Real-time Surface Reconstruction on a Mobile Device using Chunked Truncated Signed Distance Fields

Matthew Klingensmith
The Robotics Institute, Carnegie Mellon University
mklingen@andrew.cmu.edu

Ivan Dryanovski
City College of New York
ivan.dryanovski@gmail.com

Fig. 1. An indoor apartment scene reconstructed in real-time on a mobile device (inset) at a resolution of 3 cm.

Summary. In this work, we present an efficient algorithm for reconstructing 3D surfaces on a depth sensor equipped mobile device in real-time using chunked, truncated signed distance fields.

What is the problem? We are interested in the problem of real-time 3D reconstruction. The task is to extract the true 3D geometry and color of a real scene from a sequence of noisy sensor readings from a handheld mobile device.

Why is it interesting? Solutions to this problem are useful for mobile robot navigation, indoor localization, mapping, and object scanning. As far as we know, high-quality, real-time 3D surface reconstruction on a mobile device has never been attempted before.

Why is it hard? The problem is ill-posed, since it requires localization as well as mapping (and hence is a variant of the SLAM problem.) In 3D, memory requirements for high quality 3D reconstruction are prohibitive. Real-time 3D reconstruction remains challenging on mobile devices. Compared to full-sized machines, mobile devices have severely limited processing power, memory requirements, and graphics capabilities. Further, the depth sensors available on these devices are much more limited than their full-sized counterparts; they are typically lower resolution, have slower refresh rates, and have much more undesirable nonlinear distortion and noise.

What is our insight? Previous work on high quality real-time 3D reconstruction has focused on much more capable and high-powered computing platforms with much more accurate sensing, and is therefore not well suited for low-powered mobile devices. These approaches, such as *Kinect Fusion* [1], and *Kintinuous* [2], work by costructing a truncated signed distance field of the scene.

So far, real-time 3D reconstruction on mobile devices has been limited to simple occupany grid mapping, using fixed 3D arrays [1], [3], and octrees [4] which is not suitable for high quality 3D suface reconstruction.

Our main insight is to use a hybrid data structure containing a dynamic map of fixed 3D "chunks" of signed distance field data. Unlike fixed 3D arrays, this data structure has much better memory performance; and, unlike octrees, it has much better cache performance and boasts $\mathcal{O}(1)$ access times. By chunking the distance field dynamically, we are able to asynchronously reconstruct surfaces for rendering while new sensor data is coming in.

Additionally, by sacrificing the physical accuracy of ray-casting for a more efficient projection-based technique of constructing the distance field, we are able to use *negative information* in the form of space carving [5] to produce a less noisy surface estimate.

Contributions. We make the following contributions:

- An efficient data structure for representing very large volumetric data sets.
- An example framework and interface for doing large 3D reconstructions on a mobile device.
- Efficient techniques for real-time rendering and interaction with large reconstructions on a mobile device.

Limitations. Our work depends on localization as a pre-requisite, and uses a visual intertial odometry system as a black box input. We make no attempt to improve localization using mapping, and thus avoid the SLAM problem. As such, our reconstructions suffer from localization errors that we accumulate over time. How can we use the map itself to improve pose estimates? We are interested in using the volumetric signed distance data directly to compute global localization updates.

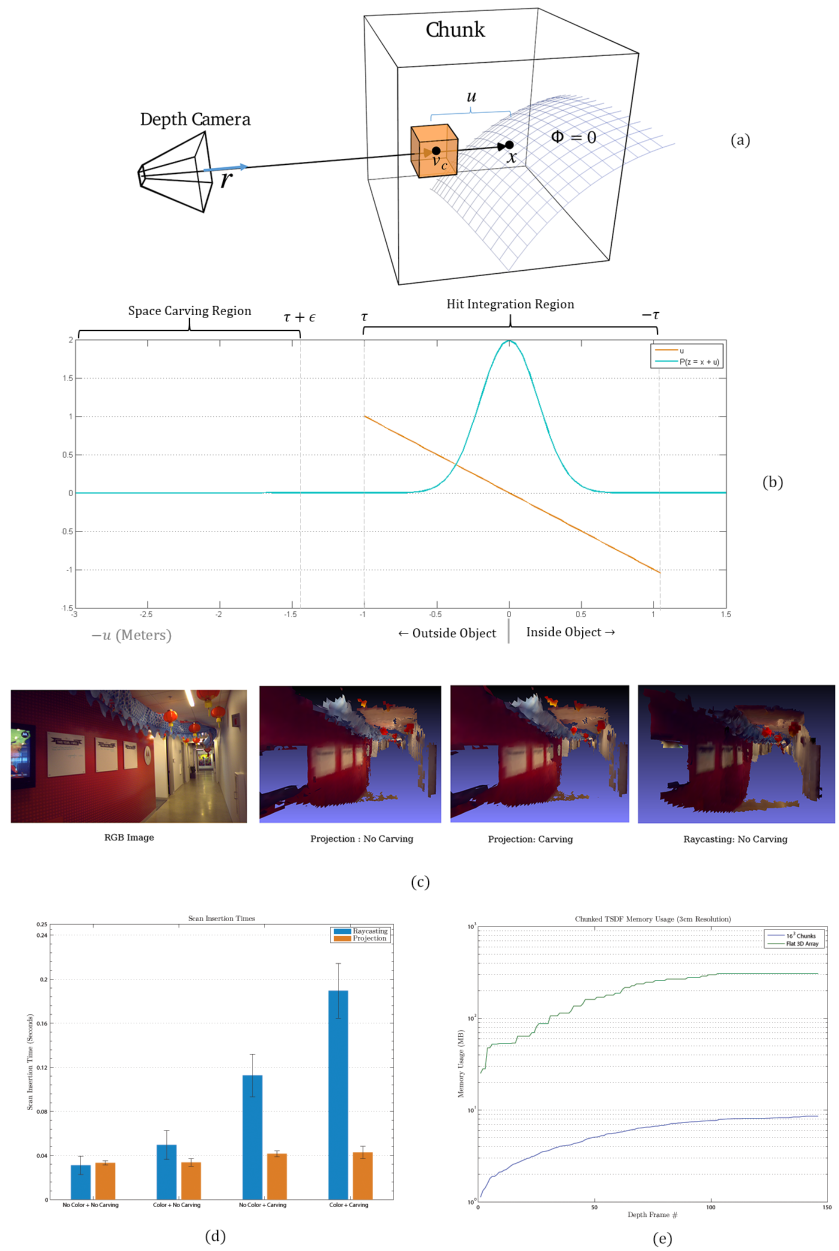

Fig. 2. We divide the world into a series of chunks that contain N_c^3 voxels each. In each voxel we store the value of a signed distance function, $\Phi(x)$. Wherever $Phi = 0$, we have the surface. Wherever $\Phi < 0$, it is inside an object, and vice versa wherever $\Phi > 0$. The distance field is estimated near the surface by the distance along the depth ray from the surface to the camera (a), within a threshold τ called the truncation distance (b). In all parts of the space closer than $\tau + \epsilon$ to the camera, we "carve" voxels there. In all parts of the space within τ of the end of a depth ray, we increment a running weighted average of voxels in the distance field. Raycasting is approximated by a projection technique which has resolution dependant accuracy (c), but which is much faster when space carving is used (d). By only storing "chunks" of data near the surfaces of objects, we use orders of magnitude less memory than *Kinect Fusion* (e).

References

[1] R. Newcombe and A. Davison, "KinectFusion: Real-time dense surface mapping and tracking," *...(ISMAR), 2011 10th ...*, 2011. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6162880

[2] T. Whelan and H. Johannsson, "Robust real-time visual odometry for dense RGB-D mapping," *...(ICRA), 2013 IEEE ...*, no. i, 2013. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6631400

[3] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer (Long. Beach. Calif).*, 1989. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=30720

[4] K. Wurm, "OctoMap: A probabilistic, flexible, and compact 3D map representation for robotic systems," *...Pract. 3D ...*, 2010. [Online]. Available: http://ais.informatik.uni-freiburg.de/publications/papers/wurm10octomap.pdf

[5] M. Klingensmith, M. Herrmann, and S. S. Srinivasa, "Object Modeling and Recognition from Sparse , Noisy Data via Voxel Depth Carving," in *ISER*, no. d, 2014.