Outlier Removal in Stereo Reconstruction of Orbital Images

Marvin Smith[1] and Ara Nefian[2]

[1] University of Nevada, Reno
[2] NASA Ames Research Center

Abstract. NASA has recently been building 3-dimensional models of the moon based on photos taken from orbiting satellites and the Apollo missions. One issue with the stereo reconstruction is the handling of "outliers", or areas with rapid and unexpected change in the data. Outliers may be introduced by issues such as shadows on the surface, areas with low amounts of surface detail, or flaws in the camera systems. These errors may result in elevation spikes which cause the model to differ significantly from accurate ground truth. We are seeking to remove outliers from reconstructions by using a pair of filters which target the characteristics of these outliers. The first filter will use edge detection to filter areas with low detail and the second filter will remove areas in the disparity map which differ too far from their surrounding neighbors.

1 Introduction

The NASA Vision Workbench and the Ames Stereo Pipeline have created accurate, high resolution 3D maps of the Lunar surface. These maps support a variety of NASA missions as well as commercial products. Examples include finding future landing sites for human missions, developing computer-based landing systems[1], and Google Moon. One aspect of the Ames Stereo Pipeline that we are looking to improve upon is in the removal of outliers, or areas of the disparity map which are extreme to the points around them.

Outliers can cause unsightly and inaccurate results on a disparity map when ignored. Figure 1 shows the effect of outliers on an image. Also shown is the effect of the current outlier removal system in Figure 2.

Currently, there is a system in place which seeks to filter outliers. This system relies on a morphological erosion-like operation which is described in the following steps.

- For each pixel in disparity map $D(i, j)$
  - Search a surrounding window of $M \times N$ pixels
  - Compare each pixel in window to input pixel s.t. $|D(i, j) - D(i - m, j - n)|$
  - If difference is greater than threshold, increment count of invalid pixels
- If invalid pixels divided by total pixels is greater than threshold, invalidate pixel
- Else, return original pixel

G. Bebis et al. (Eds.): ISVC 2010, Part III, LNCS 6455, pp. 181–188, 2010.

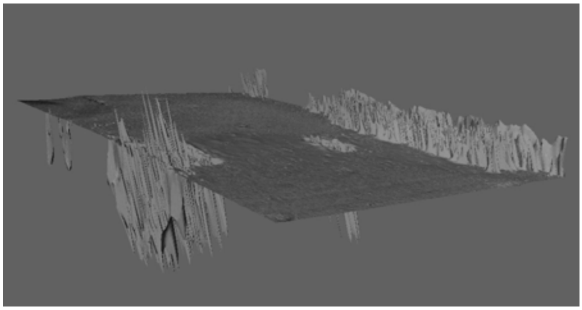


Fig. 1. Original image

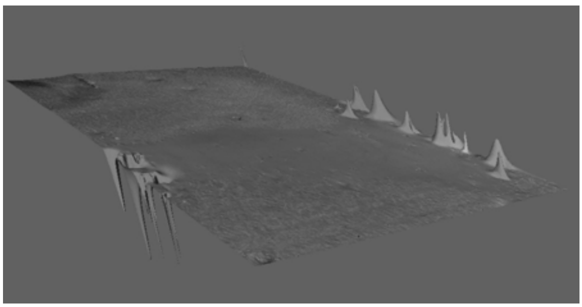


Fig. 2. Result of the current method

The current method occurs after the sub-pixel refinement stage[2]. Other outlier rejection techniques were analyzed and found not to fit within the framework of the Ames Stereo Pipeline [3][4].

There are several aspects of the original method which we seek to improve. The first issue is that there is not a preprocessing method to invalidate regions which have little to no detail. This method does not directly address this issue itself either as regions which have no detail, may appear the same, therefore giving the appearance of the same disparity. Another challenging alternative is an outlier being created when the pixel correlator takes a pixel inside a shadow and is required to guess which pixel is the corresponding match. As the correlation window searches for a match [5], if it searches over an area where many of the pixels are similar, there is no guarantee as to where in the window a match may be found.

The next issue to improve is the iterative process which treats every pixel in the disparity map as a logical value based on the difference in disparity from the center of the window. This may incorrectly invalidate regions which have rapid change, but a consistent gradient. If the pixels inside the window are different from the pixel of interest beyond a threshold, the pixels will be added to the invalid count. A process which compares the pixel to the window *average* would allow for more flexibility.

These solutions will be included in different phases. The Edge Density filter will be located at the start of this process. It will filter the initial image. Figure 3 shows the proposed changes to the Ames Stereo Pipeline.

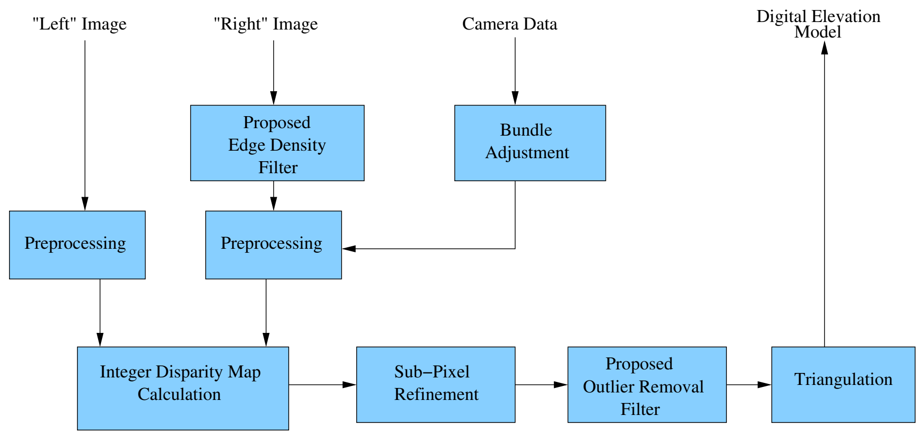


Fig. 3. The Ames Stereo Pipeline with the proposed outlier filtering techniques

1.1 Edge Density Filter

The Edge Density filter is a preprocessing filter which is designed to remove regions of the image which have little to no extractable detail. This filter will be applied to the original image before being sent to the Integer Disparity Map module of the Ames Stereo Pipeline. A general outline of the process is shown in Figure 4.

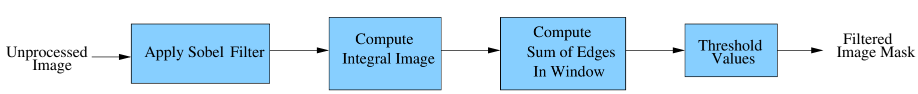
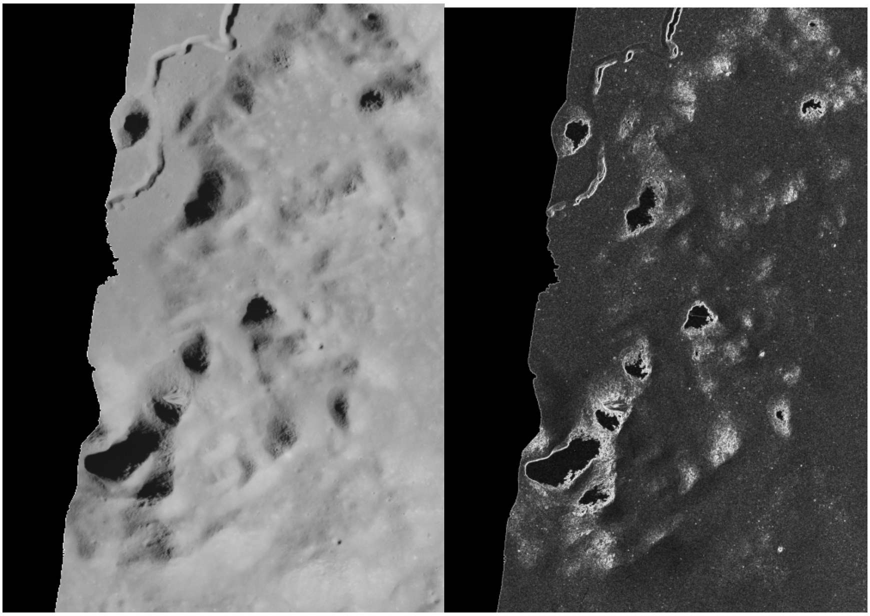


Fig. 4. The Edge Density Filter pipeline

The first and primary technique is to perform the Sobel edge detector [6]. The purpose is to describe and quantify the amount of detail or change thereof. Once we compute the edges, we compute the integral image[4]. This will allow us to rapidly and efficiently compute the sum of edges around a window for every pixel. Due to the size of the images and the large landscapes that are being dealt with, window sizes around 35 pixels are used. Next divide each result by the area of the window. Once these values have been computed, we apply a final threshold to create a binary mask. The binary mask will be intersected with the results from the sub-pixel correlation to prevent the invalidated regions from being computed in the triangulation module. Figure 5 shows an example output of the Edge Density Filter prior to thresholding.



(a) Original image (b) Filtered image

Fig. 5. Effect of Edge Density Filter prior to thresholding

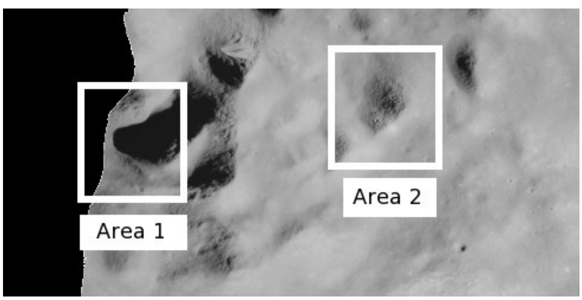
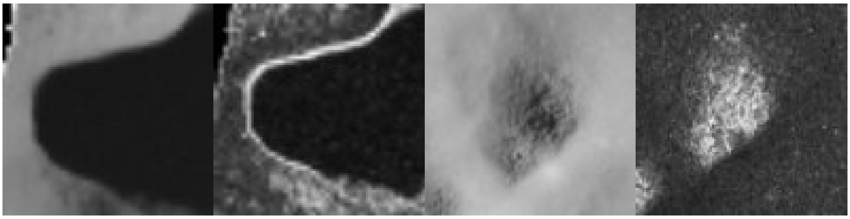


Fig. 6. Areas under investigation

What is relevant about this example is that regions which are completely black correspond to areas inside a shadow or have extremely low detail. Also note the distinction between shadows which persist over large regions such as craters, and small shadows which relate to the rapid elevation change of a hill or other natural surface. In Figure 6, regions marked *Area 1* and *Area 2* show two types of shadows which we want to analyze with respect to the Edge Density Filter.

Area 1, as seen in Figure 7a and Figure 7b, is a large blanketing shadow which shows no discernible detail inside of it. This will cause undesirable effects when



(a) Area 1 Original (b) Area 1 Filtered (c) Area 2 Original (d) Area 2 Filtered

Fig. 7. Effect of the Edge Density Filter on regions investigated in Figure 6. Note that both regions (a) and (c) are covered inside a shadow, yet the filtering effects are noticeably different (b & d). Area 1 is completely occluded inside the shadow, whereas Area 2 has a much lighter covering. The resulting contrast from the shadow in (c) enhances the detail, making the filtering results (d) possibly even stronger than regions not in a shadow.

correlated. In comparison, the shadow in Area 2 as seen in Figure 7c and Figure 7d is very unique and has much detail. The Edge Density Filter is very effective for this purpose as shown in Figure7.

As a result, different types of shadows can result in completely opposite reactions. This is advantageous for comparing shadows of different structures.

1.2 Outlier Removal Filter

The next phase is the outlier removal filter. The outlier removal filter will search the disparity map and invalidate pixels where the disparity between itself and its surrounding average are different beyond a threshold. This is similar to the original method described in Section 1, but differs in that we want to compare it to the average inside the window and not compare it to each pixel independently. The difference between these methods become apparent in Figure 8.

Another strength of computing an average is that it is more likely to find and remove turbulent areas much like regions in Figure 1. If there is rapid change in a large window, the threshold may allow a number of outliers to validate a region. This makes the average much more secure to threats as the remaining 'holes' would affect the average value.

The first step of this process is to convert the horizontal and vertical disparities into a single scalar magnitude. This will allow for a single pass through the image and treats the separate disparities as a combined result. Once the scalar magnitude is computed, an integral image is created and the density of the disparities is calculated. This is very similar to the process outlined in the Edge Density Filter in Section 1.1. Finally, for each pixel, compare the scalar magnitude of the disparities to the results of the disparity density. If the difference is beyond a threshold, invalidate the pixel. Figure 9 shows the results from the new filter.

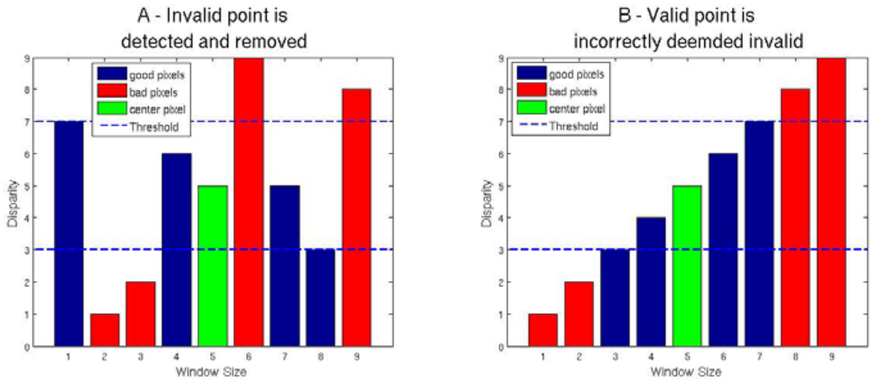
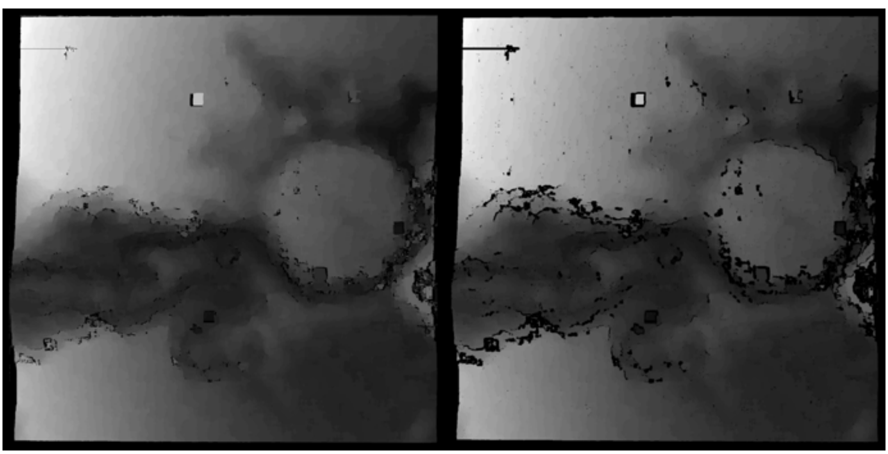


Fig. 8. Example of pixels rejected using the current method. (A) is an obvious outlier. The range of data shows no pattern and has extreme pixel values, therefore the system works. In (B), the rapid but constant increase in disparity could mean a legitimate increase in depth, making the increase a natural phenomenon. However, with the current system, the outer limits are rejected since they lie outside of the threshold. For the Outlier Rejection Filter, the average would be equal to the center pixel value, making this a legitimate disparity.



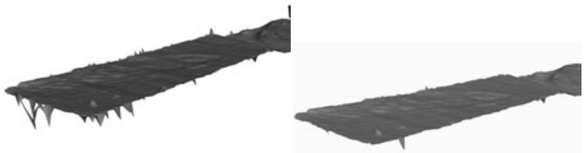
(a) Original image (b) Filtered image

Fig. 9. Sample results from the Outlier Filter on a disparity map

2 Results

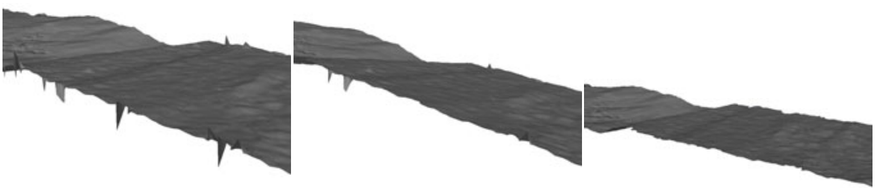
Our results showed improved results as compared to the original and unfiltered systems.

One element that is incredibly important is choosing proper thresholds as well as window sizes [7]. If the window size is too small, large outliers may survive as it may be gradual enough to exploit the example in Figure 8(b). Likewise,



(a) Unfiltered (b) Current method (c) Proposed method

Fig. 10. Results from a trial run on similar segment using different filter. Notice that the new system performs equal, if not better than current system.



(a) Unfiltered (b) Current method (c) Proposed method

Fig. 11. Results from a trial run on similar segment using different filter. The new filter performs clearly better than current system.

(a) Unfiltered (b) Current method (c) Proposed method

Fig. 12. Results from a trial run on similar segment using the different filters. The new system performed equally to the current system.

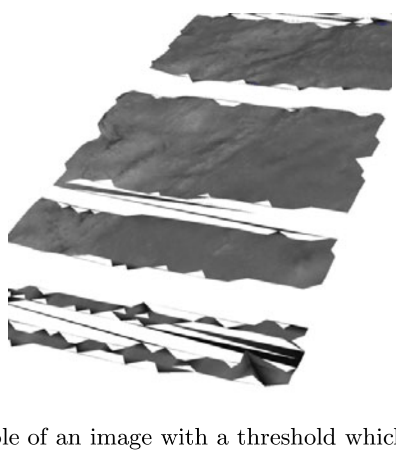


Fig. 13. Example of an image with a threshold which is too selective

if the window size is too large, then the average will be minimally impacted by large outliers, thus some outliers will survive. The thresholds are similar, as very high thresholds make the filters very selective. Figure 13 shows an example of selecting a threshold for the Edge Density Filter which is too selective.

3 Conclusion

We have achieved a successful set of cascade filters for outlier detection and removal. The Edge Density Filter and the Outlier Removal Filter were able to increase the rejection rate of true outliers as well as decrease the rate of false positives. These filters are currently being implemented in the Ames Stereo Pipeline with direct application on the Apollo lunar reconstructions. This system may be implemented on other types of orbital imagery or other stereo vision applications. Examples include the Mars satellite orbiters and rovers.

4 Future Work

One aspect that would further increase run-time performance would be to run the filter on just a segment of each image. With orbital images, there may be only a small amount of each image which overlaps, making full computation unnecessary. Another interesting concept would be to create a method of automatically computing a threshold or window size. Currently, the best method for choosing an appropriate threshold or window size is by observation.

References

1. Broxton, M.J., Nefian, A.V., Moratto, Z., Kim, T., Lundy, M., Segal, A.V.: 3d lunar terrain reconstruction from apollo images. In: ISVC 2009: Proceedings of the 5th International Symposium on Advances in Visual Computing, pp. 710–719. Springer, Heidelberg (2009)
2. Nefian, A.V., Husmann, K., Broxton, M., To, V., Lundy, M., Hancher, M.D.: A bayesian formulation for sub-pixel refinement in stereo orbital imagery. In: ICIP 2009: Proceedings of the 16th IEEE International Conference on Image Processing, Piscataway, NJ, USA, pp. 2337–2340. IEEE Press, Los Alamitos (2009)
3. Koch, R., Pollefeys, M., Gool, L.J.V.: Multi viewpoint stereo from uncalibrated video sequences. In: Burkhardt, H.-J., Neumann, B. (eds.) ECCV 1998. LNCS, vol. 1406, pp. 55–71. Springer, Heidelberg (1998)
4. Viola, P., Jones, M.J.: Robust real-time face detection. Int. J. Comput. Vision 57, 137–154 (2004)
5. Menard, C.: Robust stereo and adaptive matching in correlation scal-space. Technical report (1997)
6. Jain, R., Kasturi, R., Schuck, B.: Machine Vision. McGraw-Hill, New York (1995)
7. Gerrits, M., Bekaert, P.: Local stereo matching with segmentation-based outlier rejection. In: Canadian Conference Computer and Robot Vision, p. 66 (2006)