Linear Embeddings in Non-Rigid Structure from Motion

Vincent Rabaud Serge Belongie
Department of Computer Science and Engineering
University of California, San Diego
{ vrabaud, sjb }@cs.ucsd.edu

Abstract

This paper proposes a method to recover the embedding of the possible shapes assumed by a deforming non-rigid object by comparing triplets of frames from an orthographic video sequence. We assume that we are given features tracked with no occlusions and no outliers but possible noise, an orthographic camera and that any 3D shape of a deforming object is a linear combination of several canonical shapes. By exploiting any repetition in the object motion and defining an ordering between triplets of frames in a Generalized Non-Metric Multi-Dimensional Scaling framework, our approach recovers the shape coefficients of the linear combination, independently from other structure and motion parameters. From this point, a good estimate of the remaining unknowns is obtained for a final optimization to perform full non-rigid structure from motion. Results are presented on synthetic and real image sequences and our method is found to perform better than current state of the art.

1. Introduction

The problem of reconstructing the 3D structure of a rigid object from a monocular video sequence has seen significant progress in recent literature. In the case where the object is non-rigid, however, a few methods perform well but the problem is still wide open.

It has been shown that for many practical objects (*e.g.* faces), it is valid to assume that the deforming object adopts 3D shapes defined by a linear combination of basis shapes [4]. In this case, the non-rigid structure from motion (NRSFM) problem is an optimization over the camera parameters, the shape basis and the shape coefficients. In prior work (*e.g.* [21]), this optimization is usually initialized by assuming that the object has a dominant rigid component, hence treating the initialization problem as an instance of rigid structure from motion. The different parameters are then refined by alternating minimization or via an EM procedure. Such approaches are known to suffer from two principal drawbacks. First, the alternating optimization framework can exhibit instability and second, the underlying assumption is highly restrictive, as an object might not even exhibit a dominant rigid component at all. In this paper, we propose an approach to NRSFM that overcomes these drawbacks.

To motivate the discussion, let us perform a thought experiment. Suppose we put a cat in a room and let it move and "deform". If we observe it long enough, we should capture all its possible 3D shapes from every possible viewpoint. The first contribution of this paper is based on this intuition: we propose to exploit any repetitions in shape deformation within a video sequence to compare triplets of frames; by combining these orderings in a Generalized Non-metric Multi-Dimensional Scaling framework [1], our method recovers an embedding of the possible shapes that can be assumed by a non-rigid object. By focusing on this subspace only, our method appears complementary to more general manifold-based techniques [3, 20, 16] that recover the whole 3D shapes and the camera parameters at the same time. The other main contribution of this paper is a new NRSFM algorithm that uses this shape embedding as initialization.

After reviewing the related work in Section 2 and formulating the problem in Section 3, we demonstrate how the shape embedding can be recovered in Section 4 by using comparisons of triplets of frames, as described in Section 5. Finally, Section 6 explains how the final reconstruction can be obtained and Section 7 presents our experimental results.

2. Previous Work

References are here restricted to the general case where there is no physical or learned prior on the object.

Modern structure from motion started with the study of a rigid object under orthographic camera [18]. It was then extended to the projective case [17], multiple bodies [7] and articulated bodies [26].

Solving for the structure of a non-rigid object had its first breakthrough with [4, 21] where it is assumed that an object's 3D shape can be explained by a linear combination

of elements of a shape basis. This helps in formulating the problem as a factorization problem of the measurement matrix which can be solved by first assuming that the object is globally rigid [10] or has a dominant rigid component. This provides an initialization to gradient descent or an EM-type algorithm [19]. These techniques usually have two problems: they require a good initialization (which might not be provided if the object does not comply to their assumptions) and their optimization usually does not consider all the parameters at once. Nonetheless, they have been improved by combining a feature tracker [21, 8], including noise models [20] and approching the problem in a coarse-to-fine manner [2]. Finally, while [23] proved some theoretical limitations, the same formulation was used to extend the analysis to the projective case [25, 14, 22].

In our method, we propose to exploit motion repetitions for SFM as it was used for action recognition [12], motion segmentation [11] and sequence alignment [5]. Our method also shares similarities with [3, 16] where a criterion was optimized on a locally linear manifold, but we here focus on a globally linear manifold, differently from probabilistic PCA [20].

3. Problem Formulation

In our setup, n features are tracked over f frames with no occluions under an orthographic camera: their 2D projected positions are known and can be stacked in f $2 \times n$ matrices W_t (t indexes time). The problem consists of recovering the 3D positions of these features (stacked in a $3 \times n$ shape matrix S_t at each frame) as well as the camera parameters: rotation R_t^* and translation $\mathbf{t}_t^*$. We define R_t and $\mathbf{t}_t$ as the respective top two rows of R_t^* and $\mathbf{t}_t^*$.

In the shape basis assumption, every shape S_t can be expressed as a linear combination of a mean shape $\mathtt{S}^0$ and s deformation modes $\mathtt{S}^i$ ($3 \times n$ matrices). s is a given parameter, works like [2] can estimate it. The measurements can then be explained as follows:

$$W_t = R_t \left(\mathtt{S}^0 + \sum_{i=1}^{s} l_t^i \mathtt{S}^i \right) + \mathbf{t}_t \mathbf{1}^\top \tag{1}$$

where $\mathbf{1}$ is a $n \times 1$ vector of ones. We choose the notation: $\mathbf{l}_t = \begin{bmatrix} l_t^1 & \dots & l_t^s \end{bmatrix}^\top$.

As in [18], we can eliminate the $\mathbf{t}_t$ by subtracting the mean of the measurements. From now on, we will consider the centered measurements $\overline{W}_i$. By imposing the $\mathtt{S}^i$ to also be centered, the measurement matrix can be factorized as follows:

$$W = \begin{bmatrix} \overline{W}_1 \\ \vdots \\ \overline{W}_f \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} 1 & \mathbf{l}_1^\top \end{bmatrix} \otimes R_1 \\ \vdots \\ \begin{bmatrix} 1 & \mathbf{l}_f^\top \end{bmatrix} \otimes R_f \end{bmatrix} \begin{bmatrix} \mathtt{S}^0 \\ \vdots \\ \mathtt{S}^s \end{bmatrix} = MS \tag{2}$$

where $\otimes$ is the Kronecker product.

We propose to recover the $\mathbf{l}_t$'s first, independently from other coefficients as explained in Section 4 and 5. This will provide us with a good initialization for our final optimization in Section 6.

Algorithm 1 Comparison-based SFM

1: {the a_2 functions are known polynomials in $\mathbf{l}_t$'s}
2: **for** every pair of frames (i, j) **do**
3: compute $a_{\min}(i, j)$ such that $a_{\min}(i, j) \leq a_2(i, j)$. (Section 5.1)
4: **end for**
5: compute pairs $((i, j, k), (i', j', k'))$ of frame triplets such that $a_2(i, j, k) \leq a_2(i', j', k')$ (Section 5)
6: Use those pairwise and triplet-wise inequalities in a GNMDS framework to estimate the $\mathbf{l}_t$'s (Section 4)
7: Use those $\mathbf{l}_t$ estimates to have an approximation of the $\mathtt{S}^i$'s and R_t's (Section 6.1 and Section 6.2)
8: Optimize the reprojection error in a bundle adjustment manner (Section 6.3)

4. Outline

The proposed approach assumes that a non-rigid object deforms in 3D shapes that can be observed several times in a video sequence. Intuitively, if the 3D reconstruction from a set of frames has a low reconstruction error, the frames should probably represent a similar 3D shape (of course, outliers could be present as depth is removed during camera projection, and our method should account for them). Similarly a high reconstruction error would be due to a poor matching of the different views.

As two views have ambiguous rigid 3D reconstructions (due to Necker reversal and bas-relief ambiguity), we decide to focus on triplets of frames (which only lead to a sign ambiguity). Also, due to the impossibility of relating this reconstruction error to a metric, we propose to only get orderings between triplets of frames and next use those in a Multi-Dimensional Scaling (MDS) framework. The approach is detailed in Algorithm 1.

4.1. Ordering Set $\mathcal{F}$

In traditional linear NRSFM methods, the basis elements $\mathtt{S}^i$ are only assumed to be centered at the origin. The Gram-Schmidt process can orthonormalize any basis but it also preserves the centering. Therefore, if a basis exists, a centered orthonormal basis can be built from it: the $\mathtt{S}^i$'s can consequently be assumed to be centered and orthonormal. But, as we use 1 for the first coordinate, we cannot impose $\left\|\mathtt{S}^0\right\|_F = 1$. Therefore, we can only impose the $\mathtt{S}^i$'s to be orthogonal to each other, and $\left\|\mathtt{S}^i\right\|_F = 1, \forall i \neq 0$

Now, in this basis, every shape S_t has coordinates $[1, \mathbf{l}_t^\top]$ and therefore:

$$\|S_i - S_j\|_F = \|\mathbf{l}_i - \mathbf{l}_j\|_2 \tag{3}$$

Let us consider the function:

$$a_F(i,j,k) = \sum_{h \in \{i,j,k\}} \left\| S_h - \frac{S_i + S_j + S_k}{3} \right\|_F^2 \tag{4}$$

Let us assume for now that we can build a set of pairs of triplets of frames as follow:

$$\mathcal{F} = \{((i,j,k),(i',j',k')) \,|a_F(i,j,k) \leq a_F(i',j',k')\} \tag{5}$$

Section 5 will focus on the construction of such a set. It will also demonstrate that not all pairs $\{(i,j,k),(i',j',k')\}$ can be compared with a_F. Therefore, only certain triplets of frames have to be considered; this makes Generalized Non-metric Multi-Dimensional Scaling (GNMDS) from [1] the appropriate choice to solve for the $\mathbf{l}_t$'s.

4.2. Generalized Non-metric Multi-Dimensional Scaling Overview

Let us consider the positive semi-definite Gram matrix $K = [K_{ij}]_{1 \leq i,j \leq f} = [\mathbf{l}_i^\top \mathbf{l}_j]_{1 \leq i,j \leq f}$ and let us define:

$$a_2(i,j,k) = \sum_{h \in \{i,j,k\}} \left\| \mathbf{l}_h - \frac{\mathbf{l}_i + \mathbf{l}_j + \mathbf{l}_k}{3} \right\|_2^2 \tag{6}$$

By using Equation (3), $a_F(i,j,k) = a_2(i,j,k)$.

Solving for the $\mathbf{l}_t$'s can therefore be reduced to finding a positive semi-definite matrix K such that:

$$a_2(i,j,k) \leq a_2(i',j',k') \text{ if } ((i,j,k),(i',j',k')) \in \mathcal{F} \tag{7}$$

By introducing slack variables $\xi_{i,j,k,i',j',k'}$, solving for the $\mathbf{l}_t$'s is equivalent to solving the following Semi-Definite Programming (SDP) problem:

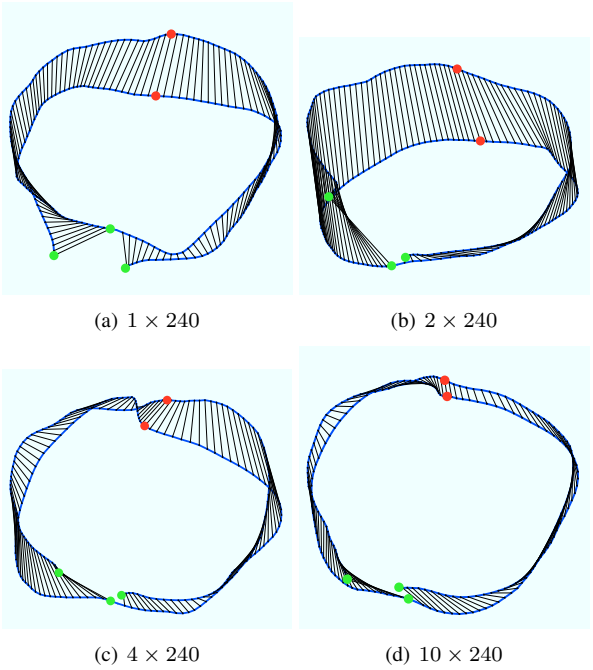
$$\begin{aligned} \min_{K, \xi_{i,j,k,i',j',k'}} \quad & \sum_{((i,j,k),(i',j',k')) \in \mathcal{F}} \xi_{i,j,k,i',j',k'} \\ \text{subject to} \quad & a_2(i,j,k) \leq a_2(i',j',k') + \xi_{i,j,k,i',j',k'} \\ & \xi_{i,j,k,i',j',k'} \geq 0, \quad K \succeq 0 \end{aligned} \tag{8}$$

Also, the constraints are linear in the elements of K as:

$$a_2(i,j,k) = \frac{2}{3}(K_{ii} + K_{jj} + K_{kk} - K_{ij} - K_{ik} - K_{jk}) \tag{9}$$

We can also notice for the future that this is equivalent to:

$$a_F(i,j,k) = \frac{1}{3}\left(\|S_i - S_j\|_F^2 + \|S_i - S_k\|_F^2 + \|S_j - S_k\|_F^2\right) \tag{10}$$



(a) 1×240 (b) 2×240

(c) 4×240 (d) 10×240

Figure 1. **Recovered $\mathbf{l}_t$'s**. This figure concerns the synthetic data from [20] where 91 features are tracked from a shark bending its tail twice during 240 frames. The features were created from a basis of exact dimensionality 2. The figure illustrates the recovered $\mathbf{l}_t$'s for $s = 2$, each linked to its previous and next temporal neighbors (in blue) as well as to the same point a period after (in black). Below each figure is indicated the number of triplet samples chosen to compute the embedding (the number of pairs is chosen to be the same) and, as expected, the more the better. What we see in the last case is one path almost overlapping with itself once (hence the periodicity of the motion). In green are displayed the points 1, 121 and 240, while in red are displayed the points 61 and 181 (these points are important as the video sequence has 240 frames and a period of 120). The green points should indicate when the shark returns to its initial state (hence their proximity) while the red ones indicate the half period (hence the fact they are furthest from the green points and actually symmetric). While these points are not perfectly overlapping, they are close and enough for us to perform full SFM.

4.3. Constraints

Note the K_{ij}'s can be recovered up to a similarity. We choose to constrain the rotation/translation/scale differently from [1]:

- **Scale**. If $(\mathbf{l}_t, R_t, \mathtt{S})$ is a solution of Equation (2), so is $(\alpha \mathbf{l}_t, R_t, \frac{1}{\alpha}\mathtt{S}), \forall \alpha \neq 0$. Therefore, scale ambiguity is already inherent to our formulation. Nonetheless, to prevent the $\mathbf{l}_t$'s from collapsing to the origin, and as we have chosen the $\mathtt{S}^i$'s to be of norm 1, we choose to enforce a scale constraint as follows:

$$\|\mathbf{l}_i - \mathbf{l}_j\|_2^2 = \|S_i - S_j\|_2^2 \geq a_{\min}(i,j) \tag{11}$$

where $a_{\min}(i,j)$ is defined in Section 5.1.

- **Rotation**. The rotation ambiguity is also inherent to our formulation as the shape formation equation contains an invertible matrix ambiguity: for every valid $(\mathbf{l}_t, R_t, \mathtt{S})$, $(Q^{-1}\mathbf{l}_t, R_t, \left[\text{vec}\left(\mathtt{S}^0\right)\text{vec}\left(\mathtt{S}^1\right)Q\cdots\text{vec}\left(\mathtt{S}^s\right)Q\right])$ is also a solution (where $\text{vec}\left(\cdot\right)$ is the operator that stacks the columns into a vector).

- **Translation**. Finally, the translation ambiguity is also inherent to our formulation, and therefore unimportant, as for every valid $(\mathbf{l}_t, R_t, \mathtt{S})$, $(\begin{bmatrix}1\\ \mathbf{l}_t+\mathbf{t}\end{bmatrix}, R_t, \left[\text{vec}\left(\mathtt{S}^{0\prime}\right)\cdots\text{vec}\left(\mathtt{S}^s\right)\right])$ is also a solution, with $\text{vec}\left(\mathtt{S}^{0\prime}\right) = \text{vec}\left(\mathtt{S}^0\right) - \left[\text{vec}\left(\mathtt{S}^1\right)\cdots\text{vec}\left(\mathtt{S}^s\right)\right]\mathbf{t}$. Nonetheless, to avoid a drifting of the $\mathbf{l}_t$'s during their computation, we impose that the $\mathbf{l}_t$'s be centered:

$$\sum_{t=1}^{f}\mathbf{l}_t = 0 \Longleftrightarrow \sum_{t=1}^{f}\mathbf{l}_t^\top\sum_{t=1}^{f}\mathbf{l}_t = 0 \Longleftrightarrow \sum_{t,t'}K_{tt'} = 0 \tag{12}$$

We also add a regularization term to impose a smooth deformation of the object over time: $\lambda\sum_{t=2}^{f-1}\Delta''(\mathbf{l}_t)$, where λ weighs the regularization and Δ'' is a finite difference approximation of the second derivative of $\mathbf{l}_t$ at t (and can be expressed in terms of $K_{tt'}$). We therefore have the new SDP formulation:

$$\begin{aligned}\min \quad & \sum_{((i,j,k),(i',j',k'))\in\mathcal{F}}\xi_{i,j,k,i',j',k'} + \lambda\sum_{t=2}^{f-1}\Delta''(\mathbf{l}_t)\\ \text{subject to} \quad & \text{constraints of Equation (8)}\\ & \|\mathbf{l}_i-\mathbf{l}_j\|_2^2 \geq a_{\min}(i,j), \quad \forall(i,j)\\ & \sum_{t,t'}K_{tt'} = 0\end{aligned} \tag{13}$$

The $\mathbf{l}_t$'s are then obtained from K by using its $r_\mathtt{S}$-rank SVD decomposition (where $r_\mathtt{S} = \text{rank}\left(\mathtt{S}\right) = \text{rank}\left(W\right)$), as shown on Figure 1.

5. Ordering Set $\mathcal{F}$

We have shown in the previous section how the coordinates of every shape S_t in the basis $\mathtt{S}$ can be computed by using an ordering set $\mathcal{F}$ between triplets of frames. This section focuses on the computation of such a set. To this end, we need to define the 3D measurement matrix W_i^* whose first two rows are W_i, but whose third row is what would be obtained if the depth could be measured. If there is no noise, $W_i^* = R_i^* S_i$. For the sake of simplicity, we will assume that there is no noise, but what follows would hold by adding an extra term of lower order.

5.1. Triplet Distance Infimum

Considering two frames i and j, we have:

$$\|S_i - S_j\|_F^2 = \left\|R_i^{*\top}W_i^* - R_j^{*\top}W_j^*\right\|_F^2 \tag{14}$$

$$= \left\|W_i^* - R_i^*R_j^{*\top}W_j^*\right\|_F^2 \tag{15}$$

Therefore:

$$\begin{aligned}\|S_i - S_j\|_F^2 \geq \quad & \min_{R^*,\mathbf{x},\mathbf{y}} \quad \left\|\begin{bmatrix}W_i\\ \mathbf{x}^\top\end{bmatrix} - R^*\begin{bmatrix}W_j\\ \mathbf{y}^\top\end{bmatrix}\right\|_F^2\\ & \text{subject to} \quad R^* \text{ is a rotation matrix}\\ & \qquad\qquad \text{mean}(\mathbf{x}) = \text{mean}(\mathbf{y}) = 0\end{aligned} \tag{16}$$

We name this minimum: $a_{\min}(i,j)$. Its definition can be interpreted as a pose estimation problem where the third coordinates/depths are unknown. $\mathbf{x}$ and $\mathbf{y}$ can be computed in closed form with respect to R^* and the problem then boils down to a simple optimization over the quaternion of R^*.

We can now define an infimum for $a_F\left(i,j,k\right)$, that we name $a_{\min}(i,j,k)$, as follows:

$$a_F\left(i,j,k\right) \geq \frac{1}{3}\left(a_{\min}(i,j) + a_{\min}(j,k) + a_{\min}(k,i)\right) \tag{17}$$

5.2. Triplet Distance Supremum

For the supremum, we can notice that:

$$a_F\left(i,j,k\right) = \min_S \frac{1}{3}\left(\|S_i - S\|_F^2 + \|S_j - S\|_F^2 + \|S_k - S\|_F^2\right) \tag{18}$$

Therefore:

$$a_F\left(i,j,k\right) \leq \frac{1}{3}\left(\|S_i - S_{ijk}\|_F^2 + \|S_j - S_{ijk}\|_F^2 + \|S_k - S_{ijk}\|_F^2\right) \tag{19}$$

where S_{ijk} is the optimal reconstruction for the 3 frames i,j and k. We define R_i', R_j', R_k' are the corresponding optimal rotation matrices (different from R_i, R_j, R_k but close for frames representing close-by shapes).

Now, if we assume that the reconstruction error in depth is similar to the reprojection errors in abscissa and ordinate, each term $\|S_i - S_{ijk}\|_F$ can be approximated by:

$$\|S_i - S_{ijk}\|_F \simeq \|W_i^* - R_i'S_{ijk}\|_F \simeq \frac{3}{2}\left\|W_i - R_i'^*S_{ijk}\right\|_F \tag{20}$$

The terms on the right in the expression above can be computed as part of the 3D reconstruction of frames i,j,k and therefore, we obtain a supremum for $a_F\left(i,j,k\right)$:

$$a_F\left(i,j,k\right) \leq a_{\max}(i,j,k) \tag{21}$$

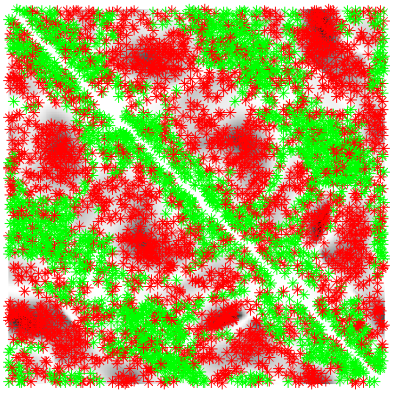


Figure 2. This figure is also based on the shark sequence. **Frame Triplet Selection Procedure.** The sampling is performed as follows: for every frame i in the vertical axis, two frames j and k leading to a low reconstruction error are chosen: these two are plotted in green on the line number i. Then, two views j' and k' are chosen so that $a_{\max}(i,j,k) \leq a_{\min}(i,j',k')$: they are plotted in red on the same line. If no such pair (j',k') exists, the pair (j,k) is not considered. As we can see, the green couples are almost only present on the diagonals and sub-diagonals while the red are located anywhere. 2000 triplets are sampled and represented on this figure. This amount was chosen with respect to limitations of the SDP solver.

5.3. Computing the Ordering Set $\mathcal{F}$

Now, given two triplets of pairs (i,j,k) and (i',j',k'), we can compute:

$$\begin{aligned} a_{\min}(i,j,k) \leq a_F(i,j,k) \leq a_{\max}(i,j,k) \\ a_{\min}(i',j',k') \leq a_F(i',j',k') \leq a_{\max}(i',j',k') \end{aligned} \tag{22}$$

Therefore, if $a_{\max}(i,j,k) \leq a_{\min}(i',j',k')$, we have $((i,j,k),(i',j',k')) \in \mathcal{F}$ (and respectively with $a_{\max}(i',j',k') \leq a_{\min}(i,j,k)$). Such triplets exist as $a_{\max}(i,j,k) = 0$ for pairs representing the same shape. Similarly, $a_{\min}(i,j,k) \gg 0$, usually, for frames representing very different shapes.

5.4. Practicality

In practice, the $a_{\min}(i,j,k)$ and $a_{\max}(i,j,k)$ are computed for several triplets and if an ordering exists, it is used in the SDP problem.

While we could compute $a_{\max}(i,j,k)$ for every triplet, the task is computationally very expensive (it is $O(f^3)$). Therefore, to speed up the process and obtain low $a_{\max}(i,j,k)$, we first compute all the $a_{\min}(i,j)$'s, and, given a frame i, we draw j and k from a distribution based on this error, leading to i,j and k more likely to be views of the same shape. Figure 2 illustrates this procedure.

6. Locally Optimal Shape Basis and Rotation

Now that we have obtained a good approximation of the $\mathbf{l}_t$'s, we are going to use them to first find a good estimate of the optimal shape basis $\mathtt{S}$ and then obtain the R_t's. We will then use these parameters as an initialization to a gradient descent procedure.

The proposed full structure from motion method now detailed does not suffer from the rank problems from [23] and that were approached but not clearly solved in [24]. It also does not face problems when some shape coefficients are small when trying to recover the rotation matrices (like in [24]).

In the following, we will assume that $\mathtt{S}$ is of rank $r_{\mathtt{S}}$ ($r_{\mathtt{S}} \leq 3(s+1)$; it was assumed that $r_{\mathtt{S}} = 3(s+1)$ in [23]) and that it is given by the user or guessed from W (as $\text{rank}\,(\mathtt{S}) = \text{rank}\,(\overline{W})$). Now, let us consider an optimal rank decomposition of $\overline{W}$ (*e.g.* provided by SVD): $\overline{W} = AB$ with A and B full rank (of rank $r_{\mathtt{S}}$). As the rows of B and $\mathtt{S}$ span the same spaces, we have $\mathtt{S} = GB$ where G is a $3(s+1) \times r_{\mathtt{S}}$ ambiguity matrices.

The goal is now to recover G first, and then the R_t's.

6.1. Kronecker Constraint

By using the definition of $\mathtt{S}$ in Equation (2), $\mathtt{S}$ must verify for every t:

$$\begin{aligned} &\left(\begin{bmatrix} 1 & \mathbf{l}_t^\top \end{bmatrix} \otimes R_t\right) \mathtt{S} = \overline{W}_t \\ \Rightarrow &R_t \left(\begin{bmatrix} 1 & \mathbf{l}_t^\top \end{bmatrix} \otimes \mathrm{I}_3\right) G = \overline{W}_t B^+ \end{aligned} \tag{23}$$

as B has full row rank. By dropping, for now, the rotation constraint on the R_t's, we obtain the following bilinear problem:

$$\min_{R_t,G} \sum_{t=1}^{f} \left\| R_t \left(\begin{bmatrix} 1 & \mathbf{l}_t^\top \end{bmatrix} \otimes \mathrm{I}_3\right) G - \overline{W}_t B^+ \right\|_F^2 \tag{24}$$

We also add a regularization term so that the rotations do not change much from frame to frame: $\lambda_R \sum_{t=2}^{f} \|R_t - R_{t-1}\|_F^2$. In order not to have the R_t's shrink to 0 and G diverge to infinity (as for any (R_t, G), $(\alpha R_t, \frac{1}{\alpha} G)$ is also a solution), we also need a counterbalancing regularization term: $\lambda_G \|G\|_F^2$. In practice, we choose $\lambda_R = \lambda_G = 1$. Hence the new bilinear problem:

$$\begin{aligned} \min_{R_t,G} \sum_{t=1}^{f} \left\| R_t \left(\begin{bmatrix} 1 & \mathbf{l}_t^\top \end{bmatrix} \otimes \mathrm{I}_3\right) G - \overline{W}_t B^+ \right\|_F^2 \\ + \lambda_R \sum_{t=2}^{f} \|R_t - R_{t-1}\|_F^2 + \lambda_G \|G\|_F^2 \end{aligned} \tag{25}$$

While recent work like [6] improves the solving of bilinear problems if one of the two sets of variables as a much

lower dimensionality, it is impractical in our case as the dimensions of G are too high. We therefore solve it by generating several random initializations for G and proceed by alternate optimization between the R_t's and G (a closed form can easily be obtained for one, if the other is fixed). In practice, we use 10 random initializations and 50 alternate optimization iterations.

It is worth noting that after this optimization, an approximation to the best solution is obtained up to an ambiguity matrix Q as:

$$R_t \left(\begin{bmatrix} 1 & \mathbf{l}_t^\top \end{bmatrix} \otimes \mathrm{I}_3 \right) G = R_t Q \left(\begin{bmatrix} 1 & \mathbf{l}_t^\top \end{bmatrix} \otimes \mathrm{I}_3 \right) \left(\mathrm{I}_{s+1} \otimes Q^{-1} \right) G \tag{26}$$

6.2. Rotation Constraint

So far, the R_t's have not been imposed to be rotation matrices. We now seek the ambiguity matrix Q such that the R_t's are as close to rotation matrices as possible by optimizing the simple SDP problem:

$$\min_Q \sum_{t=1}^{f} \left\| R_t Q Q^\top R_t^\top - \mathrm{I}_2 \right\|_F^2 \tag{27}$$

6.3. Final Optimization

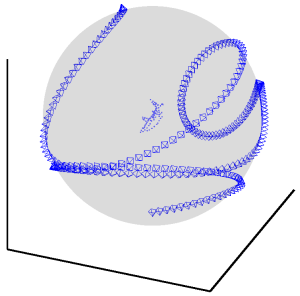
G is first recovered using the Kronecker and the rotation constraints. From there, $\mathtt{S}$ is recovered and therefore all the S_t. Recovering an initial estimate of R_t's is then solving multiple instances of pose estimation. While we could perform the full computation for every frame, we first only perform it for a few frames, but very accurately (we had little chance with EPnP from [13] so we used our own implementation relying on a simple polynomial formulation and a solving by [9]). We then compute the optimal R_t's by performing gradient descent on $\|R_t S - W_t\|_F^2$ with an initial estimate of R_{t-1} and R_{t+1}.

Once an initial estimate of all the parameters $\mathbf{l}_t$, R_t and $\mathtt{S}$ is obtained, what follows is an optimization over all the parameters at once to minimize the reprojection error. To take advantage of the sparsity of the problem, we interpret it as a sparse bundle-adjustment with points of dimensionality $3(s+1)$ instead of 3 in the normal 3D-case (the camera parameters being extended to $(R_t, \mathbf{t}_t, \mathbf{l}_t)$) and then use the sprase Levenberg-Marquardt of [15] to obtain a fast and accurate solution.

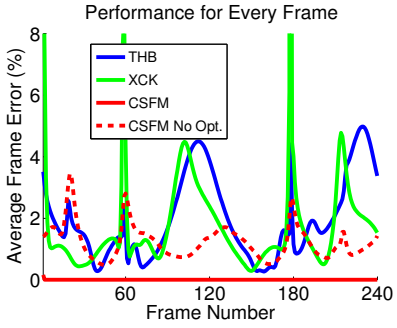
7. Experiments

We tested our approach on two synthetic datasets: the classical *Shark Data* from [19] and the *Roller Coaster* from [16]. We also experimented with real data.

The reconstruction error considered in these experiments is computed in percentage points: the average distance of the reconstructed point to the correct point divided by the



(a) Camera Positions



(b) Depth Error over Time

Figure 3. **Shark Dataset from from [19].** Figure (a) shows the camera positions used in our experiment: the camera has a smooth random path over a sphere. Figure (b) shows the reconstruction error produced by Xiao-Chai-Kanade CVPR04 (XCK), Torresani-Hertzmann-Bregler PAMI08 (THB) and our algorithm (CSFM), with and without the final gradient descent.

span of the shape, as defined in [21]. We also compare our approach (named CSFM for Comparison-based Structure From Motion) to two standard algorithms: [20] (with 100 EM steps if not specified) and [24]. For the latter, we have our own implementation of the code which uses an SDP formulation and does reach an error of 0 (as it is a closed-form solution) in the assumptions of the paper (that actually do not fit the shark data). We ran 10 trials for each experiment.

The running time for these experiments is usually 25 minutes on a 3GHz machine with the following bottlenecks:

- The computation of the $a_{\min}$ takes 10 minutes. It is optimized Matlab code and results in an average of 3 steps of gradient descent for every of the $O(f^2)$ pairs.
- The SDP problem is solved in 10 minutes. SDP is a very active area of research and faster algorithms are to be expected.
- The final gradient descent takes a few minutes. The efficient Sparse Bundle Adjustment code from [15] is used. The computation time seems hard to improve but it is worth noticing that 50 iterations are used while much fewer could be used (*cf*.Figure 4(a)).

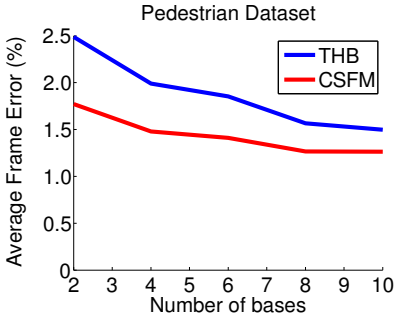


Figure 5. **Pedestrian Dataset**. This figure shows the 3D reconstruction errors for THB and CSFM on the pedestrian dataset from [20]. Our method achieves a lower error and seems to scale with higher orders of deformation.

7.1. Shark Data

The shark data was first introduced in [19]. It features a synthetic shark created from a basis where $s = 2$ rotating its tail twice over 240 frames. In terms of performance, our method reaches 0.00% (actually $10^{-27}\%$) of error while [20]'s with 100 EM iterations reaches 2%. Details are given in Figure 3 and more insight is given in Figure 4.

7.2. Pedestrian Data

We also ran CSFM on the pedestrian dataset from [20] to compare with THB and also see how our method deals with more modes of deformation. The results are shown in Figure 5

7.3. Bending Sheet

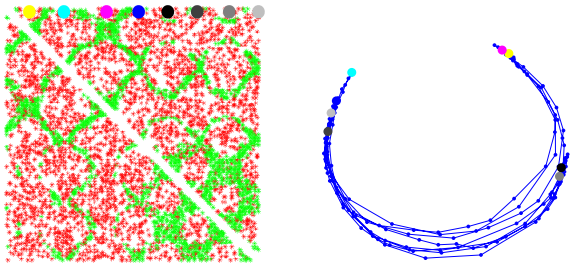
The final experiment is a video of a sheet of paper twisted laterally back and forth. 51 features are tracked during the 190 frames of the sequence. The motion is repetitive but not circular: the sheet is bent to an extreme before being bent to normal. Figure 6 illustrates how this impacts the reconstruction of the shape embedding.

This deformation seems fairly easy to explain with the shape basis and choosing $s = 2$ was enough to reproduce equivalent 3D deformations.

8. Conclusion

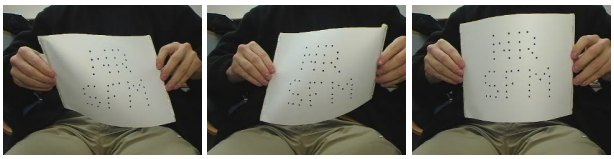
In this paper, we have presented a novel approach to recover the basis coefficients of a shape deforming in an orthographic video sequence, independently from the other parameters. Using triplets of frames, we are able to define proximities between 3D shapes which can be used in a semi-definite programming formulation to recover the linear embedding of the shapes.

While these embeddings are interesting in themselves, we showed they can provide a very good initialization for further 3D-reconstruction which competes with the current



(a) Sampling (b) Recovered Embedding

Figure 6. **Bending Sheet Dataset** Figure (a) illustrates the sampling as described in Figure 2. We can notice the repetition of the motion with the different green descending diagonals. On the other hand, the ascending green diagonals illustrates the fact that when reaching an end, it looks equivalent to having the sequence going forward or backward in time. This is even more obvious when looking at the recovered manifold in Figure (b): he sheet is twisted between two extremes. It is worth noting that the black point and its neighbor do not belong to an end of the manifold as the sheet was not twisted fully then.



(a) Sample Frames from the Sheet video sequence

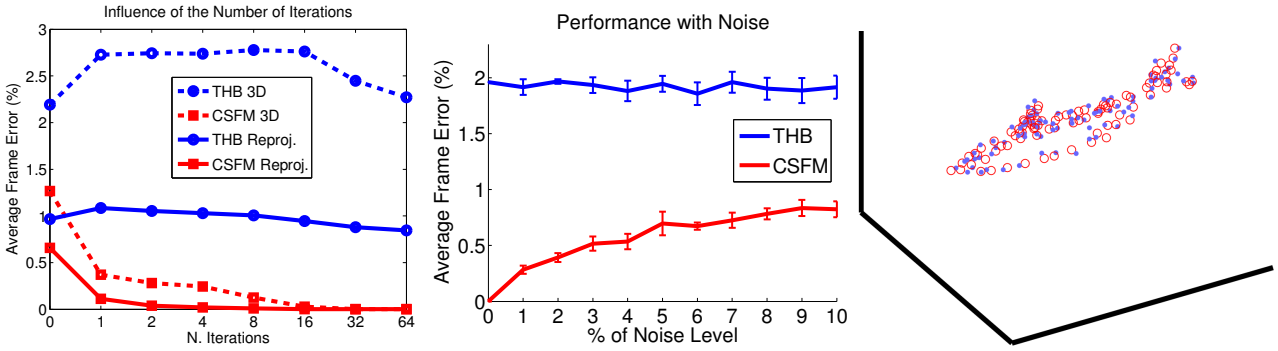
(b) Reconstructions corresponding to the above frames, rendered from novel view points

Figure 7. Figure (a) shows 3 frames from the original sequence where the sheet is twisted. In Figure (b), three reconstructed views are presented under a novel view point: a quadratic surface was fit through the points and illuminated so as to highlight the bending.

state of the art. Further improvement includes the study of a regularization term in the final optimization to conserve the shape deformation smoothness recovered in the shape embedding.

Acknowledgements

This work was funded in part by NSF Career Grant #0448615, the ONR MURI Grant #N00014-08-1-0638 and the UCSD Division of the California Institute for Telecommunications and Information Technology, Calit2. The authors would also like to thank Sameer Agarwal, Ben Ochoa, Steve and Kristin Branson as well as Manmohan Chandraker for very useful discussions and help.



(a) Importance of the Number of Iterations (b) Dealing with noise (c) Example of a noisy frame

Figure 4. **Importance of the Number of Iterations.** In Figure (a), the reprojection error (full line) and the 3D reconstruction error (dashed) are represented for the THB algorithm and our approach. The iterations for THB correspond to the EM iterations while the ones for CSFM are the ones for the gradient descent. CSFM converges very quickly to the best solution. **Robustness to Noise** Figure (b) displays the 3D reconstruction error of THB and CSFM with respect to noise (XCK was excluded because of its poor performance). The amount of noise (in percentage) is $\|err\| \ / \ \|W\|$. As an example of how much noise is 10%, we display a frame in Figure (c).

References

[1] S. Agarwal, J. Wills, L. Cayton, G. Lanckriet, D. Kriegman, and S. Belongie. Generalized non-metric multidimensional scaling. In *AISTATS*, 2007.

[2] A. Bartoli, V. Gay Bellile, U. Castellani, J. Peyras, S. Olsen, and P. Sayd. Coarse-to-fine low-rank structure-from-motion. In *CVPR*, pages 1–8, 2008.

[3] M. Brand. A direct method for 3d factorization of nonrigid motion observed in 2d. In *CVPR*, 2005.

[4] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3d shape from image streams. In *CVPR*, pages 690–696, 2000.

[5] R. L. Carceroni, F. L. C. Padua, G. A. M. R. Santos, and K. N. Kutulakos. Linear sequence-to-sequence alignment. *CVPR*, 01:746–753, 2004.

[6] M. Chandraker and D. Kriegman. Globally optimal bilinear programming for computer vision applications. In *CVPR*, 2008.

[7] J. P. Costeira and T. Kanade. A multibody factorization method for independently moving objects. In *IJCV*, volume 29, September 1998.

[8] A. Del Bue, F. Smeraldi, and L. Agapito. Non-rigid structure from motion using ranklet-based tracking and non-linear optimization. In *IVC*, volume 25, pages 297–310, 2007.

[9] D. Henrion and J.-B. Lasserre. Gloptipoly: Global optimization over polynomials with matlab and sedumi. *ACM Trans. Math. Softw.*, 29(2):165–194, 2003.

[10] T. Kim and K.-S. Hong. Estimating approximate average shape and motion of deforming objects with a monocular view. In *IJPRAI*, volume 19, pages 585–601, 2005.

[11] I. Laptev, S. J. Belongie, P. Perez, and J. Wills. Periodic motion detection and segmentation via approximate sequence alignment. In *ICCV*, volume 1, pages 816–823, 2005.

[12] I. Laptev and P. Perez. Retrieving actions in movies. In *ICCV*, pages 1–8, 2007.

[13] V. Lepetit, F. Moreno-Noguer, and P. Fua. Epnp: An accurate o(n) solution to the pnp problem. In *IJCV*, 2008.

[14] X. Llado, A. Del Bue, and L. Agapito. Non-rigid 3d factorization for projective reconstruction. In *BMVC*, 2005.

[15] M. Lourakis and A. Argyros. The design and implementation of a generic sparse bundle adjustment software package based on the levenberg-marquardt algorithm. Technical Report 340, Institute of Computer Science - FORTH, 2004.

[16] V. Rabaud and S. Belongie. Re-thinking non-rigid structure from motion. In *CVPR*, 2008.

[17] S. Soatto and P. Perona. Recursive estimation of camera motion from uncalibrated image sequences. In *ICIP*, pages III: 58–62, 1994.

[18] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. In *IJCV*, volume 9, November 1992.

[19] L. Torresani, A. Hertzmann, and C. Bregler. Learning non-rigid 3d shape from 2d motion. In *NIPS*, 2003.

[20] L. Torresani, A. Hertzmann, and C. Bregler. Non-rigid structure-from-motion: Estimating shape and motion with hierarchical priors. In *PAMI*, 2008.

[21] L. Torresani, D. Yang, E. Alexander, and C. Bregler. Tracking and modeling non-rigid objects with rank constraints. In *CVPR*, pages I–493– I–500, 2001.

[22] R. Vidal and D. Abretske. Nonrigid shape and motion from multiple perspective views. In *ECCV*, pages II: 205–218, 2006.

[23] J. Xiao, J. Chai, and T. Kanade. A closed-form solution to non-rigid shape and motion recovery. In *IJCV*, volume 67, April 2006.

[24] J. Xiao and T. Kanade. Non-rigid shape and motion recovery: degenerate deformations. In *CVPR*, pages I–668– I–675 Vol.1, 2004.

[25] J. Xiao and T. Kanade. Uncalibrated perspective reconstruction of deformable structures. In *ICCV*, pages 1075– 1082 Vol. 2, 2005.

[26] J. Yan and M. Pollefeys. A factorization-based approach to articulated motion recovery. In *CVPR*, pages 815– 821, 2005.