Accurate, Dense, and Robust Multi-View Stereopsis

Girish Chandar G — 16110057

July 12, 2020

1 Aim

Implementation of the algorithm proposed by Furukawa et al in [1] for multi-view stereopsis.

2 Introduction

Multi-View stereopsis is a much sort after problem in computer vision and has applications in 3D Photography and Augmented Reality. [1] proposes a novel algorithm for 3D reconstruction of the scene from multiple images of the scenes. The algorithm involves 3 steps namely: match, expand and filter. In this project I have implemented the 'match' step of the algorithm. One important aspect of this algorithm is that it does not require any initial mesh model. The algorithm delivers the orientation of the surface along with its depth.

3 Key Elements

This section gives a brief overview of the functions and terminologies and how they are defined.

3.1 Patch Model:

Every patch p is defined by three parameters namely $\mathbf{c}(p)$(the centre of the patch),$\mathbf{n}(p)$(the normal to the patch) and $R(p)$(the reference image). The patch is taken to be a 5x5 grid with one of its axis parallel to the axes of the reference image. In my implementation I have taken the x-axis to be parallel to the x-axis of the reference image w.r.t to the camera centre of the image.

3.2 Photometric Discrepancy Function:

The photometric discrepancy function for a patch p is defined as follows:

$$g(p) = \frac{1}{|V(p) \backslash R(p)|} \sum_{I \epsilon V(p) \backslash R(p)} h(p, I, R(p)$$

where $V(p)$ is the set of images in which the patch p is visible and $h(p, I_1, I_2)$ is defined as pairwise discrepancy function between I_1 and I_2. $\mathbf{q}(p, I_i)$ is calculated by laying 5x5 grid on I_i and sampling pixels values by bilinear interpolation[2]. $h(p, I_1, I_2)$ is calculated from $\mathbf{q}(p, I_1)$ and $\mathbf{q}(p, I_2)$ as follows:

$$h(p, I_1, I_2) = 1 - \mathbf{NCC}(\mathbf{q}(p, I_1), \mathbf{q}(p, I_2))$$

where **NCC** calculates the normalised cross correlation score. The set of visible images $V(p)$ is further refined by retaining only thse images in $V(p)$ whose $g(p)$ is less an threshold(α).

3.3 Patch Optimization:

The initialization of parameters for patch p is described in the next section. Once $\mathbf{c}(p)$ and $\mathbf{n}(p)$ are initialized, these parameters are optimized by conjugate gradient method. $\mathbf{c}(p)$ is constrained on the line joining the optical centre and its projection on one of its visible images and therefore only depth has to be optimized. $\mathbf{n}(p)$ is parametereized by yaw and pitch. Altogether optimizing $\mathbf{c}(p)$ and $\mathbf{n}(p)$ boils down to a 3 variable optimzation problem.

4 Approach

As mentioned in Section 2 the full algorithm consists of matching, expansion and filtering steps. The following description is for my implementation of the matching step of the algorithm.

4.1 Feature Detection:

The paper uses DoG(Difference of Gaussian) and Harris operators for detecting features. In my implementation I have used SIFT for feature detection and extraction.

4.2 Feature matching:

For every feature in the reference image, its feature correspondence in the other images is found by calculating its corresponding epipolar line in the other images and selecting the features within 2 pixel distance from the line. For every feature f, its corresponding featuresf' are sorted w.r.t to the distance from the epipoles of the optical centre of the reference image on their respective images. Every pair (f, f') is taken in the sorted order and used in trying to reconstructing a patch until we succeed.

4.3 Patch Initialization:

Every f in image I_i with optical centre $O(I_i)$ a patch is initialized for every pair (f, f'), where f' is found as mentioned in Section 4.2. Patch initialization is done by initializing the 3 parameters $\mathbf{c}(p)$, $\mathbf{n}(p)$ and $R(p)$ as follows.

$$\begin{aligned}
\mathbf{c}(p) &\longleftarrow \text{triangulation of } (f, f') \\
\mathbf{n}(p) &\longleftarrow \frac{\overrightarrow{\mathbf{c}(p)O(I_i)}}{|\mathbf{c}(p)O(I_i)|} \\
R(p) &\longleftarrow I_i
\end{aligned}$$

Figure 1: Reference Image

4.4 Visible Images Initialization:

A patch is said to be visible in image I if the angle between the ray joining $\mathbf{c}(p)$ and $O(I)$(the optical centre of image I) and the $\mathbf{n}(p)$ is less than a threshold $tau = \pi/3$.

$$V(p) = \{I | \mathbf{n}(p).\frac{\overrightarrow{\mathbf{c}(p)O(I)}}{|\mathbf{c}(p)O(I)|} > \cos(\tau)\}$$

Once $V(p)$ is in initialized then $V^*(p)$ as mentioned in Section 3.2. Then $\mathbf{c}(p)$ and $\mathbf{n}(p)$ are refined as described in Section 3.3. $V(p)$ and $V^*(p)$ are updated after $\mathbf{c}(p)$ and $\mathbf{n}(p)$ are refined. The patch reconstruction is considered a success only if the cardinality of $V^*(p)$ is greater than a certain threshold γ. The value for γ has not been mentioned in the paper, I have used $t = 8$ for my implementation. The patches that succeed the patch reconstruction is stored in P($Patches$ dictionary in my implementation).

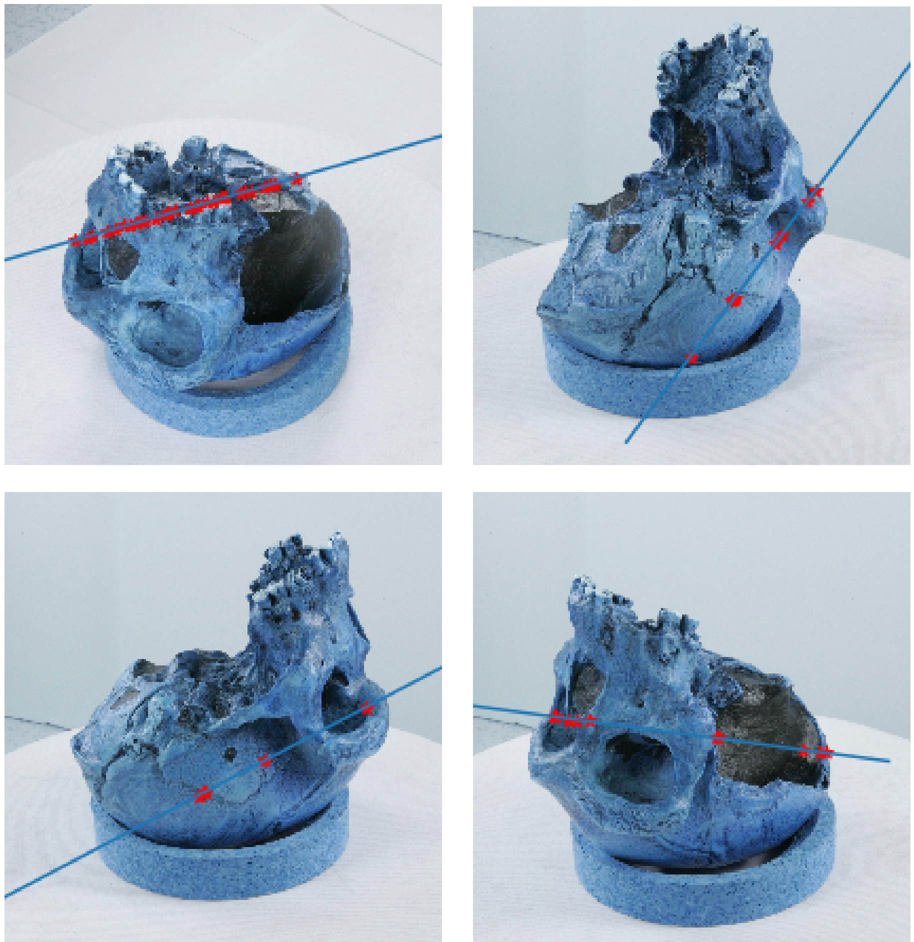

Figure 2: Other Images

5 Results

I have not been able to display the reconstructed patches but below I have shown the feature detection and feature matching with corresponding epipolar lines below. Figure 1 shows the reference image along with the feature under consideration. Figure 2 shows the epipolar lines correspoinding to featuer displayed in the reference image and the features in those images that are within 2 pixels distance from the line. The patch parameters for all the patches reconstructed is stored in the *Patches* dictionary in the implemented code.

References

[1] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 8, pp. 1362–1376, 2009.

[2] A. Iserles, "Numerical recipes in c—the art of scientific computing, by wh press, bp flannery, sa teukolsky and wt vetterling. pp 735.£ 27· 50. 1988. isbn 0-521-35465-x (cambridge university press)," *The Mathematical Gazette*, vol. 73, no. 464, pp. 167–170, 1989.