3D Scanner C++ Project

Weekly Report (25/12/2017 - 31/12/2017)

Meldrick REIMMER, Selma BOUDISSA , Jaafar Al-TUWAYYIJ and Danie SONIZARA

Assessment of the week

This week we researched on several filters in which we could use for the mesh generated, to enable us for a good results of it. We also started with the Gui interface and made an draft on how it should look like and what it must need and what it must not have.

Previous Objectives

1. Research Filters to use for the mesh
2. Start with the GUI (In progress)
3. Try to include Open-CV on the previous year program (In progress)

Filters (The Bilateral Filter)

The bilateral filter is a nonlinear, feature preserving image filter, proposed by Smith and Brady [1997], and separately by Tomasi and Manduchi [1998]. Although proposed as an alternative to anisotropic diffusion, the close connection between the two was not well understood until recently. The connection between robust estimation, anisotropic diffusion, and the bilateral filter was investigated by Durand and Dorsey [2002], as well as Barash [2001] via an extension of intensity to include spatial position, and Elad [2002] as the output of an iterative minimization.

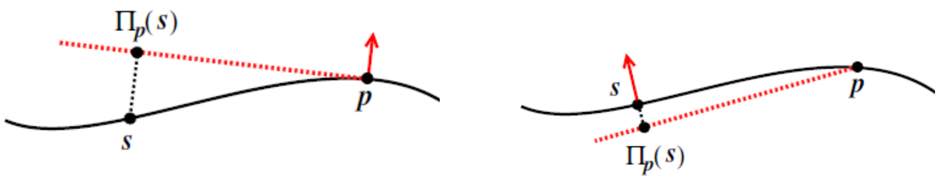
Bilateral Filter for 3D Surfaces

The success of the bilateral filter for image de-noising encourages us to use similar methods for surfaces. This follows the pattern of much of the previous work in which image techniques have been successfully extended to polygonal meshes and other 3D data.

Such extensions are nontrivial because of a fundamental difference between 2D images and 3D surfaces: Images have an inherent parameterization that is well separated from the images signal. In the usual case, pixels are arranged on a rectangular grid, giving a simple and global parameterization. This separation of position vs.signal (or domain vs. range) in images simplifies operations such as smoothing and feature detection. Surfaces in 3D lack this conceptual separation. The spatial position of a point on a surface and the surface's "signal" at that point are the same.We must find some way to separate position and signal to apply methods from image processing.

Predictors

In order to make our extension concrete, we much choose a predictor Î p(s). If we assume that we have points with normals, then there are two natural choices for the predictor, projecting s to the plane of p, or building a local frame with s and its normal and predicting s to have the same height in that local frame as p.



(a) Our predictor from Jones et al. [2003].

(b) Predictor from Fleishman et al. [2003]

Figure 4-1: 3D predictors for point **s** due to point **p**.

The main difference between the two predictors is which normals are used to form predictions. In the first, each prediction for a point uses a different normal, while the second uses a single normal, the one at the point for which predictions are being made.

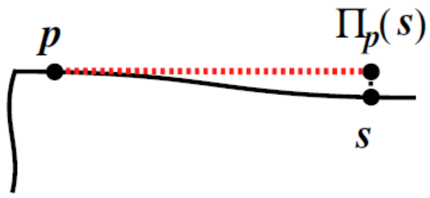


Figure 4-2: When $\mathbf{s}$ and $\mathbf{p}$ are on the same side of the corner, $\Pi_{\mathbf{p}}(\mathbf{s})$ is near $\mathbf{s}$.

This Week Objectives

1. Research Filters to use for the mesh (Continue)
2. Progress on the GUI
3. Try to include Open-CV on the previous year program (In progress)

Reference

- **Trello Account** `https://trello.com/b/MaBdGQ7p/software-engineering`
- **GitHub Account** `https://github.com/MSCV1-2017/3D-ScannerProject`
- **Previous Project** `https://github.com/umaatgithub/3D-KORN`