



Projective Reconstruction from Multiple Views with Minimization of 2D Reprojection Error

Y.S. HUNG AND W.K. TANG

Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam Road, Hong Kong
 yshung@eee.hku.hk
 wktang@eee.hku.hk

Received June 11, 2004; Revised May 31, 2005; Accepted June 28, 2005

Abstract. The problem of projective reconstruction by minimization of the 2D reprojection error in multiple images is considered. Although bundle adjustment techniques can be used to minimize the 2D reprojection error, these methods being based on nonlinear optimization algorithms require a good starting point. Quasi-linear algorithms with better global convergence properties can be used to generate an initial solution before submitting it to bundle adjustment for refinement. In this paper, we propose a factorization-based method to integrate the initial search as well as the bundle adjustment into a single algorithm consisting of a sequence of weighted least-squares problems, in which a control parameter is initially set to a relaxed state to allow the search of a good initial solution, and subsequently tightened up to force the final solution to approach a minimum point of the 2D reprojection error. The proposed algorithm is guaranteed to converge. Our method readily handles images with missing points.

Keywords: multiple views, projective reconstruction, structure and motion, sub-space method, factorization method, projective bundle adjustment

1. Introduction

There are many existing approaches for reconstructing 3D Euclidean structure from multiple 2D images (Hartley, 1993; Faugeras, 1995; Pollefeys and Gool, 1999; Han and Kanade, 2000, 2001; Triggs et al., 2000; Chen and Medioni, 2002). Often, a projective reconstruction is a necessary step in the process whereby matched correspondences between 2D planar images are used to recover the 3D motion and structure in a projective space. There has been considerable interest in the factorization approach (for affine, Tomasi and Kanade, 1992) to projective reconstruction proposed by Sturm and Triggs (1996). The factorization approach has the advantage that it is a multi-view approach that handles all of images uniformly without preferential treatment for any image.

Suppose a set of 3D points with homogeneous coordinates $X_j = [x_j y_j z_j 1]^T$ ($j = 1, \dots, n$) are projected onto m cameras with projection matrices P_i ($i = 1, \dots, m$). Let $x_{ij} = [u_{ij} v_{ij} 1]^T$ be the projection of the j th point on the i th view, i.e.,

$$\lambda_{ij} x_{ij} = P_i X_j \quad (1)$$

where λ_{ij} represents the *depth* of X_j measured along the optical axis of i th camera. Then, the projection of all the 3D points onto all the cameras can be represented as

$$\begin{bmatrix} \lambda_{11} x_{11} & \cdots & \lambda_{1n} x_{1n} \\ \cdots & \cdots & \cdots \\ \lambda_{m1} x_{m1} & \cdots & \lambda_{mn} x_{mn} \end{bmatrix} = P X \in \mathfrak{R}^{3m \times n} \quad (2)$$

where

$$P = [P_1^T, P_2^T, \dots, P_m^T]^T \in \mathfrak{R}^{3m \times 4}$$

is the *joint projection matrix* and

$$X = [X_1, X_2, \dots, X_n] \in \mathfrak{R}^{4 \times n}$$

is the *projective shape*. Since P and X are at most rank 4, the *scaled measurement matrix* $[\lambda_{ij}x_{ij}]$ is *at most rank 4 too*. In general, the depths $\{\lambda_{ij}\}$ are unknown. In the factorization approach, a consistent set of projective depths need to be estimated such that the scaled measurement matrix is of rank 4 and factorizable in the form of (2). Many different methods have been proposed for estimating λ_{ij} . Sturm and Triggs (1996) proposed to recover the projective depths by means of the epipolar constraints between two views, which has the advantage of being non-iterative, but the method is indirect requiring the estimation of the fundamental matrix between pairs of views. Triggs (1996) extended the method by refining projective depths and factorizing (2) iteratively until the projective depths converge. Most of the other approaches use iterative methods for estimating the unknown λ_{ij} in (2). Sparr (1996) proposed an iterative factorization algorithm for simultaneous scene and motion reconstruction. Chen and Medioni (1999, 2002) developed an iterative eigen algorithm which minimizes a weighted version of the algebraic error in Eq. (2) by repeatedly performing intersection for the points followed by resection for the cameras. Heyden et al. (1999) proposed a subspace method for estimating the projective depths with the advantage that the result is independent of the coordinate representation of image points. Mahamud and Herbert (2000) applied a subspace constraint to the columns of the scaled measurement matrix to estimate the projective depths by an iterative factorization algorithm. Since the solutions in these works are iterative, convergence of the algorithms is an important issue, which however is not always addressed in existing methods. Discussions of the convergence of factorization algorithms are given in Oliensis (1996) and Mahamud et al. (2001). Most of the above iterative methods are based on quasi-linear algorithms where P and X are estimated alternately (together with λ_{ij}) using linear least-squares techniques.

Although the 2D reprojection error is the most sensible measure for minimization in a projective reconstruction, existing methods are mostly based on minimization of some algebraic error or subspace proximity

measure whose relationship with the 2D reprojection error is not clear. Bundle adjustment can be applied subsequently to minimize the 2D reprojection error. Bundle adjustment (e.g. Hartley, 1993; Morris et al., 1999; Shum et al., 1999; Triggs et al., 2000; Bartoli and Sturm, 2001) is a method for refining the 3D structure and camera motions simultaneously by minimizing the sum of squared distances between the reprojected points and measured points. In practice, bundle adjustment is accomplished by non-linear optimization algorithms. The reconstruction result relies on the initial estimate since most existing algorithms are based on local descent methods. Shum et al. (1999) used a hierarchical approach to perform bundle adjustment as proposed in Hartley (1993) on small sub-sequences and then merge the results into a complete reconstruction so that faster convergence is achieved over conventional bundle adjustment methods. Bartoli and Sturm (2001) developed three different bundle adjustment methods in a projective frame for simpler implementation and minimal parametrization but they are applicable to two-view cases only. Chen and Medioni (2002) proposed to solve a sequence of iterative eigen problems instead of using standard optimization techniques for bundle adjustment in a projective frame. However, convergence is not guaranteed.

In general, existing factorization methods suffer from one or more of the following drawbacks:

- (i) requires a quasi-linear algorithm to generate a good initial solution before submitting it to bundle adjustment;
- (ii) convergence of quasi-linear algorithms not guaranteed;
- (iii) a lack of provision for missing points.

In this paper, we propose a factorization-based method which integrates the initial search and the projective bundle adjustment into a single algorithm that addresses all the above problems. Our approach aims at minimizing the 2D reprojection error by solving a sequence of relaxed problems each of which approximately minimizes the 2D reprojection error. A control parameter is used in the sequence of problems to force the solutions of the relaxed problems to approach a minimum of the 2D reprojection error. A key feature in our solution is that the inverse depth rather than the depth is estimated. Theoretical results are provided to ensure convergence of the algorithmic solution. The problem of missing points is readily handled in our method. Compared with existing bundle adjustment

methods, our method does not rely on the provision of a good initial estimate.

The paper is organized as follows. The factorization problem for projective reconstruction is discussed in Section 2 where a relaxed version of the problem of minimizing 2D reprojection errors is introduced. In Section 3, a discussion of how the solution to the relaxed problem is related to the 2D reprojection error is given and an iterative procedure is developed for minimizing the 2D reprojection error. A discussion of how missing points are handled is given in Section 4. Simulation results using synthetic data and experimental results based on real images are given in Section 5 to illustrate the performance of our algorithm in comparison with existing factorization methods. Section 6 contains some concluding remarks.

Notation: The Hadamard product of two matrices $A = [a_{ij}]$ and $B = [b_{ij}]$ of the same size is denoted

$$A * B = [a_{ij}b_{ij}].$$

2. Problem Formulation

Given image coordinates x_{ij} ($i = 1, \dots, m; j = 1, \dots, n$), the factorization problem is to determine projective depths λ_{ij} so that the scaled measurement matrix can be factorized into two rank-4 matrices as in (2). If the image coordinates contain noise, the factorization can only be approximate and the results will depend on the criterion used in the factorization. An obvious quantity to minimize is the algebraic error:

$$\min_{\lambda_{ij}, P_i, X_j} \sum_{i,j} \|\lambda_{ij}x_{ij} - P_i X_j\|^2. \quad (3)$$

where the summation is taken over $1 \leq i \leq m$ and $1 \leq j \leq n$. For (3) to be a meaningful minimization problem, constraints need to be imposed on the size of λ_{ij} , P_i or X_j to avoid the trivial solution $\lambda_{ij} = 0$, $P_i = 0$ and $X_j = 0$.

The algebraic error has the advantage of being linear in λ_{ij} and bilinear in P_i and X_j enabling various iterative schemes for minimizing (3). However, the algebraic error does not have a geometric meaning and so the minimization problem (3) does not have a clear physical interpretation. Other criteria (Heyden et al., 1999; Mahamud and Hebert, 2000; Mahamud et al., 2001) have been proposed based on measures of closeness of column or row space of $[\lambda_{ij}X_{ij}]$ to the subspaces spanned by the columns of P_i or the rows of X_j , respectively.

These methods provide some geometric meanings of the error being minimized in terms distance between subspaces. However, the most appropriate quantity for minimization is the 2D reprojection error. Let P_j^k denote the k th row of P_j ($k = 1, 2, 3$). The problem of minimizing the 2D reprojection error can be stated as

$$\min_{\lambda_{ij}, P_i, X_j} \sum_{i,j} \left[\left(u_{ij} - \frac{1}{\lambda_{ij}} P_i^1 X_j \right)^2 + \left(v_{ij} - \frac{1}{\lambda_{ij}} P_i^2 X_j \right)^2 \right] \quad (4)$$

subject to

$$\lambda_{ij} = P_i^3 X_j. \quad (5)$$

The difficulty of the optimization problem (4) is that it is a constrained problem with a nonlinear cost function. To overcome this difficulty, we propose to replace the depth λ_{ij} by the inverse depth:

$$\beta_{ij} = \frac{1}{\lambda_{ij}} \quad (6)$$

so that (4) can be written equivalently as

$$\min_{\beta_{ij}, P_i, X_j} \sum_{i,j} \left[(u_{ij} - \beta_{ij} P_i^1 X_j)^2 + (v_{ij} - \beta_{ij} P_i^2 X_j)^2 \right] \quad (7)$$

subject to

$$\beta_{ij} P_i^3 X_j = 1. \quad (8)$$

The 2D reprojection error in (7) is now trilinear in β_{ij} , P_i and X_j , which is amenable to an iterative solution. Although one may enforce the constraint $\beta_{ij} P_i^3 X_j = 1$ in an iterative solution to the minimization problem (7), the constrained problem is stiff making the speed of convergence very slow. This difficulty can be alleviated by relaxing the hard constraint and replacing it by means of a penalty term in the cost function. A relaxed version of (7) is:

$$\min_{\beta_{ij}, P_i, X_j} \sum_{i,j} \left[(u_{ij} - \beta_{ij} P_i^1 X_j)^2 + (v_{ij} - \beta_{ij} P_i^2 X_j)^2 + \gamma^2 \gamma_{ij}^2 (1 - \beta_{ij} P_i^3 X_j)^2 \right] \quad (9)$$

where $\gamma_{ij} \neq 0$ ($i = 1, \dots, m; j = 1, \dots, n$) are constant weighting factors for individual image points, and γ is an overall weighting factor for adjusting the degree to which the constraint $\beta_{ij} P_i^3 X_j = 1$ is to be enforced. In (9), γ_{ij} is used to scale $(1 - \beta_{ij} P_i^3 X_j)$ to a magnitude compatible with the 2D reprojection error, e.g., $\gamma_{ij} = \max(|u_{ij}|, |v_{ij}|)$.

Let

$$\beta = \begin{bmatrix} \beta_{11} & \cdots & \beta_{1n} \\ \cdots & \cdots & \cdots \\ \beta_{m1} & \cdots & \beta_{mn} \end{bmatrix} \in \mathfrak{R}^{m \times n} \quad \text{and} \quad \bar{\gamma}_{ij} = \begin{bmatrix} 1 \\ 1 \\ \gamma \gamma_{ij} \end{bmatrix}. \quad (10)$$

We shall denote the cost function in (9) as

$$F_\gamma(P, X, \beta) = \sum_{i,j} \|\bar{\gamma}_{ij} * (x_{ij} - \beta_{ij} P_i X_j)\|^2. \quad (11)$$

The minimization problem (9) can be expressed more succinctly as

$$\min_{P, X, \beta} F_\gamma(P, X, \beta). \quad (12)$$

Given $\gamma > 0$, denote an optimal solution to (12) by $(P(\gamma), X(\gamma), \beta(\gamma))$. (i.e., P, X and β written with an argument γ mean that they minimize $F_\gamma(P, X, \beta)$ for that γ). The corresponding minimum will be denoted

$$\begin{aligned} F^*(\gamma) &= F_\gamma(P(\gamma), X(\gamma), \beta(\gamma)) \\ &= \min_{P, X, \beta} F_\gamma(P, X, \beta). \end{aligned} \quad (13)$$

For a fixed γ , we may use the following iterative algorithm to solve for $(P(\gamma), X(\gamma), \beta(\gamma))$. In each iteration, we alternately solve for one of the three variables P, X and β as a free parameter while fixing the other two variables. Since $F_\gamma(P, X, \beta)$ is trilinear in P, X and β , the minimization with respect to each variable is a weighted linear least-squares problem solvable by standard techniques.

2.1. Algorithm 1

1. Put $k=0$ and assign initial values to β° (e.g. set $\beta_{ij}^\circ = 1 \forall i, j$; perform an SVD of $[\frac{1}{\beta_{ij}^\circ} x_{ij}]$ and

obtain P° and X° from a rank-4 approximation retaining only the largest four singular values).

2. Put $k = k+1$.

Fix X^{k-1} and β^{k-1} and determine P^k by solving

$$\varepsilon'_k = \min_{P^k \in \mathfrak{R}^{3m \times 4}} F_\gamma(P^k, X^{k-1}, \beta^{k-1}). \quad (14)$$

3. Fix β^{k-1} and P^k and determine X^k by solving

$$\varepsilon''_k = \min_{X^k \in \mathfrak{R}^{4 \times n}} F_\gamma(P^k, X^k, \beta^{k-1}). \quad (15)$$

4. Fix P^k and X^k and determine β^k by solving

$$\varepsilon_k = \min_{\beta^k \in \mathfrak{R}^{m \times n}} F_\gamma(P^k, X^k, \beta^k). \quad (16)$$

5. Repeat steps 2, 3 and 4 until ε_k converges.
6. Output $\beta(\gamma) = \beta^k, P(\gamma) = P^k, X(\gamma) = X^k$ and stop.

Clearly, the cost ε_k in the above algorithm is monotonic decreasing satisfying

$$\varepsilon'_k \geq \varepsilon''_k \geq \varepsilon_k \geq \cdots \geq \varepsilon_{k+1} \geq 0.$$

Hence, the algorithm is guaranteed to converge. Since the constraint $\beta_{ij} P_i^3 X_j = 1$ has been relaxed, the solution to (12) does not minimize the 2D reprojection error, but provides only an approximate solution to (7) and (8). How good the approximation is depends on the weighting factor γ . We may force the unconstrained problem (12) to approach the constrained problem (7) by letting $\gamma \rightarrow \infty$. However, using a large value of γ to start with in the above algorithm will bring about the same difficulties as the constrained problem (7). Instead, we propose to solve a series of problems of the form (12) with increasing values of γ taken from a monotonic increasing sequence $\{\gamma_k\}$. A recommended strategy is to choose $\gamma_k = \alpha^k$, where $\alpha > 1$ is a constant (e.g. $\alpha = 1.1$ is used in the examples of Section 5). As the initial values in Step 1 of Algorithm 1 are computed under the explicit assumption that all $\beta_{ij} = 1$ and the implicit assumption that $\gamma = 1$, it is appropriate to start the iterative solution for the problem with an initial value of $\gamma_o = 1$. To enforce the constraint (8), γ is progressively increased by taking successive values of $\{\gamma_k\}$ and the solution to the previous problem is used as the starting point for re-solving (12) with the increased γ . As γ increases, the solution to the series of problems (12) will approach a minimum point for the 2D reprojection error. In the next section, we will

develop some theoretical results to justify the proposed approach.

3. Minimization of 2D Reprojection Error

To consider how the solution to the unconstrained problem (9) is related to the 2D reprojection error, we denote the 2D reprojection error for any solution pair (P, X) by

$$E(P, X) = \sum_{i,j} \left[\left(u_{ij} - \frac{P_i^1 X_j}{P_i^3 X_j} \right)^2 + \left(v_{ij} - \frac{P_i^2 X_j}{P_i^3 X_j} \right)^2 \right].$$

From (11), we have

$$F_\gamma(P, X, \beta) = \sum_{i,j} \left[\left(u_{ij} - \beta_{ij} P_i^1 X_j \right)^2 + \left(v_{ij} - \beta_{ij} P_i^2 X_j \right)^2 + \gamma^2 \gamma_{ij}^2 \left(1 - \beta_{ij} P_i^3 X_j \right)^2 \right].$$

Note that $F_\gamma(P, X, \beta)$ can always be reduced to the 2D reprojection error by setting β_{ij} equal to the inverse depths, i.e.,

$$E(P, X) = F_\gamma(P, X, \beta) \Big|_{\beta_{ij} = \frac{1}{P_i^3 X_j} \forall i,j}. \quad (17)$$

Suppose $E(P, X)$ achieves the minimum 2D reprojection error at (P^*, X^*) . Denote

$$E^* = E(P^*, X^*) = \min_{P, X} E(P, X). \quad (18)$$

Some properties of $F^*(\gamma)$ are given in the next theorem.

Theorem 1.

- $F^*(\gamma)$ is a monotonic increasing function of γ .
- $\forall \gamma > 0, F^*(\gamma) \leq E^* \leq E(P(\gamma), X(\gamma))$
- $\sum_{i,j} \gamma_{ij}^2 (1 - \beta_{ij}(\gamma) \lambda_{ij}(\gamma))^2$ is a monotonic decreasing function of γ , where $\beta_{ij}(\gamma)$ is the (i, j) th entry of $\beta(\gamma)$ and $\lambda_{ij}(\gamma)$ is the $(3i, j)$ th entry of $[P(\gamma)X(\gamma)]$.
- For any (i, j) such that $\gamma_{ij} \neq 0$, $\lim_{\gamma \rightarrow \infty} \beta_{ij}(\gamma) \lambda_{ij}(\gamma) = 1$.

A proof of Theorem 1 is given in the Appendix.

Remark 1.

- By Theorem 1 (a) and (b), since $F^*(\gamma)$ is monotonic increasing with γ and is bounded above by E^* , it follows that $F^*(\gamma)$ will converge as $\gamma \rightarrow \infty$ to a solution also bounded above by E^* , i.e.,

$$\lim_{\gamma \rightarrow \infty} F^*(\gamma) \leq E^*. \quad (19)$$

- Part (d) of the Theorem 1 shows that as $\gamma \rightarrow \infty$, $\beta_{ij}(\gamma)$ will converge to the inverse depth of the image point x_{ij} as long as the weighting factor γ_{ij} is non-trivial.

We can in fact go further than (19) and show that $F^*(\gamma)$ converges to the 2D reprojection error E^* as $\gamma \rightarrow \infty$. This will be established in Theorem 2 below. First, we note that the expression in (11) can be decomposed as

$$x_{ij} - \beta_{ij} P_i X_j = \left(x_{ij} - \frac{1}{\lambda_{ij}} P_i X_j \right) + \left(\frac{1}{\lambda_{ij}} - \beta_{ij} \right) P_i X_j \quad (20)$$

where λ_{ij} is given by (5). The first component on the right hand side of (20) represents 2D reprojection error whereas the second component is called the truncation error in Triggs (1998). For any (P, X, β) , we will denote the weighted total truncation error by

$$T_\gamma(P, X, \beta) = \sum_{i,j} \left\| \left(\frac{1}{\lambda_{ij}} - \beta_{ij} \right) (\tilde{\gamma}_{ij} * P_i X_j) \right\|^2. \quad (21)$$

Theorem 2.

- Let $(P(\gamma), X(\gamma), \beta(\gamma))$ be an optimal solution to (12). Then,

$$E(P(\gamma), X(\gamma)) = F^*(\gamma) + T_\gamma(P(\gamma), X(\gamma), \beta(\gamma)) \quad (22)$$

- $\lim_{\gamma \rightarrow \infty} T_\gamma(P(\gamma), X(\gamma), \beta(\gamma)) = 0$
- $\lim_{\gamma \rightarrow \infty} F^*(\gamma) = E^*$

A proof of Theorem 2 can be found in the Appendix.

Suppose the problem (12) is solved for values of γ taken from a monotonic increasing sequence $\{\gamma_k\}$. By Theorem 1(a) and Theorem 2(c), $F^*(\gamma_k)$ is a monotonic increasing sequence converging to E^* . Furthermore, $F^*(\gamma_k)$ is bounded above by the sequence $E(P(\gamma_k), X(\gamma_k))$. Note that $E(P(\gamma_k), X(\gamma_k))$ overbounds E^* and is also convergent to E^* . Hence, as k increases, the two sequences $\{F^*(\gamma_k)\}$ and $\{E(P(\gamma_k), X(\gamma_k))\}$ should approach E^* from opposite sides, squeezing an interval containing E^* . The difference $[E(P(\gamma_k), X(\gamma_k)) - F^*(\gamma_k)]$ can therefore be used as a measure for indicating convergence. The following algorithm provides an iterative procedure for obtaining a solution (P^*, X^*, β^*) where the 2D reprojection error is a minimum.

3.1. Algorithm 2

1. Put $k = 1$.
2. Use Algorithm 1 to solve for $(P(\gamma_k), X(\gamma_k), \beta(\gamma_k))$ from

$$F^*(\gamma_k) = \min_{P, X, \beta} F_{\gamma_k}(P, X, \beta)$$

where $(P(\gamma_{k-1}), X(\gamma_{k-1}), \beta(\gamma_{k-1}))$ is used as the starting point for the minimization if $k > 1$.

3. Evaluate $E(P(\gamma_k), X(\gamma_k))$.
If $E(P(\gamma_k), X(\gamma_k)) - F^*(\gamma_k) > \varepsilon$ (a prescribed threshold),
 put $k = k + 1$ and return to Step 2;
else
 output $(P^*, X^*, \beta^*) = (P(\gamma_k), X(\gamma_k), \beta(\gamma_k))$
 and stop.

Theorems 1 and 2 provide the basis for deducing the asymptotic behavior of $F^*(\gamma)$. If the global minimum to the trilinear problem in Step 2 of Algorithm 2 can be obtained for each γ_k , then asymptotically $F^*(\gamma_k)$ will approach the global minimum of the 2D reprojection error as $\gamma_k \rightarrow \infty$. However, while Algorithms 1 and 2 are proven to converge, there is no guarantee that they converge to the global minimum. As Algorithm 1 may converge to a local minimum of the trilinear problem, Algorithm 2 will at best converge to a local minimum of the 2D reprojection error. Our experience suggests that Algorithm 2 is often able to find a solution which for practical purposes is (very close to) the global minimum. There are, however, instances where

the algorithm converges to a local minimum of the geometric reprojection error. In this case, the property that $F^*(\gamma_k)$ and $E(P(\gamma_k), X(\gamma_k))$ bracket the local minimum of the geometric reprojection error remains valid, and the size of the bracket will still approach 0 as $\gamma_k \rightarrow \infty$.

4. Missing Data

The ability to handle missing points is essential for any multi-view techniques, as there are bound to be missing data in real images due to occlusion. Missing data results in ‘holes’ in the measurement matrix (2) and therefore creates difficulties for methods that operate on (2) as a matrix. Let the set of indices of available image points be

$$\mathcal{A} = \{(i, j) \mid x_{ij} \text{ is observed as point } j \text{ on view } i\}. \quad (23)$$

Various methods are available for handling missing data in projective reconstruction, such as the sequential updating method of Beardsley (1997), the linear fitting method of Jacobs (2001), and the parametric approach of Shum et al. (1995). As far as missing points are concerned, our method is similar to the approach of Shum et al. (1995) in that the cost function for minimization is defined over available data only and so there is no need to pay special attention to the missing data as they simply do not feature in the minimization problem.

Since our algorithms do not rely on any matrix operation on (2) (except for an initialization step in Algorithm 1), the algorithms can be readily applied despite missing data. If there are missing points, then in Step 1 of Algorithm 1, the matrix $[\frac{1}{\beta_{ij}^0} x_{ij}]$ is formed by setting all missing entries x_{ij} to the centroids of the visible 2D points in each image. All the results and Theorems 1 and 2 remain valid if we replace the summation over all (i, j) by a summation over available measurements with indices in the set \mathcal{A} . Once the joint projection matrix P^* and the shape matrix X^* are estimated using available data, missing points x_{ij} can be filled in by projecting X_j^* onto view i by means of P_i^* if required.

5. Experimental Results

In this section, the proposed method is first evaluated by means of synthetic images for which ground truth is known. Corresponding results using two other

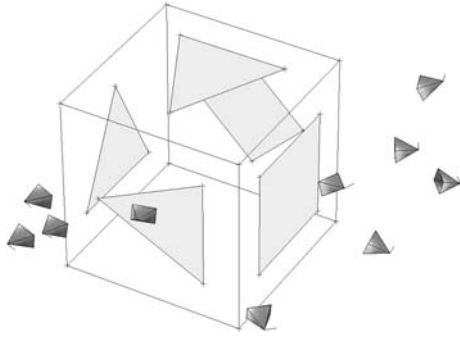


Figure 1. A synthetic scene with a virtual box.

established methods, namely Sturm-Triggs' algorithm (Sturm and Triggs, 1996) and Heyden's subspace method (Heyden et al., 1999), are given for the sake of comparison. We note here that the Sturm-Triggs' algorithm is one of the very first factorization methods proposed and is not expected to perform well on the criteria used in later works. The results of Sturm-Triggs' algorithm are nevertheless given below as a reference for comparison. Finally, an example using real images will be provided.

5.1. Synthetic Example

A synthetic scene is made up consisting of a virtual box as shown in Fig. 1. The box has size $60 \times 60 \times 60$ cm and contains a total of 25 feature points including corners of the box and patterns on its sides. Twelve images of the box are synthesized for cameras placed around two sides of the box at distances in the range of 70 cm to 100 cm from the centre of the box. All cameras have the same intrinsic parameters and the size of each image is 1080×720 . The cameras are oriented with their optical axes pointing towards the centre of the box. The location of the cameras are otherwise selected in a random manner.

Gaussian noise of different noise levels (with standard deviation ranging from 0 to 4 pixels in 0.5 pixel increments) are introduced independently to the x and y coordinates of each 2D image point. The algorithms are run repeatedly for 30 trials using different randomly generated noise and the graphs to be given show the mean values of the simulation results.

5.1.1. Performance on 2D Reprojection Error. The 2D reprojection errors (relative to the noisy data) are plotted in Fig. 2. We have also applied the

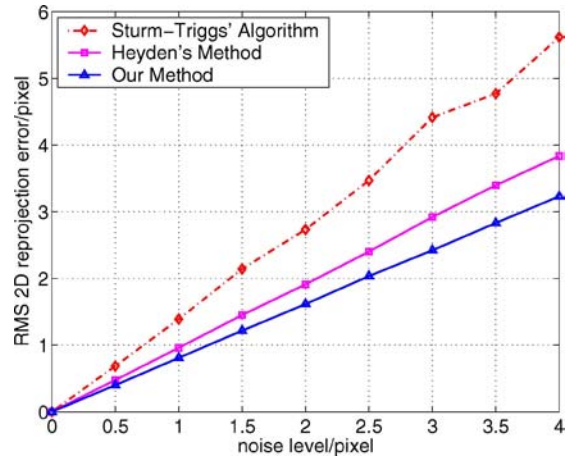


Figure 2. 2D reprojection error.

Sturm-Triggs' algorithm (Sturm and Triggs, 1996) and Heyden's subspace method (Heyden et al., 1999) to the same data set. For Sturm-Triggs' method, we have taken the 1st camera (leftmost in Fig. 1) as the reference for the estimation of fundamental matrices. Since neither of the above methods minimizes the 2D reprojection error, it is expected that our method produces a smaller 2D reprojection error. On average, the 2D reprojection error increases almost linearly with the noise level and has magnitude roughly matching (in fact lower than) the noise level.

5.1.2. Performance on Estimating Projective Depths.

Since the projective depths cannot be uniquely recovered in a projective reconstruction, in order to compare how good the estimated projective depths are, we make use of the cross ratio of projective depths defined as:

$$\hat{c}_{ij} = \frac{\hat{\lambda}_{ij}\hat{\lambda}_{i+1,j+1}}{\hat{\lambda}_{i,j+1}\hat{\lambda}_{i+1,j}}, \quad (i = 1, \dots, m-1; j = 1, \dots, n-1) \quad (24)$$

where $\hat{\lambda}_{ij}$ represents the estimated projective depths. The mean-squared cross-ratio error (MSCRE) is then defined as

$$\frac{1}{(m-1)(n-1)} \sum_{i,j} (c_{ij} - \hat{c}_{ij})^2 \quad (25)$$

where c_{ij} are the cross ratios computed using the ground truth depths. The MSCRE of the results of our method as well as Sturm-Triggs' and Heyden's methods are shown in Fig. 3.

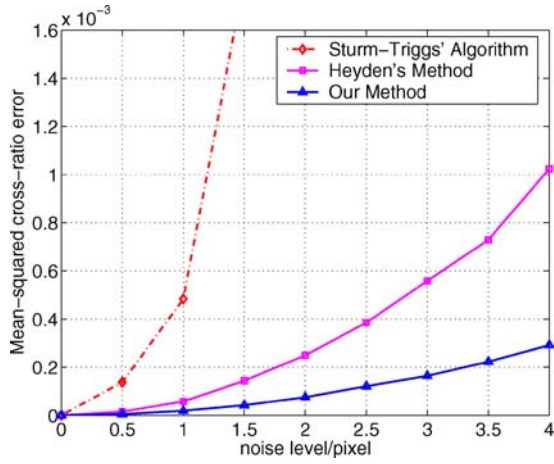


Figure 3. Projective-depth cross-ratio error.

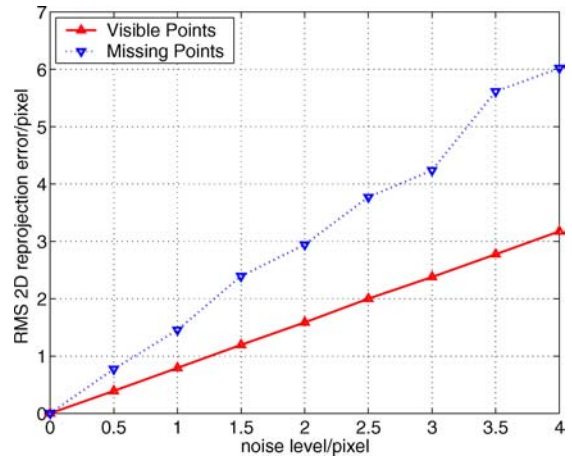


Figure 5. 2D reprojection error when there are missing data.

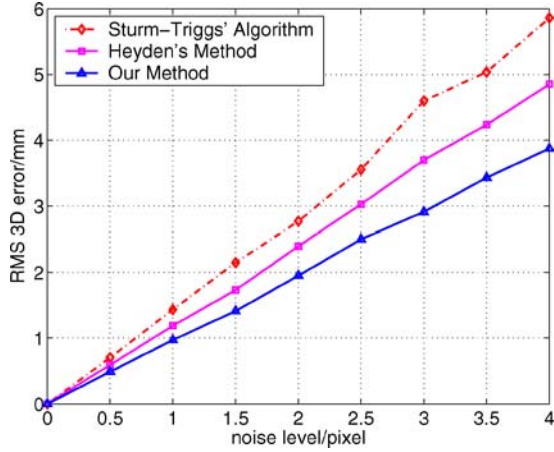


Figure 4. 3D error.

5.1.3. Performance on 3D Error. The evaluation on 3D error is performed by upgrading the reconstructed 3D points \hat{X}_j in the projective space to a Euclidean space by means of a collineation $T \in \mathbb{R}^{4 \times 4}$ obtained by minimizing

$$e_{3D} = \sqrt{\min_T \frac{1}{n} \sum_j \|X_j - \alpha_j T \hat{X}_j\|^2} \quad (26)$$

where X represents the ground truth 3D points and α_j is a scaling factor for normalizing the 4th component of $T \hat{X}_j$ to 1. The RMS 3D error e_{3D} plotted against different noise levels are shown in Fig. 4. In general, 3D points for reconstructions with smaller 2D reprojection error are closer to the ground truth.

5.1.4. Performance on Missing Data. To simulated missing data, 17 image points are randomly selected and removed from the measurement matrix and these are regarded as missing data. Only our method is used to perform a projective reconstruction, as the issue of missing points is not dealt with explicitly in Sturm and Triggs (1996) or Heyden et al. (1999). The root-mean-squared 2D reprojection errors for both visible points and missing points are shown in Fig. 5. The RMS 2D reprojection error for the visible points are measured relative to the noisy data whereas the error for the missing points are measured relative to the ground truth as noisy data for these points supposedly do not exist. The errors of the estimated missing points are comparable with the noise level corrupting the visible points.

Figure 6 shows the performance of our method in terms of 3D error for varying percentages of missing data from 0 to 40% with 5% increments. The data set is contaminated by Gaussian noise with standard deviation $\sigma = 2$ pixels. There are 30 trials for each percentage of missing data with 2D points randomly marked as missing data. We additionally restrict that there are at least 7 points visible to every 3 consecutive views for each trial. This condition becomes increasingly hard to satisfy when the percentage of missing points exceeds 40% as the scene contains only 25 feature points. The 3D error is computed as in (26) and the average RMS and MAX 3D errors are shown in Fig. 6. The RMS 3D errors are reasonably small compared with the size of the box.

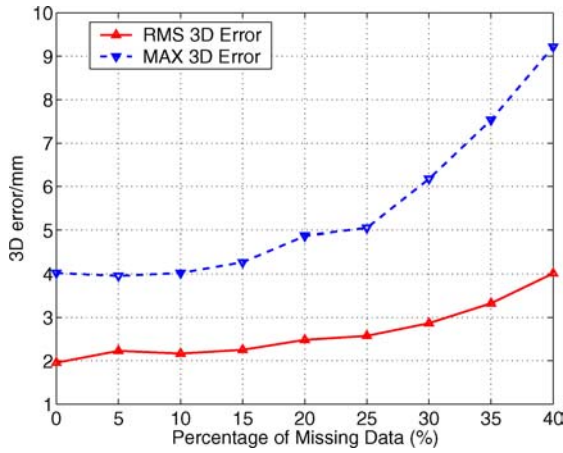


Figure 6. 3D error versus percentage of missing data (noise level $\sigma = 2$ pixels).

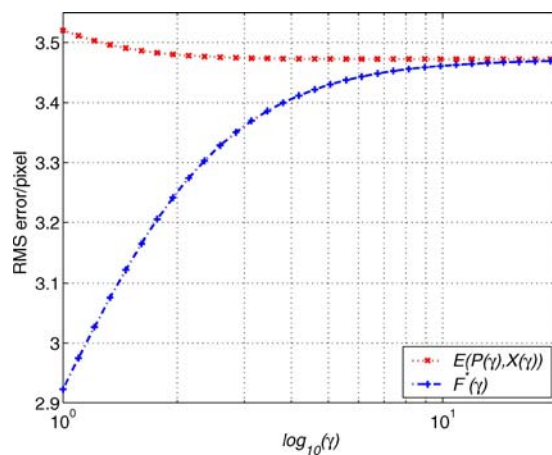


Figure 7. Convergence of algorithm.

5.1.5. Convergence of Algorithm. As far as convergence is concerned, our method compares favourably with existing methods. In Algorithm 2, we set $\gamma_k = (1.1)^k$ and Algorithm 1 is run for each γ_k until convergence is attained. To illustrate the use of Theorem 2, a plot of the cost function $F^*(\gamma_k)$ being minimized and the 2D reprojection error $E(P(\gamma_k), X(\gamma_k))$ versus γ_k (in log scale) is shown in Fig. 7. By the results of Section 3, $F^*(\gamma_k)$ is monotonic increasing and should converge to a minimum point E^* of the 2D reprojection error. As $E(P(\gamma_k), X(\gamma_k))$ converges towards E^* from above, $F^*(\gamma_k)$ should increase towards the same value, thereby squeezing E^* in between. The closeness of $F^*(\gamma_k)$ to $E(P(\gamma_k), X(\gamma_k))$ can be used to judge whether convergence has been attained. This is in contrast to existing algorithms which does not provide any

Table 1. Comparisons of algorithms.

	No. of iterations	Run-time (sec)	RMS 2D reprojection error (pixel)	RMS 3D error (mm)
Our method	72	0.87	3.236	3.881
Heyden's method	17	0.44	3.837	4.850
Bundle adjustment (BA) only	18	67.6	3.446	4.639
(Heyden +) BA	(17+)5	14.3	3.250	3.888

means of determining convergence other than changes in the cost function being minimized.

A comparison of our method with Heyden's method and standard bundle adjustment in terms of convergence, the accuracy of solution and run-times is given in Table 1, which contains results obtained by averaging 30 trials on data sets with a noise level of 4 pixels. All algorithms are run on a 2.4 GHz Pentium PC. In the case of the proposed method, all iterations within Algorithm 1 have been counted. Heyden's method converges in fewer iterations and is faster but both the 2D and 3D errors are larger. The bundle adjustment used here is based on Powell's dog-leg method (Powell, 1970) and is implemented using the Matlab Optimization Toolbox. The bundle adjustment started from the same initial condition as our method converges to a solution which is slightly worse than but close to the solution from our method. Despite the apparently small number of iterations, bundle adjustment requires a much longer run-time because the algorithm contains a nested iterative process



Figure 8. The second image of the image sequence 'Arc de Triomphe'.

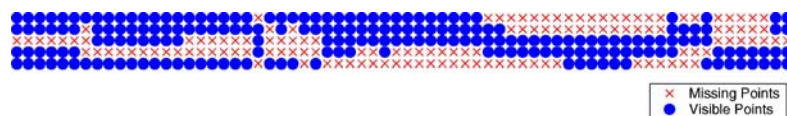


Figure 9. The distribution of available and missing data in the measurement matrix of the Arc de Triomphe sequence. The i th row represents the i th image and the j th column represents the j th 3D point.

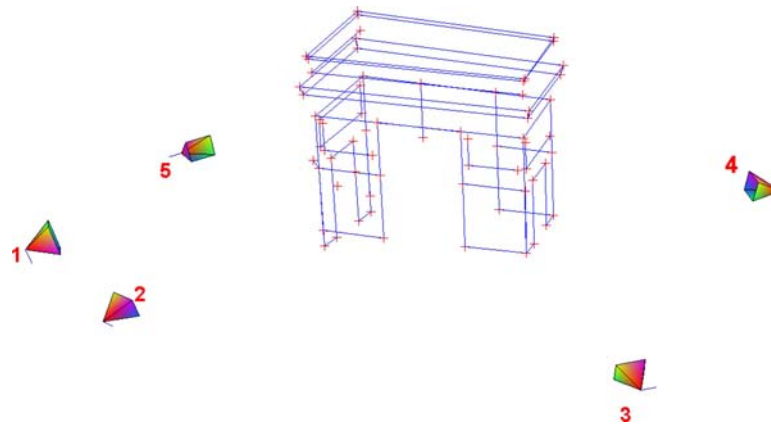


Figure 10. A scene of the reconstructed wire-frame of 'Arc de Triomphe'.

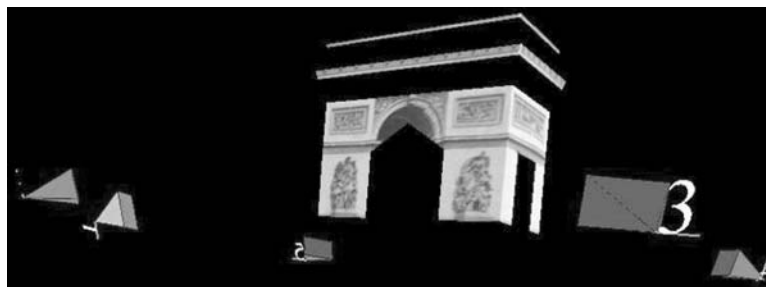


Figure 11. A view of 'Arc de Triomphe' with texture.

where each outer-loop iteration requires several tens of inner-loop iterations. We have also used the result from Heyden's method as the initial solution for bundle adjustment, which produces a solution performing equally well as our method, but requires a run-time more than 10 times that of our method. We have also run the bundle adjustment with an initialization provided by our method, which shows that no further improvement can be made, and therefore confirms that our method indeed reaches a local minimum.

5.2. 3D Reconstruction Using Real Images

Five photographs are captured by a camera which was moved a complete revolution around the Arc de

Triomphe so that each of its four facades is visible in at least one image. The photographs are digitized by a film scanner to give images of size 1850×1205 pixels. Figure 8 shows the second image of the sequence. A total of 68 feature points are matched across the five images. Figure 9 shows a map of the available and missing points in the measurement matrix. There are a total of 133 missing points in the 5 images. Due to the high percentage (39.1%) of missing data, it requires a total of about 5,000 iterations for our algorithm to converge, which takes about 1 minute for our implementation of the algorithm in Matlab 6.5 running on a 2.4 GHz Pentium PC. The maximum 2D reprojection error among all views is 2.129 pixels while the RMS 2D reprojection error is 0.769 pixels. For comparison, standard bundle adjustment requires about 60 minutes

to converge to a solution with RMS 2D reprojection error of 1.97 pixels.

To see whether our projective reconstruction is reasonable, we upgrade the projective reconstruction to a Euclidean space by means of the normalization method of Han and Kanade (2000, 2001). Assuming that principal points of the cameras are fixed and the skew ratios are zero, a scene of the wire-frame reconstruction is shown in Fig. 10, and a synthesized scene with some of the surfaces textured is given in Fig. 11.

6. Conclusion

In this paper, a projective reconstruction method for multiple views is developed to estimate the (inverse) projective depths while minimizing the 2D reprojection error. It is shown that the algorithm is guaranteed to converge to a minimum E^* of the 2D reprojection error. An indicator for the attainment of convergence is provided in terms of two different qualities, namely $F^*(\gamma_k)$ and $E(P(\gamma_k), X(\gamma_k))$, both of which can be monitored during the algorithm, squeezing towards E^* . Furthermore, missing data can be readily handled by the proposed method. Simulations show that the method is robust to image noise, giving solutions with mean 2D reprojection error matching the level of noise corrupting the image points. The projective reconstruction obtained using the method of this paper can be used as a basis for a Euclidean reconstruction.

Appendix

Proof of Theorem 1: Consider $0 < \gamma_1 < \gamma_2 < \infty$.

- (a) Noting that each of the triple $(P(\gamma_k), X(\gamma_k), \beta(\gamma_k))$ ($k = 1, 2$) achieves the minimum of its own objective function $F_{\gamma_k}(P, X, \beta)$, we have

$$\begin{aligned} F^*(\gamma_1) &= F_{\gamma_1}(P(\gamma_1), X(\gamma_1), \beta(\gamma_1)) \\ &\leq F_{\gamma_1}(P(\gamma_2), X(\gamma_2), \beta(\gamma_2)) \\ &\leq F_{\gamma_2}(P(\gamma_2), X(\gamma_2), \beta(\gamma_2)) \\ &\leq F_{\gamma_2}(P(\gamma_1), X(\gamma_1), \beta(\gamma_1)) \\ &= F^*(\gamma_2) \end{aligned}$$

where the second inequality follows from the fact that $F_{\gamma}(P, X, \beta)$ is monotonic increasing in γ by definition (11). It follows that $F^*(\gamma)$ is a monotonic increasing function of γ .

- (b) For any $\gamma > 0$, we have

$$\begin{aligned} F^*(\gamma) &= \min_{P, X, \beta} F_{\gamma}(P, X, \beta) \\ &\leq \min_{P, X} F_{\gamma}(P, X, \beta) \Big|_{\beta_{ij} = \frac{1}{P_i^3 X_j} \forall i, j} \\ &= \min_{P, X} E(P, X) = E^*. \end{aligned}$$

Clearly, $E^* \leq E(P(\gamma), X(\gamma))$ for any γ .

- (c) From the inequalities given in (a), we have

$$\begin{aligned} &F_{\gamma_2}(P(\gamma_2), X(\gamma_2), \beta(\gamma_2)) - F_{\gamma_1}(P(\gamma_2), \\ &X(\gamma_2), \beta(\gamma_2)) \leq F_{\gamma_2}(P(\gamma_1), X(\gamma_1), \beta(\gamma_1)) \\ &\quad - F_{\gamma_1}(P(\gamma_1), X(\gamma_1), \beta(\gamma_1)) \\ &\Rightarrow (\gamma_2^2 - \gamma_1^2) \sum_{i, j} \gamma_{ij}^2 (1 - \beta_{ij}(\gamma_2) \lambda_{ij}(\gamma_2))^2 \\ &\leq (\gamma_2^2 - \gamma_1^2) \sum_{i, j} \gamma_{ij}^2 (1 - \beta_{ij}(\gamma_1) \lambda_{ij}(\gamma_1))^2 \\ &\Rightarrow \sum_{i, j} \gamma_{ij}^2 (1 - \beta_{ij}(\gamma_2) \lambda_{ij}(\gamma_2))^2 \\ &\leq \sum_{i, j} \gamma_{ij}^2 (1 - \beta_{ij}(\gamma_1) \lambda_{ij}(\gamma_1)) \end{aligned}$$

which shows that $\sum_{i, j} \gamma_{ij}^2 (1 - \beta_{ij}(\gamma) \lambda_{ij}(\gamma))^2$ is a monotonic non-increasing function of γ .

- (d) From (b), we have

$$\begin{aligned} \lim_{\gamma \rightarrow \infty} F^*(\gamma) &\leq E^* \\ \Rightarrow \lim_{\gamma \rightarrow \infty} \sum_{i, j} \gamma^2 \gamma_{ij}^2 (1 - \beta_{ij}(\gamma) P_i^3(\gamma) X_j(\gamma))^2 &\leq E^*. \end{aligned} \tag{27}$$

Hence, provided $\gamma_{ij} \neq 0$, we must have

$$\lim_{\gamma \rightarrow \infty} (1 - \beta_{ij}(\gamma) P_i^3(\gamma) X_j(\gamma)) = 0$$

which completes the proof of Theorem 1. \square

Proof of Theorem 2:

- (a) Since $(P(\gamma), X(\gamma), \beta(\gamma))$ is a minimum solution for $F_{\gamma}(P, X, \beta)$, at $\beta = \beta(\gamma)$, we have

$$\frac{\partial F_{\gamma}(P, X, \beta)}{\partial \beta_{ij}} = -2 \left[(u_{ij} - \beta_{ij} P_i^1 X_j) P_i^1 X_j \right]$$

$$+(v_{ij} - \beta_{ij} P_i^2 X_j) P_i^2 X_j + \gamma^2 \gamma_{ij}^2 (1 - \beta_{ij} P_i^3 X_j) P_i^3 X_j \Big] = 0 \quad (28)$$

From (20),

$$\begin{aligned} \bar{\gamma}_{ij} * \left(x_{ij} - \frac{1}{\lambda_{ij}} P_i X_j \right) &= \bar{\gamma}_{ij} * (x_{ij} - \beta_{ij} P_i X_j) \\ &- \left(\frac{1}{\lambda_{ij}} - \beta_{ij} \right) (\bar{\gamma}_{ij} * P_i X_j) \end{aligned}$$

Hence,

$$\begin{aligned} \begin{bmatrix} u_{ij} - \frac{1}{\lambda_{ij}} P_i^1 X_j \\ v_{ij} - \frac{1}{\lambda_{ij}} P_i^2 X_j \\ 0 \end{bmatrix} &= \begin{bmatrix} u_{ij} - \beta_{ij} P_i^1 X_j \\ v_{ij} - \beta_{ij} P_i^2 X_j \\ \gamma \gamma_{ij} (1 - \beta_{ij} P_i^3 X_j) \end{bmatrix} \\ &- \left(\frac{1}{\lambda_{ij}} - \beta_{ij} \right) \begin{bmatrix} P_i^1 X_j \\ P_i^2 X_j \\ \gamma \gamma_{ij} P_i^3 X_j \end{bmatrix}. \quad (29) \end{aligned}$$

By (28), the two vectors on the right hand side of (29) are orthogonal when $(P, X, \beta) = (P(\gamma), X(\gamma), \beta(\gamma))$. Hence, taking the norm of both side of (29) and summing over (i, j) yields (22).

- (b) Substituting $\lambda_{ij} = P_i^3 X_j$ into (21), the truncation error can be written

$$T_\gamma(P, X, \beta) = \sum_{i,j} \left\| (1 - \beta_{ij} \lambda_{ij}) \begin{bmatrix} \hat{u}_{ij} \\ \hat{v}_{ij} \\ \gamma \gamma_{ij} \end{bmatrix} \right\|^2 \quad (30)$$

where

$$(\hat{u}_{ij}, \hat{v}_{ij}) = \frac{1}{\lambda_{ij}} (P_i^1 X_j, P_i^2 X_j).$$

By Theorem 1(b), $F^*(\gamma) \leq E^*$. Applying Theorem 1(d) and (27) to (30) shows that

$$\lim_{\gamma \rightarrow \infty} T_\gamma(P(\gamma), X(\gamma), \beta(\gamma)) \leq E^*.$$

Hence, by 22, $\lim_{\gamma \rightarrow \infty} E(P(\gamma), X(\gamma)) \leq 2E^*$. This implies that when $(P, X, \beta) = (P(\gamma), X(\gamma), \beta(\gamma))$ and for a sufficiently large γ , $(u_{ij} - \hat{u}_{ij}, v_{ij} - \hat{v}_{ij})$ and hence also $(\hat{u}_{ij}, \hat{v}_{ij})$ are bounded.

From (28), we have

$$\beta_{ij} = \frac{u_{ij} P_i^1 X_j + v_{ij} P_i^2 X_j + \gamma^2 \gamma_{ij}^2 P_i^3 X_j}{(P_i^1 X_j)^2 + (P_i^2 X_j)^2 + \gamma^2 \gamma_{ij}^2 (P_i^3 X_j)^2} \quad (31)$$

Substituting (31) into (30), the third component of $T_\gamma(P, X, \beta)$ can be written

$$\begin{aligned} &(1 - \beta_{ij} \lambda_{ij}) \gamma \gamma_{ij} \\ &= - \frac{(u_{ij} - \hat{u}_{ij}) \hat{u}_{ij} + (v_{ij} - \hat{v}_{ij}) \hat{v}_{ij}}{\hat{u}_{ij}^2 + \hat{v}_{ij}^2 + \gamma^2 \gamma_{ij}^2} \gamma \gamma_{ij} \end{aligned} \quad (32)$$

which implies

$$\lim_{\gamma \rightarrow \infty} (1 - \beta_{ij} \lambda_{ij}) \gamma \gamma_{ij} = 0. \quad (33)$$

Letting $\gamma \rightarrow \infty$ in (30) and making use of Theorem 1(d) and (33) yields

$$\lim_{\gamma \rightarrow \infty} T_\gamma(P(\gamma), X(\gamma), \beta(\gamma)) = 0. \quad (34)$$

- (c) Note that for any $\gamma > 0$, $E(P(\gamma), X(\gamma)) \geq E^*$. Making use of (34) and taking limit in (22) gives

$$\lim_{\gamma \rightarrow \infty} F^*(\gamma) = \lim_{\gamma \rightarrow \infty} E(P(\gamma), X(\gamma)) \geq E^*$$

which together with (19) imply that

$$\lim_{\gamma \rightarrow \infty} F^*(\gamma) = E^*.$$

This completes the proof of Theorem 2. \square

Acknowledgment

The work described in this paper was partially supported by a grant from the Research Grants Council of Hong Kong Special Administrative Region, China (Project No. HKU7058/02E) and partially by the CRCG of the University of Hong Kong. We would like to thank the reviewers for valuable comments.

References

- Bartoli, A. and Sturm, P. 2001. Three new algorithms for projective bundle adjustment with minimum parameters. Technical Report 4236, INRIA.
- Beardsley, P. Zisserman, A., and Murray, D. 1997. Sequential updating of projective and affine structure from motion. *Int. J. Computer Vision*, 23(3):235–259.
- Chen, G. and Medioni, G. 1999. Efficient iterative solutions to M-view projective reconstruction problem. In *Int. Conf. on Computer Vision & Pattern Recognition*, Vol. II, pp. 55–61.

- Chen, G. and Medioni, G. 2002. Practical algorithms for stratified structure-from-motion, 20:103–123.
- Faugeras, O.D. 1995. Stratification of 3-dimensional vision: Projective, affine, and metric representations. *J. of the Optical Society of America-A*, 12(3):465–484.
- Han, M. and Kanade, T. 2000. Scene reconstruction from multiple uncalibrated views. Technical Report CMU-RI-TR-00-09, Robotics Institute, Carnegie Mellon University.
- Han, M. and Kanade, T. 2001. Multiple motion scene reconstruction from uncalibrated views. In *IEEE Int. Conf. Computer Vision*, Vol. 1, pp. 163–170.
- Hartley, R.I. 1993. Euclidean reconstruction from uncalibrated views. In *Applications of Invariance in Computer Vision*, J. Mundy and A. Zisserman (Eds.), vol. LNCS 825, pp. 237–256.
- Heyden, A., Berthilsson, R., and Sparr, G. 1999. An iterative factorization method for projective structure and motion from image sequences. *Image and Vision Computing*, 1713:981–991.
- Jacobs, D.W. 2001. Linear fitting with missing data for structure from motion. *Computer Vision and Image Understanding*, 82:57–81.
- Mahamud, S. and Hebert, M. 2000. Iterative projective reconstruction from multiple views. In *Int. Conf. on Computer Vision & Pattern Recognition*, vol. 2, pp. 430–437.
- Mahamud, S., Hebert, M., Omori, Y., and Ponce, J. 2001. Provably-convergent iterative methods for projective structure from motion. In *Int. Conf. on Computer Vision & Pattern Recognition*, Kauai, Hawaii, pp. 1018–1025.
- Morris, D.D., Kanatani, K., and Kanade, T. 1999. Uncertainty modeling for optimal structure from motion. In *Vision Algorithms Theory and Practice*, Springer LNCS.
- Oliensis, J. 1996. Fast and accurate self-calibration. In *IEEE Int. Conf. Computer Vision*, pp. 745–752.
- Pollefeys, M. and Gool, L.V. 1999. Stratified self-calibration with the modulus constraint. *IEEE Trans. Pattern Analysis & Machine Intelligence*, 21(8):707–724.
- Powell, M.J.D. 1970. A hybrid method for non-linear equations. In *Numerical Methods for Non-Linear Algebraic Equations*, P. Rabinowitz (Ed.), 87ff.
- Shum, H.Y., Ikeuchi, K., and Reddy, R. 1995. Principal component analysis with missing data and its application to polyhedral object modeling. *IEEE Trans. Pattern Analysis & Machine Intelligence*, 17(9):854–867.
- Shum, H.Y., Ke, Q., and Zhang, Z. 1999. Efficient bundle adjustment with virtual key frames: A hierarchical approach to multi-frame structure from motion. In *Int. Conf. on Computer Vision & Pattern Recognition*.
- Sparr, G. 1996. Simultaneous reconstruction of scene structure and camera locations from uncalibrated image sequences. In *Int. Conf. Pattern Recognition*.
- Sturm, P. and Triggs, B. 1996. A factorization based algorithm for multi-image projective structure and motion. In *European Conf. on Computer Vision*. Cambridge, England, pp. 709–720.
- Tomasi, C. and Kanade, T. 1992. Shape and motion from image streams under orthography: A factorization method. *Int. J. Computer Vision*, 9(2):137–154.
- Triggs, B. 1996. Factorization methods for projective structure and motion. In *Int. Conf. on Computer Vision & Pattern Recognition*, San Francisco, pp. 845–851.
- Triggs, B. 1998. Some notes on factorization methods for projective structure and Motion, unpublished.
- Triggs, B., McLauchlan, P., Hartley, R., and Fitzgibbon, A. 2000. Bundle adjustment—A modern synthesis. In *Vision Algorithms: Theory and Practice*, W. Triggs, A. Zisserman, and R. Szeliski (Eds.), vol. LNCS 1883, Springer Verlag, pp. 298–375.