3D STRUCTURE FROM VIDEO STREAMS WITH PARTIALLY OVERLAPPING IMAGES

Rui F. C. Guerreiro *Pedro M. Q. Aguiar*

Institute for Systems and Robotics, IST, Lisboa, Portugal
{rfcg,aguiar}@isr.ist.utl.pt

ABSTRACT

The majority of methods available to recover 3D structure from video assume that a set of feature points are tracked across a large number of frames. This is not always possible in real videos because the images overlap only partially, due to the occlusion and the limited field of view. This paper describes a new method to recover 3D structure from videos with partially overlapping views.

The well known factorization method [1] recovers 3D rigid structure by factoring an observation matrix that collects trajectories of feature points. We extend this method to the more challenging scenario of observing incomplete trajectories. This way, we accommodate not only the features that disappear, but also features that, although not visible in the first image, become available later. Under this scenario, the observation matrix has missing entries. We develop three new algorithms to factor out matrices with missing data. Experiments with synthetic data and real video images demonstrate the viability of our approach to recover 3D structure.

1. INTRODUCTION

The problem of recovering 3D structure (3D shape and 3D motion) from video finds a wide range of applications in fields such as robotics, digital video, and virtual reality. Since the strongest cue to infer 3D structure from a sequence of images is the 2D motion of the brightness pattern, the recovery 3D structure is commonly referred as *structure from motion* (SFM).

Obviously, the accuracy of the 3D reconstructions of stationary scenes improves with the number of video frames available. In turn, the SFM problem becomes more difficult due to the larger number of unknowns (the larger set of camera positions). An elegant well known approach to the multi-frame SFM problem is the so called *factorization method* [1]. In [1], the authors collect the trajectories of the projections of a set of tracked feature points into an observation matrix. They show that the observation matrix is highly rank deficient due to the 3D rigidity of the scene. The 3D rigid shape of the scene and the 3D motion of the camera are then computed from the best rank deficient approximation to the noisy observation matrix. The factorization method was later extended to more general geometric projection models [2] and to more general parametric descriptions of the 3D shape [3].

Any of the works cited above assume that the projections of the features to be processed are available in all frames, i.e., that they are seen during the entire video sequence. In real world applications, this assumptions limits severely the feature candidates because very often important regions that are seen in some frames are not seen in others due to the scene self-occlusion and the limited field of view. Under this scenario, the observation matrix contains incomplete information, i.e., it misses the entries corresponding to the unobserved projections. Very few attempts have been done to extend the factorization methods to this more general scenario. In fact, while the rank deficient matrix that best approximates a completely known matrix is easily obtained from its *Singular Value Decomposition* (SVD), there is no equivalent for matrices with missing elements. Reference [1] suggests a computationally heavy procedure, requiring O(max(#frames,#features)) computations of the SVD [4], to "fill in", in a sequential way, the missing values of the observation matrix. Reference [4] proposes a sub-optimal method to combine the constrains that arise from the observed submatrices of the original matrix. A bidirectional optimization scheme was proposed in [5].

In this paper, we propose three new algorithms to find the best rank deficient approximation of a matrix with missing elements. The first algorithm can be seen as an initialization – it computes in an efficient way an initial estimate of the complete matrix. The other two algorithms are iterative schemes that converge to the optimal solution when properly initialized. The first iterative algorithm is based in a well known method to deal with missing data – the *Expectation-Maximization* (EM) [6]. The second iterative scheme computes, alternately, in closed form, two matrices whose product is the solution matrix. We call this the *Two-Step* (TS) iterative method. Our three algorithms are general in the sense that they compute approximations of any rank. Thus, although developed under the framework of SFM, our algorithms are adequate to other signal/image processing tasks that require finding rank deficient approximations of matrices with missing elements.

Besides their generality, the algorithms presented in this paper have other advantages over the above mentioned ones. Our initialization algorithm is a contribution by itself since it requires a much smaller number of SVD computations than the "filling in" method of [1]. Although the authors of [5] don't mention EM, their bidirectional scheme is in fact an EM-like method. Our EM algorithm relates thus to [5] but, as detailed below, our E-step is simpler. Our TS algorithm proves to work as efficiently as the EM with the advantage of being computationally cheaper. Finally, the combination of a first estimate with iterative algorithms is also an advantage of our method – while the good behavior of the iterative algorithms makes unnecessary to find a very accurate starting matrix (a computationally expansive task), the existence of an initial estimate assures they converge in a few iterations.

MatLab© implementations of the algorithms we propose in this paper are available at `www.isr.ist.utl.pt/~aguiar`.

Paper organization In section 2, we formulate the problem of approximating a matrix with missing data by a rank deficient matrix. Sections 3, 4, and 5 describe, respectively, the initialization, the EM, and the TS algorithms. In section 6 we describe experiments with real and artificial data. Section 7 concludes the paper.

Work partially supported by FCT project POSI/SRI/41561/2001.

2. FACTORIZATION WITH MISSING DATA

In the original factorization method [1], the authors track P feature points over F frames and collect their trajectories in the $2F \times P$ observation matrix $\boldsymbol{W}$. The observation matrix is written in terms of the parameters that describe the 3D structure as

$$\boldsymbol{W} = \boldsymbol{R}\boldsymbol{S}^T + \boldsymbol{t}\boldsymbol{1} = \left[\begin{array}{c|c} \boldsymbol{R} & \boldsymbol{t} \end{array}\right] \left[\begin{array}{c} \boldsymbol{S}^T \\ \hline \boldsymbol{1} \end{array}\right], \tag{1}$$

where $\boldsymbol{R}$ and $\boldsymbol{t}$ represent the rotational and translational components of the 3D motion of the camera and $\boldsymbol{S}$ represents the 3D shape of the scene. $\boldsymbol{R}$ is $2F \times 3$. It collects entries of the $2F$ 3D rotation matrices that code the camera orientations. The $2F \times 1$ vector $\boldsymbol{t}$ collects the $2F$ camera positions. $\boldsymbol{S}$ is $P \times 3$. It contains the 3D coordinates of the P feature points. See [1] for the details. The problem of recovering SFM is then: given $\boldsymbol{W}$, compute $\boldsymbol{R}$, $\boldsymbol{S}$, and $\boldsymbol{t}$. Since in a noiseless situation the observation matrix $\boldsymbol{W}$ in (1) is rank 4, the method in [1] recovers SFM by using the SVD to compute the best rank 4 approximation to the matrix $\boldsymbol{W}$.

Suppose now that we have an image stream in which some feature points disappear due to the occlusion or the limited field of view and/or new feature points become available along the video sequence. This means that there are frames in which the coordinates of some feature points are unknown, leading to an observation matrix $\boldsymbol{W}$ with missing data. To extend the factorization method to this scenario, we must compute the best rank 4 approximation $\widetilde{\boldsymbol{W}}$ to the observation matrix $\boldsymbol{W}$ that is now partially unknown. If the noise is white and Gaussian, the *Maximum Likelihood* (ML) solution leads to the minimization problem

$$\min_{\widetilde{\boldsymbol{W}} \in \mathcal{S}_4} \left\| \left(\boldsymbol{W} - \widetilde{\boldsymbol{W}}\right) \odot \boldsymbol{M} \right\|_F, \tag{2}$$

where $\mathcal{S}_4$ denotes the space of the $2F \times P$ rank 4 matrices; $\odot$ represents the elementwise product, also known as the Hadamard product; the matrix $\boldsymbol{M}$ is a binary mask that accounts for the known entries of the observation matrix $\boldsymbol{W}$, i.e., $m_{ij} = 1$ if w_{ij} is known and $m_{ij} = 0$ otherwise; and $\|.\|_F$ represents the Frobenius norm. After solving (2) for $\widetilde{\boldsymbol{W}}$, the recovery of SFM is straightforward by using the factorization method of [1].

When the matrix $\boldsymbol{M}$ contains only ones, i.e., when the observation matrix $\boldsymbol{W}$ is completely known, the solution $\widetilde{\boldsymbol{W}}$ of (2) is obtained from the SVD of $\boldsymbol{W}$ after selecting the 4 larger singular values as in the factorization method of [1]. We denote this optimal rank reduction operation by $\boldsymbol{W} \downarrow \mathcal{S}_4$:

$$\widetilde{\boldsymbol{W}} = \boldsymbol{W} \downarrow \mathcal{S}_4 = \boldsymbol{U}_{2F\times 4}\boldsymbol{\Sigma}_{4\times 4}\boldsymbol{V}_{4\times P}. \tag{3}$$

In opposition, the existence of unknown entries in $\boldsymbol{W}$ prevents us to minimize (2) by using the SVD of $\boldsymbol{W}$ as in (3). This makes nontrivial the recovery of SFM in our scenario. The following sections deal with the nonlinear minimization of expression (2). Note that, although particularized for rank 4 matrices, our algorithms are valid for any other order rank deficient approximations.

3. INITIAL ESTIMATE

Any rank 4 matrix $\widetilde{\boldsymbol{W}}$ can be written as the matrix product

$$\widetilde{\boldsymbol{W}} = \boldsymbol{A}_{2F\times 4}\boldsymbol{B}_{4\times P} \in \mathcal{S}_4, \tag{4}$$

where $\boldsymbol{A}$ determines the column space of $\widetilde{\boldsymbol{W}}$ and $\boldsymbol{B}$ its row space. We find an initial estimate of $\widetilde{\boldsymbol{W}}$ by computing in an expedite way estimates of the matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ from the data in $\boldsymbol{W}$.

Suboptimal subspace estimation Before addressing the general case, we consider the simpler case where a number of columns of $\boldsymbol{W}$ are entirely known, i.e., a number of feature points are present in all frames, and a number of rows of $\boldsymbol{W}$ are entirely known, i.e., and a number of frames contain all feature projections. We collect those known columns in a submatrix $\boldsymbol{W}_c$ and those known rows in a submatrix $\boldsymbol{W}_r$. From the data in $\boldsymbol{W}_c$, the ML estimate of the column space matrix $\boldsymbol{A}$ is

$$\boldsymbol{A} = \boldsymbol{W}_c \downarrow \mathcal{S}_4. \tag{5}$$

From the data in $\boldsymbol{W}_r$ and the column space matrix $\boldsymbol{A}$, the ML estimate of the row space matrix $\boldsymbol{B}$ is the known *Least Squares* (LS) solution that uses to the Moore-Penrose pseudoinverse, see [7],

$$\boldsymbol{B} = \left(\boldsymbol{A}_r^T\boldsymbol{A}_r\right)^{-1}\boldsymbol{A}_r^T\boldsymbol{W}_r, \tag{6}$$

where $\boldsymbol{A}_r$ collects the rows of $\boldsymbol{A}$ that correspond to the rows of $\boldsymbol{W}_r$. We see that the matrix $\boldsymbol{A}_r^T\boldsymbol{A}_r$ must be nonsingular so the matrix $\boldsymbol{W}_c$ must have at least 4 linearly independent columns and the matrix $\boldsymbol{W}_r$ must have at least 4 linearly independent rows.

Subspace combination In the general case, however, it is not possible to find 4 entire columns and 4 entire rows without missing elements in the matrix $\boldsymbol{W}$. We must then estimate the column and row spaces matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ by combining the spaces that correspond to smaller submatrices of $\boldsymbol{W}$. We describe the algorithm for combining two column/row space matrices. The process is then repeated until the entire matrix $\boldsymbol{W}$ has been processed.

Select from $\boldsymbol{W}$ two submatrices $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ that have at least 4 columns and 4 rows without missing elements. We factorize $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ using (5) to obtain the corresponding column space matrices $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$. If the observation matrix $\boldsymbol{W}$ is in fact well modelled by a rank 4 matrix, the submatrices of $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ that correspond to common rows in $\boldsymbol{W}$, denoted by $\boldsymbol{A}_{12}$ and $\boldsymbol{A}_{21}$, are related by a linear transformation,

$$\boldsymbol{A}_{12} \simeq \boldsymbol{A}_{21}\boldsymbol{N}, \tag{7}$$

where $\boldsymbol{N}$ is a 4×4. We compute $\boldsymbol{N}$ from (7) as the LS estimate

$$\boldsymbol{N} = \left(\boldsymbol{A}_{21}^T\boldsymbol{A}_{21}\right)^{-1}\boldsymbol{A}_{21}^T\boldsymbol{A}_{12} \tag{8}$$

and assemble a combined column space matrix $\boldsymbol{A}$ for the rows corresponding to $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ as

$$\boldsymbol{A} = \left[\begin{array}{c} \boldsymbol{A}_1 \\ \hline \boldsymbol{A}_{2\backslash 1}\boldsymbol{N} \end{array}\right], \tag{9}$$

where $\boldsymbol{A}_{2\backslash 1}$ denotes the submatrix of $\boldsymbol{A}_2$ that collects the rows that do not correspond to rows of $\boldsymbol{W}_1$. We compute the combined row subspace matrix $\boldsymbol{B}$ by using an analogous procedure with (6). We define the initial estimate of the matrix $\widetilde{\boldsymbol{W}}$ as $\widetilde{\boldsymbol{W}}^{(0)} = \boldsymbol{A}\boldsymbol{B}$.

4. EXPECTATION-MAXIMIZATION ALGORITHM

The EM approach to estimation problems with missing observations works by enlarging the set of parameters to estimate – the data that is missing is jointly estimated with the other parameters.

The joint estimation is performed iteratively in two alternate steps: i) the E-step estimates the missing data given the previous estimate of the other parameters; ii) the M-step estimates the other parameters given the previous estimate of the missing data, see [6].

In our case, given the initial estimate $\widetilde{W}^{(0)}$, computed as described in section 3, the EM algorithm estimates in alternate steps: i) the missing entries of the observation matrix W; ii) the rank 4 matrix $\widetilde{W}$ that best matches the data. The algorithm performs these two steps until convergence, i.e., until the error measured by the Frobenius norm (2) stabilizes.

E-step – estimation of the missing data Given $\widetilde{W}^{(k-1)}$, the ML estimates of the missing entries $\{w_{ij} : m_{ij} = 0\}$ of W are simply the corresponding entries $\widetilde{w}_{ij}^{(k-1)}$ of $\widetilde{W}^{(k-1)}$. We then build a complete observation matrix $\widehat{W}^{(k)}$, whose entry $\widehat{w}_{ij}^{(k)}$ equals the corresponding entry w_{ij} of the observation matrix W if w_{ij} was observed or its estimate $\widetilde{w}_{ij}^{(k-1)}$ if w_{ij} is unknown,

$$\widehat{w}_{ij}^{(k)} = \begin{cases} w_{ij} & \text{if } m_{ij} = 1, \\ \widetilde{w}_{ij}^{(k-1)} & \text{if } m_{ij} = 0, \end{cases} \tag{10}$$

or, in matrix notation,

$$\widehat{W}^{(k)} = W \odot M + \widetilde{W}^{(k-1)} \odot [1 - M]. \tag{11}$$

M-step – estimation of the rank deficient matrix We are now given the complete observation matrix $\widehat{W}^{(k)}$ with the estimates of the missing data from the E-step. The ML estimate of the rank 4 matrix $\widetilde{W}^{(k)}$, i.e., the rank 4 matrix that best matches $\widehat{W}^{(k)}$ in the Frobenius norm, is then obtained from the SVD of $\widehat{W}^{k}$, see (3),

$$\widetilde{W}^{(k)} = \widehat{W}^{(k)} \downarrow \mathcal{S}_4. \tag{12}$$

In reference [5], the authors treat 3D translation separately, rather than including it in the factorization process as we do, remember (1). Their bidirectional algorithm is then developed to that specific strategy. In opposition, our EM algorithm is general, i.e, it solves any rank defficient matrix approximation problem with missing data. Furthermore, our E-step in (11) is simpler than the corresponding step of [5] that requires inverting matrices.

5. TWO-STEP ALGORITHM

This section describes the TS algorithm, a new iterative scheme to compute the rank 4 matrix that best matches the data. From our experience, the TS algorithm is computationally cheaper then EM – avoids SVD computations and exhibits slightly faster convergence. For the TS algorithm, we parameterize the rank 4 matrix $\widetilde{W}$ as in (4), and compute the column space matrix A and the row space matrix B directly from the minimization of the global cost (2),

$$\min_{A,B} \|(W - AB) \odot M\|_F. \tag{13}$$

Through this parameterization, we map the constrained minimization (2) wrt $\widetilde{W} \in \mathcal{S}_4$ into the unconstrained minimization (13) wrt A and B.

We minimize (13) in two alternate steps. In step i), we assume the column space matrix A is known and estimate the row space matrix B. In step ii), we estimate B for known A. The algorithm is initialized by recovering A from the initial estimate $\widetilde{W}^{(0)}$ computed in section 3, $A = \widetilde{W}^{(0)} \downarrow \mathcal{S}_4$, and it runs until the value of the Frobenius norm in (13) stabilizes. When there is no missing data, our TS algorithm implements the *power method* [7] that is widely used to find the best rank deficient approximation without computing the SVD. We will see that, even when there is missing data, both steps i) and ii) admit closed-form solution and the overall algorithm results very simple.

Step i) estimate of B for known A If A is known, minimizing (13) wrt each entry of B leads to a LS problem. We compute the LS solution of each column b_p of B as

$$b_p = \left[A^T (A \odot M_p)\right]^{-1} A^T (w_p \odot m_p), \tag{14}$$

where the lowercase boldface letters denote columns of the matrices with the same uppercase letters and M_p is a $2F \times 4$ matrix with all 4 columns equal to m_p, $M_p = m_p 1_{1\times 4}$. The vector m_p in (14) selects from W the known data and from A the corresponding relevant entries. Note that the set of $p = 1 \ldots P$ equalities like (14) is the generalization of the known LS solution based on the pseudoinverse (6) for the possibility of having missing data.

Step ii) estimate of A for known B Given B, we estimate each row a_f of A, $f = 1 \ldots 2F$, in a similar way,

$$a_f = (w_f \odot m_f) B^T \left[(B \odot M_f) B^T\right]^{-1}, \tag{15}$$

where now, for commodity, the lowercase boldface letters denote rows rather than columns, and $M_f = 1_{4\times 1} m_f$.

6. EXPERIMENTS

We describe three experiments. First, we demonstrate the efficiency of our method to compute general rank deficient approximations of matrices with missing data. The second experiment recovers 3D shape and 3D motion from synthetic 2D trajectories from which data was removed. Finally, we recover 3D structure from a real-life video clip with partially overlapping images.

Rank deficient approximation We have generated rank deficient matrices, added noise, and removed a subset of their entries. Then we used our methods to recover the original matrix. We used matrices of dimensions ranging from 2 to 50 and ranks from 1 to 6. In all experiments both our methods estimate the original matrix with a mean square error smaller than the variance of the observation noise. We illustrate the behavior of the algorithms when recovering a 40×40 rank 6 matrix W_6 from a noisy observation W (noise variance $\sigma^2 = 1$) with 30×30 missing entries. Figure 1 shows plots of the evolution of the estimation error, measured by the Frobenius norm (2) for both the EM and TS algorithms of sections 4 and 5. While for the left hand side plot we used a random initialization, for the one in the right we used the complete procedure, i.e., we initialized the process by using the method of section 3. Since the error for the *true* matrix W_6 is given by $\|(W_6 - W) \odot M\|_F = \sigma\sqrt{40^2 - 30^2} \simeq 26.5$, we see from the plots of figure 1 that both the EM and TS algorithms provide good estimates. From the right hand side plot, we conclude that the initial estimate of section 3 enables a faster convergence (in 2 or 3 iterations) to a solution with lower error.

3D Structure from 2D motion with missing data We synthesized noisy versions of the 2D trajectories of 372 feature points located on the 3D surface of a rotating cylinder. Then, we simulated occlusion and inclusion by removing significant segments of those

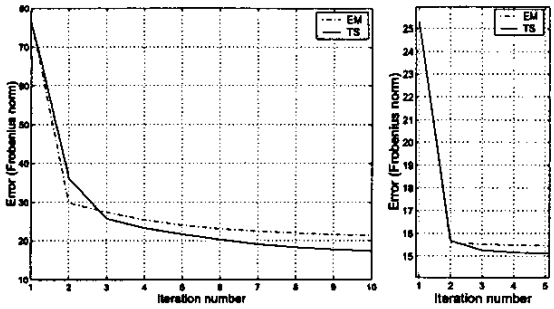

Fig. 1. Evolution of the estimation error (2) for a 40×40 rank 6 matrix with 30×30 missing entries, with a random initialization (left plot) and with the initialization of section 3 (right plot).

trajectories. Figure 2 show one of the 50 synthesized frames. The small circles denote the noiseless projections and the points denote their noisy version, i.e., the data that is observed. Note that only an incomplete view portion of the cylinder is observed in each frame.

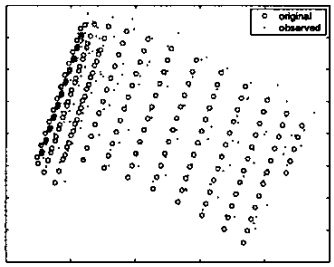

Fig. 2. One frame of the cylinder sequence.

The data from the cylinder sequence was then collected on a 100×372 observation matrix $\boldsymbol{W}$ with 9537 unknown entries ($\approx$ 26% of the total number). We applied our method with the TS algorithm to the matrix $\boldsymbol{W}$ and recovered 3D SFM by using the factorization method [1]. Figure 3 plots the final estimate of the 3D shape. We see that the complete cylinder is recovered. Due to the incorporation of the rigidity constraint, the 3D positions of the features points are accurately estimate even in the presence of very noisy observations (compare Figures 2 and 3).

Real video We used a real-life video clip available at the computer vision site `www-2.cs.cmu.edu/~cil/vision.html`. This clip show a rotating ping-pong ball with painted dots. The left image of figure 4 shows the first of the 52 video frames of the ball sequence. We tracked a set of 64 feature points. Due to the camera-ball 3D rotation, the region of the ball that is visible changes across time, leading to an observation matrix with $\simeq$ 41% missing entries. We proceeded as described for the previous experiment and recovered the 3D shape shown on the right image of figure 4. We see that our method succeeded in recovering the spherical surface of the ball.

7. CONCLUSION

We presented a new general, efficient, and computationally simple method to find the best rank deficient approximation of a matrix with missing data. Our method combines an expedite initialization with two iterative algorithms that converge in few iterations. Our experiments demonstrate that this approach is well suited to the recovery of 3D rigid structure from videos that, due to the occlusion and inclusion effects, exhibit partially overlapping views.

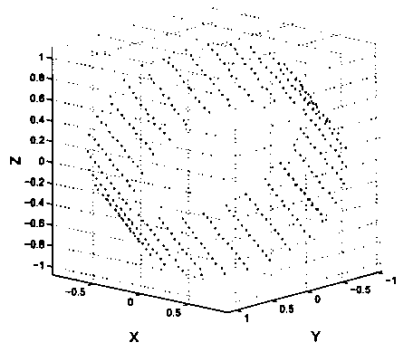

Fig. 3. Cylinder recovered from the frames as in figure 2.

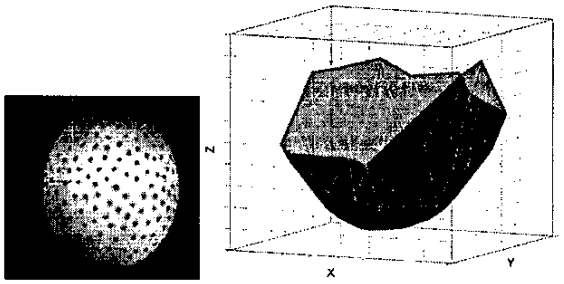

Fig. 4. Left: first frame of the ball video clip. Right: 3D shape recovered from the ball video clip.

8. REFERENCES

[1] C. Tomasi and T. Kanade, "Shape and motion from image streams under orthography: a factorization method," *International Journal of Computer Vision*, vol. 9, no. 2, 1992.

[2] C. J. Poelman and T. Kanade, "A paraperspective factorization method for shape and motion recovery," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 3, 1997.

[3] P. M. Q. Aguiar and J. M. F. Moura, "Three-dimensional modeling from two-dimensional video," *IEEE Transactions on Image Processing*, vol. 10, no. 10, 2001.

[4] D. Jacobs, "Linear fitting with missing data: Applications to structure-from-motion and to characterizing intensity images," in *IEEE Computer Vision Pattern Recognoition*, 1997.

[5] M. Maruyama and S. Kurumi, "Bidirectional optimization for reconstructing 3D shape from an image sequence with missing data," in *IEEE ICIP*, Kobe, Japan, 1999.

[6] T. K. Moon and W. C. Stirling, *Mathematical Methods and Algoritms for Signal Processing*, Prentice Hall, 1999.

[7] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, 1996.