http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 7 (1): 220-223, 2008
ISSN 1812-5638

A Linear Resection-Intersection Bundle Adjustment Method

[1]Shigang Liu, [2]Jiancheng Sun and [2]Jianwu Dang
[1]School of Computer Science, Shaanxi Normal University, Xi'an 710062, China
[2]School of Eletronics, Jiangxi University of Finance and Economics, Nanchang, 330013, China

Abstract: Bundle adjustment is one of the main tools used for multiple view reconstruction. It seeks to refine the estimation of the 3D scene and the visual parameters. The main drawback of this technique, due to nonlinear optimization, is its large amount of computation. To solve the problem, we propose a linear resection-intersection bundle adjustment method (LRI) for photogrammetric bundle adjustment. The method refines linearly the projection matrices and the space points alternately and the jointly optimal estimations are finally obtained. Important features are that (i) the original cost function is preserved and (ii) the method is linear. The method has been tested with simulated data and the results demonstrate its accuracy and efficiency.

Key words: Bundle adjustment, resection-intersection, alternated optimization

INTRODUCTION

Bundle adjustment is an effective method of refining a visual reconstruction to produce jointly optimal 3D structure and visual parameter (camera pose and/or calibration) estimations (Lourakis and Argyros, 2005; Yu *et al*., 2006). Optimal means that the parameter estimations are found by minimizing some cost function that quantifies the model fitting error and jointly that the solution is simultaneously optimal with respect to both structure and camera variations. Adjustment computations are used in camera calibration, auto-calibrated reconstruction and many other fields of computer vision (Schweighofer and Pinz, 2006; Kahl and Henrion, 2007).

Bundle adjustment is an optimization problem that is performed over a set of parameters that represents structure and motion using usually non-linear Newton-type optimization techniques such as Levenberg-Marquardt (LM) (Triggs *et al*., 2000). This technique has the following drawbacks: Firstly, it requires the computation of at least the first order Jacobian matrix of the residuals with respect to structure and motion parameters that might be non-trivial. Secondly, the computational cost may be non-negligible since the Hessian matrix has to be inverted, even if specific techniques have been proposed to speed up the process (Del *et al*., 2007). To overcome the drawbacks of the traditional technique, Bartoli adapts the resection-intersection method which is an instance of a class of technique for structure from motion in photogrammetry (Zhang *et al*., 2006; Mahamud *et al*., 2001) and performs bundle adjustment using quasi-linear optimizations method (QLM) (Barloli, 2002). The principle of his paper is to rewrite the Euclidean distance used in bundle adjustment as a weighted algebraic distance. The relationship between the Euclidean and the algebraic distance is the basis for quasi-linear optimization. The method is to initialize weight factors to unity and compute a biased estimation that corrects the bias of the algebraic distance with respect to the Euclidean one using the weighted algebraic distance. When such an estimate has been obtained, weight factors can be computed using the relationship between the Euclidean and the algebraic distance. The process is iterated until convergence. The drawbacks of QLM are that (i) the non-linearity is hidden in the weight factors that can be thought of as a bias of the algebraic distance with respect to the Euclidean one and (ii) the weight factors are updated after refining structure and motion, so the speed of convergence slows down.

In the study, we present a linear resection-intersection bundle adjustment method that includes two steps: firstly, the projective matrices are refined with points in the projective space being constant. Then, inversely, the projective points are adjusted with the projective matrices unchanged. The process is iterated until convergence. Typically 3 or 4 iterations are enough. Our contributions differ from previous ones in that (i) we use the original cost function of bundle adjustment, which preserves optimality and (ii) all the parameters adjustment is linear. Experimental results reveal that it performs as well as foregoing methods in terms of convergence accuracy while it greatly reduces the computational cost.

Corresponding Author: Shigang Liu, School of Computer Science, Shaanxi Normal University, Xi'an 710062, China

PROJECTION MODEL

A commonly used model for perspective camera is a pinhole model. The homogeneous coordinate of a space point in the world frame is expressed as the vector:

$$X_j = [x_j\ y_j\ z_j\ w_j]^T \tag{1}$$

Where, subscript j is the j-th space point.

The corresponding image point is:

$$m_{ij} = [u_{ij}\ v_{ij}\ 1]^T \tag{2}$$

Where, the subscript i represents the i-th image and superscript j the j-th image point, respectively.

The transformation between the image points and the world point is the well-known perspective projection relation:

$$\lambda_{ij}\, m_{ij} = P_i\, X_j \tag{3}$$

Where, P_i is a 3×4 matrix of rank 3 and λ_{ij} is the scale factor.

To estimate the unknown space points X_j and camera projection matrices P_i from the image points m_{ij}, we minimize some measures of their total prediction error. Bundle adjustment is the model refinement part of this process with initial parameter estimations (e.g., from some approximate reconstruction method). Hence, it is essentially a matter of optimizing a complicated cost function (the total prediction error) over a large parameter space (the scene and camera parameters).

LINEAR RESECTION-INTERSECTION

Linear adjustment of the projection matrices: For the convenience of description, the camera projection matrix P_i is denoted as follows:

$$P_i = \begin{pmatrix} p_i^{1T} \\ p_i^{2T} \\ p_i^{3T} \end{pmatrix}$$

Then Eq. 3 becomes

$$\lambda_{ij}\begin{pmatrix} u_{ij} \\ v_{ij} \\ 1 \end{pmatrix} = \begin{pmatrix} p_i^{1T} \\ p_i^{2T} \\ p_i^{3T} \end{pmatrix} X_j = \begin{pmatrix} p_i^{1T}X_j \\ p_i^{2T}X_j \\ p_i^{3T}X_j \end{pmatrix}$$

Thus,

$$\begin{cases} \dfrac{p_i^{1T}X_j}{p_i^{3T}X_j} = u_{ij} \\[2ex] \dfrac{p_i^{2T}X_j}{p_i^{3T}X_j} = v_{ij} \end{cases} \tag{5}$$

When we optimize the projection matrix, Eq. 5 is rearranged as follows:

$$\begin{pmatrix} \dfrac{X_j^{\ T}}{p_i^{3T}X_j} & 0 & -\dfrac{X_j^{\ T}u_{ij}}{p_i^{3T}X_j} \\[2ex] 0 & \dfrac{X_j^{\ T}}{p_i^{3T}X_j} & -\dfrac{X_j^{\ T}v_{ij}}{p_i^{3T}X_j} \end{pmatrix} \begin{pmatrix} p_i^1 \\ p_i^2 \\ p_i^3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{6}$$

When all space points are considered for *i*-th image, Eq. 6 becomes:

$$\begin{pmatrix} \dfrac{X_1^{\ T}}{p_i^{3T}X_1} & 0 & -\dfrac{X_1^{\ T}u_{11}}{p_i^{3T}X_1} \\[2ex] 0 & \dfrac{X_1^{\ T}}{p_i^{3T}X_1} & -\dfrac{X_1^{\ T}v_{11}}{p_i^{3T}X_1} \\[2ex] \dfrac{X_2^{\ T}}{p_i^{3T}M_2} & 0 & -\dfrac{X_2^{\ T}u_{12}}{p_i^{3T}X_2} \\[2ex] 0 & \dfrac{X_2^{\ T}}{p_i^{3T}X_2} & -\dfrac{X_2^{\ T}v_{12}}{p_i^{3T}X_2} \\ \vdots & \vdots & \vdots \\ \dfrac{X_m^{\ T}}{p_i^{3T}X_m} & 0 & -\dfrac{X_m^{\ T}u_{1m}}{p_i^{3T}X_m} \\[2ex] 0 & \dfrac{X_m^{\ T}}{p_i^{3T}X_m} & -\dfrac{X_m^{\ T}v_{1m}}{p_i^{3T}X_m} \end{pmatrix}_{2m\times 12} \begin{pmatrix} p_i^1 \\ p_i^2 \\ p_i^3 \end{pmatrix}_{12\times 1} = 0_{2m\times 1}$$

Let A_i be the 2m×12 matrix, q_i the 12×1 vector. Then, for all the projection matrices and all the space points, Eq. 6 becomes:

$$A_{2mn\times 12n}\, Q_{12n\times 1} = 0_{2mn\times 1} \tag{7}$$

Where:

$$A_{2mn\times 12n} = \begin{pmatrix} A_1 & 0 & \cdots & 0 \\ 0 & A_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_n \end{pmatrix}, Q_{12n\times 1} = \begin{pmatrix} q_1 \\ q_2 \\ \vdots \\ q_n \end{pmatrix}$$

We adjust the parameters in so small range that the values of projection matrices P_i (1 = 1,2,3...n) and the space points X_i (i = 1,2,3...n) in matrix A can be substituted by their values in the last iteration. So Q, namely the entire projection matrix P_i, can be solved, linearly and iteratively.

Linear adjustment of the space points: In the similar way, the space points can also be adjusted linearly and iteratively. Equation 5 is rearranged and becomes as follows:

$$\begin{pmatrix} \dfrac{p_i^{1T}}{p_i^{3T}X_j} \\ \dfrac{p_i^{2T}}{p_i^{3T}X_j} \end{pmatrix} X_j = \begin{pmatrix} u_{ij} \\ v_{ij} \end{pmatrix} \tag{8}$$

When all space points are considered for i-th image, Eq. 8 becomes:

$$\begin{pmatrix} \dfrac{p_i^{1T}}{p_i^{3T}X_1} & 0 & \cdots & 0 \\ \dfrac{p_i^{1T}}{p_i^{3T}X_1} & 0 & \cdots & 0 \\ 0 & \dfrac{p_i^{1T}}{p_i^{3T}X_2} & \cdots & 0 \\ 0 & \dfrac{p_i^{1T}}{p_i^{3T}X_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \dfrac{p_i^{1T}}{p_i^{3T}X_m} \\ 0 & 0 & \cdots & \dfrac{p_i^{1T}}{p_i^{3T}X_m} \end{pmatrix}_{2\times 4m} \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_m \end{pmatrix}_{4m\times 1} = \begin{pmatrix} u_{i1} \\ v_{i1} \\ u_{i2} \\ v_{i2} \\ \vdots \\ u_{im} \\ v_{im} \end{pmatrix}_{2m\times 1} \tag{9}$$

Let B_i be the 2×4 m matrix on the left side, I_i the 2 m $\times$ 1 vector on the right side. Then, for all the projection matrices and all the space points, Eq. 9 becomes:

$$B_{2mn\times 4m} M_{4m\times 1} = I_{2mn\times 1} \tag{10}$$

Where:

$$B_{2mn\times 4m} = \begin{pmatrix} B_1 \\ B_2 \\ \vdots \\ B_n \end{pmatrix}_{2mn\times 4m}, M_{4m\times 1} = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_m \end{pmatrix}_{4m\times 1}, I_{2mn\times 1} = \begin{pmatrix} I_1 \\ I_2 \\ \vdots \\ I_n \end{pmatrix}_{2mn\times 1}$$

In the similar way that the projection matrices are solved, the values of space points X_i (i = 1,2,3...m) in the matrix $B_{2mn\times 4m}$ can be substituted by their values in last iteration. So the value of vector $M_{4mn\times 1}$, namely the space points, can be linearly solved.

By the foregoing analysis, the projection matrices and the space points can be precisely solved based on the linear resection-intersection bundle adjustment in turn.

EXPERIMENT

In order to verify the performance of the proposed LRI, we conducted the following experiments by using synthetic image data in Matlab on PC with Intel pentium4 CPU 1.7GHz. We first generate 100 points at random in a unit sphere. Under the condition that the camera intrinsic parameters are assumed to be $f_u = 800$, $f_v = 800$, $s = 0$, $u_0 = 320$, $v_0 = 240$, the 3D points are projected to the images with size 640×480 by different external parameters. We corrupt the position of the imaged points by zero-mean Gaussian noise with different standard deviations. After the initial values of the projection matrices and the space points can be obtained by the factorization method (Furukawa *et al*., 2006) from all the image points, LRI, QLM and LM are used to optimize the projection matrices and the space points, respectively. The experiments are repeated 100 times in the noise circumstances with different standard deviations. The experiment results of the mean re-projection errors (Fig. 1).

LRI is as accurate as QLM and LM in terms of convergence accuracy and the re-projection errors increase linearly with the noise level. At the same time, the CPU time of LRI, QLM and LM is obtained in the foregoing simulative experiment, respectively. The CPU time of the LRI method is about one third that of the LM method and three fourths that of the QLM method (Fig. 2). In the LM method, the computation of Hessian matrix costs much CPU time. The LRI method runs faster than the QLM method and the LM method, since the LRI method is linear.

Figure 3 shows the convergence performance of the LRI method. The different curves are obtained from different levels of noise, from below:0.5, 1.0, 1.5, 2.0 pixels standard deviation. The LRI method can converge after several iterations, typically 3 or 4 iterations (Fig. 3).

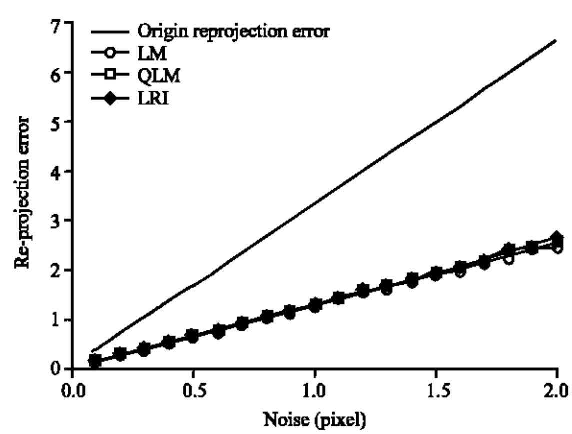

Fig. 1: Re-projection error versus noise levels

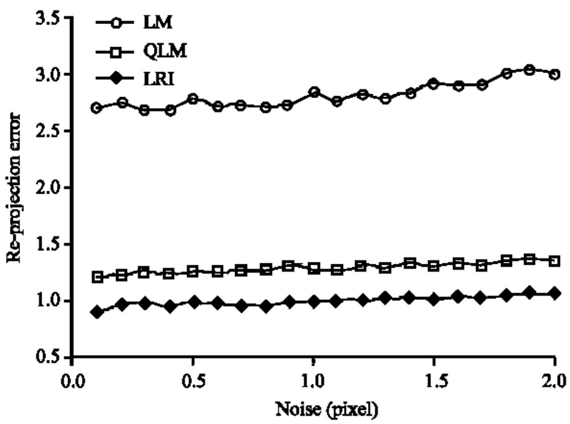

Fig. 2: Re-projection error versus noise levels

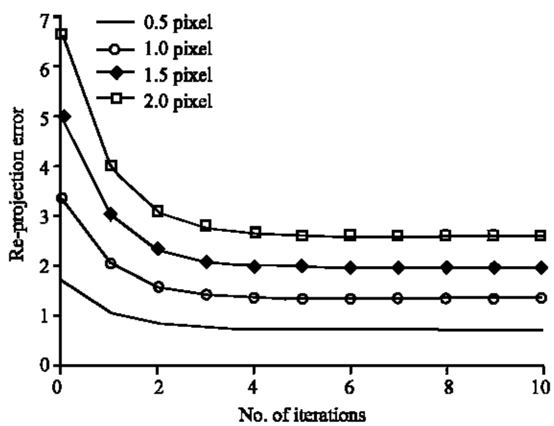

Fig. 3: Re-projection error versus No. of iterations

CONCLUSIONS

We present a linear resection-intersection bundle adjustment method to compute the cost function for multiple view reconstruction. We make use of the property that the parameters adjusted are in so small range that the values of some parameters can be substituted by their values in last iteration to refine linearly the projection matrices and the space points alternately. Experiments with both simulate data show that the proposed the LRI method is as accurate as LM and QLM in terms of convergence accuracy while it is much faster in terms of computational cost.

ACKNOWLEDGMENT

This study is supported by Natural Science Foundation of China (No.60602034).

REFERENCES

Barloli, A., 2002. A Unified Framework for Quasi-Linear Bundle Adjustment. ICPR, Quebec City, Canada, pp: 560-563.

Del, B., F. Smeraldi and L. Agapito, 2007. Non-rigid structure from motion using ranklet-based tracking and non-linear optimization. Image and Vision Comput., 25: 297-310.

Furukawa, Y., A. Sethi, J. Ponce and D. Kriegman, 2006. Robust structure and motion from outlines of smooth curved surfaces. IEEE Transac. Pattern Analysis Mach. Intell., 28: 302-315.

Kahl, F. and D. Henrion, 2007. Globally optimal estimates for geometric reconstruction problems. Int. J. Comput. Vision, 74: 3-15.

Lourakis, M. and A. Argyros, 2005. Is levenberg-marquardt the most efficient optimization algorithm for implementing bundle adjustment? ICCV, pp: 1526-1531.

Mahamud, S., M. Herbert, Y. Omori and J. Ponce, 2001. Provably convergent iterative methods for projective structure and motion. CVPR. Kauai Hawaii, pp: 1018-1025.

Schweighofer, G. and A. Pinz, 2006. Robust pose estimation from a planar target. IEEE Trans. Pattern Analysis Mach. Intell., 28: 2024-2030.

Triggs, B., P. McLauchlan, R. Hartley and A. Fitzgibbon, 2000. Bundle adjustment- A Modern Synthesis. In: Vision Algorithms: Theory and Practice, LNCS, Springer-Verlag, pp: 298-375.

Yu, Y., K. Wong and M. Chang, 2006. Merging artificial objects with marker-less video sequences based on the interacting multiple model method. IEEE Trans. Multimedia, 8: 521-528.

Zhang, J., M. Boutin and D. Aliaga, 2006. Robust bundle adjustment for structure from motion. ICIP, Atlanta, GA, USA., pp: 2185-2188.