered depth maps. In this paper, we improve an existing fusion algorithm towards a more ideal solution. Our method builds a nonredundant point cloud from a sequence of depth maps so that the new measurements are either added to the existing point cloud if they are in an area which is not yet covered or used to refine the existing points. The method is robust to outliers and erroneous depth measurements as well as small depth map registration errors due to inaccurate camera poses. The results show that the method overcomes its predecessor both in accuracy and robustness.

Keywords: Depth map merging, RGB-D reconstruction

1 Introduction

Merging partially overlapping depth maps into a single point cloud is an essential part of every depth map based 3-dimensional (3D) reconstruction software. A simple registration of depth maps may lead to a huge number of redundant points even with relatively small objects. That will make the further processing very slow.

The amount of points could be reduced afterwards by simplifying the cloud but it is more reasonable to aim directly at a nonredundant point cloud. This will save both time and needed memory capacity.

In this paper, we further develop a method which merges a sequence of depth maps into a single nonredundant point cloud [7]. The method takes the measurement accuracy of obtained depths into account and merges nearby depth measurements into a single point in 3D space by giving more weight to the more certain measurement. Thus, only those points that do not have other neighbouring points are added to the cloud. The proposed method significantly reduces the amount of outliers in the depth maps and rejects incorrectly measured or badly registered points.

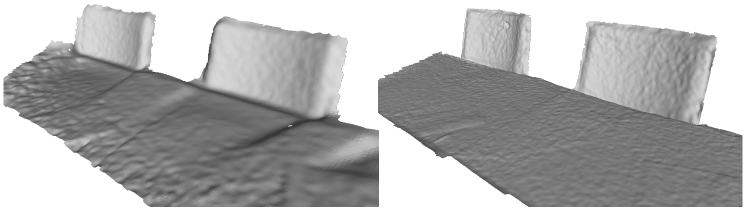

Fig. 1. Illustration of the multipath interference error in time-of-flight cameras. Left: A Poisson reconstructed surface [6] created from a point cloud which was back projected from a single depth map. Right: the same surface part but now created from the output point cloud of the proposed method.

One major issue in time-of-flight cameras, such as Kinect V2, is the problem called multipath interference [13]. It occurs when the depth sensor receives multiple scattered or reflected signals from the same direction and causes a positive bias to the depth measurements. As illustrated in the left part of Figure 1, the problem especially occurs in concave corners, which in this case are formed by the table and the backrests of the chairs. Our method, proposed in this paper, is able to correct those errors as shown in the right part of Figure 1.

2 Related work and our contributions

Fusion of depth maps from the aspect of 3D reconstruction has been studied widely during recent years [4, 11, 21, 9, 18]. The most relevant work regarding to our work is the one presented in [11]. There, the authors proposed a depth map fusion method which is capable of building 3D reconstructions from live video in real time. The method is designed for passive stereo depth maps, and thus, does not use uncertainties for depth measurements.

Since the release of Microsoft Kinect, the interest in the real-time reconstruction has increased widely. These methods mostly represent the models as voxels [16, 19, 17, 14, 1] which means that their resolution is limited by the available memory. However, this restriction is successfully avoided especially in [14] but this method is designed for live video, and therefore, it may not work that well with wide baseline depth maps. Choi et al [1] have also achieved impressive results recently. In their method, the loop closures play a significant role which have to be taken into account when capturing the data. The voxel based approach is also used in [3] in the merging of depth maps with multiple scales but the depth maps were acquired with a range scanner or with a multi-view stereo system.

Kyöstilä et al proposed a method where the point cloud is created iteratively from a sequence of depth maps so that the added depth maps do not increase the redundancy of the cloud [7]. That is, starting with a point cloud, back projected from a single depth map, the method either creates new points to the cloud from other depth maps if they are in an area which has not yet been covered by other points or uses the new measurements to refine the existing points. The refinement merges nearby points by giving more weight to measurements that have lower empirical, depth dependent variances.

However, Kyöstilä's method is mainly designed for merging redundant depth maps and it cannot handle outliers. In addition, the method was designed for the first generation Kinect device (Kinect V1), and thus, it does not take all the characteristics of the newer Kinect device (Kinect V2) into account. These differences and our solutions are discussed in more detail in Section 2.1.

2.1 Our contributions

As described in Section 2, the method in [7] cannot handle outliers and does not work properly with Kinect V2. Regarding to our method, the most essential difference between the Kinect devices is the depth measuring technique. Kinect V1 calculates the depths using an infrared dot pattern projected into the space, whereas Kinect V2 is based on time-of-flight (ToF) technique and predicts the depths from the phase shift between an emitted and received infrared signals [15]. Generally, the measurements acquired with Kinect V2 are more accurate, but in certain cases the sensor might receive multiple reflected or scattered signals from the same direction which might cause significant measurement errors as presented in Figure 1. This multipath interference problem [13] is not taken into account in [7].

Thus, in this paper we propose three extensions to the method in [7] to overcome its weaknesses. The extensions provide three different ways to measure the errors which occur in ToF measurements and our method tries to replace and refine the erroneous points with more accurate measurements from other redundant depth maps. That, is the contributions of this paper are

1. pre-filtering of depth maps to reduce the amount of outliers,
2. improved uncertainty covariance to compensate for the measurement variances and make the method more accurate and
3. filtering of the final point cloud based on a simple visibility violation rule to reduce the amount of erroneous and badly registered measurements due to the multipath interference [13] and incorrect camera poses, respectively.

The experiments show that the extensions significantly improve the results when compared with [7] which make the proposed method a potential post-processing step for methods like ORB-SLAM [12] or [2]. In addition, the nonredundant point clouds produced with the proposed method can be further transformed into a mesh, like e.q. in [11, 1], using [6] or [8] for example.

3 Method

As presented in Figure 2, the proposed method takes a set of depth maps and calibrated RGB images with known camera poses as input and outputs a point cloud. The method improves the algorithm described in [7] with three extensions which are marked with darker boxes in Figure 2. Similarly to [7], our method

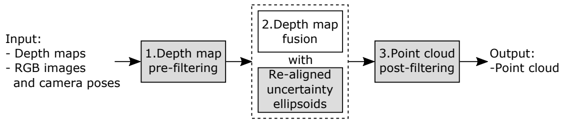

Fig. 2. An overview of the proposed fusion pipeline. In this paper, we propose three extensions (rectangles with a grey background) to the fusion algorithm in [7].

can be used as a pipeline to process one depth map at a time and therefore the only thing that limits the size of the reconstruction is the available memory for storing the created point cloud.

The pipeline consists of three steps: 1) depth map pre-filtering, 2) actual depth map fusion with re-aligned uncertainty ellipsoids and 3) post-filtering of the final point cloud. The steps are described in more detail in the following sections. Section 3.1, describes the pre-filtering step. The re-aligned uncertainty ellipsoids extension is described together with the fusion step in Section 3.2 and the filtering of the final point cloud is presented in Section 3.3.

3.1 Pre-filtering of depth maps

Typically, backprojected Kinect depth maps (both V1 and V2) have outliers or inaccurate measurements near depth edges and near the corners of the depth image. Usually, their distances to the nearest neighbouring points are much above the average. To remove such measurements from the depth maps, we first calculate a reference curve which describes the average distance from a point to its nth nearest neighbour (NN) ($n = 4$ in all our experiments) in the 3D space at a certain depth. The left part of Figure 3 presents the calculation of a reference distance at depth d_z for one pixel. The final reference distance at depth d_z is the average of such distances of all pixels. The average distances are calculated for depths from 0.5m to 4.5m with 0.1m interval and the reference curve (blue solid line in the right sub figure) is then acquired by fitting a line to these values.

Now in the pre-filtering, the distance d_m to the 4th nearest neighbour is calculated for every pixel in the input depth map and compared with the reference value d_{ref} at the same depth. The measurement is removed as an outlier if

$$d_m > \frac{d_{ref}}{\sqrt{0.3}}. \tag{1}$$

The red dashed line in the right part of Figure 3 illustrates the equation. That is, the measurements whose distance value is above the line are removed.

3.2 Improved depth map fusion

The actual depth map fusion is based on [7] with two exceptions: 1) the device dependent parameter values were calibrated for Kinect V2 and 2) the orientations of uncertainty ellipsoids were improved to match with the ToF measuring technique. The details are described later in the section.

That is, starting with an initial point cloud, backprojected from a single depth map, the next depth maps are merged with the existing cloud so that the new measurements are either added to cloud if there is no other points nearby or used to refine the existing measurements without increasing the point count. As described in Section 2, the refinement gives more weight to the measurement with lower empirical variance, i.e. uncertainty. The uncertainty of a measurement is described as a covariance $\mathbf{C}$ which determines the location uncertainty of the measurement in x, y and z directions as depth dependent variances

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

where z is the measured depth and λ_1, λ_2, β_x, β_y, α_2, α_1 and α_0 are parameters which were calibrated for Kinect V2 using the approach presented in [7].

The covariance matrix corresponds to an ellipsoid in the 3D space and in [7] it is aligned so that the z-axis of the ellipsoid is parallel to the optical axis of the camera. However, as described in Section 2.1, Kinect V2 measures the depth by comparing the phase shift between the emitted and received signals which travel to the sensor along the line of sight. Therefore, in the proposed method, the covariance ellipsoids are aligned parallel to the line of sights, which means that their orientations depend on the locations of the measurements in the original depth maps. That is, given the rotations $\mathbf{R}$ between the world frame and the camera coordinate frame and $\mathbf{R}_{los}$ between the optical axis of the camera and the line of sight, the covariance $\mathbf{C}$ can be expressed in the world frame with

$$\mathbf{C}_{world} = \mathbf{R}^T \mathbf{R}_{los}{}^T \mathbf{C} \mathbf{R}_{los} \mathbf{R} \tag{3}$$

As in [7], an existing measurement is refined by the new measurement nearby. First, the refined location is calculated using the best linear unbiased estimator (BLUE) [10], which gives

$$\mathbf{p}'_e = \mathbf{p}_e + \mathbf{C}'_e \mathbf{C}_n{}^{-1}(\mathbf{p}_n - \mathbf{p}_e), \tag{4}$$

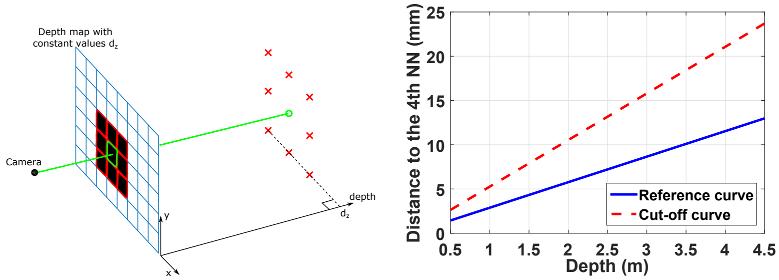

Fig. 3. Illustration of the calculation of the reference distances (left) and the reference and cut-off curves (right). The reference distance for the backprojected pixel (green circle) is the distance to its 4th nearest neighbour. The value of reference curve (blue solid line) at the depth d_z is the average of such distances of all backprojected pixels at the depth d_z. The pre-filtering removes points whose distance to the 4th nearest neighbour is above the cut-off curve (red dashed line).

where $\mathbf{p}_e$ is the location estimation of the existing point which has been added to the cloud earlier, $\mathbf{p}_n$ is the new measurement with the covariance $\mathbf{C}_n$ and $\mathbf{C}'_e$ is the covariance of the refined point defined by

$$\mathbf{C}'_e = (\mathbf{C}_e{}^{-1} + \mathbf{C}_n{}^{-1})^{-1}, \tag{5}$$

where $\mathbf{C}_e$ is the covariance of the existing measurement estimation.

Now, the Mahalanobis distances d_1 and d_2 between $\mathbf{p}'_e$ and $\mathbf{p}_e$ and $\mathbf{p}'_e$ and $\mathbf{p}_n$, respectively, are calculated using the corresponding covariances

$$d_1 = \sqrt{(\mathbf{p}'_e - \mathbf{p}_e)\mathbf{C}_e{}^{-1}(\mathbf{p}'_e - \mathbf{p}_e)} \tag{6}$$

$$d_2 = \sqrt{(\mathbf{p}'_e - \mathbf{p}_n)\mathbf{C}_n{}^{-1}(\mathbf{p}'_e - \mathbf{p}_n)} \tag{7}$$

If both distances are below the threshold τ, the existing estimate is updated with

$$\mathbf{p}_e \leftarrow \mathbf{p}'_e \quad \text{and} \quad \mathbf{C}_e \leftarrow \mathbf{C}'_e. \tag{8}$$

3.3 Post-filtering of the final point cloud

If in the refinement part of the fusion, at least one of the distances d_1 and d_2 (Eqs. (6) and (7), respectively) is bigger than the threshold τ, the existing measurement is not updated but the measurements might violate the visibility of each other depending on their locations. To solve possible visibility violations, we need normals for every point. The normals are estimated by a plane fitted to the k nearest neighbours of the point ($k = 50$ in all our experiments) in the original back projected depth map.

In this paper, we consider three alternatives, illustrated in Figure 4, how the measurements may locate with respect to each other. In the first case, point **A** occludes point **B** but they are far away from each other so that is not a visibility violation. Next, the point **C** is occluding point **D** nearby but this time the normal of measurement **D** is not pointing towards the half space where the camera under consideration is located, and therefore, this is not a visibility violation either. In the third case, the point **E** occludes the nearby point **F** whose normal is towards the camera. In this case, there is a visibility violation because it is very unlikely that both of these measurements really exist in the scene. In practice, the points are near enough when the distance between them is less than 10% of the depth of the new measurement. This kind of violation may occur due to the inaccuracy of the camera poses or calibration, noise or the multipath interference.

The post-filtering consists of two parts. The first part is built-in to the depth map fusion and it collects some point-wise statistics which are utilized in the second part that does the actual filtering after the fusion. The statistics are two values which record the number of merges and the number of visibility violations.

That is, if two points that project onto the same pixel are not close enough to be merged together but still violate the visibility of each other in the 3D space, either the existing measurement or the new one is probably an outlier or too

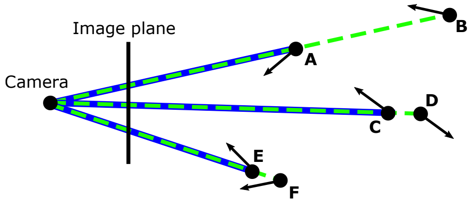

Fig. 4. Three alternatives considered in this paper how the points that project onto the same pixel may locate in the 3D space. The lowest case is the only one causing a visibility violation between points because the points are nearby and their normals point towards the same half space where the camera under consideration is located.

inaccurate to be added to the final cloud of points. If the existing measurement has already been merged with another point more than once, it can be considered more reliable and the visibility violation value of the new measurement is incremented by one. Otherwise, the reliability is based on an unreliability weight $w = (1/cos(\alpha))^2$, where α is the angle between the line of sight and the normal of the point, i.e. the bigger the angle the more unreliable the point is and the violation value of the more unreliable measurement is incremented.

Finally in the second part when the fusion has stopped, the points whose visibility violation count is bigger than the value which measures the count of merges, are removed from the cloud.

4 Experiments

The experiments were carried out using three data sets captured with Kinect V2: CCorner, Office1 and Office2. The last two are complicated office environments whereas the first one is a simple concave corner bounded by floor and two walls. Figure 5 presents a sample image of each data set. The checker boards on CCorner data set were used to acquire the poses of the cameras as well as to create a ground truth for quantitative evaluation. The data sets consist of RGB images and depth maps and they were captured with Kinect by moving the device around the room and holding it still while capturing. The sets were captured so that the depth maps had redundant measurements and sequential RGB images had common areas with rich texture in order to gain as good camera pose estimations as possible as described below.

Kinect device was calibrated using the method in [5] which was slightly modified in order to use it with Kinect V2. In the office data sets, the camera poses were obtained via structure from motion using VisualSFM[1]. The calibration parameters, the depth maps and the sparse point cloud, produced by SfM, were used to set the scale of the obtained poses to match with the metric system used by the depth sensor of Kinect V2. The poses, the depth maps and the RGB images were then fed to the algorithm pipeline.

The method in [7] was used as a baseline in the evaluations. The results presented in the following sections show step by step how each extension iter-

[1] http://ccwu.me/vsfm/

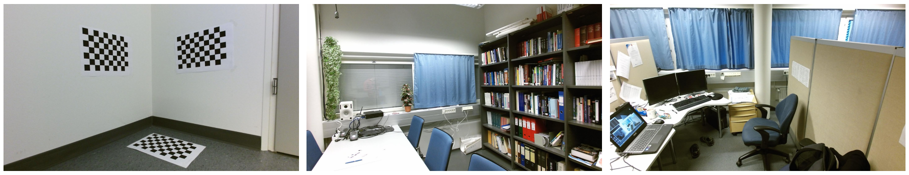

Fig. 5. Sample images of the data sets used in the experiments. From left to right: CCorner, Office1 and Office2.

atively enhances the results made with the baseline algorithm. In Section 4.1, we present the enhancement achieved with pre-filtering. Then, Section 4.2 compares the results produced by our method and the baseline extended with the pre-filtering, and finally, in Section 4.3 the influence of every extension, including the pre-filtering (PRF), re-aligned covariances (RAC) and post-filtering (POF), is shown by three quantitative analyses.

4.1 Depth map pre-filtering

Figure 6 illustrates the pre-filtering result on a single depth map. As expected, the filtering removes measurements near depth edges and image corners where the outliers typically exist or the measurement accuracy is worse due to the lens distortion. In addition, the filtering removes points in darker areas where measurement noise is bigger and on surfaces whose normal create too big angle with the optical axis of the camera and thus are unreliable. The redundancy in the input depth maps guarantees that the removed points does not make holes in the final point cloud.

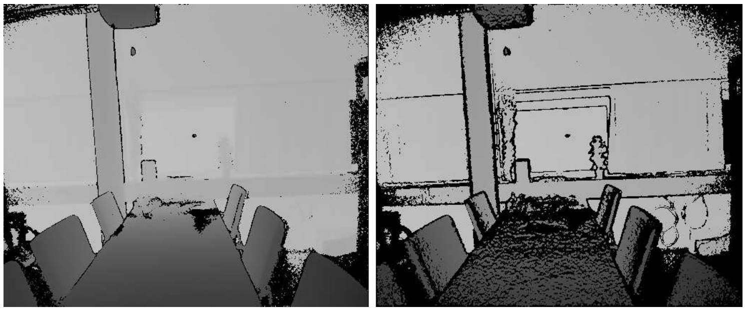

Fig. 6. An illustration of the depth map pre-filtering. Left: the original depth map, right: the filtered depth map. The filter removes incorrect or inaccurate measurements near depth edges and image corners. In the fusion, the holes are filled with more accurate points from other depth maps.

In Figure 7, we present the results of Office2 data set after the depth map fusion made with the baseline algorithm and the baseline extended with the pre-filtering. In the figure, the improvement is clearly visible. The filtering has removed a vast majority of the outliers around the laptop as well as from the air beyond the wall (the green solid ellipses) for example. The filtering also improves the details in the view like the area in front of the two computer screens (the red dashed ellipse).

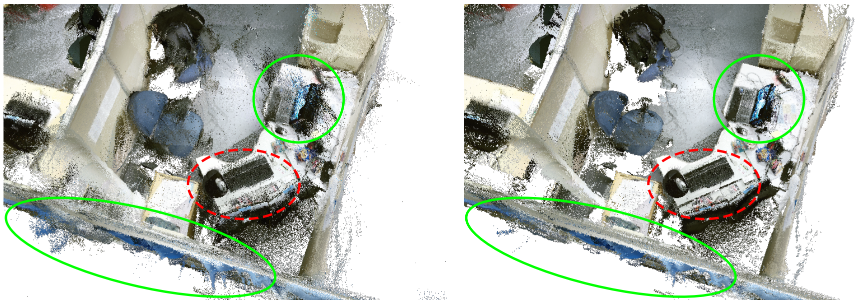

Fig. 7. Comparison between Office2 results made with the baseline algorithm without (left) and with the depth map pre-filtering. The filtering removes a great number of outliers around the laptop and from the air beyond the wall (the green solid ellipses). Pre-filtering also enhances the visibility of the details (the red dashed ellipse).

4.2 Re-aligned covariances and post-filtering

Figure 8 shows the comparison of the results made with the baseline method extended with the pre-filtering and the proposed method. Our method is able to remove the outliers between the backrests of the chairs and the table as shown in the top part of the figure (green rectangles), but as the bottom part of the figure illustrates, the method is also able to remove the incorrect measurements under the table (red dashed ellipses) and the misplaced measurements above (green solid ellipses). The incorrect measurements below the table have suffered from the multipath interference via backrest of the chair and Kinect had obtained too long distances for those measurements (cf. Figure 1). The misplaced measurements above the table exists due to an inaccurate pose of the camera where the measurements originate from.

The noise on the measured surfaces is usually parallel to the line of sight and therefore, by re-aligning the covariance also parallel to that line helps the refinement to move the point in the right direction. The post-filtering instead handles the overlapping points by preserving the one which seems to be more reliable. For example, if the same surface is captured in two ways; first so that the surface in perpendicular to the camera and second so that the surface is slanted and may more probably suffer from multipath interference, the perpendicularly measured points remain in the cloud and others are removed.

4.3 Quantitative analyses

In the last experiment, the methods and extensions were tested against each other with three quantitative analyses. First, Table 1 illustrates an overview of the sizes of the used data sets and the sizes of the final results. The abbreviations PRF, RAC and POF refer to the proposed extensions to the baseline method, i.e. pre-filtering, re-aligned covariances and post-filtering, respectively. As the table shows, every extension increases the ratio of reduction of the point count.

Then the results from CCorner data set were compared against the ground truth consisting three planes defined by the backprojected checker board corners. The extrinsic parameters of the cameras were obtained by the non-linear

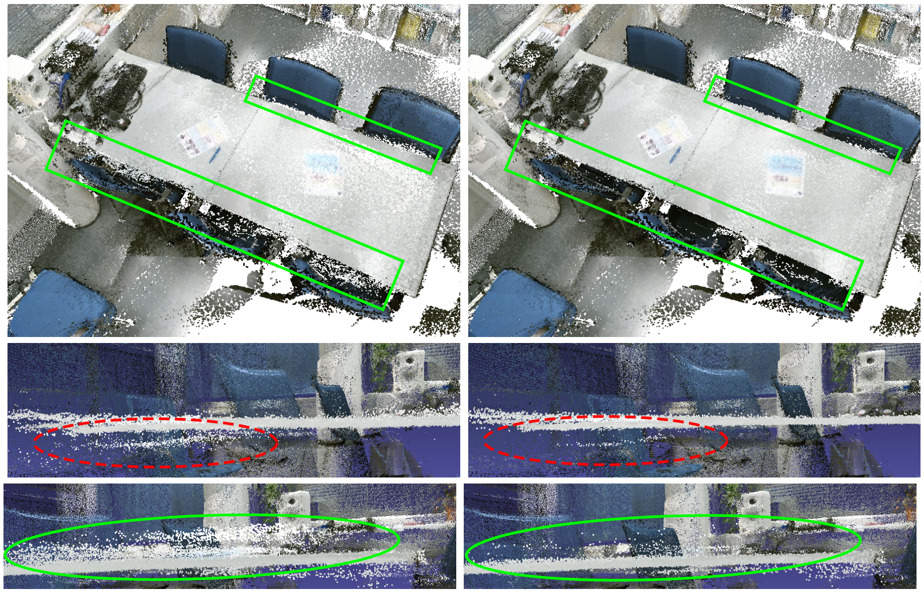

Fig. 8. Comparison between Office1 results made with the baseline algorithm with pre-filtering extension (left) and with the proposed method (right). The proposed method is able to significantly decrease the amount of outliers between the table and the backrests of the chairs (green rectangles) as well as incorrect measurements below the table (red dashed ellipses) and misplaced measurements above (green solid ellipses).

optimization where the errors between the detected and projected checker board corners were minimized while the intrinsic parameters of the camera were kept constant. Now for each fusion result, the distances from the points to the nearest plane (floor, right wall or left wall) were calculated and presented as cumulative error curves shown on the left in Figure 9. The value on the y-axis is the percentage of points whose error is below the value on the x-axis. 100% contains all the points in each fused point cloud (absolute point counts are listed in Table 1).

As shown in the left sub figure, each extension enhances the accuracy of the fusion. Especially the re-aligned covariance extension significantly improves the result (the red square curve versus the green diamond curve). Pre-filtering and post-filtering bring only moderate improvement in this data set because, due to the simplicity of the data set, the amount of outliers is moderate and practically there are no badly misplaced measurements because the camera poses were acquired relatively accurately as described earlier.

Table 1. An overview of the sizes of used data sets and achieved point reduction ratios.

Dataset	**CCorner**				**Office1**				**Office2**			
Method	[7]	PRF	PRF+RAC	PRF+RAC+POF	[7]	PRF	PRF+RAC	PRF+RAC+POF	[7]	PRF	PRF+RAC	PRF+RAC+POF
View count	59				98				114			
Original point count	9 307 296				16 690 662				20 400 588			
Final point count	1 299 555	1 123 701	955 161	939 730	5 930 663	4 533 418	4 408 269	4 352 962	6 777 222	5 578 252	5 382 525	5 221 117
Ratio of reduction	86.0%	87.9%	89.7%	89.9%	64.5%	72.8%	73.6%	73.9%	66.7%	72.7%	73.6%	74.4%

The right part of Figure 9 was produced with the voxel based evaluation method presented in [20]. The figure illustrates the coverage and the compactness of the reconstructions. The coverage is presented as Jaccard index indicating the proportion of the ground truth which is covered by the reconstruction within a certain threshold. The coverage value is calculated between the voxel representations of the ground truth and the reconstruction so the above-mentioned threshold is the width of a voxel edge. The compactness is presented as a compression ratio which is the ratio of the number of points in the ground truth and the reconstruction. Now, one can see from Figure 9 that the completeness of the result made with the proposed method is at least equal to that of the baseline method depending on the width of a voxel while the compression ratio is clearly better. That is although the pre-filtering may also have removed some possible correct points on slanted surfaces, that did not make any holes in the reconstruction.

5 Conclusion

In this paper, we proposed a method for merging a sequence of overlapping depth maps into a single non-redundant point cloud. Starting with a point cloud back projected from a single depth map, the method iteratively adds points from other depth maps so that the new measurements refine the existing points in overlapping areas. The refinement is based on an uncertainty covariance calculated for every measurement. The proposed method improves the algorithm [7] with three extensions: 1) depth map pre-filtering, 2) depth map fusion with directed uncertainty covariances and 3) post-filtering of the final point cloud. The performance of each extension was demonstrated with several experiments. The proposed method outperformed the baseline algorithm both in robustness and accuracy.

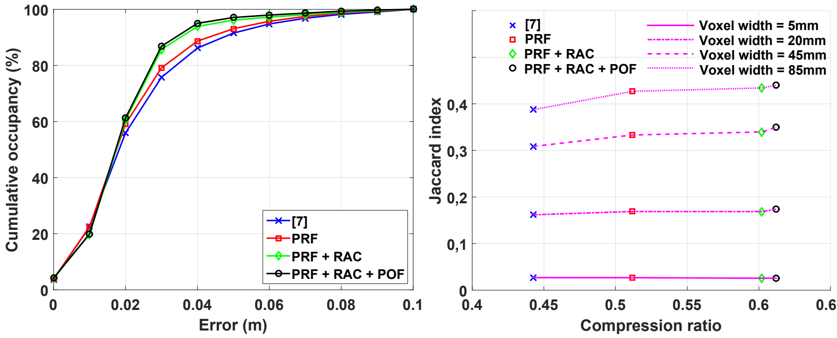

Fig. 9. Evaluation of the leftover errors after the fusion pipeline (left) and evaluation of the coverage (Jaccard index) and compactness (compression ratio) of the reconstructions (right) [20]. PRF, RAC and POF refer to the proposed extensions, i.e. pre-filtering, re-aligned covariances and post-filtering, respectively. Jaccard index indicates the proportion of the ground truth which is covered by the reconstruction within the certain threshold represented by the width of a voxel. Compression ratio is the ratio of the number of points in the ground truth and the reconstruction.

References

1. Choi, S., Zhou, Q.Y., Koltun, V.: Robust reconstruction of indoor scenes. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 5556–5565 (2015)
2. Córdova-Esparza, D.M., Terven, J.R., Jiménez-Hernández, H., Herrera-Navarro, A.M.: A multiple camera calibration and point cloud fusion tool for kinect v2. Inpress: Science of Computer Programming (2017)
3. Fuhrmann, S., Goesele, M.: Fusion of depth maps with multiple scales. In: Proceedings of the 2011 SIGGRAPH Asia Conference. pp. 148:1–148:8. ACM (2011)
4. Goesele, M., Curless, B., Seitz, S.M.: Multi-view stereo revisited. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2006)
5. Herrera C, D., Kannala, J., Heikkilä, J.: Joint depth and color camera calibration with distortion correction. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 34(10), 2058–2064 (2012)
6. Kazhdan, M., Bolitho, M., Hoppe, H.: Poisson surface reconstruction. In: Eurographics Symposium on Geometry Processing (2006)
7. Kyöstilä, T., Herrera C., D., Kannala, J., Heikkilä, J.: Merging overlapping depth maps into a nonredundant point cloud. In: Scandinavian Conference on Image Analysis (SCIA). pp. 567–578 (June 2013)
8. Labatut, P., Pons, J.P., Keriven, R.: Robust and efficient surface reconstruction from range data. Computer Graphics Forum (CGF) 28(8), 2275–2290 (2009)
9. Li, J., Li, E., Chen, Y., Xu, L., Zhang, Y.: Bundled depth-map merging for multi-view stereo. In: IEEE Conf. on Computer Vision and Pattern Recognition (2010)
10. Mendel, J.: Lessons in Estimation Theory for Signal Processing, Communications and Control. Prentice Hall (1995)
11. Merrell, P., et al.: Real-time visibility-based fusion of depth maps. In: IEEE International Conference on Computer Vision (ICCV) (2007)
12. Mur-Artal, R., Montiel, J.M.M., Tardós, J.D.: ORB-SLAM: a versatile and accurate monocular SLAM system. IEEE Transactions on Robotics 31(5) (2015)
13. Naik, N., Kadambi, A., Rhemann, C., Izadi, S., Raskar, R., Kang, S.B.: A light transport model for mitigating multipath interference in time-of-flight sensors. In: IEEE Conf. on Computer Vision and Pattern Recognition. pp. 73–81 (2015)
14. Nießner, M., Zollhöfer, M., Izadi, S., Stamminger, M.: Real-time 3d reconstruction at scale using voxel hashing. ACM Transactions on Graphics (TOG) (2013)
15. Pagliari, D., Pinto, L.: Calibration of kinect for xbox one and comparison between the two generations of microsoft sensors. Sensors 15(11), 27569–27589 (2015)
16. Richard A., N., Shahram, I., Otmar, H., David, M., David, K., Andrew J., D., Pushmeet, K., Jamie, S., Steve, H., Andrew, F.: Kinectfusion: Real-time dense surface mapping and tracking. In: IEEE International Symposium on Mixed and Augmented Reality (ISMAR). pp. 127–136 (October 2011)
17. Roth, H., Vona, M.: Moving volume kinectfusion. In: British Machine Vision Conference (2012)
18. Tola, E., Strecha, C., Fua, P.: Efficient large-scale multi-view stereo for ultra high-resolution image sets. Machine Vision and Applications 23(5), 903–920 (2012)
19. Whelan, T., Kaess, M., Maurice, F., Johannsson, H., Leonard, J., McDonald, J.: Kintinuous: Spatially extended kinectfusion. Tech. rep. (2012)
20. Ylimäki, M., Kannala, J., Heikkilä, J.: Optimizing the Accuracy and Compactness of Multi-view Reconstructions, pp. 171–183 (September 2015)
21. Zach, C., Pock, T., Bischof, H.: A globally optimal algorithm for robust TV-L^1 range image integration. In: IEEE International Conference on Computer Vision (ICCV) (2007)