Automatic Registration of Mobile LiDAR and Spherical Panoramas

Ruisheng Wang
NOKIA
Chicago, IL, USA
ruisheng.wang@nokia.com

Frank P. Ferrie
Centre for Intelligent Machines
McGill University, Canada
ferrie@cim.mcgill.ca

Jane Macfarlane
NOKIA
Chicago, IL, USA
jane.macfarlane@nokia.com

Abstract

We present an automatic mutual information (MI) registration method for mobile LiDAR and panoramas collected from a driving vehicle. The suitability of MI for registration of aerial LiDAR and aerial oblique images has been demonstrated in [17], under an assumption that minimization of joint entropy (JE) is a sufficient approximation of maximization of MI. In this paper, we show that this assumption is invalid for the ground-level data. The entropy of a LiDAR image cannot be regarded as approximately constant for small perturbations. Instead of minimizing the JE, we directly maximize MI to estimate corrections of camera poses. Our method automatically registers mobile LiDAR with spherical panoramas over an approximate 4 kilometer drive, and is the first example we are aware of that tests mutual information registration in large-scale context.

1. Introduction

Image-to-range registration is prerequisite for many applications. The registration result is critical not only for texture-mapping 3D models of large-scale scenes, but also for applications such as image based upsampling of range data [6, 8, 21, 24], image-guided range segmentation [4, 2], and 3D scene modeling [5]. The problem of image-to-range registration involves the alignment of 2D images with 2D projections of 3D range data, consisting of estimating the relative camera pose with respect to the range sensors.

There has been a considerable amount of research in registering images with range data. Existing methods range from keypoint-based matching [3, 7, 11], structural features based matching [13, 14, 20, 23], to Mutual Information based registration [17]. The range data include terrestrial or aerial LiDAR, and the images include vertical or oblique aerial images, and ground-level images.

Keypoint based matching [3, 11] is based on the similarity between laser intensity images and corresponding camera images. First, each pixel of the laser intensity image is encoded with its corresponding 3D coordinate. Then feature points are extracted by using either SIFT [16] or Förstner operators [10] from both images. A robust matching strategy based on RANSAC [9] and/or epipolar geometry constraint is employed to determine the correspondence pairs for computing the fundamental matrix. Sensor registration is then achieved based on a robust camera spatial resection. Ding et al. [7] registered oblique aerial images with a 3D model generated from aerial LiDAR data based on 2D and 3D corner features in the 2D images and 3D LiDAR model. The correspondence between extracted corners was based on a Hough transform and generalized M-estimator sample consensus. The resultant corner matches are used in Lowe’s algorithm [15] to refine camera parameters estimated from a combination of vanishing point computation and GPS/IMU readings. In general, the feature point extraction and robust matching are the key to a successful registration for this type of approaches.

Instead of matching points, structural feature based methods [13, 14, 20, 23] match structural features in both 2D and 3D space to estimate the relative camera pose. Direct matching single line features is error-prone because of the noise in both LiDAR and image data as well as the robustness of the detection algorithms. High-level structural features are helpful to increase the robustness of both detection and matching. Wang and Neumann [23] registered aerial images with aerial LiDAR based on matching so-called “3 Connected Segments” in which each linear feature contains 3 segments connected into a chain. They used a two-level RANSAC algorithm to refine the tentative feature matches, and estimated camera pose using the method described in [12]. Liu et al. [13, 14, 20] extracted so called “rectangular parallelepiped” features, which are composed of vertical or horizontal 3D rectangular parallelepipeds in the LiDAR and 2D rectangles in the images, to estimate camera translation with a hypothesis-and-test scheme. Camera rotation was estimated based on at least two vanishing points. Since vanishing points are required, their methods work well for ground-level data but are not efficient to handle aerial data with a weak perspective effect.
All the above methods are dependent on either the strong presence of parallel lines to infer vanishing points, or availability of feature pair correspondence, which limits their applicability and robustness. A recent method [17] using statistical metrics, such as Mutual Information [22], as a similarity measure for registering oblique aerial images and aerial LiDAR does not require any feature extraction process. This method searches for the optimal camera pose through maximizing the mutual information between camera images and different attributes of LiDAR such as the LiDAR intensity images, depth maps, or a combination of both. Instead of using features, the mutual information method evaluates statistical measures using all the pixels in both images, which avoids the problems of feature extraction and correspondence. Thus mutual information registration method holds potential to be a robust solution.

This paper deals with the problem of the registration between mobile LiDAR and spherical panoramas. We adopted mutual information metric for the similarity measure. The paper [17] registers aerial LiDAR with aerial oblique images. The assumption is that the entropy of the LiDAR images remains approximately constant for small perturbations, and minimizing the joint entropy is equivalent to maximizing the mutual information. This may be the case for the airborne data, but is not suitable for our ground based applications. The small perturbations may have larger effect on the LiDAR rendering compared with airborne case. Instead of minimizing the joint entropy, we maximize the mutual information with a different implementation to [17]. Our method is complementary, since we deal with the data collected from ground level. Second, our algorithm is fully automatic and has been designed to run efficiently on a metropolitan scale LiDAR/spherical image database using different representations of LiDAR from [17]. To the best of our knowledge, our work is the first registration example that automatically runs through a large-scale database in the context of mutual information registration.

2. Data Acquisition

Data is collected from a mobile mapping system shown in Figure 1, which is composed of a 360° LiDAR sensor (Velodyne HDL-64E), six high-resolution cameras, a Ladybug 3 camera, GPS, Inertial Measurement Unit (IMU), and Distance Measurement Instrument (DMI). The Velodyne LiDAR sensor consists of 64 lasers mounted on upper and lower blocks of 32 lasers each and the entire unit spins, and generates over one million points per second. The Ladybug 3 covers more than 80 percent of a full sphere, with six high quality 1600x1200 Sony CCD sensors, and provides up to 12 MP images at 15 Frames Per Second (FPS). All of these sensors are geo-referenced through a GPS and IMU.

3. The method

We start with the work [17] that registers aerial LiDAR with aerial oblique images based on MI. LiDAR intensity images normally look very similar to gray-scale camera images with, of course, a much lower resolution. This correlation makes MI a suitable measure to evaluate their similarity. In [17], Mastin et al. define \( p(x, y) \) and \( l(x, y) \) as the intensity camera image and projected LiDAR features respectively on the \( xy \) image plane. For a specific camera matrix \( T \), the projected LiDAR features is given by \( l_T \).

MI based registration methods find the optimal camera matrix that maximizes the MI between camera images and projected LiDAR features:

\[
T_{MI} = \arg \max_T I(p; l_T). \tag{1}
\]

Mutual information is expressed in terms of entropy.

\[
I(p; l_T) = H(p) + H(l_T) - H(p, l_T). \tag{2}
\]

where \( H(p) \) is the entropy of the optical features, \( H(l_T) \) is the entropy of the LiDAR features, and \( H(p, l_T) \) is the JE. Mastin et al. [17] assume that the minimization of the JE is an approximation of the maximization of the mutual information according to Equation 2. They minimize Equation 3 instead of maximizing MI over \( T \) for the registration.

\[
H(p, l_T) \approx \sum_{i=1}^{N_u} \sum_{j=1}^{N_v} \hat{d}(p_i, l_j; T) \log(\hat{d}(p_i, l_j; T)), \tag{3}
\]

where \( \hat{d} \) denotes a joint histogram estimated of a density and \( N_u \) and \( N_v \) denote the number of distinct bins for each modality. In their case, \( N_u, N_v = 256 \).

They use a generic camera model, the finite projective camera as described in [12]. Under this camera model, a point in space is mapped to the point on the image plane by

\[
P = KR[I | -C]. \tag{4}
\]
Where \( C = [C_x, C_y, C_z]^T \) is the camera center, \( I \) is the identity matrix, and \( R \) is the camera rotation matrix. The matrix \( K \) is the camera calibration matrix.

In this paper, we show that their assumption is invalid for our case. The small perturbations in camera poses may have larger effect on the LiDAR rendering in the ground-level data. The entropy of a LiDAR image can not be regarded as approximately constant for small perturbations. This is demonstrated by a perturbation analysis shown in Figure 2, which shows how the normalized MI and JE vary around the initial registration in terms of altered camera poses. The left columns show the results for the normalized MI, and the right columns show the results for the normalized JE. We use four representative scenes for this test as shown in Figure 2. Since the correct registration value should be near the initial registration, we set all parameters at their initial values and vary each parameter to view the shape of the cost functions. The range of camera parameter perturbations are \( \pm 2 \) units, meters for translation and degrees for orientation. The step size for the perturbation analysis is 0.1 units. So we have 40 measurements for each camera parameter perturbation analysis, and the the number 0 in X axis corresponds to the initial registration, where camera pose corrections are set to zero. The data 1, 2, 3, 4 in Figure 3 show the results of the probe analysis, which corresponds to the image 1, 2, 3, 4 of Figure 2, respectively. The data 5 in Figure 3, which are indicated by solid black lines, are mean curves of the four curves showing a general trend. For all the probe experiments, a global maximum within the tested camera parameter range in terms of normalized MI can be always found around the initial registration. However, this is not the case for normalized JE, which a global minima can not be found around the initial registration. The minimization of the JE does not work on our ground level mobile LiDAR data. Instead, we maximize the MI to obtain correct camera solutions.

Figure 2: Probe analysis on four representative scenes

Figure 3: Probe analysis on camera translation and orientation parameters for evaluation of normalized MI and JE
3.1. Coordinate Framework

Our optical images are panoramic images, which are originally taken from Ladybug III consisting of 6 Fish eye cameras. The 6 individual images are stitched via $\alpha$ blending. For storage purpose, the stitched images are transformed to panoramic images via a Cylindrical Equidistant Projection as shown in the left image of Figure 4, which is the original panoramic images. We map each panoramic image onto a sphere and view it from the sphere center (the center of the camera), to generate spherical panoramas (linear perspective images) as shown in the right image of Figure 4.

![Figure 4: Panoramic and spherical view](image)

(Left) A panoramic view; (Right) A spherical view

Both LiDAR and image data are geo-referenced. We first convert the geographic coordinates into Earth-Centered, Earth-Fixed (ECEF) coordinates, and then transform into local tangent plane coordinates. Each LiDAR point $p = (x, y, z)^T$ in LTP coordinates is converted to spherical coordinates $(\theta, \varphi)$ by Equation 5,

$$\theta = \arccos\left(\frac{z}{\sqrt{x^2 + y^2 + z^2}}\right), \quad \varphi = \tan^{-1}(y, x), \tag{5}$$

where $\theta$ is the inclination ($\theta \in [0, \pi]$), and $\varphi$ is the azimuth ($\varphi \in (-\pi, \pi]$). Each point’s corresponding location in the panoramic image $(r, c)$ is computed by Equation 6,

$$r = \text{int}[\frac{\theta}{\pi}H], \quad c = \text{int}[\frac{\varphi}{2\pi} + 0.5]W, \tag{6}$$

where $H$ and $W$ are the height and width of the panoramic images respectively.

3.2. Mutual Information Registration

Mutual Information methods have been widely used for the multi-modal registration problem in the medical imaging domain (e.g., registration of CT and MRI). Recently they also have been applied to the problem of registering airborne LiDAR data with oblique aerial images [17]. The mutual information of two random variables $X$ and $Y$ can be defined as

$$I(X; Y) = \int_Y \int_X p(x, y) \log\left(\frac{p(x, y)}{p_1(x)p_2(y)}\right) dx dy, \tag{7}$$

where $p(x, y)$ is the joint probability density function of $X$ and $Y$, and $p_1(x)$ and $p_2(y)$ are the marginal probability density functions of $X$ and $Y$ respectively. We assume that an approximate camera pose is provided from the imperfect GPS/IMU solution. The problem here is to estimate the correction of the relative camera pose between the LiDAR sensor and the Ladybug camera. The spherical panorama is chosen as a fixed image because the camera view point has to stay in the center of the sphere to generate perspective images. Once the camera moves out of the sphere center, the spherical image will be distorted. The LiDAR image is selected as a moving image, where new LiDAR images are generated at each iteration during the optimization process. Both LiDAR and spherical images are rendered onto a plane from the camera center using OpenGL for the MI evaluation under a pinhole camera model. The perspective camera image is generated by rendering the spherical panorama with a view port of 940 by 452 pixels. The LiDAR data set is normally very large. In our experiments, each scene contains 8 million LiDAR points. To make 3D rendering efficient, we also integrate the OpenGL rendering of the LiDAR features into the registration pipeline to speed up the optimization process.

We use three different representations of the LiDAR data with spherical panoramas for evaluating mutual information. The first representation of LiDAR is the projected LiDAR points with intensity information (see Figure 5b). We call it a LiDAR intensity image which looks similar to a gray-scale camera image. The second representation is the projected LiDAR points without intensity information (see Figure 5c). The third is the depth map of the LiDAR point cloud (see Figure 5d). The point cloud is rendered with depth intensities, where brighter points indicate a further distance to the camera center. We use gray scale camera images instead of color images (see Figure 5a) for the mutual information computation. The Nelder-Mead downhill simplex method [18] is used to optimize the cost function as it does not require derivatives of the function.

4. Experiments

For the experiments, we made the algorithm automatically run through an approximate 4 kilometer drive. The driving routine is shown with light blue line in Figure 6a. An illustration of the collected data is shown in Figure 6b, where the distance between each bubble is around 4 meters. The test data were collected in the northwestern suburban of Chicago, Illinois, which includes residential, urban streets, and highway scenes. The data is in binary format containing around 4 GB LiDAR data (about 226 million points) and 1 GB panoramic images (814 spherical panoramas). We use the camera views perpendicular to or parallel to the vehicle driving direction to generate perspective images. For qualitative analysis, we selected 10 representative urban scenes.
for the evaluation using the three different representations of the LiDAR data described earlier.

We start with an approximate initial registration that is available from the GPS/IMU system. The initial camera pose corrections are set to zero. The optimization will compute the final camera corrections. The experiments for the qualitative analysis were performed on a laptop PC with a dual core 2.60GHz processor and 2 GB of RAM. The NVIDIA Quadro NVS 135M video card was used. The registration algorithms were implemented in C++, and the implementations of Mutual Information and Amoeba optimization were from ITK [1]. We adjust the tolerances on the optimizer to define convergence. The tolerance on the 6 parameters is 0.1 (the unit for translation parameters is meter and degree for orientation parameters). We also set the tolerance on the cost function value to define convergence. The metric returns the value of mutual information, which we set the tolerance to be 0.001 bits. The initial size of the simplex is set to 1, and the maximum iteration number is set to 200. In our experiments, almost all registrations converged in less than 150 iterations.

4.1. Performance Evaluation

To quantitatively measure the registration results, we compare the registration accuracy in terms of pixel offset between LiDAR and camera images before and after the registration. We manually selected 15 correspondence points in each spherical image and LiDAR intensity image. Figure 7 shows an example of a spherical image and a LiDAR intensity image marked with 15 correspondence points. Figure 8 shows computed Euclidean Distance histogram of the correspondence points for all the 10 images before and after registration. The horizontal axis stands for the pixel offsets, and the vertical axis stands for the frequency. Before the MI registration, most correspondence points have 2-3 pixel errors. After the registration, most of the correspondence points are within 1 pixel. The pixel offset histograms using other LiDAR representations are similar. Table 1 show the run time for the 4 representative scenes. Testing on the 4km drive shows that using the LiDAR points without intensity normally runs fast with less iterations. Using LiDAR points with intensity normally performs the most robustly followed by using LiDAR points without intensity and using the LiDAR depth maps. We also study the convergence of the optimization using three different measures of mutual information. Without loss of generality, we choose the data shown in Figure 5 as an example. Figure 9 shows the sequence of metric values computed as the optimizer searched the parameter space using these three different representations of the LiDAR data. The measure initially increases overall with the number of iter-

Figure 5: Camera image and three representations of LiDAR point clouds in a same scene

Figure 6: Test data.

Figure 7: Optical image (a) and LiDAR image (b) with 15 manually selected correspondence points
ations. After about 50 iterations, the metric value reaches steady state without further noticeable convergence.

Figure 8: Histogram of Euclidean distance of pixel offset for before registration (a) and after registration (b). The histograms were generated with samples for all ten test images.

Figure 9: Mutual information values produced during the registration process.

An example of the registration is shown in Figure 10. After MI registration, the misalignment is not noticeable.

<table>
<thead>
<tr>
<th>MI Measure</th>
<th>LiDAR intensity</th>
<th>LiDAR without intensity</th>
<th>LiDAR depth map</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image 1</td>
<td>0.93</td>
<td>0.50</td>
<td>1.08</td>
</tr>
<tr>
<td>Image 2</td>
<td>0.87</td>
<td>0.55</td>
<td>0.88</td>
</tr>
<tr>
<td>Image 3</td>
<td>0.70</td>
<td>0.4</td>
<td>1.06</td>
</tr>
<tr>
<td>Image 4</td>
<td>0.73</td>
<td>0.38</td>
<td>0.87</td>
</tr>
<tr>
<td>Mean</td>
<td>0.88</td>
<td>0.47</td>
<td>0.87</td>
</tr>
</tbody>
</table>

Table 1: Registration times in minutes for the 4 images in Figure 2

4.2. Perturbation Analysis

We plot the normalized MI of Figure 5a in Figure 11 using the three LiDAR attributes. Figure 11 (red-intensity, green-point only, yellow-depth map) demonstrates that each curve has a single peak over a subset of the displacement parameters around the initial registration, which demonstrates the effectiveness of maximization of MI for computing optimal camera corrections.

Figure 11: Plots of normalized mutual information
We also investigate the failure cases. In our experiments, the algorithm works well in feature-rich environments such as residential areas, but often fails in scenes with sparse features or containing moving objects like cars, particularly highway scenes. In our case, the highway scenes mostly fail. The partial overlap between LiDAR point clouds and camera images is another reason. The LiDAR scanner only can reach up to 120 meters, while the camera can always have a larger field of view than the LiDAR scanner. Figure 12a and Figure 12b shows one typical failure case in a highway scene. The cars in the camera image (Figure 12a) don’t appear in the LiDAR image (Figure 12b). The LiDAR image only partially covers the camera image, for instance, the trees and buildings in the far distance in the camera image don’t appear in the LiDAR image. In [17], the authors claim that they only use the image pixels with corresponding projected LiDAR points for mutual information calculation and others are considered background points and discarded. We tried to discard the background points and only use overlapping regions for mutual information computation, but the results were worse than using entire images for MI computation. When using entire images for MI computation, the background such as sky appears similar in both LiDAR and camera images, which largely contributes the MI score. When using overlapping regions for MI computation, the LiDAR images contain no sky. Therefore the background is not used in the MI computation, which affects the MI evaluation. Failure cases are also due to overexposed images (Figure 12c and Figure 12d), particularly in the case where the vehicle drives through/out of a tunnel.

4.3. Camera Pose Corrections

One of our interests is to investigate how camera pose errors change during the data collection. To do so, we manually selected 100 successful registrations (using the registrations from camera views vertical to the vehicle driving direction) by carefully examining the alignment of major features in the registered images, and plot the camera pose corrections shown in Figure 13. Figure 13a shows the camera translation corrections, and Figure 13b shows the camera orientation corrections. Our observation is that almost all the camera translation corrections are within 0.1 meter, while orientation corrections are within 1 degree.

5. Conclusions and Future Work

In this paper, we have investigated MI registration for ground level LiDAR and images. The existing method [17] for registering airborne LiDAR with aerial oblique images does not work on the LiDAR and images collected from the mobile mapping system because the assumption used in [17] is violated in the case of mobile LiDAR data. Instead of the minimization of the joint entropy, we use maximization of mutual information for computing optimal camera corrections. The algorithms work with unstructured LiDAR data and perspective rectified panoramic images generated by rendering panorama into an image plane using spheric views. We tested the algorithm on various urban scenes using three different representations of LiDAR data with camera images for the mutual information calculation. Our mutual registration algorithm automatically runs through a large-scale mobile LiDAR and panoramic images collected over a metropolitan scale. It is the first example we are aware of that tests mutual information registration in large-scale context. With the initiative of urban 3D modeling from location based service providers such as Nokia and Google, this work is particularly important for combining ground-level range and visual data for large-scale photorealistic city modeling.

We generated perspective images from spherical images using the view either perpendicular or parallel to the vehicle driving direction. Therefore, we just used 1/6 of the entire spherical image for the mutual information registration, which does not efficiently use all the available information containing in the 360° panoramic images. For the future work, one possible approach is to project entire Li-
DAR points along with spherical images onto 6 cube faces using a Quadrilateralized Spherical Cube mapping [19] or other linear projections. Because the sky and the ground do not provide much useful information, we actually just need 4 faces for the mutual information registration. To speed up the computation, a multi-resolution approach can be employed by establishing image pyramids on both images. This coarse-to-fine strategy can improve the performance of the registration algorithm and also increases robustness by eliminating local optima at coarser level. One of the limitations of MI metric is that the intensity histograms contain no spatial information. One possible direction is to incorporate spatial context information into the metric to improve the robustness of the similarity measure.

6. Acknowledgement

The first author would like to thank Roman Ostrovskiy for his help and discussion.

References