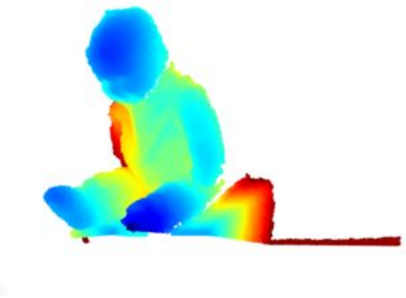
CIS II Journal Club

Group 15: 3D Reconstruction of Infants' Cranial Shape using Mobile Devices

David Shi

Project Summary

Create a software pipeline to reconstruct an accurate 3D model of a baby's head from depth information for accurate detection of cranial deformities



Depth Map

RGB Image

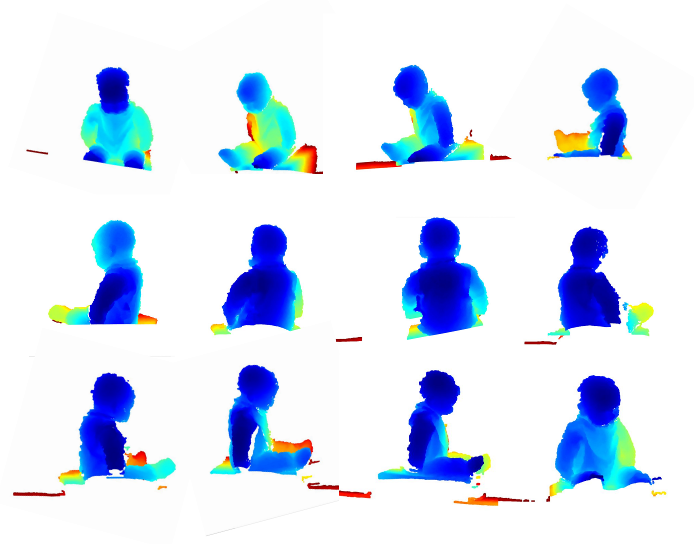
Figures by us

Background and Motivation

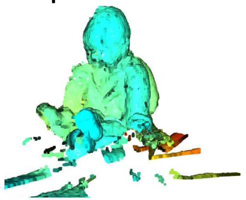
- With the rise of RGB-D cameras, RGB images and depth maps are easy to capture, making data for 3D reconstruction of a scene/object accessible
- Depth maps are like RGB images, except the intensity dimension is replaced with depth information
- 3D reconstruction of object/scene using RGB pictures and/or depth maps is a classic problem
- Fusion of these depth maps affects how the reconstruction looks, bad registrations can lead to bad fusions

Simple Method

- Consider all depth maps as point clouds
- Calculate the transformations between each consecutive cloud
- Register all depth maps/clouds into a single coordinate frame
- When everything is finished, fuse all depth maps together into one large point cloud
    - Is this the best idea?



Depth maps captured converted to point clouds

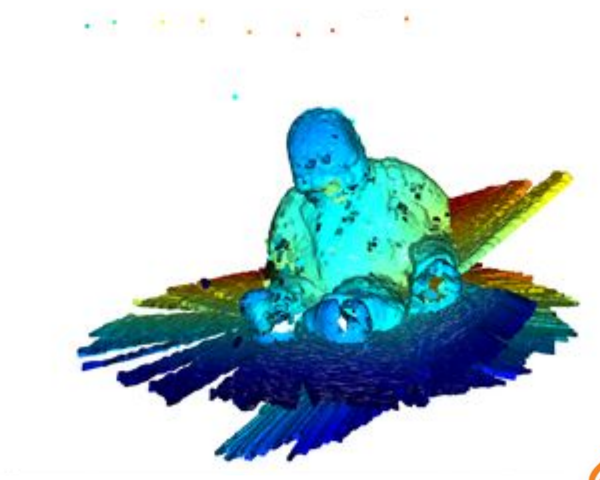


Fused Point Cloud

Figures by us

Problems with Simple Fusion

- Poor registrations of depth maps can increase outliers that impact reconstruction negatively
- Creates a huge amount of points in the point cloud that overlap: redundant points in point cloud
    - Slows down further processing



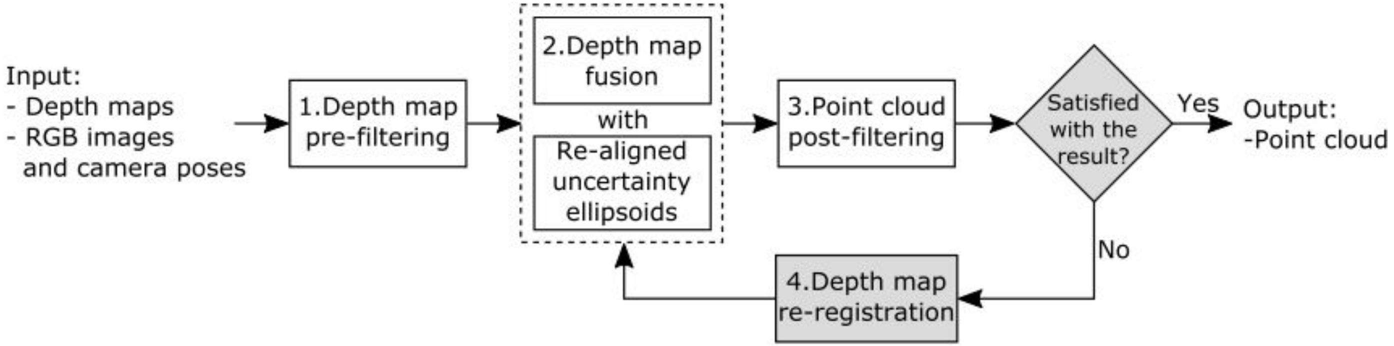
Single misaligned registration leads to poor reconstruction

Figures by us

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

Ylimäki, Markus, Janne Heikkilä, and Juho Kannala.

Paper Pipeline



- Contribution is re-registering the original depth maps back into the fused point cloud to refine camera poses
- Repeat the fusion step again

Figure from Ylimäki et al., 2018

1. Pre-filtering

- Assume that outlier depths are located in areas of low point density
- Establish a reference distance based on the average distance of a point to its 4th nearest neighbor
- Point is removed if its depth and nearest neighbors with respect to a reference distance is longer than a threshold

2. Depth Map Fusion with Uncertainty

- Calculate an ellipse of uncertainty around each depth point
- If the new point is outside of any ellipses, just add them to the point cloud
- If the new point is in one or more ellipses, combine the two or more measurements into a new point, weighted by the measurements

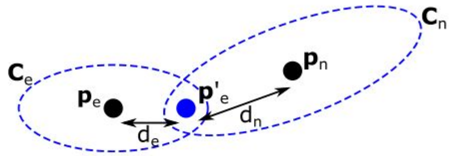


Fig. 3. Fusion of two nearby depth measurements. $\mathbf{C}_e$ and $\mathbf{p}_e$ are the covariance and location of the existing measurement. $\mathbf{C}_n$ and $\mathbf{p}_n$ are the covariance and location of the new measurement. These two measurements are merged together to a point p'_e if the merged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

Figure from Ylimäki et al., 2018

3. Post-filtering

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

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

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.

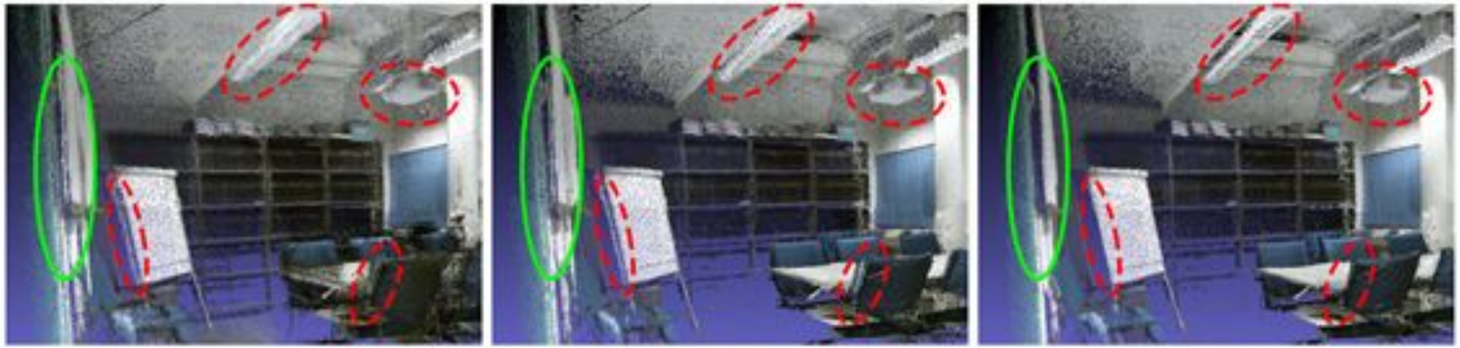
- Visibility violation
- Facing same direction
- Distances from camera are less than a threshold from each other

Figure from Ylimäki et al., 2018

4. New Contribution

- Align original depth maps with last fused point cloud
- Use iterative closest point (ICP) algorithm to align the original depth maps with the last fused point cloud to minimize the distance between the two
- Refine camera poses and repeat fusion
- Repeat as necessary until satisfied or nothing changes
- Basically refines the ICP with the fused depth map

Experiments: Qualitative

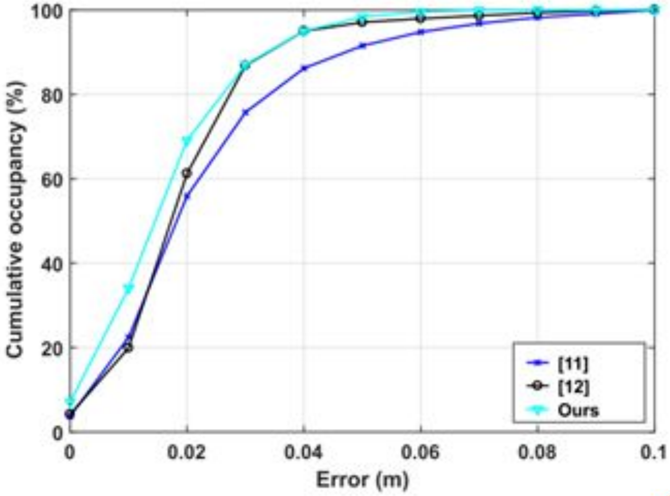


- The red highlighted areas where object boundaries were sharpened compared to baseline methods
- The green highlighted surfaces that were reconstructed better

Figure from Ylimäki et al., 2018

Experiments: Quantitative

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%



- Table showing reduction in points
- Graph showing error comparison to ground truth

Figure from Ylimäki et al., 2018

Thoughts

Good:

- Provides useful tricks to reduce the redundant points in a point cloud
    - Uncertainty around depth, visibility violation, and the re-registering are both interesting ideas to consider
    - Not too hard to implement

Bad:

- Might be slow, since we have to do the re-registering step each time
- No convergence criteria are discussed for the re-registration step. No meaningful quantification of why they chose six to stop at.
- Pre-filtering step described poorly
- Lacking a GitHub repository implementing their methods

References

[1] Ylimäki, Markus, Janne Heikkilä, and Juho Kannala. "Accurate 3-D Reconstruction with RGB-D Cameras using Depth Map Fusion and Pose Refinement." 2018 24th International Conference on Pattern Recognition (ICPR). IEEE, 2018.

[2] Kyöstilä, Tomi, et al. "Merging Overlapping Depth Maps into a Nonredundant Point Cloud." Scandinavian Conference on Image Analysis. Springer, Berlin, Heidelberg, 2013.

[3] Ylimäki, Markus, Juho Kannala, and Janne Heikkilä. "Robust and practical depth map fusion for timeof-flight cameras." Scandinavian Conference on Image Analysis. Springer, Cham, 2017.