Joint 48th IEEE Conference on Decision and Control and
28th Chinese Control Conference
Shanghai, P.R. China, December 16-18, 2009

Using Dynamics to Recover Euclidian 3–Dimensional Structure from 2–Dimensional Perspective Projections

Mario Sznaier Mustafa Ayazoglu Octavia Camps
Electrical and Comp. Engineering Department,
Northeastern University,
Boston, MA 02115.

Abstract— **In this paper we consider the problem of recovering the 3–dimensional Euclidian structure of a rigid object from multi-frame point correspondence data in a sequence of 2–D images obtained under perspective projection. The main idea is to recast the problem as the identification of an LTI system based on partial data. The main result of the paper shows that, under mild conditions, the lowest order system whose projections interpolate the 2-D data, yields (up to a *single* scaling constant) the correct 3 dimensional Euclidean coordinates of the points. Finally, we show that the problem of finding this system (and hence the associated 3-D data) can be recast into a rank minimization form that can be efficiently solved using convex relaxations. In contrast, existing approaches to the problem, based on iterative matrix factorizations can recover structure only up to a projective transformation that does not preserve the Euclidian geometry of the object.**

I. Introduction

The problem of structure from motion (SfM) consists of recovering the 3D shape (structure) of a rigid object or scene from a set of correspondences of features in a sequence of 2D images captured by a camera. This is a central problem in computer vision with many applications including image-based modeling for computer graphics and animation, autonomous navigation, and human computer interfaces. An extensive review and explanation of SfM methods can be found in the textbooks [1], [2].

The most popular algorithms for SfM are based on the factorization method for 3D reconstruction under orthographic projection proposed by Lucas and Kanade in [3]. In this approach, a *measurement matrix* consisting of the image coordinates of the set of point features tracked over the sequence is factored through a singular value decomposition (SVD) into a *motion matrix* with the motion matrices for each frame stacked on top of each other and a *structure matrix* with the 3D coordinates of the tracked scene points. Since the original formulation of this method, several additions and improvements have been proposed to enhance its performance and expand its application. Morita and Kanade [4], proposed sequentially updating the factorization to improve run time, and Fransen et al [5] used system identification techniques to improve robustness to tracking errors. Several extensions have been proposed to deal with non rigid objects [6]–[8] and to segment multiple moving objects [9]–[19].

The factorization method was also expanded to handle paraperspective projection in [20] and perspective projections in [21], [22]. However, these methods are not longer based on a simple SVD due to the nonlinearity introduced by the projection. Instead, the factorization is found after an iterative procedure to search for the unknown projective depths. Furthermore, they can only recover 3D structure *up to a projectivity transformation at each frame*. That is, if a set of unknown depths Z_{ij} is a solution, then the same set multiplied by a time and point varying scaling factor λ_{ij} is also a solution. Recovering the Euclidian geometry entails an additional computationally challenging non–linear, non–convex optimization.

Motivated by these difficulties, in this paper, we present a convex–optimization based solution to the SfM problem under perspective projection capable of recovering the *Euclidean* 3D structure up to a *single* constant scaling factor across the entire motion sequence. This is accomplished by exploiting the dynamical information encoded in the temporal ordering of the frames. Specifically, the main result of the paper shows that, under mild conditions, the lowest order system whose projections interpolate the 2-D data, yields (up to a *single* scaling constant) the correct 3 dimensional Euclidian coordinates of the points. Finally, we show that the problem of finding this system (and hence the associated 3-D data) can be recast into a rank minimization form that can be efficiently solved using convex relaxations.

The paper is organized as follows. Section II summarizes the notation used in the paper, introduces some required definitions and formally states the structure from motion problem. Section III presents the main result of the paper, showing that the 3-D geometry can be recovered, up to an overall scaling function, by finding a minimum rank operator that interpolates a set of trajectories constructed from the 2-D data. These results are illustrated in section IV with an example. Finally, section V summarizes our conclusions and point out to directions for future research.

II. Preliminaries

A. Notation and definitions

In this section we summarize the notation used in the paper and introduce some definitions required to formally state the

This work was supported in part by NSF grants ECS–0648054, IIS–0713003, and AFOSR grant FA9550–09–1–0253..

problem under consideration.

$\|x\|_\infty$	∞ norm of the vector $x \in R^n$: $\|x\|_\infty \doteq \max_i \|x_i\|$.
$\|x\|_2$	2 norm of the vector $x \in R^n$: $\|x\|_2^2 \doteq \sum_{i=1}^n \|x_i\|^2$.
$\det(\mathsf{M})$	determinant of the matrix M.
ℓ_∞^n	extended Banach spaces of vector valued real sequences $\{y\}_0^\infty \in R^n$ equipped with the norm $\|y\|_{\ell_\infty} \doteq \sup_i \|y_i\|_\infty$.
$\mathcal{P}_k^n$	projection operator from ℓ_∞^n to R^n $$\mathcal{P}_k^n([x_o \dots x_k \dots]) = x_k$$
$\mathbf{H}_y(k,l)$	$k \times l$ Hankel matrix associated with the vector sequence $\mathbf{y}$ $$\mathsf{H}_y(k,l) \doteq \begin{bmatrix} \mathbf{y}_1 & \mathbf{y}_2 & \cdots & \mathbf{y}_l \\ \mathbf{y}_2 & \mathbf{y}_3 & \cdots & \mathbf{y}_{l+1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{y}_k & \mathbf{y}_{k+1} & \cdots & \mathbf{y}_{k+l-1} \end{bmatrix}$$
$X(z)$	z–transform of the sequence $\{x\}_k$. $X(z) \doteq \sum_o^\infty x_k z^{-k}$.

Definition 1: A linear time invariant operator $\mathcal{L}$: $x_o \in R^n \rightarrow \{x_k\} \in \ell_\infty^n$ is said to be *pointwise rigid* if $\det(\mathcal{P}_k^n \mathcal{L}) = 1$ for all k (note that this implies $\|x_k\|_2 = \|x_o\|_2$ but rules out operators such as specular symmetry)

In this paper we consider a class of rigid objects defined as follows:

Definition 2: N_p points $P_1, \dots, P_{N_p} \in R^3$ are said to belong to a rigid body if there exist a point $O \in R^3$ (not necessarily in the object) and a linear time invariant, pointwise rigid operator $\mathcal{L}$: $R^n \rightarrow \ell_\infty^n$ such that for all points and all time instants, the corresponding trajectories satisfy:

$$P_{ki} - O_k = \{\mathcal{L}\,[P_{oi} - O_o]\}_k,\ k = 1, 2, \dots$$

where P_{ki} and $\{\mathcal{L}[x]\}_k$ denote the coordinates of point P_i and the output of $\mathcal{L}$ at time k, respectively. In the sequel we will refer to O as the center of the motion.

A simple example of the definition above is the case of a constant rotation R about a moving axis. In this case, L_k, the Markov parameters (impulse response) of the operator $\mathcal{L}$ are given by $L_k = R^k$ and $\mathcal{L}(z) = z(zI - R)^{-1}$.

B. SfM from Perspective Image Sequences

Given an image sequence of a rigid scene captured by a camera under perspective projection, the problem of *structure and motion from perspective image sequences* seeks to determine the 3D structure of the scene and the relative motion between the camera and the scene from a set of feature correspondences established by matching the images of N_p scene points across N_F frames. This problem is formalized below.

Consider a camera Cartesian coordinate system defined with its origin at the center of projection and its Z axis along the camera optical axis. Let N_p be the number of tracked points from the scene and let $\mathbf{Q}_{ij} = (X_{ij}, Y_{ij}, Z_{ij})^T$ be the 3D Cartesian camera coordinates of point $\mathbf{Q}_j$, $j = 1, \dots, N_p$, at the time frame i, $i = 1, \dots, N_F$. Then, the 2D homogeneous coordinates of the images of these points at frame i, $\bar{\mathbf{q}}_{ij} = (x_{ij}, y_{ij}, z_{ij})^T$ and the corresponding Cartesian image coordinates, $\mathbf{q}_{ij} = (u_{ij}, v_{ij})^T$, are given by

$$\bar{\mathbf{q}}_{ij} = \mathbf{P}\mathbf{Q}_{ij} = \begin{bmatrix} f & 0 & 0 \\ 0 & \alpha f & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{Q}_{ij} \quad (1)$$

$$u_{ij} = \frac{x_{ij}}{z_{ij}} = f\frac{X_{ij}}{Z_{ij}} \quad (2)$$

$$v_{ij} = \frac{y_{ij}}{z_{ij}} = \alpha f\frac{Y_{ij}}{Z_{ij}} \quad (3)$$

where $\mathbf{P}$ is the 3×3 projection matrix associated with the camera, f is the focal length, and α is the pixel aspect ratio. In this context, the problem of interest here can be precisely stated as:

Problem 1: Given the above setup, recover the scene structure $\mathbf{Q}_{ij}$ and the matrix $\mathbf{P}$, $i = 1, \dots, N_F$, j= $1, \dots, N_P$, from the $N_P \times N_F$ feature correspondences $\mathbf{q}_{ij}$.

Classically, the problem above has been solved using the Strum Triggs Factorization Algorithm [21], based on iteratively computing the best rank 4 approximation to a matrix constructed from the image data, and the associated depths. Since the problem is not jointly convex, this algorithm is guaranteed to converge only to a local solution. In addition, the algorithm as stated above can only recover the 3D structure up to an arbitrary projectivity.

III. The Role of Dynamics in Structure Recovery

A. Recovering geometry via Rank Minimization

The proposed method is based on the following result, showing that for rigid objects, dynamics encapsulates geometry, in the sense that the latter can be recovered from the *simplest* dynamical system that explains the available data. Formally:

Theorem 1: Consider the trajectories P_{ki}, $i = 1, 2, 3$ of three points from a rigid moving under some point-wise rigid LTI motion operator $\mathcal{L}$ with center O. Let $\alpha_{ki} \geq \epsilon > 0$, $i = 1, 2$ be arbitrary constants and define the *difference* vectors

$$\mathbf{y}_k^{i,\alpha_{ki}} = \begin{bmatrix} X_{ki} - \alpha_{ki} X_{k3} \\ Y_{ki} - \alpha_{ki} Y_{k3} \\ Z_{ki} - \alpha_{ki} Z_{k3} \end{bmatrix},\ i = 1, 2 \quad (4)$$

where $\mathbf{P}_{ki} = (X_{ki}, Y_{ki}, Z_{ki})^T$, represents the location at time k of the 3D point $\mathbf{P}_i$, measured with respect to a coordinate system attached to a pin hole camera and α_{ki} is a time–varying scaling factor. Denote by $\mathcal{L}(\alpha_{ti})$ the pointwise rigid LTI operator that maps $\mathbf{y}_o^{i,\alpha_{oi}}$ to $\mathbf{y}_t^{i,\alpha_{ti}}$. Then, under mild conditions:

$$\operatorname{argmin}_{\alpha_{ti} \geq \epsilon} \operatorname{rank}\{\mathcal{L}(\alpha_{ti})\} \equiv 1,\ \text{for all } t, i$$

where for a linear time varying operator $\mathcal{L}$ its rank is defined as $\operatorname{rank}\{\mathcal{L}\} \doteq \sup_t \operatorname{rank}\{\mathsf{W}_t^o \mathsf{W}_t^c\}$, e.g the supremum of the

rank of the product of its observability and controllability Grammians.

Proof: see the Appendix ■

Corollary 1: Let $\mathsf{H}(\alpha) = \begin{bmatrix}\mathsf{H}_{\mathbf{y}^{1,\alpha_{t1}}} & \mathsf{H}_{\mathbf{y}^{2,\alpha_{t2}}}\end{bmatrix}$, where $\mathsf{H}_{\mathbf{y}}$ denotes the Hankel matrix associated with the vector sequence $\{\mathbf{y}\}$. Then $\operatorname{argmin}_{\alpha_{ki}\geq\epsilon}\operatorname{rank}\{\mathsf{H}(\alpha_{ki})\} = 1$.

Proof: see the Appendix ■

Corollary 2: Assume now that the measurements of all 3 points are affected by unknown scaling factors, potentially different in each direction, e.g. only $\mathcal{S}_{ki}P_{ki}$ are available, where $\mathcal{S}_{ki} = \operatorname{diag}(\alpha^x_{ki}, \alpha^y_{ki}, \alpha^z_{ki})$, with $\alpha^x_k, \alpha^y_k, \alpha^z_k > \epsilon > 0$ Then the rank of the matrix H is minimized by taking $\mathcal{S}_{ki} = \mathcal{S}_o$. That is, the geometry is recovered up to a (direction dependent) overall scaling factor. The proof follows immediately by applying the Theorem above to the trajectories $\mathcal{S}_{k1}(P_{k1} - \mathcal{S}^{-1}_{k1}\mathcal{S}_{k3}P_{k3})$ and $\mathcal{S}_{k2}(P_{k2} - \mathcal{S}^{-1}_{k2}\mathcal{S}_{k3}P_{k3})$ and noting that the rank of the observability matrix is invariant under left multiplication by the full rank matrix $\mathcal{S} \doteq \operatorname{blockdiag}(\mathcal{S}_{ki})$. Note also that the proof can be extended (at the price of a more complicated notation) to any number of points.

Remark 1: Briefly, Corollary 2 above states that if the 3-D trajectories of 3 points from a rigid are available up to unknown scaling factors (possibly different for each point), then the 3-D geometry can be recovered (up to an overall scaling factor) by minimizing the rank of the corresponding Hankel matrix. This result forms the basis of the proposed method.

B. Finding the unknown depths

Theorem 1 can be applied to solve Problem 1, as follows. From (1)–(3) it follows that, given the 2-D image coordinates (u_{ki}, v_{ki}) of the points P_i, $i = 1, \ldots, N_P$, the corresponding (scaled) 3-D coordinates are given by:

$$\tilde{P}_{ki} \doteq \begin{bmatrix} fX_{ki} \\ \alpha f Y_{ki} \\ Z_{ki} \end{bmatrix} = Z_{ki} \begin{bmatrix} u_{ki} \\ v_{ki} \\ 1 \end{bmatrix} \tag{5}$$

where Z_{ki}, f and α are unknown. From Corollary 2, it follows that the unknown Z_{ki} can be found (up to an overall scaling factor) by using the following algorithm.

Algorithm 1: (CONCEPTUAL) RANK MINIMIZATION BASED 3D-STRUCTURE RECOVERY

Input: (u_{ij}, v_{ij}), the 2-D coordinates of N_P points in N_F frames.

Output: 3-D depths Z_{ij} up to a an overall scaling constant.

1. Form the *difference* vectors:
$\mathbf{y}^i_k \doteq \tilde{P}_{ki} - \tilde{P}_{k,N_P},\ i = 1, \ldots, N_{P-1}$
and the corresponding Hankel matrices

$$\mathsf{H}_{y^i} \doteq \begin{bmatrix} \mathbf{y}^i_1 & \mathbf{y}^i_2 & \cdots & \mathbf{y}^i_l \\ \mathbf{y}^i_2 & \mathbf{y}^i_3 & \cdots & \mathbf{y}^i_{l+1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{y}^i_k & \mathbf{y}^i_{k+1} & \cdots & \mathbf{y}^i_{k+l-1} \end{bmatrix},\ i = 1, \ldots, N_P - 1$$

2. Solve the following rank minimization problem in Z_{ij}
$\operatorname{minrank}\begin{bmatrix}\mathsf{H}_{y^1} \ldots \mathsf{H}_{y^{N_p-1}}\end{bmatrix}$

C. Computational Complexity and Robustness Considerations.

In principle, Algorithm 1 will recover the unknown Z_{ij} in a single optimization step. Moreover, although it is well known that rank minimization is generically NP–hard, efficient convex relaxations in the form of a Linear Matrix Inequality optimization are available [23]. A potential problem here is the computational complexity entailed in solving for all Z_{ki} at the same time, since the computational complexity of conventional LMI solvers scales as (number of decision variables)^5. On the other hand, using larger sets of points minimizes the effects of outliers. To balance these effects we will pursue a sequential approach, where the coordinates of 4 points are found first. These coordinates are subsequently used to find the unknown calibration parameters and the coordinates of the other points, one at a time. Clearly, in the presence of noisy data, the performance of such an approach will depend on the choice of the initial 4-tuple. Thus, robustness can be improved by combining this algorithm with a Random Sample Consensus (Ransac) type approach [24], where N_s 4–tuples are randomly selected from the complete set of points, and the best one (in a sense made precise below) is chosen. These considerations lead to the following algorithm:

Algorithm 2: LMI MINIMIZATION BASED BASED 3D-STRUCTURE RECOVERY

Input: (u_{ij}, v_{ij}), the 2-D coordinates of N_P points in N_F frames.

Output: 3-D depths Z_{ij} up to an overall scaling constant, and camera intrinsic parameters f and α.

0. Select N_s, the number of Ransac iterations, set $\epsilon_{best} = \infty$, $iter = 1$,$f_{best} = 0, \alpha_{best} = 0$, $\mathcal{J}_{best} = [0, 0, 0, 0]$.

1. Randomly select a 4-tuple
$\mathcal{J} \doteq \{j_1, j_2, j_3, j_4\} \in [1, N_p]^4$.

2. Form the *difference* vectors:
$\mathbf{y}^i_k \doteq \tilde{P}_{ki} - \tilde{P}_{k,j_4},\ i = j_1, j_2, j_3$
and the corresponding Hankel matrices

$$\mathsf{H}_{y^i} \doteq \begin{bmatrix} \mathbf{y}^i_1 & \mathbf{y}^i_2 & \cdots & \mathbf{y}^i_l \\ \mathbf{y}^i_2 & \mathbf{y}^i_3 & \cdots & \mathbf{y}^i_{l+1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{y}^i_k & \mathbf{y}^i_{k+1} & \cdots & \mathbf{y}^i_{k+l-1} \end{bmatrix},\ i = j_1, j_2, j_3$$

3. Find $Z_{ki}, i = j_1, j_2, j_3$ by (approximately) minimizing $\operatorname{rank}[\mathsf{H}(Z_{ki})] \doteq \begin{bmatrix}\mathsf{H}_{y^1} \ldots \mathsf{H}_{y^3}\end{bmatrix}$ by solving the following convex problem in Z_{ki} R, S:

$$\text{minimize } Tr(\mathsf{R}) + Tr(\mathsf{S})$$
$$\text{subject to } \begin{bmatrix} \mathsf{R} & \mathsf{H}(Z) \\ \mathsf{H}(Z)^T & \mathsf{S} \end{bmatrix} \geq 0$$

4.- Find the calibration parameters f and α by solving the following optimization in the variables $\frac{1}{f^2}$, α^2, ϵ, for instance via least squares:
min ϵ subject to:
$|d(k+1, i, j) - d(k, i, j)| \leq \epsilon$ for all $k = 1, \ldots, N_F - 1$, $i = 1, \ldots, 4;\ j = 1, \ldots, 4,\ j \neq i$
where

$d(k,i,j) = \big[(Z_{ki}u_{ki} - Z_{ki}u_{kj})^2 +$
$\alpha^2(Z_{ki}v_{ki} - Z_{ki}v_{kj})^2 + \frac{1}{f^2}(Z_{ki} - Z_{kj})^2\big]$

5. If $\epsilon < \epsilon_{best}$ then
set $\epsilon_{best} = \epsilon$, $\mathcal{J}_{best} = \mathcal{J}$, $\alpha_{best} = \alpha$, $f_{best} = f$.

6. Set $iter = iter + 1$. If $iter <= N_s$ go to step 1.

7.- Use $\mathcal{J}_{best}$ as the initial 4-tuple and find the depth for the remaining points by solving, for all $i \in [1\ N_P]$, $i \notin \mathcal{J}_{best}$, the following convex problem in Z_{ki}, R and S:

minimize $Tr(\mathsf{R}) + Tr(\mathsf{S})$

subject to $\begin{bmatrix} \mathsf{R} & \mathsf{H}(Z) \\ \mathsf{H}(Z)^T & \mathsf{S} \end{bmatrix} \geq 0$

where $\mathsf{H}(Z)] \doteq \big[\mathsf{H}_{y^1} \ldots \mathsf{H}_{y^3}\mathsf{H}_{y^i}\big]$

IV. Illustrative Example

In this section we illustrate the proposed method using data[1] generated by animating an artificially generated object that is a standard benchmark in the computer graphics community. The data consists of 10 frames of the trajectories of 288 points taken from a 32–surface rendering of the Utah teapot shown in Figure 1, with 9 points selected from each surface. In this experiment the teapot underwent a constant velocity rotation around an axis slowly translating with constant velocity and 2 dimensional data was generated by projecting the 3-D coordinates of the points using a pinhole camera model with $f = 2$ and $\alpha = 1$. Sample frames are shown in Figure 1.

Figure 2 (a)–(b) show the results of applying Algorithm 2 to the 2–dimensional trajectories. As expected, the 3-D structure is recovered up to an overall scaling constant. Indeed, scaling back the reconstruction to its original size and computing the total reconstruction error yields $e^2 = \sum_i \|P_{orig} - P_{recons}\|^2 = 2.7 \times 10^{-12}$.

V. Conclusions

This paper considered the problem of recovering the 3–dimensional structure of a rigid object from a sequence of 2–D images obtained under perspective projection. The main idea is to recast the problem into a dynamical systems interpolation form: finding a minimal order system that interpolates the data. As we show in the paper, the lowest order interpolant (amongst all possible time–varying ones) is Linear Time Invariant and recovers the 3-D geometry up to an overall scaling factor. Exploiting the well known connection between system order and the rank of the associated Hankel matrix allows for recasting the reconstruction problem into a rank minimization form that can be relaxed to an efficient convex optimization form. In contrast, existing approaches to the problem exploit only geometrical constraints, discarding the information encapsulated in the temporal ordering of the frames (e.g. solutions are invariant to any arbitrary frame reordering). As a consequence these techniques can recover structure only up to a projective transformation that does not preserve the Euclidian geometry of the object. While in principle the 3-D geometry can be extracted from these solutions, this entails a very challenging non-linear optimization.

These results were illustrated with an example involving synthetic trajectories of an object used as a benchmark in the computer graphics community, showing virtually perfect reconstruction. Research is currently underway seeking to extend these results to multiple, not necessarily rigid objects.

[1]Additional experiments, omitted for space reasons, can be obtained by contacting the authors.

References

[1] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge Univeristy Press, 2003.

[2] D. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, 2003.

[3] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9(2):137–154, November 1992.

[4] T. Morita and T. Kanade. A paraperspective factorization method for recovering shape and motion from image sequences. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(8):858–867, August 1997.

[5] B. Fransen, O. I. Camps, and M. Sznaier. Robust structure from motion and identified dynamics. In *International Conference on Computer Vision*, pages 772–777, 2005.

[6] L. Torresani, D. Yang, G. Alexander, and C. Bregler. Tracking and modelling non-rigid objects with rank constraints. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 493–500, Kaui, Hawaii, December 2001.

[7] H. Zhou and T. S. Huang. Recovering articulated motion with a hierarchical factorization method. In 5^{th} *International Workshop on Gesture and Sign Language based Human–Computer Interaction*, Genoa, April 2003.

[8] R. Lublinerman, M. Sznaier, and O. I. Camps. Dynamics based robust motion segmentation. In *IEEE Computer Vision and Pattern Recognition*, pages 1176–1184, 2006.

[9] J. Costeira and T. Kanade. A multibody factorization method for independently moving objects. *International Journal of Computer Vision*, 29(3):159–179, September 1998.

[10] C. Gear. Multibody grouping from motion images. *International Journal of Computer Vision*, 29(3):133–152, August 1998.

[11] M. Han and T. Kanade. Multiple motion scene reconstruction with uncalibrated cameras. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25(7):884–894, July 2003.

[12] K. Kanatani. Motion segmentation by subspace separation: Model selection and reliability evaluation. *International Journal of Image and Graphics*, 2(2):179–197, April 2002.

[13] P. H. S. Torr. Geometric motion segmentation and model selection. *Phil. Trans. Royal Society of London*, 356(1740):1321–1340, 1998.

[14] L. Zelnik-Manor and M. Irani. Degeneracies, dependencies and their implications in multi-body and multi-sequence factorization. In *IEEE Computer Vision and Pattern Recognition*, pages 287–293, 2003.

[15] Jing Xiao, Jinxiang Chai, and Takeo Kanade. A closed-form solution to non-rigid shape and motion recovery. In *The 8th European Conference on Computer Vision (ECCV 2004)*, May 2004.

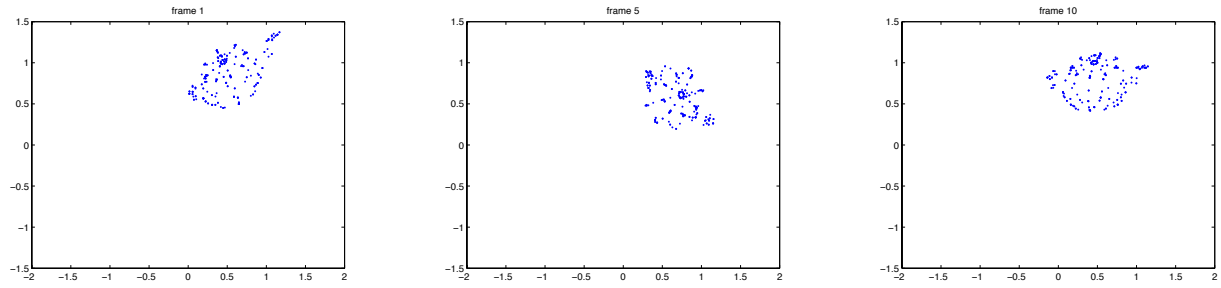
[16] R. Vidal, Y. Ma, S. Soatto, and S. Sastry. Two–view multibody structure from motion. *International Journal of Computer Vision*, 68(1):7–25, 2006.

[17] R. Hartley and R. Vidal. The multipbody trifocal tensor: Motion segmentation from three perspective views. In *IEEE Computer Vision and Pattern Recognition*, volume 1, pages 769–775, 2004.

[18] Y. Wu, Z. Zhang, T. S. Huang, and J. Lin. Multibody grouping via orthogonal subspace decomposition. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume II, pages 252–257, Kaui, Hawaii, December 2001.

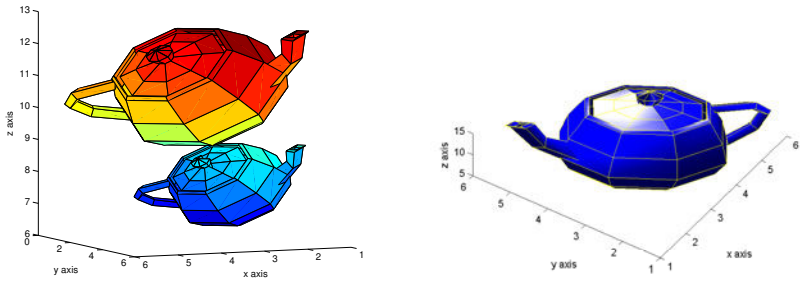
[19] K. Schindler, J.U., and H. Wang. Perspective n-view multibody structure-and-motion through model selection. In *9th European Conference on Computer Vision*, 2006.

[20] Conrad J. Poelman and Takeo Kanade. A paraperspective factorization method for shape and motion recovery. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(3):206–218, 1997.



(a) (b) (c)

Fig. 1. First (a), fifth (b) and tenth (c) frames of Utah teapot set



(a) (b)

Fig. 2. (a) Original and reconstructed teapot at Frame 1. (b) Scaled reconstruction superimposed on the ground truth.

[21] Peter Sturm and Bill Triggs. A factorization based algorithm for multi-image projective structure and motion. In B. Buxton and Roberto Cipolla, editors, *Proceedings of the 4th European Conference on Computer Vision, Cambridge, England*, volume 1065 of *Lecture Notes in Computer Science*, pages 709–720. Sprin-ger-Ver-lag, April 1996.

[22] J. Oliensis and R. Hartley. Iterative extensions of the sturm/triggs algorithm: Convergence and nonconvergence. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29(12):2217–2233, 2007.

[23] M. Fazel, H. Hindi, and S. P. Boyd. Log-det heuristic for matrix rank minimization with applications to hankel and euclidean distance matrices. In *Proceedings of American Control Conf. 2003*, volume 3, pages 2156–2162. AACC, 2003.

[24] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981.

[25] T. Kailath. *Linear Systems*. Prentice Hall, 1980.

APPENDIX

The proof consists of three parts:

(i) Building a pointwise rigid operator $\mathcal{L}$ such that its impulse response interpolates the trajectories of the differences $\mathbf{y}_k^{i,\alpha_{ki}} \doteq (\mathsf{P}_{ki} - \alpha_{ki}\mathsf{P}_{k3})$. For reasons that will become apparent later, we will seek an operator with two inputs such that its output at time k in response to an impulse applied at the i^{th} input at time $k = 0$, is precisely $\mathbf{y}_k^{i,\alpha_{ki}}$.

(ii) Finding a controllable (but not necessarily minimal) realization of $\mathcal{L}$ and the associated observability Grammian W_t^o.

(iii) Showing that, since the realization above is controllable, the rank of $\mathcal{L}$ is given by the rank of W_t^o and that this rank is minimized for $\alpha_{ik} \equiv 1$.

Begin by assuming, without loss of generality[2], that the Markov parameters of the operator $\mathcal{L}$ and the trajectory of O_k, the "center" of the motion associated with $\mathcal{L}$, satisfy an arma model of the form

$$\begin{aligned} \mathsf{L}_t &= \textstyle\sum_{i=1}^{n_L} \mathsf{A}_i^L \mathsf{L}_{t-i}, \\ \mathsf{O}_t &= \textstyle\sum_{i=1}^{n_O} \mathsf{A}_i^O \mathsf{O}_{t-i}, \qquad \mathsf{A}_i^L, \mathsf{A}_i^O \in R^{3\times 3} \end{aligned} \tag{6}$$

Let $\mathbf{x}_t^i \doteq P_{ti} - O_t$. From the above, it follows that the trajectories $\mathbf{x}_k^i$ also satisfy a model of the form

$$\mathsf{x}_t^i = \sum_{j=1}^{n_L} \mathsf{A}_j \mathsf{x}_{t-j}, \tag{7}$$

or, in compact form:

$$\begin{aligned} \xi_{t+1}^i &= \mathcal{A}_L \xi_t^i, \\ \omega_{t+1} &= \mathcal{A}_O \omega_t \end{aligned} \tag{8}$$

where

$$\mathcal{A}_L \doteq \begin{bmatrix} \mathsf{A}_1^L & \mathsf{A}_2^L & \dots & \mathsf{A}_{n_L-1}^L & \mathsf{A}_{n_L}^L \\ \mathsf{I} & 0 & \dots & \dots & 0 \\ 0 & \mathsf{I} & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & \mathsf{I} & 0 \end{bmatrix} \quad \xi_t^i \doteq \begin{bmatrix} \mathsf{x}_{t-1}^i \\ \mathsf{x}_{t-2}^i \\ \vdots \\ \mathsf{x}_{t-n_L}^i \end{bmatrix}$$

and a similar definition holds for ω_i, $\mathcal{A}^O$, involving A_i^O and the past values O_{t-i}. Thus, it follows that the trajectories of

[2]Note that, since we are working over finite horizons, any trajectory L_k can be interpolated with an arma model of sufficiently high order.

the vectors $\mathbf{y}^{i,\alpha}$ are given by the impulse response of the following state space system

$$\begin{array}{rcl} \zeta_{t+1} & = & \mathcal{A}\zeta_t + \mathcal{B}u; \;\; u \in R^2 \\ y_t & = & \mathcal{C}_t\zeta_t \end{array} \tag{9}$$

where

$$\mathcal{A} = \begin{bmatrix} \mathcal{A}_L & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathcal{A}_L & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathcal{A}_O & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathcal{A}_L & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathcal{A}_L & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathcal{A}_O \end{bmatrix} \mathcal{B} = \begin{bmatrix} \xi_o^1 & 0 \\ \xi_o^3 & 0 \\ \omega_o & 0 \\ 0 & \xi_o^2 \\ 0 & \xi_o^3 \\ 0 & \omega_o \end{bmatrix}$$

$$\mathcal{C}_t = \begin{bmatrix} \mathcal{C}_L & -\alpha_{t1}\mathcal{C}_L & (1-\alpha_{t1})\mathcal{C}_O & \mathcal{C}_L & -\alpha_{t2}\mathcal{C}_L & (1-\alpha_{t2})\mathcal{C}_O \end{bmatrix}$$

$$\mathcal{C}_L = \begin{bmatrix} \mathsf{I} & 0 \dots 0 \end{bmatrix}, \; \mathcal{C}_O = \begin{bmatrix} \mathsf{I} & 0 \dots 0 \end{bmatrix} \tag{10}$$

Note that the system (9) is *time–varying*, due to the presence of α_{ti} in $\mathcal{C}$. From a PBH argument (see [25], page 366) it can be shown that the pair $(\mathcal{A}, \mathcal{B})$ in (10) is generically controllable (except possibly in situations where for all i the vectors $P_{ki} - O_k$ are orthogonal to one eigenvalue of A_k for $k = 0, 1, \dots, n_L$). Assessing observability of the pair $(\mathcal{C}_t, \mathcal{A})$ requires considering the observability Grammian W_t^o (see for instance [25], Chapter 9). It can be easily shown that in this case, W_t^o is given by

$$\mathsf{W}_t^o = \sum_{j=1}^{t} \left(\mathcal{A}^{(t-j)}\right)^T \mathcal{C}_{j-1}^T \mathcal{C}_{j-1} \mathcal{A}^{(t-j)} = \left(\mathcal{K}_t\right)^T \mathcal{K}_t \tag{11}$$

where

$$\mathcal{K}_t = \begin{bmatrix} \mathcal{C}_{t-1} \\ \mathcal{C}_{t-2}\mathcal{A} \\ \vdots \\ \mathcal{C}_o\mathcal{A}^{t-1} \end{bmatrix} = \tag{12}$$

Finally, using the explicit expressions for $\mathcal{A}$ and $\mathcal{C}$ yields, for each block-row of $\mathcal{K}_t$:

$$\left(\mathcal{K}_t\right)_j = \begin{bmatrix} \left(K_{obs}^L\right)_j & -\alpha_{(t-j)1}\left(K_{obs}^L\right)_j & (1-\alpha_{(t-j)1})\left(K_{obs}^O\right)_j \\ \left(K_{obs}^L\right)_j & -\alpha_{(t-j)2}\left(K_{obs}^L\right)_j & (1-\alpha_{(t-j)2})\left(K_{obs}^O\right)_j \end{bmatrix}$$

where $(\mathsf{M})_j$ denotes the j^{th} block–row of a matrix M, and K_{obs}^L, K_{obs}^O denote the observability matrices of the pairs $(\mathcal{C}_L, \mathcal{A}_L)$ and $(\mathcal{C}_O, \mathcal{A}_O)$, respectively. Since by construction both of these realizations are observable, it follows that, if the motion of the center O_k has at least one mode not contained in the operator $\mathcal{L}$ (the relative motion of the rigid with respect to O) then:

$$\operatorname{rank}\{\mathcal{K}_t\} = \begin{cases} n_L & \text{if } \alpha_{ti} = 1 \\ n_L + n', \, n' \geq 1 & \text{if } 0 < \epsilon \leq \alpha_{ti}, \alpha_{ti} \neq 1 \end{cases}$$

Hence, the minimum rank solution (over the class of LTV systems considered here) corresponds to the LTI case where $\alpha_{ti} \equiv 1$. Further, a simple computation shows that in this case, $\mathcal{C}_k\mathcal{A}^k\mathcal{B} = [\mathbf{y}_k^{1,\alpha_{k1}} \;\; \mathbf{y}_k^{2,\alpha_{k2}}]$. It follows then, that for $\alpha_{ki} \equiv 1$, the order of a minimal realization of the operator $\mathcal{L}$ is given precisely by the rank of $\mathsf{H}_y = \begin{bmatrix} \mathsf{H}_{\mathbf{y}^1} & \mathsf{H}_{\mathbf{y}^2} \end{bmatrix}$.

Proof of Corollary 1.

Assume by contradiction that the minimum above is achieved by some sequence $\tilde{\alpha}_{ki}$. Let $L(\tilde{\alpha}_{ki})$ denote the associated LTI operator. From the theorem above, it follows that

$$\begin{array}{rcl} \operatorname{rank}\{\mathsf{H}(\alpha_{ki})\}|_{\alpha_{ki}\equiv 1} & = & \operatorname{rank}\{\mathcal{L}(\alpha_{ki})\}|_{\alpha_{ki}\equiv 1} \\ & < & \operatorname{rank}\{\mathcal{L}(\tilde{\alpha}_{ki})\} = \operatorname{rank}\{\mathsf{H}(\tilde{\alpha}_{ki})\} \end{array}$$

which contradicts the hypothesis that $\tilde{\alpha}_{ki}$ was the minimizing solution.