CS6476 Final Project Proposal
Dense Mapping using Feature Matching and Superpixel Clustering

Mandy Xie, Shicong Ma, Gerry Chen

October 2019

1 Problem Statement

One of the fundamental tasks for robot autonomous navigation is to perceive and digitalize the surrounding 3D environment.[1] To be usable in mobile robot applications, the mapping system needs to fast and densely recover the environment in order to provide sufficient information for navigation.

Unlike other 3d reconstruction methods that reconstructs the environment as a 3D point cloud, we hope to extract surfels [2, 3, 4] based on extracted superpixels from intensity and depth images and construct a surfel cloud. This approach is introduced by [5] which can greatly reduces the memory burden of mapping system when applied to large-scale missions. More importantly, outliers and noise from low-quality depth maps can be reduced based on extracted superpixels.

The **goal** of our project is to reproduce results of Wang et al's, namely implementing superpixel extraction, surfel initialization, and surfel fusion to generate a surfel-based reconstruction given a camera poses from a sparse SLAM implementation. The **input** to our system will be an RGB-D video stream with accompanying camera poses and the **output** will be a surfel cloud map of the environment, similar to Figures 4b or 8 of the original paper [5].

2 Approach

The idea behind dense mapping is to first generate frame related poses, then reconstruct the dense map based on pre-generated poses and surfels.

1. Select a RGB-D dataset [https://vision.in.tum.de/data/datasets/rgbd-dataset/download]
2. Read pose information from the dataset / Use a sparse SLAM system (VINS [6]/ORB-SLAM2 [7]) to estimate camera poses
3. **(Suggested implementation)** – Single frame Superpixels extraction from RGB-D images using a k-means approach adapted from SLIC [8] - IV.D section in [5]
4. **(Suggested implementation)** – Single frame surfel generation based on extracted superpixels. - IV.E section in [5]
5. **(Suggested implementation)** – Surfel fusion and Surfel Cloud update. - IV.G section in [5]

3 Experiments and Results

The intended goal is to produce a surfel map. As such, our chief datasets will be RGB-D video datasets, preferably with camera pose ground truths, such as the ICL-NUIM [1], TUM [9], and KITTI [10] datasets. We will then run our code to generate surfel clouds and quantitatively compare our surfel clouds to those generated by the original paper's implementation [11] to show we produce similar results. We will then perform qualitative comparisons of our surfel reconstructions for the different datasets and locations within the datasets. Finally, we will introduce non-idealities in the input data such as camera pose error and depth error to quantitatively and qualitatively evaluate the robustness of our surfel implementation.

Since we aim to replicate the results of the paper, we expect that our surfels should match reasonably well with those generated by the original implementation. Since we do not consider map deformations, however, we expect that our implementation may be more prone to input data error than the original implementation but the degree of robustness is dependent upon the specifics of the surfel fusion implementation. Finally, we expect that the resulting surfel cloud should produce qualitatively better results in camera trajectories without loops or backtracking since the robustness of surfel fusion is limited and most accurate when reprojection error is minimal. A successful project would produce similar results as the original paper although with potentially reduced robustness to camera pose error.

We intend to implement all superpixel and surfel extraction and fusion ourselves as well as some data processing, visualization, and experimental evaluation. Codes which we do not intend to implement include extraction of camera poses (if not provided by the dataset) and map deformation.

References

[1] A. Handa, T. Whelan, J. McDonald, and A. J. Davison, "A benchmark for rgb-d visual odometry, 3d reconstruction and slam," in *2014 IEEE international conference on Robotics and automation (ICRA)*, IEEE, 2014, pp. 1524–1531.

[2] T. Schöps, T. Sattler, and M. Pollefeys, "Surfelmeshing: Online surfel-based mesh reconstruction," *ArXiv preprint arXiv:1810.00729*, 2018.

[3] H. Pfister, M. Zwicker, J. Van Baar, and M. Gross, "Surfels: Surface elements as rendering primitives," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., 2000, pp. 335–342.

[4] I. Tobor, C. Schlick, and L. Grisoni, "Rendering by surfels," in *Proceedings of Graphicon*, vol. 3, 2000, p. 25.

[5] K. Wang, F. Gao, and S. Shen, "Real-time scalable dense surfel mapping," in *2019 International Conference on Robotics and Automation (ICRA)*, IEEE, 2019, pp. 6919–6925.

[6] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[7] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[8] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.

[9] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, 2012.

[10] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[11] https://github.com/HKUST-Aerial-Robotics/DenseSurfelMapping.