Robust Tumbling Target Reconstruction through Fusion of Vision and LIDAR for Autonomous Rendezvous and Docking On-Orbit

Jose Padial
Aerospace Robotics Laboratory, Stanford University
450 Lomita Mall, Stanford, CA 94305
jpadial@stanford.edu

Abstract

A preliminary framework for 3D tumbling target reconstruction through fusion of vision and line-scanning LIDAR data is outlined. The utility of using both vision and LIDAR for on-orbit target reconstruction is first presented. The technical approach, including a new camera-LIDAR Structure from Motion (SfM) framework, is next detailed, though this technical framework remains in development. Preliminary results indicate that the method holds potential to perform well, though there remains significant development work ahead to make this a viable, robust solution.

1. Introduction

Target reconstruction is a necessary capability for safe and reliable autonomous rendezvous and docking capability on orbit. Vision is a natural sensor for object reconstruction as it is capable of providing frame-to-frame point correspondence and texture information. The field of Structure from Motion (SfM) is a well-developed one, providing the capability to map a target (structure) and recover the camera motion, up to a similarity transformation (unknown overall scale), assuming calibrated cameras. This scale ambiguity is a problem for real operation on orbit, and cannot be resolved without more information.

Range data (LIDAR) provide 3D structure data directly. When using 3D LIDAR technology (*e.g*. Flash LIDAR), it is possible to solve for scan-to-scan correspondence through alignment of point clouds (typically, with a form of Iterative Closest Point algorithm [1]). Conversely, line-scanning (2D) LIDAR can only solve the correspondence problem in loop closure situations, other than for the degenerate case where the axis of target rotation is perpendicular to the line-scan plane. In terrestrial applications, the use of 3D LIDAR is most likely the correct choice. However, for applications on small satellite chaser vehicles, limitations on power, size, and weight may/will dictate use of the line-scanning LIDAR. The estimation framework proposed here is designed for the camera and line-scanning LIDAR sensor configuration. However, the results will be extensible to use with more complex LIDAR technology.

It should be explicitly noted that this is not a triangulated active illumination system. Active light systems such as the Microsoft Kinect have found great uses in 3D reconstruction. However, these sensor modalities are, in general, highly sensitive to natural lighting conditions - the harsh lighting environment of space all but precludes this form of sensor.

2. Related Work

3D reconstruction is a well-researched field. In computer vision, the problem of recovering camera motion and scene geometry from temporal sequences of images is generally referred to as Structure from Motion (SfM), and has been the focus of a good deal of work. Hartley and Zisserman provide a comprehensive overview of multiple-view geometry algorithms[4], as do Ma *et al*. [9]. SfM algorithms can coarsely be segmented into algebraic and factorization solutions. Algebraic solutions typically rely on iterative techniques whereby the full SfM solution is achieved through alternating solution of camera motion (holding projective depths constant) and projective depth estimation (holding camera motion constant) [9].

Factorization methods, first proposed by Tomasi and Kanade [12], offer a clever solution to the SfM problem whereby the tracking matrix is factorized via the singular value decomposition (SVD) into motion and structure matrices. Affine factorization assumes an affine camera model, and as such does not need to include projective depths in the tracking matrix. Affine factorization methods typically work best for scenes that are distant from the observing cameras and that have little relief. Projective factorization methods, as first proposed by Triggs [13], use a similar factorization scheme with perspective camera models, where the tracking matrix includes a projective depth for each ob-

served feature. The difficulty of using this projective factorization method is in obtaining sufficiently accurate projective depths.

A problem with using factorization methods in general is that they require that all features in the tracking matrix be observed in all of the frames. For situations where lighting is variable or there is the potential for self-occlusion, it cannot be gauranteed that a sufficient number of features will be visible across multiple frames for population of the tracking matrix. In the case of tumbling target observation on-orbit, this is an especially dangerous assumption.

The fusion of range and vision data for 3D reconstruction is not new. Liu *et al*. proposed a method of automatic alignment of 2D image sequences with 3D range data [7]. 3D-3D range registration is performed to produce a dense range point cloud, and an SfM vision solution generates a more sparse (and scale ambiguous) point cloud. The vision and range data is aligned under the assumption that there will be strong horizontal and/or vertical lines present in the scene structure. While this assumption is suitable for many urban/man-made scenes, this is clearly unsuitable for observation of natural terrain/debris (e.g. asteroid) or targets for which we cannot assume there are strong edges. Mastin *et al*. proposed a 2D-3D registration technique based on the maximization of mutual information between 2D images and 3D LIDAR features projected onto the 2D image plane [10]. Their method is specifically aimed at registration of airborne LIDAR measurements with aerial imagery of urban scenes. They explored different methods for evaluating mutual information between images and LIDAR projections, e.g. the mutual information between elevation in LIDAR and luminance in the optical image, where higher elevations of the point cloud are rendered with higher intensities.

Registration of co-located camera/LIDAR systems has been investigated in the context of camera-LIDAR calibration. Zhang and Pless [14] developed a framework for extrinsic calibration of a line-scanning LIDAR to a camera with strong results. The calibration method of Zhang and Pless is used in our work to estimate camera-LIDAR extrinsics. Extensions to their method have been developed to incorporate the use of multi-planar LIDAR [5].

There has been some great progress made in high-resolution 3D reconstruction through fusion of optical imagery and time-of-flight (ToF) ranging. In [3] the authors present an algorithm for improved 3D resolution ("'super resolution"') using Markov Random Fields. More recently, these authors and colleagues have demonstrated in [6] a multi-view system that fuses imagery and ToF sensing for impressive, dense reconstruction. The approach presented in this paper differs from that of [6] in that we do not have a 3D ToF ranging sensor. Instead, for our problem we are restricted to the use of a line-scanning LIDAR. As such, we

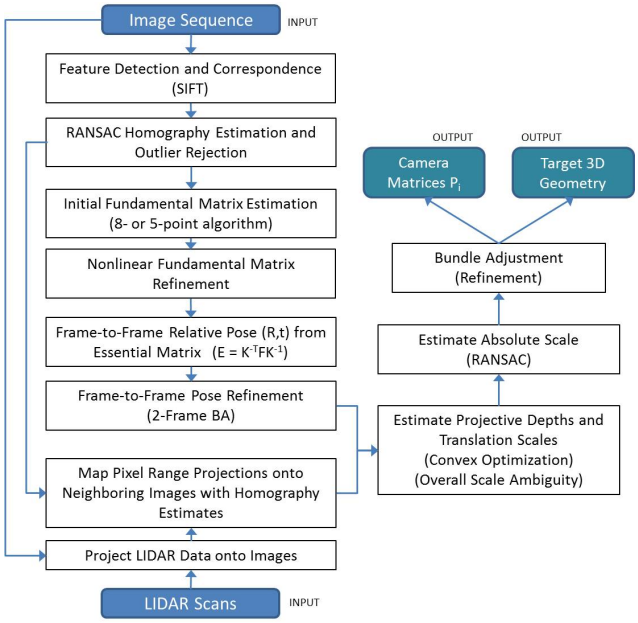


Figure 1: Schematic of algorithmic pipeline.

are provided far less information from the ranging sensor, and must rely more heavily upon the visual SfM solution framework for correspondence.

3. Approach

The approach presented here toward 3D reconstruction through fusion of visual imagery and simple (line-scanning) LIDAR data is evolving. Preliminary results indicate that the foundation of the methodology is promising, but significant development work must be completed before this will be a viable and robust solution for 3D reconstruction. Figure 1 shows the current algorithmic pipeline. It is key to note that this formulation is developed under the assumption that this will be run offline — this is not a real-time formulation, and computational efficiency has not been expressly considered.

3.1. Camera-LIDAR Calibration

The reconstruction algorithm outlined in this paper assumes a calibrated camera-LIDAR system. Extrinsic calibration of the camera to LIDAR (i.e. rotation and translation from LIDAR frame L to camera frame C) is accomplished using the method of Zhang and Pless [14]. This method utilizes standard camera calibration [2]. Hence, we assume for all the following algorithmic detail that we have a fully calibrated system consisting of the camera intrinsic matrix K, radial distortion parameters k_c, LIDAR-to-camera frame rotation matrix ${}^{C}R^{L}$ and camera-to-LIDAR frame translation

${}^{C}t^{L/C}$.

3.2. Robust Frame-to-Frame Feature Correspondence

Frame-to-frame feature correspondence is the first step in our reconstruction pipeline. Feature correspondences can be generated by any suitable method, *e.g*. SIFT, SURF, Harris-Laplace. The choice of feature to use does not affect the structure of our reconstruction solution, with the requirement that the feature be sufficiently robust to illumination and affine transformation variation. In the current implementation SIFT features (keypoints and descriptors) are used for correspondence. Outlier rejection is critical for success of our algorithm downstream of feature correspondence. In order to prune the set of candidate matches to a more accurate inlier subset, we execute a pairwise RANSAC homography estimation. Subsets of the feature matches are randomly chosen (subsets larger than or equal to the minimum 4 required for 2D homography estimation), and a homography model (8-parameter, ambiguous scale) fit by a linear least squares pseudoinverse solution. The model is then applied to the entire candidate match set, and inliers chosen such that the pixel error between the homography-predicted pixel location and the measured pixel location is below a threshold. The homography solution with the highest inlier count is selected as the true model, and only the inlier matches are passed down the pipeline.

3.3. Frame-to-Frame Relative Pose Estimation

Frame-to-frame relative pose estimation is accomplished via two-frame epipolar constraints. Assuming a calibrated camera, the Essential matrix E is estimated using the well-known 8-point algorithm [4], provided that there are 8 or more point correspondences. In the event that less than 8 point correspondences are available (but more than 4), we estimate E using the 5-point algorithm first presented by Nister [11].

Accuracy of the Essential matrix estimation is vital to success of the overall algorithm. Estimation is significantly improved by inclusion of a nonlinear optimization step wherein the estimate of the Essential matrix is refined further. By suitable parameterization of the fundamental matrix, as presented in [9], we can form a nonlinear optimization problem for fundamental matrix estimation subject to the rank-2 constraint. For simplicity, and at the expense of computational efficiency, our fundamental matrix refinement was formulated as an outer-loop/inner-loop formulation, depicted below, where the outer-loop enforces the rank-2 constraint and the inner loop solves an unconstrained nonlinear least squares optimization. Reformulating this as a parameterized nonlinear estimation that enforces the rank-2 constraint is future work, along with performance comparison of the two formulations.

while $\epsilon \geq tol$

$$F^* = \arg\min_F \sum_{j=1}^{n} \tilde{x}_{i,j}^T F \tilde{x}_{i+1,j}$$

$$F^* = U\Sigma V^T$$

$$\Sigma(3,3) := 0$$

$$F := U\Sigma V^T$$

$$\epsilon := \frac{1}{n}\sum_{j=1}^{n} \tilde{x}_{i,j}^T F \tilde{x}_{i+1,j}$$

end while

$$E = K^{-T} F K^{-1}$$

From the Essential matrix E_i, we extract the rotation and translation (${}^{i+1}R^i$, ${}^{i+1}t^{i/i+1}$) between frames C_i, C_{i+1}. A well-known method using the SVD is used to extract rotation and translation from E, yielding four possible solutions, from which one solution is selected based on chirality (triangulated feature depths should be positive in the camera frame). The rotation and translation estimates are further refined in a two-frame bundle adjustment (minimization of re-projection error) for each frame pair.

Outlier rejection can be performed on the frame-to-frame pose estimates by enforcing some measure of camera trajectory smoothness. Especially for high frame rate data, we know that the camera linear and angular velocities should vary smoothly, so we can reject camera motions that imply discontinuous velocity profiles. This has not yet been incorporated into the solution strategy, but would be useful in cases where we know something about the smoothness of our motion, *e.g*. torque free motion of a tumbling orbital target.

3.4. Vision-Range Correspondence

Correspondence between image pixels and range returns is necessary for effective fusion of these two data sources. We assume that the camera and LIDAR are co-located such that the relative translation between the two sensors is small, and as such there is no occlusion from ranged point to the image plane. Under this assumption, and with known extrinsic calibration of the LIDAR to camera (rotation, translation), and known intrinsic calibration of the camera, we can unambiguously project range scans onto the image plane:

$${}^{C}\tilde{x}_j = K[{}^{C}R^{L} \quad {}^{C}t^{L/C}]{}^{L}\tilde{X}_j \tag{1}$$

Where K is the camera intrinsic matrix, ${}^{C}\tilde{x}_j$ is the homogeneous 2D image of ranged point j in camera frame C, and ${}^{L}\tilde{X}_j$ is the homogeneous 3D ranged point in the

LIDAR frame L. Let x_i be an image interest point pixel location, let x_j be a range projection pixel location, let α be a distance threshold, and let M be the set of vision-range matches. A vision-range match is identified by the simple Euclidean distance measure:

$$if\ \|x_i - x_j\|_2 \leq \alpha \rightarrow (x_i, x_j) \in M \tag{2}$$

There is an inherent problem of sparseness in this vision-range correspondence. If we simply look for matches between our range points and robust interest points, *e.g.* SIFT features, then we will have little matches in general for reasonable α. This will diminish the ability to successfully fuse these data.

In addition to this sparse search, we also propose a dense search of range projections between frames. The heart of this dense search lies in our previous estimation of the pairwise 2D homographies between camera frames. Using the homography H_i, which is an estimate of the mapping from frame C_i to C_{i+1}, and given our estimated range projection of 3D feature j onto image I_i, we estimate the projection of the same 3D point j onto image I_{i+1} by the linear homography relation given by (3).

$$\tilde{x}_{i+1}^j = H\tilde{x}_i^j \tag{3}$$

In order to avoid significantly inaccurate vision-range correlation, we avoid applying this homography mapping to points near significant range discontinuities. Although range discontinuities provide useful information on 3D geometry, it is near these discontinuities that errors in our camera-LIDAR calibration can produce the most drastic errors. In order to quickly estimate range discontinuities, a simple difference operator is convolved with the range-scan returns, and values above a threshold are labeled as discontinuities.

3.5. Projective Depths and Absolute Translation Scale

Unlike the canonical SfM vision-only problem, with the addition of LIDAR sensing we can directly measure projective depth to 3D features. Given frame-to-frame relative pose estimates, we can formulate a global optimization problem for the projective scale to each 3D feature, and the proper scale of the frame-to-frame translation. These initial depth and scale estimates can then be used to further refine global pose estimates.

The optimization formulation is adapted from the formulation presented in [9] for two-view triangulation with known relative pose. The formulation makes use of a clever cross-product trick in order to re-shape the problem. This paper adapts this framework into a new convex optimization problem that utilizes the direct measurements of range from range-vision correspondence in order to yield a scale unambiguous estimate of 3D structure and relative camera poses.

Let λ_i^j be the projective depth of feature j from camera frame $\{C_i\}$, measured with image pixel coordinates x_i^j. Let ${}^{i+1}R^i$, ${}^{i+1}t^{i/i+1}$ be the relative rotation and translation estimates from frame $\{C_i\}$ to $\{C_{i+1}\}$. Relative pose estimates are obtained initially from frame-to-frame Essential matrix estimation. Let $\gamma_{i,i+1}$ be the scale factor estimate between the relative translation estimate ${}^{i+1}t^{i/i+1}$ and the true relative translation. Let β_i^k be the projective depth of feature k from the camera center $\{C_i\}$, derived from the LIDAR measurement that was matched with vision feature k by the vision-range correspondence step. Finally, we define $[\tilde{x}_i^j]_x$ to be the cross-product matrix of homogeneous vector $\tilde{x}_i^j$.

$$([\tilde{x}_{i+1}^j]_x\ {}^{i+1}R^i x_i^j)\lambda_i^j + ([\tilde{x}_{i+1}^j]_x\ {}^{i+1}t^{i/i+1})\gamma_{i,i+1} = 0 \tag{4}$$

$$([\tilde{x}_{i+1}^j]_x\ {}^{i+1}t^{i/i+1})\gamma_{i,i+1} = -\beta_i^k([\tilde{x}_{i+1}^k]_x\ {}^{i+1}R^i x_i^k) \tag{5}$$

Relation (4) holds $\forall j = 1, ..., N_{i,i+1}^{\lambda}$, where $N_{i,i+1}^{\lambda}$ is the number of feature matches from frame $\{C_i\}$ to $\{C_{i+1}\}$ that have no vision-range correspondence. Relation (5) holds $\forall j = 1, ..., N_{i,i+1}^{\beta}$, where $N_{i,i+1}^{\beta}$ is the number of feature matches with vision-range correspondence in frame $\{C_i\}$.

We take the relations (4), (5) for all frames i and features j, k, and form a large linear matrix equality $M\lambda = 0$. Although we know the values β_i^k, subject to ranging noise and calibration errors, we treat these projective depths as unknown in the linear matrix equality, and denote them as ${}^{\beta}\lambda_i^k$. The reason for this relaxation is to allow for a camera-consistent estimation of projective depths, which we subsequently scale by our knowledge of β_i^k. Thus, our scale vector is $\lambda = [\lambda_1^1, ..., \lambda_1^{N_{1,2}^{\lambda}}, {}^{\beta}\lambda_1^1, ..., {}^{\beta}\lambda_1^{N_{1,2}^{\beta}}, \gamma_{1,2}, \lambda_2^1, ..., {}^{\beta}\lambda_{m-1}^{N_{m-1,m}^{\beta}}, \gamma_{m-1,m}]^T$.

Further, we formulate constraints across frame-to-frame pairs by incorporating triangulation for features that are observed over 3 (or more) frames.

$$({}^{i+1}R^i x_i^j)\lambda_{i+1}^j - x_{i+1}^j\lambda_{i+1}^j + {}^{i+1}\ t^{i/i+1}\gamma_{i,i+1} = 0 \tag{6}$$

We take these constaints (6), for all frames i, and features j for which the relations can be formed, and construct another linear matrix equality $A\lambda = 0$.

Now we are able to form the following convex optimization problem to solve for our vector λ of unknown projective feature depths and translation scale factors.

$$\begin{aligned} \underset{\lambda,\epsilon}{\text{minimize}} \quad & \|M\lambda\|_2^2 + C\|\epsilon\|_2^2 \\ \text{subject to} \quad & A\lambda + \epsilon = 0 \\ & D\lambda \succeq \zeta \end{aligned}$$

Slack variables ϵ are introduced to allow for minor deviations from the linear constraints. However, we penalize the size of the slack values by inclusion of the term $C\|\epsilon\|_2^2$ in the objective, where C is some (large) positive scalar. Further, we include the elementwise constraint that each projective feature depth is greater than ζ. This requires some knowledge of the distance of the camera from the target, for which we can use our range returns to generate a conservative lower bound ζ. The inclusion of this constraint is a safety precaution against the solution $\lambda = 0$. The matrix D selects only the projective scale depths from λ, omitting the translation scale variables.

Once we have solved for our projective depths and translation scalings λ, we can now remove the overall scale ambiguity by use our vision-range projective depths β_i^k. We solve for the global scale value by RANSAC estimation. RANSAC is preferable to pure least squares estimation in that we can reject outlier values that would weaken the solution. Scale is 1-D, and the number of scale candidates is quite moderate, so we implement RANSAC as a greedy search. For each candidate scale, we select a test scale and find the number of scale candidates within a threshold ratio of the test scale. Upon choosing the largest inlier set, we average the scale factors for our scale estimate $\hat{s}$, and scale the λ vector by $\hat{s}$. At this point we have an initial estimate of camera motion and 3D target geometry with no scale ambiguity (Euclidean solution).

The sparseness of the matrices M, A make this a quickly solvable convex optimization problem. Note that the effectiveness of this solution is entirely dependent on the accuracy of the frame-to-frame relative pose estimates. If our relative pose estimates are not sufficiently accurate, our projective depth and translation scale estimation will be poor. This perhaps points to a key flaw in this methodology versus factorization methods, which simultaneously estimate the motion and the structure. However, as previously stated, factorization methods suffer from the key flaw of requiring that a sufficient number of feature correspondences are imaged over all frames considered. Performance comparison of our algebraic method to factorization methods has not yet been conducted, and is planned as future work. It may be that this algebraic approach can serve as an initialization step to a projective factorization solution, where there is sufficient feature frame-to-frame overlap to allow for a factorization solution. This would replace other methods that seek to initialize projective depths for the projective factorization solution, while in addition removing the overall scale ambiguity.

3.6. Bundle Adjustment

Refinement of the relative pose estimates and projective depths is necessary for accurate target reconstruction. In canonical SfM, this global refinement is termed bundle adjustment (BA), and is typically a minimization of summed, squared reprojection error. We use the well-known Sparse Bundle Adjustment (SBA) software package [8] for bundle adjustment. SBA employs a variant of the Levenberg-Marquardt that is tailored to the sparse nature of the SfM problem, and as such can handle very large SfM problems quickly.

The author believes there is an opportunity to do better than pure vision-only bundle adjustment. There is an opportunity to include 3D range measurmements into the bundle adjustment solution, at the very least in loop closure. Based on our current estimate of structure and motion, which should be close to the true solution for BA to be effective, we can hypothesize loop closure. Based on this hypothesis, we can search correlation of range measurements with the prior map of the region, and fold that into a global optimization formulation. Due to time constraints, this remains as future work.

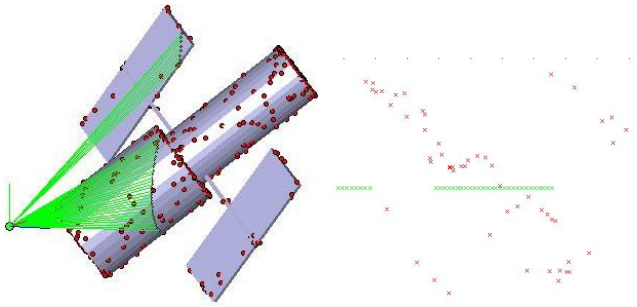
4. Experimental Results

The 3D reconstruction method was tested through simulated and experimental results. First, the framework was tested on data from our tumbling target simulation environment. In the simulation environment, a target model is flown with a specified state trajectory, *e.g.* torque free motion. An observer is populated in the environment, and simulated range returns and images (with simulated image features from 3D points) generated, as shown in Figure 2. From these simulated range returns and images, it is possible to test algorithms in a noise-free environment where perfect truth data is avaialble. Furthermore, this environment allows for noise to be injected at various stages of the measurement pipeline in a known way.

The reconstruction algorithm performed well in noise-free simulations. Unfortunately, due to time constraints in the author's rush to obtain experimental results, error analysis with simulated noise remains as future work.

The results presented herein are from hardware experiments of a target model measured by a co-located camera and URG Hokuyo line-scanning LIDAR in the Stanford Aerospace Robotics Laboratory. Figure 3 depicts the testing environment. The camera resolution and dynamic range are low, which have the dual efffects of making the reconstruction more difficult, while more faithfully approximating the sensing situation that can be expected on-orbit.

Regarding validation, the author ran into an unforeseen problem that must be mentioned. In the Aerospace Robotics Laboratory we have an IR motion detection system capa-



(a) 3D Simulation (b) 2D Synthetic Image

Figure 2: Simulation environment with simulated vision features (red) and simulated range scan (green). (a) 3D geometry and observer range scan simulation. (b) Synthetic image as viewed by observer (green ball with axes) with simulated camera intrinsics and camera-LIDAR extrinsics.

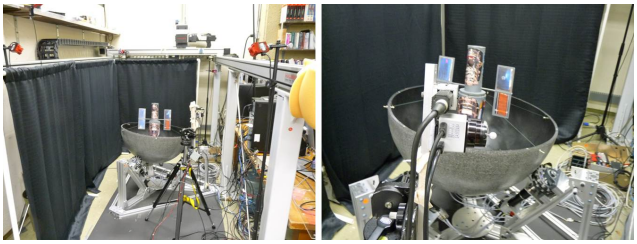


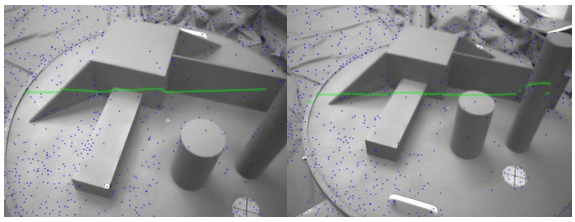
Figure 3: Testing environment in the Aerospace Robotics Laboratory.

ble of high accuracy pose estimation, and this system was planned to be used for validation of camera motion estimation. However, there was a string of problems with the IR system that rendered all of the data virtually useless for the testing conducted. Validation by the IR system remains a very attainable near-term future task, but remains future work nonetheless.

4.1. Camera-LIDAR calibration

Camera-LIDAR extrinsic calibration was accomplished by the method of [14]. Checkerboard images, along with range scans of the checkerboard plane for each image, were input to a calibration engine that includes standard camera calibration software [2]. The accuracy of the camera-LIDAR calibration method in the case of perfect measurements was validated in simulation. Evaluation of the experimental (hardware) camera-LIDAR calibration is more difficult to quantify, but meaningful qualitative evaluation can be derived from analysis of the range projections over distinct 3D geometry, as shown in Figure 4.

The left image of Figure 4 shows strong qualitative correlation between discontinuities in 3D geometry and dis-



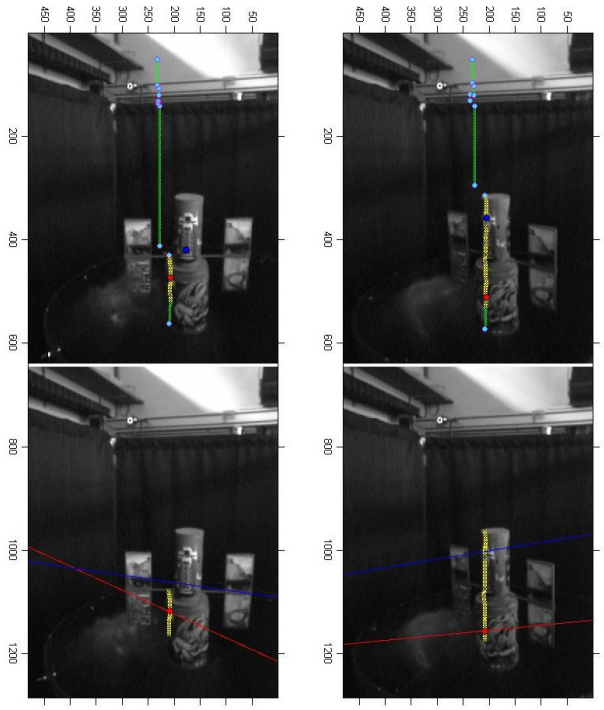
(a) Consistent Alignment (b) Noticeable Occlusion and/or Calibration Error

Figure 4: Projections of range returns on image plane.

continuities in range projections. The right image of the figure also shows good correlation in areas farther from teh camera, but there are clear errors in the range projection around the cylinders. Some of this can be directly attributed to errors in the camera-LIDAR calibration. However, the main cause of these errors is occlusion and violation of the largest assumption of camera-LIDAR correspondence—that they are imaging the same 3D geometry. The LIDAR scanner is roughly 6cm in negative camera Y and 5-6cm in camera X (up and to the right if we are looking down the boresight from the camera center). This explains why the green scan line seemingly passes through the middle cylinder in the right image—the LIDAR has direct line of sight over the edge of the cylinder, whereas the camera does not. Due to the proximity of the 3D target geometry to the camera-LIDAR sensors, these occlusions are pronounced, as the offset of the camera to LIDAR is not insignificant at this projective scale. However, in expected on-orbit sensing situations, the depth to the target is expected to be at least one to two orders of magnitude larger. At these scales, any small error in camera to LIDAR translation estimate and any occlusion due to this offset will be negligible.

4.2. Frame-to-Frame Fundamental Matrix and 2D Homography Estimation

Hardware results for frame-to-frame estimation of the Fundamental (and Essential) matrix are visualized in Figure 5, showing the the qualitative accuracy of homography and fundamental matrix estimates (when things go well, as we will see in the following section). The yellow range points in the upper image map well to the lower image through the estimated 2D homography. Furthermore, the epipolar lines estimated for each selected (red) projected range point passes very nearly through the homography-estimated point. A quantitative comparison of error metrics between two reconstructions based on epipolar constraints and homography estimation will be detailed in the following section.



(a) Frames 1,2 (b) Frames 9,10

Figure 5: Visualization of epipolar geometry estimation, range to image projection, and homography estimation. Each Upper image corresponds to its respective lower image. Green lines in upper figures are projections of LIDAR scans onto images I_i, while cyan points denote labeled range discontinuities. Yellow lines in each upper image are range projections that are mapped to the respective lower image via the homography estimate. Red Points in each upper image are members of the yellow line set, and the lower image red line is the epipolar line estimate corresponding the to upper image red point. Similar for the blue points, which are matched SIFT points between each upper-lower image pair. (a) Upper image is frame 1, lower image is frame 2. (b) Upper image is frame 9, lower image is frame 10.

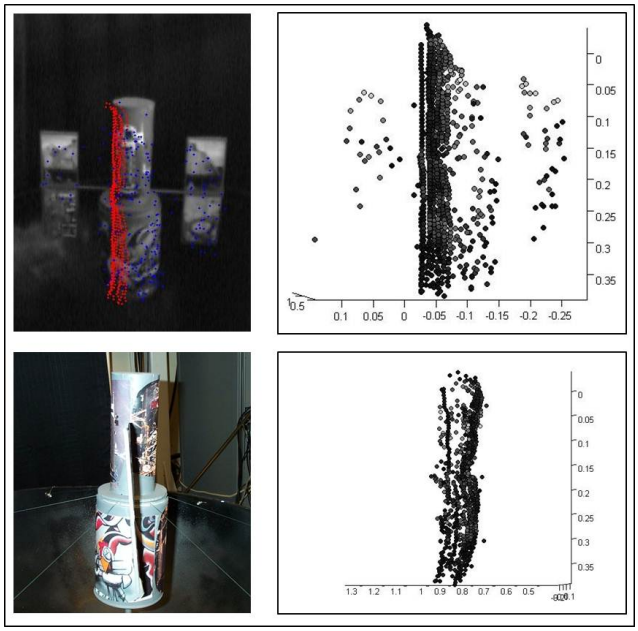


Figure 6: Quasi-successful 3D Reconstruction using 10 camera frames and LIDAR scans.*(upper left)* 3D ranged points (red) and 3D vision-SfM points (blue) are projected into one of the sequence images.*(upper right)* Front-view of 3D structure solution.*(lower left)* Side-view of Hubble target model.*(lower right)* Side-view of 3D structure solution.

4.3. 10-Frame Reconstruction: Quasi-Success and Failure

The 4 frames shown in Figure 5 bookend the 10 frames used to solve the small reconstruction shown in Figure . Though only 10 frames are solved for, the results of Figure **??** show that the algorithm produces a decent initial reconstruction of the ranged target geometry. Furthermore, the structure solution appears to be proximal to true scale. The Hubble model height is roughly 40cm, with a base diameter of 13-14cm. The height and base width of the reconstruction is most certainly in the middle of this ballpark. Note that SBA was not used in this reconstruction – the SBA solution is diverging unexpectedly, and the author suggests there is an implementation issue in his code leading to this erratic behavior (SBA is a well-known, well-validated software package).

Conversely, the 10-frame 3D reconstruction depicted in Figure 7 is clearly troubled. It is very difficult to ascertain the validity of a point cloud solution, especially looking at Matlab plots, but this reconstruction has some clear signs of gross inaccuracy. First of all, the projections of the solved 3D structure points do not correlate well with the image information. This is especially clear in the range projections, where the contours of the projections do not match those of the visual imagery at all. Furthermore, it is clear in the 3D point cloud plot that the range and vision 3D points do not sit well together.

Toward identification of what is fundamentally different between these two 10-frame reconstructions, three frame-to-frame error metrics were calculated for each solution.

$$\epsilon_{Fi} = \frac{1}{N_i} \sum_{j=1}^{N_i} \|\tilde{x}_{i,j}^T F \tilde{x}_{i+1,j}\|_2 \tag{7}$$

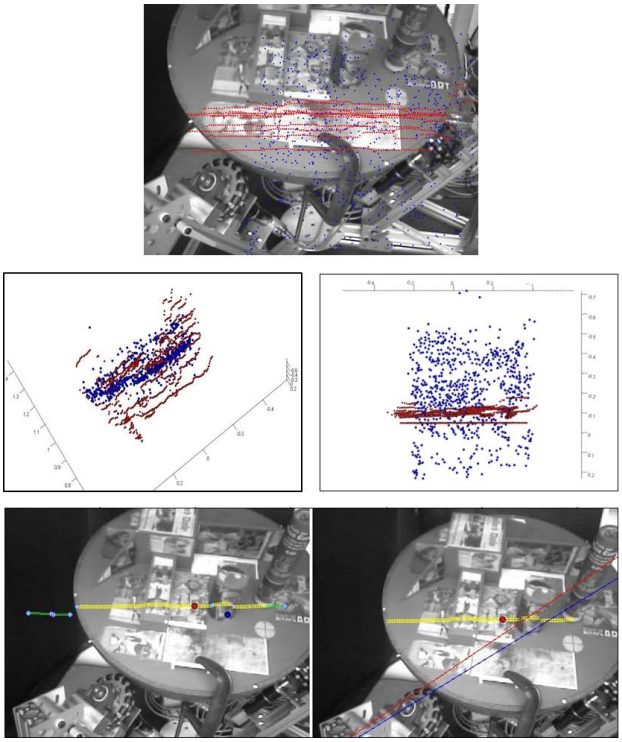


Figure 7: Unsuccessful 3D Reconstruction using 10 camera frames and LIDAR scans.*(top)* 3D ranged points (red) and 3D vision-SfM points (blue) are projected into one of the sequence images.*(middle left and right)* Two views of 3D structure.*bottom* Visualization of epipolar geometry, range projection, and homography mapping – note the lack of correspondence between homography red point and epipolar line on the right.

$$\epsilon_{Hi} = \frac{1}{N_i}\sum_{j=1}^{N_i} \|\tilde{x_i, j}^T F \tilde{x}_{i+1,j}\|_2 \tag{8}$$

$$\epsilon_{Li} = \frac{1}{N_i}\sum_{j=1}^{N_i} \|d(l_{i+1,j}, Hx_{i,j})\|_2 \tag{9}$$

Here $d(l, y)$ is defined as the minimum distance from a line l to point y. The last error metric is a mean sum of distances between the homography mapping of a point and its closest epipolar line point according to the fundamental matrix estimate. Each metric is a scalar error value tied to a particular frame-to-frame pairing. An overall mean and standard deviation were calculated over all such frame-to-frame error values for each solution.

Comparison of these error metrics between the two solutions reveals that there is a large disparity in the fit between the epipolar constraints and the homography mapping in the failure case, as presented in Table 1. These data indicate that while the estimation of the fundamental matrix satisfies the epipolar constraint for the points in the estimation, there is wide divergence from the constraint away from those points. The strange part is that there were very many feature correspondences found frame-to-frame in the failure case. It seems unlikely that a degenerate configuration was found, though many of the surfaces of the failure case hemispherical model are planar. This result remains somewhat of a puzzle to me yet, and is an exciting area to investigate as future work.

Error Metric, in pixels	Trial 1(Quasi-success)	Trial 2(Failure)
$\mu_{\epsilon_{Fi}}$	0.0065	0.0011
$\sigma_{\epsilon_{Fi}}$	0.0059	0.0014
$\mu_{\epsilon_{Hi}}$	0.4087	0.2289
$\sigma_{\epsilon_{Hi}}$	0.2558	0.1776
$\mu_{\epsilon_{Li}}$	0.8085	**22.0092**
$\sigma_{\epsilon_{Li}}$	0.5907	**16.8326**

Table 1: Error Metric Mean and Standard Deviations across Frame-to-Frame Pairs for each Reconstruction Solution

5. Conclusion

A preliminary framework for 3D tumbling target reconstruction through fusion of vision and line-scanning LIDAR data was presented. Early results indicate that the method holds potential to perform well, though there remains significant development hurdles to overcome in order to make this a viable, robust solution. One key flaw of the system is its total relience on highly accurate frame-to-frame pose estimation at the front-end of the estimation chain. This may not be a limitation given highly accurate range sensors and cameras, but how accurate the sensors need to be in order to reasonably guarantee satisfactory performance is unclear to the author. As discussed, one way to potentially overcome this flaw is to incorporate projective factorization methods into this solution methodology, whereby structure and motion is solved simultaneously. However, this introduces different flaws in and of itself, namely that all features in the tracking matrix must be present in all frames for which camera motion is computed. Even if the motion is broken up into sufficiently small submaps, there still exists the real possibility on orbit that there will not be enough concensus across multiple frames.

Projection of line-scanning LIDAR data into images is a difficult problem. Camera-LIDAR extrinsic calibration is flawed, no matter how well it is conducted. The author believes there is opportunity to improve camera-LIDAR checkerboard methodology by developing auto-calibration technology for camera-LIDAR extrinsics, much like the development of auto-calibration procedures in vision-only SfM. This has been somewhat addressed by works such as

[10], however the author believes that the robust solution for a simple LIDAR and co-located camera remains unearthed.

Overall, though 3D reconstruction is a well-research field, there remains great challenges in expanding capability for harsh environments, for targets undergoing aggresive motion, and for simple, low-power and low-weight sensor suites to do more.

References

[1] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14:239–256, February 1992.

[2] J. Y. Bouguet. Camera calibration toolbox for Matlab, 2008.

[3] J. Diebel and S. Thrun. An application of markov random fields to range sensing. In *In NIPS*, pages 291–298. MIT Press, 2005.

[4] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2 edition, 2003.

[5] L. Huang and M. Barth. A novel multi-planar lidar and computer vision calibration procedure using 2d patterns for automated navigation. In *Intelligent Vehicles Symposium, 2009 IEEE*, pages 117 –122, june 2009.

[6] Y. M. Kim, C. Theobalt, J. Diebel, and J. K. B. Miscusik. Multi-view image and tof sensor fusion for dense 3d reconstruction.

[7] L. Liu and I. Stamos. Multiview geometry for texture mapping 2d images onto 3d range data. In *In CVPR 06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2293–2300. IEEE Computer Society, 2006.

[8] M. A. Lourakis and A. Argyros. SBA: A Software Package for Generic Sparse Bundle Adjustment. *ACM Trans. Math. Software*, 36(1):1–30, 2009.

[9] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry. *An Invitation to 3D Vision: From Images to Geometric Models*. Springer Verlag, 2003.

[10] A. Mastin, J. Kepner, and J. Fisher. Automatic registration of lidar and optical images of urban scenes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2639 –2646, june 2009.

[11] D. Nistér. An efficient solution to the five-point relative pose problem. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26:756–777, June 2004.

[12] C. Tomasi. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9:137–154, 1992.

[13] B. Triggs. Factorization methods for projective structure and motion. pages 845–851, 1996.

[14] Q. Zhang and R. Pless. Extrinsic calibration of a camera and laser range finder (improves camera calibration). *2004 IEEERSJ International Conference on Intelligent Robots and Systems IROS IEEE Cat No04CH37566*, 3(314):2301–2306, 2004.