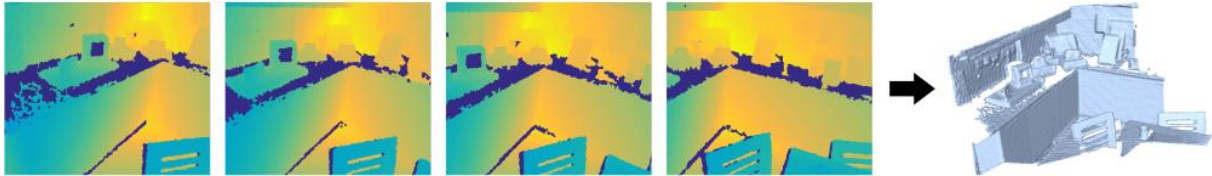

Volumetric TSDF Fusion of Multiple Depth Maps

Update: a python version of this code with both CPU/GPU support can be found [here](#).



CUDA/C++ code to fuse multiple registered depth maps into a projective truncated signed distance function (TSDF) voxel volume, which can then be used to create high quality 3D surface meshes and point clouds. Tested on Ubuntu 14.04 and 16.04.

Looking for an older version? See [here](#).

Change Log

- **Nov. 1, 2017.** Bug fix: `tsdf2mesh.m` now properly generates a mesh in camera coordinates instead of voxel coordinates.
- **Oct. 30, 2017.** Notice: changed default weight threshold for `SaveVoxelGrid2SurfacePointCloud` in demo code to enable creating point cloud visualizations with only one depth frame.
- **Aug. 30, 2017.** Bug fix: remove deprecated offsets from surface distance compute during integration.

Requirements

- NVIDIA GPU with CUDA support
- OpenCV (tested with OpenCV 2.4.11)

Demo

This demo fuses 50 registered depth maps from directory `data/rgb-d-frames` into a projective TSDF voxel volume, and creates a 3D surface point cloud `tsdf.ply`, which can be visualized with a 3D viewer like Meshlab.

Note: Input depth maps should be saved in format: 16-bit PNG, depth in millimeters.

```
1 ./compile.sh # compiles demo executable
2 ./demo # 3D point cloud saved to tsdf.ply and voxel grid saved to tsdf.bin
```

[Optional] This demo also saves the computed voxel volume into a binary file `tsdf.bin`. Run the following script in Matlab to create a 3D surface mesh `mesh.ply`, which can be visualized with Meshlab.

```
1 tsdf2mesh; % 3D mesh saved to mesh.ply
```

Seen in

- 3DMatch: Learning Local Geometric Descriptors from RGB-D Reconstructions (CVPR 2017)
- Semantic Scene Completion from a Single Depth Image (CVPR 2017)
- Deep Sliding Shapes for Amodal 3D Object Detection in RGB-D Images (CVPR 2016)

References

- A Volumetric Method for Building Complex Models from Range Images (SIGGRAPH 1996)
- KinectFusion: Real-Time Dense Surface Mapping and Tracking (ISMAR 2011)
- Scene Coordinate Regression Forests for Camera Relocalization in RGB-D Images (CVPR 2013)

Citing

This repository is a part of 3DMatch Toolbox. If you find this code useful in your work, please consider citing:

```
1 @inproceedings{zeng20163dmatch,  
2   title={3DMatch: Learning Local Geometric Descriptors from RGB-D  
3     Reconstructions},  
4   author={Zeng, Andy and Song, Shuran and Nie{\ss}ner, Matthias and  
5     Fisher, Matthew and Xiao, Jianxiong and Funkhouser, Thomas},  
6   booktitle={CVPR},  
7   year={2017}
```