Graph-based Automatic Consistent Image Mosaicking [1]

Pifu Zhang[†2] *Evangelos E. Milios*[†] *Jason Gu*[‡]
[†]Faculty of Computer Science
[‡]Department of Electrical and Computer Engineering
Dalhousie University
Halifax, Canada, B3H 1W5
{*pifu,eem*}*@cs.dal.ca* *jason.gu@dal.ca*

Abstract

Consistent image mosaicking is a potentially useful tool for robot navigation and map construction. This paper presents an automatic algorithm to generate consistent image mosaicking. During the robot processing, local optimization based on the related previous images is used for every newly added image on the baseline approach view. As soon as a loop is detected, a global optimization method is activated to generate a globally consistent image mosaicking. This method is very efficient in computation and storage saving.

Keywords: image mosaicking, robot navigation, global optimization

1 Introduction

Image mosaicking started at the beginning of photography to obtain images with a larger field of view by assembling two or more individual overlapping images. Recently, automatic image mosaicking has being attracted many researchers in computer vision, image processing, and computer graphics.

The construction of mosaics from images begins with the alignment of successive overlapping images. Using their relative positions, the images can be integrated in a single large picture. Image alignment is not a difficult problem in mosaic construction by appropriate modeling of projective transformation and lens distortion and sufficient image overlap [15]. The biggest challenge is to construct a consistent mosaic with long image sequences.

No matter how accurate image alignment can be achieved, there is always an alignment error. So with long image sequence alignment, the accumulated errors will be very significant. This kind of mis-alignment is obvious when we take the images in a cycle. The first image and the last image will not match perfectly, and a gap may exist after the pairwise alignment.

In this paper, we develop two step processes, local optimization and global optimization, to ensure the consistent image mosaic. Local optimization uses the adjacent overlapping images to get locally optimal registration; global optimization uses all the images in a loop to get a globally consistent image mosaic. Our goal is to construct a consistent mosaic in the ocean floor environment which will be used for the AUV's navigation.

2 Related Work

There are many papers in image mosaic which have dealt with the problem of alignment images [3, 7, 15, 16]. Gonzalez [3] introduced a network to represent the images in a loop, and then distributed the accumulated error to all the connections by minimizing the sum of absolute error and minimizing the biggest error of all connections.

In order to eliminate accumulated error during the processing of frame-to-frame concatenation between the adjacent frames, Hsu [7] carried out local alignment by using the feature-based technique between the frame and the partial image mosaic after the first step alignment.

In the paper [15], Shum proposed a method to construct the image mosaic with global and local alignment. In the global alignment step, he used block adjustment to modify the difference between ray directions going through corresponding points. The global alignment preserves the mosaic's consistency and the local alignment preserves the detail of the images.

Unnikrishan [16] proposed a distortion-free approach for image mosaicking in a local and global sense based on feature correspondence. Common features between

[1]Funding for this work was provided by NSERC Canada and IRIS NCE
[2]The author is also with Computer Science Department, Changsha University of Science and Technology

images are used as constraints in a global optimization, plus the additional constraints that come with cycles.

Unlike early methods that dealt with alignment of images, consistent image mosaicking is closely related to consistent map establishment of the autonomous robot. Lu and Milios [10, 11] studied consistent global estimation of a metric map constructed using laser range data. Their method maintained a history of all the local frames of sensor data used to construct the map and the network of spatial relations between the frames. A maximum likelihood algorithm was used to get a position estimate for each of the frames by minimizing the Mahalanobis distance between the actual and estimated relations over the whole network of frames. Gutmann and Konolige [4] extended this method to build maps on the environment with large cycles in a more computationally efficiency.

Data registration in three-dimensional space is very difficult because it depends on six degrees of freedom. Sharp et al [14] considered the problem of multi-view registration from range data. He firstly used the idea of decoupling the analysis of the rotation and translation, which can be analyzed separately. Then consistent rotation estimation is achieved by distributing the accumulated angular error among all the links in the cycle. The translational estimation between consecutive views involves minimizing the sum of translation around each cycle.

Madjidi and Negahdaripour [12] dealt with the $3D$ registration problem, but used a totally different way. They applied the mixed-model least squares estimation paradigm to develop recursive estimation algorithms for scene images. Position estimation is implemented with different noise levels in observation.

Our method also addresses the local and global alignment problem for image mosaicking. We follow the Lu & Milios' philosophy and use the idea by Sharp [14] and Duckett [1] to separate rotation and translation. During the locally consistent alignment step, we will use the redundant information to get the optimized solution in the maximum likelihood sense for the newly added image; when a cycle is detected, the globally consistent alignment step is activated. Unlike Sharp's approach, we establish an objective function by using maximum likelihood subject to the rotational constraints for the rotation. The solution for the rotation is used to construct the constraints for the translational optimization.

This paper is organized as following. In section 3, image transformation parameters are estimated based on the feature correspondence between images. Section 4 establishes the approach to achieve globally consistent image mosaic while a loop is detected. Section 5 presents a method to deal with local alignment based on the baseline method. In section 6 simulation results and experiment results are displayed.

3 Image Transformation Estimation

During the image mosaic construction, two of the adjacent and partially overlapping images, will be aligned by coordinate transformation. A common assumption is that the coordinate transformations between images are semi-rigid planar models, which is expressed by the following equation.

$$\begin{pmatrix} x_j \\ y_j \end{pmatrix} = s_{ij} R_{ij} \begin{pmatrix} x_i \\ y_i \end{pmatrix} + \begin{pmatrix} tx_{ij} \\ ty_{ij} \end{pmatrix} \qquad (1)$$

This equation indicates that the pixel $(x_i \ y_i)^T$ of image I_i is mapped to the pixel $(x_j \ y_j)^T$ of another image I_j by the scaling s_{ij}, rotation R_{ij} ($R_{ij} = R(\theta_{ij})$), and translation $(tx_{ij} \ ty_{ij})^T$.

This model is sufficient to match two images of a scene taken from the same viewing angle but from different positions [6]. From equation (1), we know that the most important step for the image transformation is to establish the point correspondence.

3.1 Evenly Spaced Features

Many features can be extracted from the image, such as corner, line, edge, color, illumination, texture, histogram, and SIFT feature [9]. In this paper, we use the Harris method to find corners in the image. Then the image feature correspondences can be obtained by using a SSD (Sum of Squared Differences) method. From these feature correspondences, the image transformation parameters in equation (1) can be obtained by using Least Squares method. Both of the steps are fast and stable, but the transformation parameters may not be good enough if the features are not distributed on most of the images. So an evenly spaced feature is introduced in this paper to refine the transformation parameters estimation.

Figure 1 Feature correspondences on images. Right images display the corner features with red cross, and left are the evenly distributed features.

From the image I_i, we generate an evenly spaced grid point set P_i^k, $(k = 1, \cdots, N)$, and all the grid points can be transformed to image I_j by the previous transformation, and we obtained P_j^k, $(k = 1, \cdots, M_i)$ with $(M_i \leq N)$. Since the unevenly distributed corner features in the previous steps, the point P_j^k may not the right correspondence. So we use a SSD again to search on the small range and then find the best matched feature points P_j^k. Now the evenly distributed feature points P_i^k and P_j^k can be used to estimate the transformation again, and the result is much better than the previous result.

The mismatching happens in some case, which will cause the transformation estimation far away from the right value. These outliers can be eliminated by performing RANSAC algorithm [2, 5]. So in every step of estimation the transformation parameters, we used RANSAC to eliminate the outliers.

3.2 Uncertainty modeling

In the feature point correspondence of the previous section, residual errors of the estimation arise due to image noise. If the image measurements are uncorrelated, the error can be modelled as a diagonal covariance matrix with entries σ^2. From the equation (1),if we take the tx_{ij},ty_{ij}, s_{ij}, and θ_{ij} as independent variable, then we can get the Jacobian of the correspondence as following.

$$J_{ij} = \begin{pmatrix} 1 & 0 & (-s_{ij}x_i c\theta_{ij} - s_{ij}y_i s\theta_{ij}) & (x_{ij}c\theta_{ij} - y_{ij}s\theta_{ij}) \\ 0 & 1 & (s_{ij}x_i c\theta_{ij} - s_{ij}y_i s\theta_{ij}) & (x_i s\theta_{ij} + y_i c\theta_{ij}) \end{pmatrix} \tag{2}$$

where $c\theta_{ij} = cos\theta_{ij}$ and $s\theta_{ij} = sin\theta_{ij}$. Then the uncertainty of estimation of the transformation parameters can be obtained by [8] as following.

$$P_{ij} = \sigma^2 (J_{ij}^T J_{ij})^{-1} \tag{3}$$

4 Globally Consistent Image Mosaic

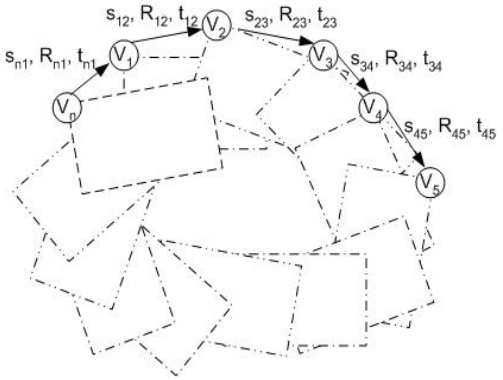

Figure 2 A cyclic graph for image mosaicking. V_i is the upper right corner of the image I_i and S_{ij}, R_i, t_{ij} are the transformation parameters (actually, the transformation center is at the image center).

Suppose that we have a cyclic graph $G = (V, e)$, which consists of vertices $V_1, V_2, \ldots, V_n$, and correspondent links shown in Figure 2. This is the very basic model to simulate the image mosaic with a loop.

Due to the error in every alignment step, the accumulated error will result in an inconsistent matching (gap) in a loop. When in the long cycle case, the misalignmnent between the first image and the last image in the circle may be very significant. So this gap must be distributed evenly to ensure the consistency of the image mosaicking.

This is the same problem as in 2D mapping with loop for an autonomous robot using laser range data [10]. An effective algorithm to construct a 2D map using compass to measure orientation was proposed by Duckett [1]. This gave us the idea to decouple the rotation and translation. Sharp [14] used this idea for $3D$ image registration. In this paper, we will follow [14] to estimate the optimal rotation and translation, respectively.

4.1 Consistent Rotation Estimation

In the case of loop, if all the measurements and calculations are perfect (see Fig. 2), then it should have the following expression:

$$R_{12}R_{23}\cdots R_{ij}\cdots R_{n1} = I \tag{4}$$

where I is the identity matrix. Since there always exist measurement error and calculation error, equation (4) will not hold exactly. According to Lu and Milios' strategy [10], the accumulated rotation error must be distributed to all the links of the cycle by the following optimization function

$$\Omega = \frac{1}{2}\sum_{i=1}^{n}(R_{ij} - \hat{R}_{ij})^T C_{ij}^{-1}(R_{ij} - \hat{R}_{ij}) + \lambda(\hat{R}_{12}\ldots\hat{R}_{n1} - I) \tag{5}$$

where $j = i + 1$ (if $j > n$, then $j = 1$), R_{ij} is the estimated value of the rotation, $\hat{R}_{ij}$ is the optimized value of the rotation, C_{ij} is the variance of the R_{ij}, which is equal to $P_{ij}(3,3)$ in equation (3), and λ is Lagrange Multiplier. In 2D planar rotation case, the rotation matrix R_{ij} can be expressed by rotation angle θ_{ij}. So the equation (5) can be transformed to

$$\Omega = \frac{1}{2}\sum_{i=1}^{n}(\theta_{ij} - \hat{\theta}_{ij})^T C_{ij}^{-1}(\theta_{ij} - \hat{\theta}_{ij}) + \lambda(\hat{\theta}_{12} + \ldots + \hat{\theta}_{n1} - 2\pi) \tag{6}$$

If we express $\theta = (\theta_{12}\ \theta_{23}\ \cdots\ \theta_{n1})^T$, $\hat{\theta} = (\hat{\theta}_{12}\ \hat{\theta}_{23}\ \cdots\ \hat{\theta}_{n1})^T$, $A = (1\ 1\ \cdots\ 1)^T$ and $C = diag(C_{12}\ C_{23}\ \cdots\ C_{n1})$, the equation (6) becomes

$$\Omega = \frac{1}{2}(\theta - \hat{\theta})^T C^{-1}(\theta - \hat{\theta}) + \lambda(A^T\hat{\theta} - 2\pi) \tag{7}$$

The Ω of equation (7)can be minimized by taking the derivative with $\hat{\theta}$ and λ and letting them equal to zero, respectively, to obtain

$$\begin{pmatrix} C^{-1} & A \\ A^T & 0 \end{pmatrix} \begin{pmatrix} \hat{\theta} \\ \lambda \end{pmatrix} = \begin{pmatrix} C^{-1}\theta \\ 2\pi \end{pmatrix} \tag{8}$$

Since the left matrix is symmetric and C is diagonal, it is possible to solve it with recursive partitioning of Gaussian elimination [13].

$$\hat{\theta} = \theta - CA(A^T CA)^{-1}(A^T\theta - 2\pi) \tag{9}$$

If the number of images in the sequence is n, the computation of θ is $O(n)$. Another benefit of the algorithm is memory saving. We do not use all the images in the loop directly for the global consistent optimization, while we only use the graph's link information.

4.2 Consistent Translation Estimation

After the consistent rotation estimation as in the previous section, we can establish the objective function to minimize the following Mahalanobis distance [10] for the consistent translation estimation.

$$W = \frac{1}{2}\sum_{i=1}^{n}(t_{ij} - \hat{t}_{ij})^T C_{ij}^{-1}(t_{ij} - \hat{t}_{ij}) +$$

$$+\frac{1}{2}\sum_{i=1}^{n}\sum_{k=1}^{M_i}(P_i^k - \hat{R}_{ij}P_j^k - \hat{t}_{ij})^T(P_i^k - \hat{R}_{ij}P_j^k - \hat{t}_{ij}) \tag{10}$$

where C_{ij} is related to the correspond entries in error model in equation (3), and P_i^k is a feature point in image I_i and $k = 1, \cdots, M_i$. This objective function must be subjected to the constraint that the estimated rotation $\hat{t}_{ij}$ will transform the vertex V_j back into V_j in the cycle [14]. This means

$$\begin{pmatrix} a_{12} & a_{23} & \cdots & 1 \end{pmatrix} \begin{pmatrix} t_{12} \\ t_{23} \\ \vdots \\ t_{n1} \end{pmatrix} = 0 \tag{11}$$

where $a_{12} = s_{n1}R_{n1}\cdots s_{23}R_{23}$ and $a_{23} = s_{n1}R_{n1}\cdots s_{34}R_{34}$. The optimal translation can be obtained by Lagrange multipliers with the following objective function

$$\Omega = \frac{1}{2}(t-\hat{t})^T C(t-\hat{t}) + \frac{1}{2}\sum_{k=1}^{M}(P^k - \hat{t})^T(P^k - \hat{t}) + \lambda(A\hat{t}) \tag{12}$$

where $t = (t_{12}\ t_{23}\ \cdots\ t_{n1})^T$, $\hat{t} = (\hat{t}_{12}\ \hat{t}_{23}\ \cdots\ \hat{t}_{n1})^T$, C is the covariance matrix for t, $P^k = (P_1^k\ \cdots\ P_n^k)^T$, and $A = (s_{n1}R_{n1}\cdots s_{23}R_{23}\ s_{n1}R_{n1}\cdots s_{34}R_{34}\ \cdots\ 1)^T$.

We use the same method of the previous section to get the solution of $\hat{t}$. The computational complexity is $O(nM)$ for consistent translation estimation. M is constant and $M \leq N$, so it is very efficient.

5 Locally Consistent Image Mosaic

In the previous section, we have got the globally consistent for image mosaic when there is a loop. If there is no loop, we still need to improve its accuracy by using the redundant information available.

Assuming that the robot navigates in an environment, it takes an image every time step. In order to have a high quality image mosaic or map, the adjacent image overlap must be at least 70% [12]. If we keep the overlap at 80%, then every image will have partially overlap with other four recent images (Figure 3 a). Assuming the given images determine the most likely pose, then append the pose to the map, and freeze it once for forever.

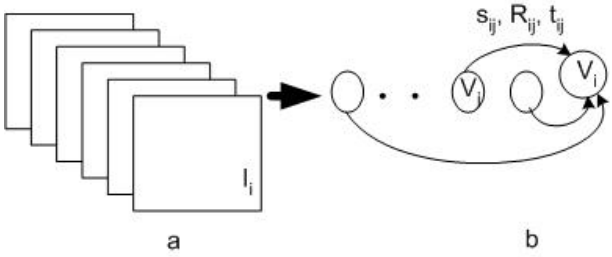

Figure 3 a. Adjacent images overlapped; b. The connection of image I_i with previous images.

For the new added image I_i, if we can find at least four feature point correspondences with any other previous images, we establish a link between them (Figure 3 b). If we can get m images having this kind of link, then we can minimize the following function for the position of image I_i.

$$\Omega = \sum_{j=1}^{m}(V_i - V_j - D_{ij})^T P_{ij}^{-1}(V_i - V_j - D_{ij}) \tag{13}$$

where $V_i = (x_i\ y_i\ \theta_i\ s_i)^T$ indicates the position of center of image I_i, $V_j = (x_j\ y_j\ \theta_j\ s_j)^T$, $(j = 1, \cdots, m)$, are the center position of the images which overlap with image I_i. P_{ij} is the covariance for the link from V_i to V_j, and D_{ij} is the measurement of relative image position from image I_i to image I_j, which is equal to $D_{ij} = (tx_{ij}\ ty_{ij}\ \theta_{ij}\ s_{ij})^T$. By taking the derivative of the equation (13) with respect to V_i and setting it to zero, we obtain

$$V_i = (\sum_{j=1}^{m} P_{ij}^{-1})^{-1}\sum_{j=1}^{m}(P_{ij}^{-1}(V_j + D_{ij})) \tag{14}$$

This is optimal solution for image I_i based on all the previous related images. The computational time is $O(m)$. Since m is usually much smaller than n, the local consistent estimation is very fast.

6 Experiments

Two steps are carried out to test our algorithm. The algorithm used in this paper is summarized in the following figure 4.

Input:	*a sequence of n images, evenly distributed feature number N*
Output:	*image mosaic of the sequence of images*
1	*for i=1 to n*
2	*get the image* I_i
3	*compute feature by using Harris method*
4	*find match features on previous images by SSD*
5	*implement RANSAC to delete outliers*
6	*find the evenly distributed feature points*
7	*if the number of matched points* $> N/2$
8	*calculate transformation between* I_i *and* I_j
9	*end if*
10	*calculate the position of image* I_i *by eq. (14)*
11	*loop check*
12	*if loop exist*
13	*global optimization by eq.(6, 7)*
14	*goto 18*
15	*else*
16	*goto 18*
17	*end if*
18	*image integration*
19	*next i*

Figure 4 Procedure for the algorithm

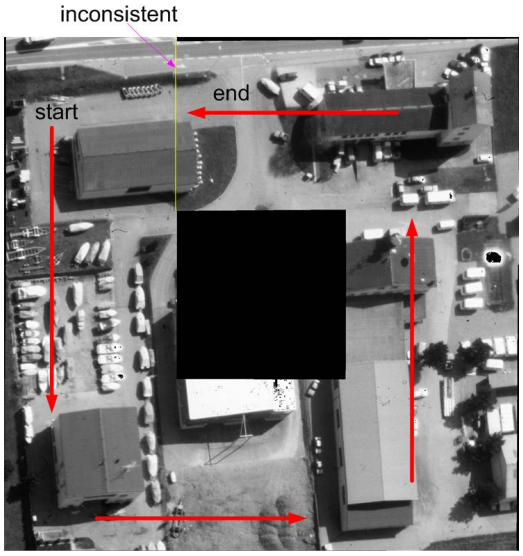

Figure 5 Image mosaicking from simple alignment between two adjacent images. The red arrow indicates the robot path, and the yellow line indicates the inconsistent merged place between the first image and the last in the loop

In order to evaluate the globally consistent algorithm, we assume that a robot navigates in a area at a constant speed counterclockwise along a rectangular trajectory, and take an image at the same time interval. Each image has 80% overlap with the previous image. Then we can clip an image sequence from a bird-view image of a factory according to the assumption. If we only use the local optimization, the image mosaic will be not consistent at the end of loop (see Fig. 5). When we use the globally consistent algorithm by adding the constraints of loop information, we get a consistent mosaic as show in Fig. 6.

Figure 6 Image mosaic after local and global optimization

In another example, we use a hand-held camera to take an image sequence at Halifax harbor which has 20 images. Since there is no cycle in this image sequence, we only used the local optimization for the mosaicking. The number of feature points $N = 400$ and search window is 15×15 pixels. The mosaic of the harbor is shown in Fig. 7.

a. Image 1, 10, 20 from image sequence

b. Image mosaicing from a sequence

Figure 7 Image mosaicking for the Halifax harbor

7 Discussion

Automatic consistent image mosaicking with local and global optimization has been addressed. The basic requirement for the consistent image mosaicking construction is the redundant information which is provided in the sequence of images by image overlapping. This redundant information makes it possible to establish constraints for the system optimization.

When the cycle is detected in the image sequence, global consistent optimization is used, in which the rotation and translation are decoupled. The worst computation time is $O(nN)$, here n is the number of images in the loop and the N is the evenly distributed feature number. while in Lu & Milios' method, the computational complexity is $O(n^3)$. And we have the same computational speed as Unnikrishan, but our method decouples rotation and translation, which will greatly reduce the computation for the $3D$ case.

When a new image is added, if there is no cycle detected, the locally consistent alignment is used with all the related abundant information. The computational complexity of our approach is $O(m)$, where m is the number of images which have enough overlapping (the matched feature points is bigger than the number of parameters in the image transformation model) with the newly added image. It is much faster than previous approaches.

The simulation results and real experiment results indicate that the algorithm is practical. The framework can be used for the $3D$ image registration and robot navigation in $3D$ environment, and this is a topic of future investigation.

References

[1] T. Duckett, S. Marsland, and J. Shapiro. Fast, on-line learning of globally consistent maps. *Autonomous Robots*, 12:287–300, 2002.

[2] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24, 1981.

[3] M. G. Gonzalez, P. Holified, and M. Varley. Improved video mosaic construction by accumulated alignment error distribution. In *British Machine Vision Conference*, pages 377–387, A7 1997.

[4] J. S. Guntmann and K. Konolige. Incremental mapping of large cyclic environments. In *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pages 318–325, Nov. 1999.

[5] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, The Edinburgh Building, Cambridge CB2 2RU, UK, 2000.

[6] C. T. Hsu and R. Beuker. Multiresolution feature-based image registration. In *Visual Communications and Image Processing 2000, Proceedings of SPIE*, pages 1490–1498, Perth, Austrilia, June 2000.

[7] C. T. Hsu, T. H. Cheng, R. A. Beuker, and J. K. Horng. Feature-based video mosaic. In *Proc. of ICIP 2000*, pages 887–890, Vancouver, Canada, Sep. 2000.

[8] A. Kelly. Some useful results for closed-form propagation of error in vehicle odometry. In *CMU-RI-TR-00-20*, Carnage Mellon University, 2000.

[9] D. G. Lowe. Object recognition from local scale-invariant features. In *Proc. of the International Confernce on Computer Vision*, Corfu, Sep. 1999.

[10] F. Lu and E. Milios. Globally consistent range alignment for envirement mapping. *Autonomous Robots*, 4:333–349, 1997.

[11] F. Lu and E. Milios. Robot pose estimation in unknown environments by matching 2d range scans. *Intelligence and Robotics Systems*, 18:249–275, 1997.

[12] H. Madjidi and S. Negahdaripour. Global alignment of sensor posistions with noisy motion measurements. In *IEEE International Conference on Robotics and Automations*, New Orleans, April 2004.

[13] P. F. McLauchlan. The variable state dimension filter applied to surface-based structure from motion. In *CVSSP Technical Report VSSP-TR-4*, 1999.

[14] G. C. Sharp, S. W. Lee, and D. K. Wehe. Toward multiview registration in frame space. In *IEEE Internatinal Conference on Robotic and Automation*, pages 3542–3547, Seoul, Korea, May 2001.

[15] H. Y. Shum and R. Szeliski. Construction and refinement of panoramic mosaics with global and local alignment. In *Six International Conference on Computer vision*, pages 953–956, 1998.

[16] R. Unnikrishan. Globally Consistent Mosaiking for Autonomous Visual Navigation. In *MSc. Thesis*, Carnegie Mellon University, Sep. 2002.