Effect of Spatial Sampling on Topological Reconstruction with Mobile Cameras

Michael K. Johnson
Stanford University
M.S. Computer Science
mikejohn@stanford.edu

Dominique S. Piens
Stanford University
M.S. Electrical Engineering
dpiens@stanford.edu

Abstract

Remaining unexplored environments may be hostile to humans and have unreliable communications. Unmanned vehicles combined with computer vision would enable simultaneous topological reconstruction and corrections to pose estimates through triangulation. Although these systems and their sources of error are well studied, the effect of spatial sampling on the accuracy of topographical reconstruction is not broadly characterized. Here, we develop a software pipeline to study the effect of spatial sampling on topological reconstruction with computer vision. The pipeline begins with a simulation to model vehicles equipped with cameras traveling over a terrain and capturing synthetic images. Then, those images and pose estimates are used to metrically refine pose estimates and reconstruct the terrain. The pipeline is demonstrated by characterizing the tradeoff curve between image overlap (spatial sampling period) and reconstruction error for an unmanned aerial vehicle with a linear flight path. The robustness of feature detectors – Harris corners, speeded-up robust features (SURF), maximally stable extremal regions (MSER) and histogram of oriented gradients (HoG) – to pose estimation error in 3D reconstruction is also assessed as a function of image overlap. The study demonstrates the usefulness of the pipeline it describes in planning exploration missions in environments with limited communications.

1. Introduction

Unmanned vehicles are essential to exploring inaccessible environments with limited communications like the deep desert or planetary surfaces [1, 2]. Limited communications introduce challenges with respect to odometry and positioning. Computer vision has provided some solutions for both challenges [3, 4]. In particular, topographical reconstruction has been proposed as a means to explore unknown environments strictly from images [5], or with the addition of light detection and ranging (LIDAR) [6]. However, the effect of spatial sampling and uncertainty in camera pose on the accuracy of topographical reconstruction is not well studied in the absence of visual markers on the terrain, GPS, or LIDAR.

Previous work exists which provides characterizations of the effect of spatial sampling on reconstruction accuracy, but these are highly dependent on the experimental setup. The effect of spatial sampling frequency is investigated under the names of image decimation, number of images, image overlap, and/or number of correspondences per point. Previous studies have characterized two cameras at ground level to reconstruct anthropogenic structures [7], up to 30 simulated cameras in various formations facing a central point [8], and a simulated UAV equipped with five cameras (Maltese cross formation) flying over a terrain with buildings [9].

In this study, we develop a software pipeline to investigate the effect of spatial sampling and pose uncertainties on the accuracy of topographical reconstruction with images captured from any team of mobile cameras. The pipeline involves simulating vehicles moving over a rough, featureless terrain, and varying the spatial frequency at which synthetic images are captured. We demonstrate the pipelines capabilities by simulating unmanned aerial vehicles (UAVs) on linear flight paths, and capturing oblique images. The images captured by one of the UAVs are used to generate a trade-off curve of spatial capture frequency and corresponding topographical reconstruction error. The modularity of the pipeline design allows for vehicle number and dynamics, number and parameters of cameras per vehicle, terrain, and spatial sampling frequency to be changed independently. This study could have implications for the planning and design of missions involving positioning and topographical reconstruction in remote areas with limited communications (e.g. planet surface exploration, military autonomous rovers, sea floor exploration).

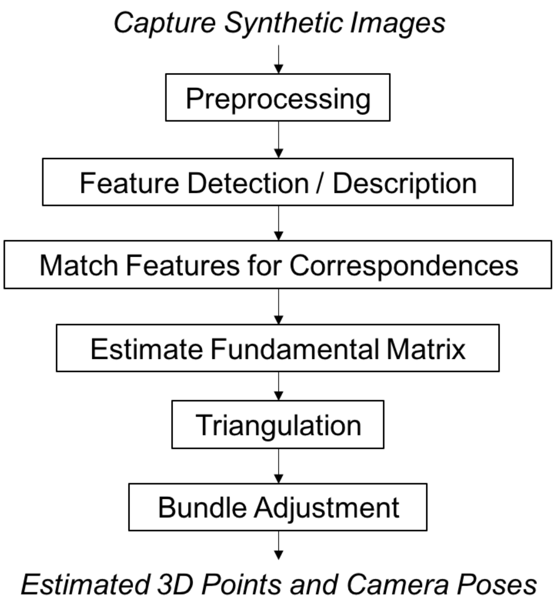

Capture Synthetic Images
Preprocessing
Feature Detection / Description
Match Features for Correspondences
Estimate Fundamental Matrix
Triangulation
Bundle Adjustment
Estimated 3D Points and Camera Poses

Figure 1. Block diagram of the Structure From Motion implementation.

2. Problem Statement

Figure 1 depicts our approach for reconstructing the topology from images. The captured images are taken from our simulation of a remote terrain using cameras with known intrinsic parameters, unknown position and orientation, and at varying amounts of overlap between images. Random uniform noise is added to the translation and orientation of the captured images to represent a more realistic environment with imperfect path control. The process outputs positions of points on the ground, and the estimated camera poses for each image, both in coordinates centered at the origin of the simulation map. The error of our estimations is assessed by the euclidean distance between the estimated 3D point and the nearest true 3D points of the simulation map. We will analyze the effects of the amount of overlap between consecutive images as well as the effects of noise of estimation. This process is defined in greater detail in the following sections.

3. Technical Content

3.1. UAV and Terrain Simulation

Open source software Height Map Editor was used to generate maps. Initially, the scale chosen is 0.1 m per pixel, and the 8-bit intensity scaled to 0 to 20 m. For development, a square map representing a 200 m^2 area was selected. The simulation is written in C++, and uses OpenCV for reading in the height map, and matrix data structures and algebra. A parameterized number of UAVs travel radially away from the center of the map with equal angular spacing between their headings. A simple physics model was used, ignoring winds or control inputs. The UAVs fly at an altitude of 30 m (relative to lowest altitude on map) with a constant velocity of 5 m/s. Each UAVs position and orientation is recorded every 0.01 s.

3.2. Generating Synthetic Images

Given a cameras position, orientation and forward vector, a rotation and translation matrix are constructed to convert world 3D coordinates (centered at the pixel in the first row and first column) to camera centered coordinates. Terrain points are projected into image coordinates. Only points which project onto an 800 by 600 pixel image are retained. The intensity of the points projected onto the image is determined by first taking the inner product of the unit normal to a 3 by 3 window at the terrain point and a direction vector from the terrain point to a simulated light source above the terrain. Then, the previous inner product is scaled by the inner product between the unit normal at the terrain point and the normalized forward vector.

3.3. Preprocessing

To emulate reconstructing topology in environments which may not contain distinctive rock formations, vegetation or buildings, the simulation generates synthetic images with few distinctive features. Thus, feature matching across multiple images to create point correspondences is challenging. To alleviate this difficulty, we first smooth the image pixels by filtering the image with an isotropic gaussian blur of variable standard deviation, and then use a top-bottom hat transform of variable morphological operations to enhance the few features that exist in the image.

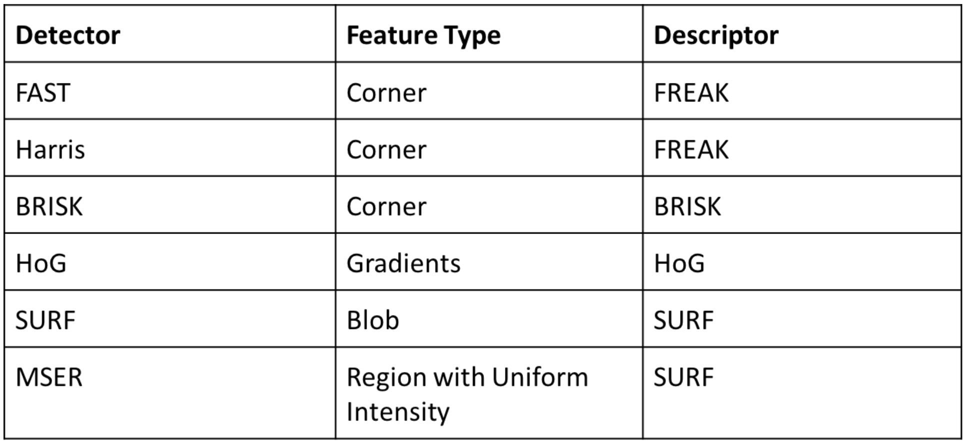
The top-bottom hat transform is a common image enhancement technique for feature detection in images with limited information [4]. The top-bottom hat transform can be described by the following equation:

$$im = im + \phi(s) - \phi(s)$$

where the $\phi(s)$ represents the morphological operation used in the transform. The addition of the operation from the image is known as the top-hat filtering of the image, while the subtraction of the operation is known as the bottom-hat filtering of the image. This transform will have the effect of increasing the contrast between the brightest and dimmest pixels in the image, allowing for improved feature detection.

3.4. Feature detection and descriptors

The viability of multiple feature detector/descriptor algorithms is assessed for our particular application. Figure 2 summarizes this.



Detector	Feature Type	Descriptor
FAST	Corner	FREAK
Harris	Corner	FREAK
BRISK	Corner	BRISK
HoG	Gradients	HoG
SURF	Blob	SURF
MSER	Region with Uniform Intensity	SURF

Figure 2. Summary of the detector/descriptor pairs evaluated for Structure from Motion with synthetic oblique images captured from a simulated UAV on a linear flight path.

3.5. Feature Matching

After using these algorithms to detect the features in each image, features in consecutive images are matched using an exhaustive approach. For each feature in an image, the euclidean distance to every feature in the consecutive image is computed and a match is accepted when the distance between a pair of features is less than a percentage from a perfect match. As an additionally test to eliminate ambiguous matches, if two features in a consecutive image are matched to one in the first, the ratio between the first and second distance is compared to a set value. If the ratio, is above this threshold, meaning the distance between the first or second feature in the image are relatively close, the match is rejected from the output. Lastly, if there are two features in the first image which match to a single feature in the second image, the second match is rejected as well. A variety of values for the two parameters described previously (i.e. percent pairwise distance from perfect match, and ratio between first and second best match) were tested to improve the quality of matches and remove outliers. Figure 3 illustrates the output of the feature matching described above.

3.6. Fundamental Matrix Estimation - Camera Pose Estimation

With correspondences across multiple images, we estimate the fundamental matrix between each pair of images using random sample consensus (RANSAC) and the normalized 8-point algorithm. To summarize, the algorithm executes as follows:

1. Repeat for a set number of trials (e.g. 1000)
   - (a) Randomly select 8 correspondences
   - (b) Compute the fundamental matrix using the normalized 8-point algorithm
   - (c) Compute the number of inliers
2. Select the fundamental matrix corresponding to the maximum number of inliers

Using the fundamental matrix, we can determine the relative translation and rotation between any two pairs of images. We then use this to compute the camera position associated with each image relative to the first image in a set. This yields the extrinsics of each camera in the reference frame of the first camera.

3.7. Multiview Triangulation

Since all the cameras are in the same coordinate system, we can triangulate the location of 3D points from the epipolar geometry of a pair of cameras under the assumption the intrinsic parameters of the cameras are known. Figure 4 depicts how a 3D estimate is computed with two images by projecting two lines into 3D space: the line from the location of the first camera to the corresponding point in the first image and the line from the location of the second camera to the corresponding point in the second image. The intersection of these two lines is the estimated 3D point which is estimated up to a scale.

We then use bundle adjustment to refine the estimations of the 3D points and camera poses. The Levenberg-Marquardt algorithm is used in this process to minimize the reprojection error of the estimated 3D point projected back into the set of images. The Levenberg-Marquardt algorithm is a nonlinear optimization algorithm that converges at a local minimum; however, we use our triangulation estimate

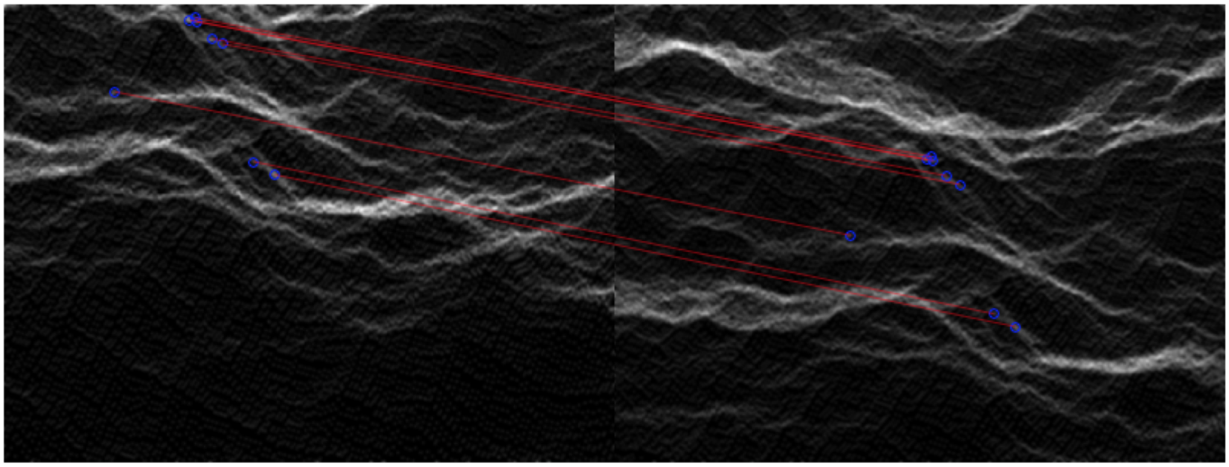


Figure 3. Example subset of correspondences found between two consecutive synthetic oblique images captured from a simulated UAV on a linear flight path. Image features are represented by blue circles, and correspondences between images are connected by red lines.

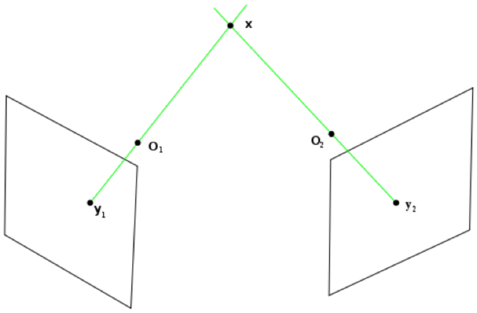


Figure 4. Illustration of 3D point triangulation (x) from correspondences in two images (y1, y2) and estimated camera centers (O1, O2).

as an initial condition to improve the result. The reprojection error is defined as the euclidean distance between the 3D estimated points projection onto the image plane and its associated correspondence point. The result of bundle adjustment is the final estimate of terrain topology and camera poses.

4. Experimental Setup and Results

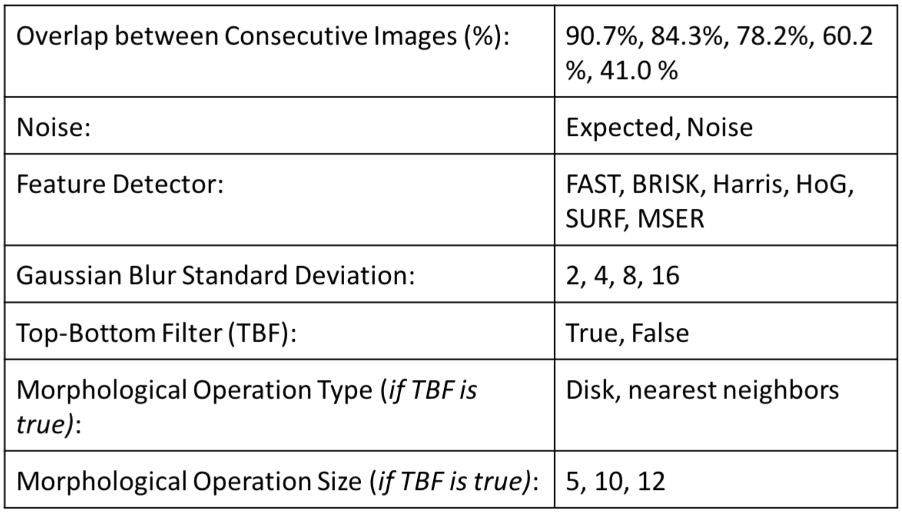
Our experiments involve testing the algorithm using a variety of parameters to determine which methods work best for our specific application. Each of these sets of parameters were tested for each feature detector / descriptor described in figure 2 and for each set of generated images associated with varying levels of noise and percentage of overlap between consecutive image. These tests are summarized in figure 5 where each test used one key-value (first column-second column) for each test.

This generated a test for each combination of percent overlap value, feature detector, and whether or not the images were noised for a total of 140 tests. The median across all 140 tests is presented as the points in figure 6 and 7. Each plot shows the median of the estimated 3D points vs. the percent overlap of the images with lines for each feature detector. Additionally, the first figure is associated with the sets of images that do not contain any noise while the second figure is associated with the sets of images that do contain noise in the rotation and translation of the camera poses.

The figures do not include the trials using the FAST or BRISK features because these techniques did not result in any estimated 3D points because the reprojection errors after bundle adjustment were poor. This may be a result of few features being detected from the images or poor feature matching between consecutive images.

Our results for estimating the location of the 3D points in the map show we could generate estimations within a centimeter or two of a true 3D point. These results may be acceptable for the navigation of an autonomous vehicle although this would depend on the specific application (e.g. planetary or ocean exploration), and the risks associated with the error.

From our experiments with no noise added to the pose of the cameras, our results (Figure 6) for the use of MSER and HoG as features show a steady increase in the error of our estimations as the separation between images increase. This result makes intuitive sense since with greater overlap, there will be many more available points and features to detect and create correspondences. We do not believe the trend for harris corner or SURF features are significant and may depend on variations in the estimation of the fundamental



Overlap between Consecutive Images (%):	90.7%, 84.3%, 78.2%, 60.2 %, 41.0 %
Noise:	Expected, Noise
Feature Detector:	FAST, BRISK, Harris, HoG, SURF, MSER
Gaussian Blur Standard Deviation:	2, 4, 8, 16
Top-Bottom Filter (TBF):	True, False
Morphological Operation Type (*if TBF is true)*:	Disk, nearest neighbors
Morphological Operation Size (*if TBF is true)*:	5, 10, 12

Figure 5. Chart of parameters used in various tests.

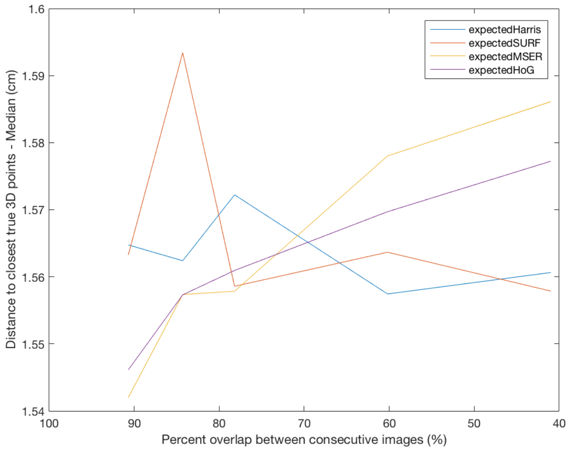


Figure 6. The median distance from a reconstructed point with low reprojection error to its closest point on the terrain as a function of the overlap between consecutive synthetic oblique images captured by a simulated ideal UAV on a linear flight path with perfect pose controls.

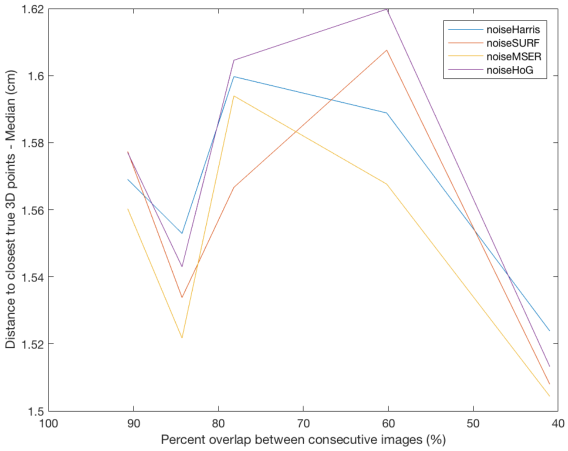


Figure 7. The median distance from a reconstructed point with low reprojection error to its closest point on the terrain as a function of the overlap between consecutive synthetic oblique images captured by a simulated UAV on a linear flight path with imperfect pose controls.

matrix for various parameter sets among other factors.

Figure 7, which is associated with the sets of images with noise added to the camera poses, shows other trends and illustrates the trade-off between errors in triangulation and feature detection / matching depending on the amount of overlap between consecutive images. Every image has different noise in its actual relative to its expected pose. Therefore, the relative heights of the curves are rough indicators of how robust each feature detector is to variance in pose between consecutive images with features like our images. SURF appears to perform best with the fewest images. Then, the curves for all feature detectors are bell-shaped, reaching their peak error at about 70% overlap between consecutive images, and decreasing in error as image overlap increases or decreases from 70%. The decrease in error can be understood from two separate explanations.

First, when there is more overlap between the images, there will be more potential features and therefore more true correspondences across two pairs of images. The increase in the number of correspondences will make execution of RANSAC more robust to outliers in the data, and therefore improve the estimation of the fundamental matrix as well as the estimation of the camera locations used in triangulation. However, since the cameras are closer together, noise will have a greater impact on the estimation of the depth.

Second, when there is less overlap between consecutive images and thus the baseline between two cameras is further apart, triangulation is more robust to noise and will improve estimation efforts; however, there are fewer potential features to be matched and the process of estimating the fundamental matrix could be more susceptible to outliers. We did achieve lowest error with this setting of the experiment where there is a longer baseline. The explanation could be that moderate pose estimation error has less of an effect when images are farther apart than when the images are closer together, but this conclusion should be tested further.

5. Conclusion

This study has developed a software pipeline consisting of a C++ simulation followed by Matlab scripts which generate synthetic images, and a 3D metric reconstruction from these. The pipeline was developed with the case of a UAV on a linear flight path taking oblique images from a single camera. For four feature detector/descriptor pairs, curves representing reconstruction error as a function of image overlap were plotted. Further, the robustness to uncertainty in the cameras pose of the descriptors was characterized for our simulation scenario. This could be useful to indicate how to best process images collected from a real UAV on a linear flight path over a barren, featureless terrain. We have demonstrated the usefulness of our pipeline to assist in designing exploratory missions which seek to map territory. Future work could add complexity to the vehicle dynamics and path planning, explore the effect of terrain roughness, and integrate images from multiple vehicles in the triangulation.

6. References

[1] Wilcox, Brian H. ”Robotic vehicles for planetary exploration.” Applied Intelligence 2, no. 2 (1992): 181-193.

[2] Stoker, Carol. ”The search for life on Mars: The role of rovers.” Journal of Geophysical Research: Planets 103, no. E12 (1998): 28557-28575.

[3] Matthies, Larry, Mark Maimone, Andrew Johnson, Yang Cheng, Reg Willson, Carlos Villalpando, Steve Goldberg, Andres Huertas, Andrew Stein, and Anelia Angelova. ”Computer vision on Mars.” International Journal of Computer Vision 75, no. 1 (2007): 67-92.

[4] Li, Linhui, Jing Lian, Lie Guo, and Rongben Wang. ”Visual Odometry for Planetary Exploration Rovers in Sandy Terrains.” International Journal of Advanced Robotic Systems 10 (2013).

[5] Otsu, K., M. Otsuki, and T. Kubota. ”A comparative study on ground surface reconstruction for rough terrain exploration.” In International Symposium on Artificial Intelligence for Robotics and Automation in Space. 2014.

[6] Ishigami, Genya, Masatsugu Otsuki, and Takashi Kubota. ”Rangedependent Terrain Mapping and Multipath Planning using Cylindrical Coordinates for a Planetary Exploration Rover.” Journal of Field Robotics 30, no. 4 (2013): 536-551.

[7] Dai, Fei, Youyi Feng, and Ryan Hough. ”Photogrammetric error sources and impacts on modeling and surveying in construction engineering applications.” Visualization in Engineering 2, no. 1 (2014): 1-14.

[8] Recker, Shawn, Mauricio Hess-Flores, Mark A. Duchaineau, and Kenneth I. Joy. ”Visualization of scene structure uncertainty in multi-view reconstruction.” In Applied Imagery Pattern Recognition Workshop (AIPR), 2012 IEEE, pp. 1-7. IEEE, 2012.

[9] Rupnik, Ewelina, Francesco Nex, Isabella Toschi, and Fabio Remondino. ”Aerial multi-camera systems: Accuracy and block triangulation issues.” ISPRS Journal of Photogrammetry and Remote Sensing 101 (2015): 233-246.