

Recovering Camera Pose from Omni-directional Images

Ada S.K. WAN¹ Angus M.K. SIU¹ Rynson W.H. LAU^{1,2}

¹ Department of Computer Science, City University of Hong Kong, Hong Kong

² Department of CEIT, City University of Hong Kong, Hong Kong

{adaskw, angus, rynson}@cs.cityu.edu.hk

Abstract

Omni-directional images are widely used in image-based walkthrough applications, in which camera pose recovery is one of the initial and important processes. Existing methods may recover camera pose of omni-directional images from lines. However, it may not work well when there is insufficient scene structure in the images. In addition, existing methods involve non-linear optimization and iterative algorithms, which may lead to the convergence problem and high computational cost.

In this paper, we propose an automatic camera pose recovery method for a network of omni-directional images. Our method only requires 2D point correspondences as input. We divide the problem into the orientation and position components, and determine them separately. The relative rotations between adjacent views are aggregated to estimate the global orientations. An algorithm is developed to adjust the rotations for avoiding global inconsistency and error accumulation. For position recovery, we have derived a linear global formulation for establishing correspondences and positions among multiple views. Globally optimized positions can be obtained simply by solving the formulation. We demonstrate the performance of our method with some experiments.

1. Introduction

Omni-directional images, such as panorama, are increasingly used for image-based walkthrough applications and 3D reconstruction [1, 14, 17]. They require camera pose estimates for arbitrary view synthesis and guided matching. Thus, camera pose recovery becomes the prerequisite for developing such image-based applications. Although the problem of camera pose recovery for planar images has been extensively studied, the methods developed cannot simply be applied to omni-directional images. It is because the linear projective relationship between the image space of planar images, such as fundamental matrix [4] and trifocal tensor [13], does not exist between the image spaces of omni-directional images. Without the linear projective relationship, most camera pose recovery methods [8, 12] that work on the projective space can no longer be used here. Thus, we need to develop new methods for omni-directional images.

Existing methods to recover camera parameters of omni-directional image can be generally classified into three types: *onsite measurement*, *interactive recovery* and *automatic recovery*. Onsite measurement [3, 17] may be the most direct method to obtain the camera pose during image capturing. However it requires expensive

equipment, such as a laser range finder. The accuracy of the acquired orientations and positions highly depends on the sensitivity of the equipment. It is usually not precise enough to bring the images into pixel-accurate registration.

To avoid expensive equipment for onsite measurement, camera pose may be estimated from images directly. [14] proposes a method for camera pose and structure recovery by manually specifying the planes, lines and points with known directions or relationship on omni-directional images. However, manual specification becomes impractical when the number of images is large. Human bias may also be introduced.

In [2], a scalable camera pose estimation method for omni-directional images with automatic edge detection and vanishing points (VPs) estimation is proposed. However, as the estimation relies on parallel lines in order to estimate VPs in every image, it may not work well when the scene lacks regular structures. Moreover, their method involves iterative algorithm and non-linear optimization. It may lead to convergence problem and expensive computational cost.

In this paper, we introduce a novel camera pose recovery approach for omni-directional images. Traditionally, camera calibration for planar images simultaneously estimates both extrinsic and intrinsic parameters of the camera. However, the strong coupling existed between the two sets of parameters cause much error in the estimated focal length. We approach the problem by first recovering the intrinsic parameters followed by the extrinsic parameters. The extrinsic parameter (6DOFs) recovery process is further decoupled into rotational registration (3DOFs) and position recovery (3DOFs). It is because omni-directional images inherit the advantage of decoupling the zero baseline problem from the wide baseline problem. Hence, the intrinsic parameters can be recovered independent of the extrinsic parameters. As such, the intrinsic parameters can be obtained more easily and accurately [15, 16]. The extrinsic parameters can also be solved more easily with point correspondences and linear algorithms.

However, the major drawback of this decoupling approach is that the error from one recovery stage may accumulate to the next. We will deal with the problem by applying the global consistency checking to minimize the error on each stage of the processes. The main contributions of this paper are as follows:

- **Correlation establishment:** Instead of correlating omni-directional images with corresponding planes and VPs, we transform image space points and establish linear relationship between the transformed domains. With this method, only several point correspondences are sufficient to establish the mapping function between two views.
- **Rotational registration:** The estimated mapping function provides only relative rotations and directions between two omni-directional images. To achieve global rotational registration, we aggregate a set of relative rotations into global orientations with the same Euclidean wireframe. Meanwhile, we have developed an orientation adjustment algorithm to ensure global consistency.
- **Position recovery:** Existing methods for position recovery requires many computer vision and statistical techniques, such as Hough transform, Markov chain Monte Carlo and expectation maximization, complicating the implementation of these methods. We have derived the global formulation and correlate the correspondences, disparity and positions among multiple views. By solving the formulated matrix, the positions of the views can be obtained easily. Our method allows camera positions to be recovered from correspondences alone. It is simple to implement. Iterative algorithms in existing methods are also avoided. It not only highly reduces the computational cost, but also prevents the convergence problems.

The rest of this paper is organized as follows. Section 2 describes our camera pose recovery method in detail. Section 3 presents some experimental results. Section 4 briefly concludes the paper.

2. Camera Pose Recovery

2.1 Preliminary Process

Without using specific equipment, groups of planar images are first taken at different optical centres (*nodes*). Images with a common node are then stitched together to form an omni-directional (panoramic) image. During the stitching process, an important intrinsic parameter, focal length, is obtained. More accurate intrinsic parameters can be derived with the method in [16]. Refer to Figure 1. After we have obtained a panorama for each node, we recover a consistent set of camera parameters for the panoramas. The nodes are triangulated with Delanuy triangulation [5] and *adjacencies* between nodes are established. To save computational cost, we only match point correspondences between each pair of adjacent images. Triangles are formed with three nodes, which constitute a *self-loop* for consistent checking.

Feature points are automatically extracted with the Harris operator [6] and matched for each adjacent image pair with Zero Normalized Cross Correlation (ZNCC) [7]. The correspondences obtained serve as the input for the camera pose recovery process.

As shown in Figure 2, our method can be divided into three modules: *image space transform*, *rotational registration*, and *position recovery*. Details of each module will be discussed next.

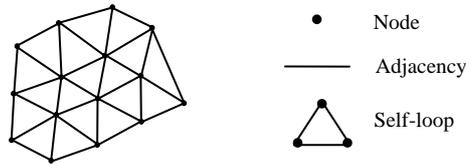


Figure 1. An image network composing of nodes, adjacencies and self-

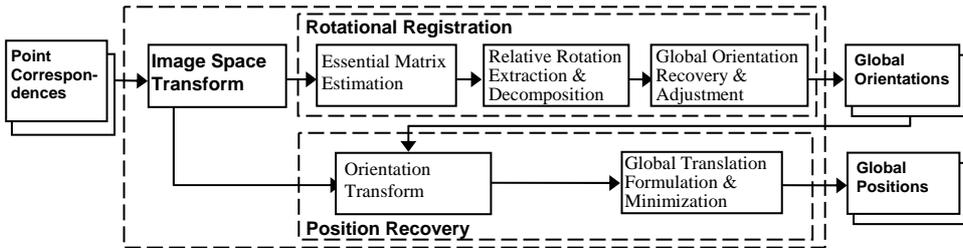


Figure 2. Our camera pose recovery method for omni-directional

2.2 Image Space Transform

As the mapping between the Euclidean space and the image space of an omni-directional image is *non-linear*, the mapping between the image spaces of an image pair is also *non-linear*. This means that we cannot directly formulate the point correspondences into a system of *linear* equations to solve for the mapping function between an image pair.

For the camera with a single mirror providing a single viewpoint, i.e., central panoramic *catadioptric* camera, [11] suggests a *non-linear* fundamental constraint on

the corresponding points in two *catadioptric* cameras. However, for the more commonly used mosaics based panorama, e.g., QTVR, they assume that the images do not provide a single viewpoint and the epipolar geometry has not been derived. In practice, it is a good approximation to assume mosaics based panorama as a cylindrical projection with a single viewpoint [9, 14], if all objects in the scene are relatively far from the tripod's rotational centre. Based on this assumption, the relationship between image space points can be established as follows. We first transform the image space points to unit vectors on a Gaussian sphere, \mathbf{S}^2 . Hence, a transformation g^j for view j is applied to image point $\mathbf{p}=[u,v]^T$ to obtain the projective ray $\mathbf{d}=[s,t,r]^T$, i.e., $g^j(\mathbf{p})=\mathbf{d}$. With this transformation, we may establish the linear mapping \mathbf{E}^{kj} between views j and k as follows:

$$g^k(\mathbf{p})^T \mathbf{E}^{kj} g^j(\mathbf{p}) = 0 \quad (1)$$

where \mathbf{E}^{kj} is a 3x3 essential matrix, with five DOFs: three for relative rotation and two for relative direction. A different transformation g would be adopted for a different kind of omni-directional images. For example, g for panoramic images would be defined as:

$$g\left(\begin{bmatrix} u \\ v \end{bmatrix}\right) = \begin{bmatrix} f \sin(u/f) \\ kv - h/2 \\ -f \cos(u/f) \end{bmatrix} = \begin{bmatrix} s \\ t \\ r \end{bmatrix} \quad (2)$$

where f is the focal length, h is the image height and k is the skew factor.

2.3 Rotational Registration

2.3.1 Essential Matrix Estimation

After transforming the image space points to rays, we can estimate \mathbf{E} for every adjacent image pair. Denoted by \mathbf{e} , the 9-vector makes up the entries of \mathbf{E} in row-major order. Let $\mathbf{d}_i^j = [s_i^j, t_i^j, r_i^j]^T$ be the projective ray at view j of the i^{th} correspondence. We can obtain a system of linear equations from a set of correspondences in the form of:

$$\mathbf{A}\mathbf{e} = \begin{bmatrix} s_1^k s_1^j & s_1^k t_1^j & s_1^k r_1^j & t_1^k s_1^j & t_1^k t_1^j & t_1^k r_1^j & r_1^k s_1^j & r_1^k t_1^j & r_1^k r_1^j \\ \vdots & \vdots \\ s_n^k s_n^j & s_n^k t_n^j & s_n^k r_n^j & t_n^k s_n^j & t_n^k t_n^j & t_n^k r_n^j & r_n^k s_n^j & r_n^k t_n^j & r_n^k r_n^j \end{bmatrix} \mathbf{e} = \mathbf{0} \quad (3)$$

By enforcing the constraint that $\text{Det}(\mathbf{E}) = 0$, we may use seven points to solve for \mathbf{e} . We also impose the condition that the two singular values of the essential matrix should be equal. The resulting \mathbf{E} is the closest solution in Frobenius norm.

We use the Random Sample Consensus (RANSAC) algorithm to estimate \mathbf{E} robustly, and determine the number of required samples adaptively by:

$$N = \log(1 - \rho) / \log(1 - \varepsilon^s) \quad (4)$$

where ρ is the probability of getting at least one sample without outlier, ε is the inlier percentage, and s is the minimum number of correspondences in each sample. The \mathbf{E} with the most inliers is chosen.

2.3.2 Relative Rotation Extraction and Decomposition

The essential matrix \mathbf{E} provides relative orientation and translation direction between two adjacent images. In order to obtain the orientations in the world coordinate system, we extract the rotational matrix \mathbf{R}^{kj} from every \mathbf{E}^{kj} by Singular Value Decomposition (SVD). There are two possible solutions for \mathbf{R}^{kj} and two for translations. The correct solution can be determined from several correspondences as suggested in [18]. \mathbf{R}^{kj} is

then further decomposed into 3 given rotations about the three coordinate axes, i.e., $\mathbf{R}^{kj} = \mathbf{R}_x^{kj} \mathbf{R}_y^{kj} \mathbf{R}_z^{kj}$, by RQ factorization.

2.3.3 Global Orientation Recovery and Adjustment

As the set of \mathbf{E}^{kj} are estimated independently, the initial set of \mathbf{R}^{kj} may not be globally consistent. The inconsistency in \mathbf{R}^{kj} may also be propagated and accumulated as the path gets long. To deal with this problem, we propose an algorithm to adjust the rotational matrices. With the adjustment, a set of globally consistent \mathbf{R} can be obtained and the inconsistency propagation can be minimized.

Let θ_{kj} be the relative rotational angle from view j to view k . The residual ε of a self-loop L of three views j, k, l (i.e., $\mathbf{R}^{kj} \rightarrow \mathbf{R}^{lk} \rightarrow \mathbf{R}^{lj}$) is defined as:

$$\varepsilon = \theta_{kj} + \theta_{lk} + \theta_{jl} \quad (5)$$

In a consistent self-loop, ε is equal to zero as shown in Figure 3(a). However, due to the independent estimation of \mathbf{R} , ε is often not equal to zero as shown in Figure 3(b). To ensure consistency, we have to adjust θ by $\Delta\theta$, such that

$$0 = \theta_{kj}' + \theta_{lk}' + \theta_{jl}' \quad (6)$$

$$\theta_{kj}' = \theta_{kj} + \Delta\theta_{kj}. \quad (7)$$



Figure 3. (a) Global consistent orientations, and (b) global inconsistent orientations.

Initially, all the adjacencies are marked as “*non-adjusted*” and adjusted one by one. We compute ε for each self-loop L and choose the one L_{curr} with the smallest ε . For every adjacency in L_{curr} , $\Delta\theta_{kj}$ is determined as:

$$\Delta\theta_{kj} = \frac{1}{n} \sum_{i=1}^n \frac{\varepsilon_i \cdot \kappa_i}{q_i} \quad (8)$$

where $n \in \{1, 2\}$ is the number of self-loops containing the adjacency, q_i is the number of non-adjusted adjacencies in the adjacent self-loop L_i , and $\kappa_i \in \{1, -1\}$ indicates the direction of edge in L_i . Then, θ_{kj} is adjusted by Equation (7) and marked as “*adjusted*”. ε 's of the adjacent self-loops L_{neig} of L_{curr} are computed. The one with the smallest ε is selected and the adjustment process is repeated until all the adjacencies are updated.

The following pseudo-code shows the orientation adjustment.

```

 $\varepsilon_{min} = \infty$ 
for i = 1 to M do
    compute  $\varepsilon_i$  for  $L_i$  by Equation (5)
    if  $\varepsilon_i < \varepsilon_{min}$  then  $L_{curr} = L_i$ 
endfor
j = M
repeat
    compute  $\Delta\theta$  for non-adjusted edges in  $L_{curr}$  by Equation (8)
    update  $\varepsilon, q$  for all  $L_{neig}$ 
    j = j - 1

```

$L_{\text{curr}} = L_{\text{neig}}$ with minimum ε
until $j = 0$

2.4 Position Recovery

The translation of a view includes direction as well as magnitude. Since only relative translation direction between two views can be inferred from \mathbf{E}^{kj} , we cannot simply obtain the global position of a view from a set of essential matrices. Here, we investigate the linear relationship among the correspondences, generalized disparity [10] and global translation among multiple views. Upon formulating these entities, we may obtain a set of globally optimized positions by solving the system of linear equations.

2.4.1 Orientation Transform

Given a 3D point, $\mathbf{X} \in \mathfrak{R}^3$ in the Euclidean space, we can define a mapping function from \mathbf{X} to a projective ray \mathbf{d} as: $\mathbf{M}: \mathbf{X} \rightarrow \mathbf{d}$. On the other hand, for a given generalized disparity, $\lambda = \delta(\mathbf{d})$, we can define another mapping function from \mathbf{d} to \mathbf{X} as: $\mathbf{N}: (\mathbf{d}, \lambda) \rightarrow \mathbf{X}$. With a 3x3 rotational matrix \mathbf{R} and the camera centre \mathbf{C} of a view, the two mapping functions can be defined as follows:

$$\begin{aligned} \mathbf{M}: \quad \mathbf{d} &\doteq \mathbf{R}\mathbf{X} - \mathbf{R}\mathbf{C} & \text{or} & \quad \mathbf{d}^\top \cdot (\mathbf{R}\mathbf{X} - \mathbf{R}\mathbf{C}) = 0 \\ \mathbf{N}: \quad \mathbf{X} &= \mathbf{C} + \lambda \mathbf{R}^\top \mathbf{d} \end{aligned}$$

We can eliminate the rotational matrix \mathbf{R} by an orientation transform $\mathbf{H}: \mathbf{d}' = \mathbf{R}^\top \mathbf{d}$. After the transformation, the two functions can be simplified as:

$$\begin{aligned} \mathbf{M}: \quad \mathbf{d}'^\top \cdot (\mathbf{X} - \mathbf{C}) &= 0 \\ \mathbf{N}: \quad \mathbf{X} &= \mathbf{C} + \lambda \mathbf{d}' \end{aligned}$$

2.4.2 Global Translation Formulation

After the orientation transform, we correlate the two mapping functions for different views and derive the formulation among multiple views.

Referring to Figure 4, \mathbf{X}_i can be inferred from a covariant view j by the function \mathbf{N} as:

$$\mathbf{X}_i = \mathbf{C}^j + \lambda_i^j \mathbf{d}_i^j \quad (9)$$

where \mathbf{d}_i^j denotes the projective ray after the orientation transform at view j for \mathbf{X}_i . λ_i^j is the corresponding generalized disparity. Meanwhile, \mathbf{X}_i can also be projected to a contravariant view k by the function \mathbf{M} as:

$$(\mathbf{d}_i^k)^\top \cdot (\mathbf{X}_i - \mathbf{C}^k) = 0 \quad (10)$$

By substituting (9) into (10), \mathbf{X}_i can be eliminated:

$$(\mathbf{d}_i^k)^\top \cdot (\mathbf{C}^j - \mathbf{C}^k + \lambda_i^j \mathbf{d}_i^j) = 0 \quad (11)$$

Let $\mathbf{d}_i^j = [s_i^j, t_i^j, r_i^j]^\top$ and $\mathbf{C}^j = [C1^j, C2^j, C3^j]$. By eliminating a redundant equation, the correspondences can be formulated with (11) as:

$$\begin{bmatrix} 0 & -r_i^k & t_i^k & 0 & r_i^k & -s_i^k & t_i^k r_i^j - r_i^k t_i^j \\ t_i^k & 0 & -s_i^k & -r_i^k & 0 & s_i^k & s_i^k r_i^j - s_i^k r_i^j \end{bmatrix} \begin{bmatrix} C1^j \\ C2^j \\ C3^j \\ C1^k \\ C2^k \\ C3^k \\ \lambda_i^j \end{bmatrix} = \mathbf{0} \quad (12)$$

By extending the formulation for multiple covariant and contravariant views, we may obtain the matrix of multiple views for global optimization.

In order to tolerate mismatches and achieve robust estimation, we again employ the RANSAC method to select a sub-set of samples for determining the global positions. The solution with the most inlier σ is selected. The number of samples N required is determined adaptively by Equation (4). Let \mathbf{C} be the right null space vector in Equation (12), the pseudo-code for position recovery to get the best solution \mathbf{C}_{best} is as follows:

```

for j = 1 to J
  for i = 1 to l
    orientation transform of  $\mathbf{d}_i^j$ 
  n =  $\sigma_{\text{max}} = 0$ 
  repeat
    select sample of corresponding  $\mathbf{d}_i^j$  from every view
    compute  $\mathbf{C}$  for the equations as in Equation (12)
    compute  $\sigma$ 
    if ( $\sigma_{\text{max}} < \sigma$ )
       $\mathbf{C}_{\text{best}} = \mathbf{C}$ 
      n = n+1
  until n = N

```

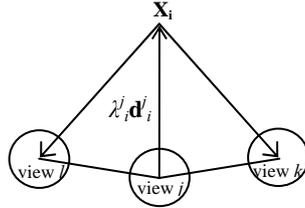


Figure 4. A 3D point X_i can be inferred from view j and then projected to views k, l .

3. Experimental Results

We have implemented our method in Java and experimented it on a Pentium 4 2.4GHz PC with two real scenes. Table 1 shows the configuration of the two scenes. We prepared 15 panoramic images, each at a resolution of 4500x450, for each of the two scenes. The 15 panoramic images form 18 self-loops and 32 adjacencies. Table 2 shows the reprojection error of the two scenes. While the reprojection error for existing methods is usually a few pixels, our method is about one pixel on average.

Node	Adjacency	Self-loop
15	32	18

Table 1. Configuration of two scenes.

Reprojection Error (pixel)	Average	Maximum	Standard Deviation
Exp 1	0.82	1.80	0.47
Exp 2	1.03	2.00	0.56

Table 2. Reprojection errors of the two experiments with 15 panoramic images.

Figures 5(a) and 5(b) show two of the 15 panoramic images for each of the two scenes. The epipolar curves, which are determined from the recovered camera pose, are superimposed on the images. Figures 5(c) and 5(e) show the epipolar curves computed by the initial \mathbf{E} (section 2.3.1) at two feature points, which do not lie on the epipolar

curves. After applying the rotational registration and the position recovery processes, the epipolar alignment is significantly improved as shown in Figures 5(d) and 5(f).

Figure 6 compares the accuracy of the camera pose obtained, with and without rotational registration, in term of inlier percentage against the number of images. We

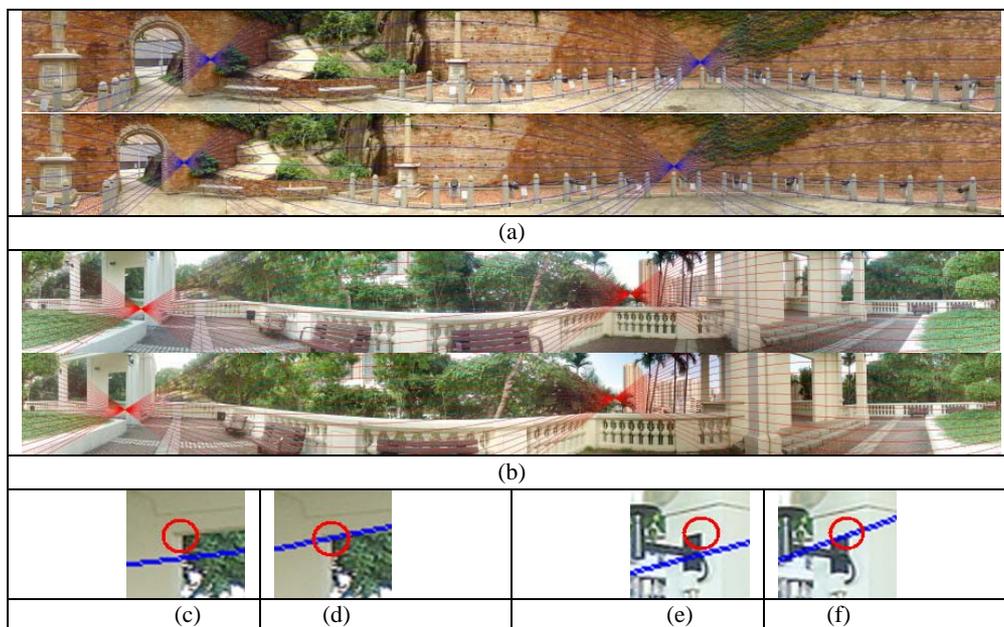


Figure 5. (a) and (b) show the Eipoplar curves of the two scenes. (c) to (f) compare the epipolar alignment before ((c), (e)) and after ((d), (f)) the orientation adjustment and position recovery processes.

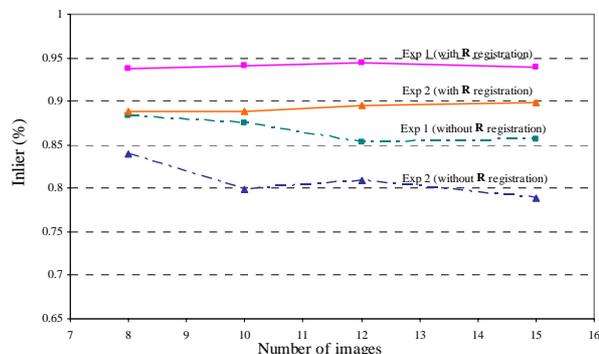


Figure 6. Inlier percentage with rotational \mathbf{R} registration is compared with that without rotational \mathbf{R} registration as a function of image number.

can see that the accuracy is improved with global orientation adjustment.

By applying SVD to \mathbf{E} , rotation and translation direction can be extracted. However, the translation directions from different nodes are inconsistent. Refer to Figure 7. The translation directions (dotted lines) do not intersect on a single node due to the global inconsistency. By applying our position recovery method, globally consistent positions and translation directions (solid line) can be obtained.

The inlier percentage against the number of images with global consistent translations computed by our method and inconsistent translations is plotted in Figure 8. From the graph, we see that the position recovery process significantly improves the accuracy of the estimated camera pose. Moreover, the trend of the curves in both Figures 6 and 8 show that the inlier percentage without either \mathbf{R} registration or position recovery decreases as the number of images increases. This is because the inconsistency is accumulated as the length of the path increases.

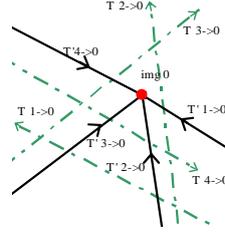


Figure 7. Global position and relative directions.

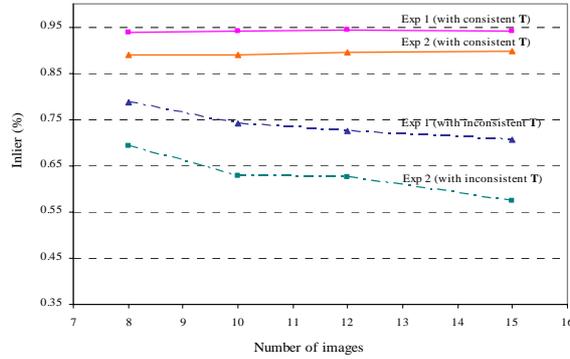


Figure 8. Inlier percentage is compared between consistent and inconsistent translation \mathbf{T} as a function of image number.

Table 3 lists the processing time of each process. The whole recovery process takes around ten minutes, which is similar in performance to existing methods. The number of samples required in the RANSAC method is inversely proportional to the number of inliers (Equation 4). As the number of inliers in experiment 1 is higher than that in experiment 2, RANSAC computation requires less samples, i.e., lower processing time.

Process	E Estimation	R Extraction and Adjustment	Position Adjustment	Total
Exp 1	1.30 mins	0.0028 mins	6.13 mins	7.43 mins
Exp 2	2.18 mins	0.0027 mins	9.78 mins	11.96 mins

Table 3. Computational time in each process with 15 panoramic images.

4. Conclusion

In this paper, we have proposed a camera pose recovery method for omni-directional images. Feature points are automatically extracted and matched across every image pair. The correspondences serve as the only input to our method. By transforming image space point, linear mapping function is established to correlate omni-directional

images. Rotational adjustment aggregates relative orientations in global consistent orientations. It effectively minimizes error accumulation. By deriving and solving the global formulation, a set of globally optimized positions from transformed points can be obtained. Because no planes and lines are required, it simplifies the implementation and does not rely on regular scene structures.

Acknowledgements

The work described in this paper was partially supported by a CERG grant from the Research Grants Council of Hong Kong (RGC Reference Number: CityU 1308/03E) and a DAG grant from City University of Hong Kong (Project Number: 9100264).

References

- [1] D. Aliaga and I. Carlbom, "Plenoptic Stitching: A Scalable Method for Reconstructing 3D Interactive Walkthroughs," *Proc. ACM SIGGRAPH*, pp. 443-450, 2001.
- [2] M. Antone and S. Teller, "Scalable Extrinsic Calibration of Omni-Directional Image Networks," *IJCV*, **49**(2/3):143-174, Sept./Oct. 2002.
- [3] R. Bunschoten and B. Kröse, "Robust Scene Reconstruction from an Omnidirectional Vision System," *IEEE Trans. on Robotics and Automation*, **19**(2):351-357, 2003.
- [4] O. Faugeras, "What can be Seen in Three Dimensions with an Uncalibrated Stereo Rig," *Proc. ECCV'92*, 1992.
- [5] S. Fortune, "Voronoi Diagrams and Delaunay Triangulations," *Computing in Euclidean Geometry*, D. Du and F. Hwang (eds.), World Scientific, pp. 193-223, 1992.
- [6] C. Harris and M. Stephens, "A Combines Corner and Edge Detector," *Alvey Vision Conference*, pp 147-151, 1988.
- [7] Z. Lan and R. Mphr, "Robust Location Based Partial Correlation," *Research Report 3186*, INRIA Sophia-Antipolis, 1997.
- [8] Q. Luong and O. Faugeras, "Self-calibration of a Moving Camera from Point Correspondences and Fundamental Matrices," *IJCV*, **1**(1):5-40, 1997
- [9] L. McMillan, G. Bishop, "Plenoptic Modelling: An Image-based Rendering System," *Proc. ACM SIGGRAPH*, pp. 39-46, 1995.
- [10] L. McMillan, "An Image-Based Approach to Three-Dimensional Computer Graphics," *Technical Report 97-013*, UNC at Chapel Hill, 1997.
- [11] T. Pajdla and T. Svoboda, "Epipolar Geometry for Central Catadioptric Cameras," *IJCV*, **49**(1):23-37, 2002
- [12] M. Pollefeys, R. Kocj, and L. van Gool, "Self-calibration and Metric Reconstruction in Spite of Varying and Unknown Internal Camera Parameters," *IJCV*, **32**(1):7-25, Aug. 1999.
- [13] A. Shashua, "Algebraic Functions for Recognition," *IEEE Trans. on PAMI*, **17**(8):779-789, Aug. 1995.
- [14] H. Shum, M. Han, and R. Szeliski, "Interactive Construction of 3D Models from Panoramic Mosaics," *Proc. CVPR*, pp. 427-433, 1998.
- [15] H. Shum and R. Szeliski, "Construction of Panoramic Image Mosaics with Global and Local Alignment," *IJCV*, **36**(2):101-130, Feb. 2000.
- [16] G. Stein, "Accurate Internal Camera Calibration Using Rotation, with Analysis of Sources of Error," *ICCV*, pp. 230-236, 1995.
- [17] S. Teller, "Automated Urban Model Acquisition: Project Rationale and Status," *IUW*, pp. 455-462, 1998.
- [18] J. Weng, T. Huang, and N. Ahuja, "Motion and Structure from Two Perspective Views: Algorithms, Error Analysis, and Error Estimation," *IEEE Trans. On PAMI*, **11**(5):451-476, May 1989.