Multi-Subspace Methods for Motion Segmentation from Affine, Perspective and Central Panoramic Cameras*

René Vidal
Center for Imaging Science
Department of Biomedical Engineering, Johns Hopkins University
308B Clark Hall, 3400 N. Charles St., Baltimore MD 21218, USA
rvidal@cis.jhu.edu

Abstract— **Many robot navigation tasks require the computation of the motion of multiple objects moving in 3-D space from a collection of images taken by a moving robot. In this paper we present a unifying theoretical framework for both infinitesimal and discrete 3-D motion segmentation from optical flow or point correspondences in multiple affine, perspective or central panoramic views. We exploit the fact that for these motion and camera models, the image measurements associated with a single object live in a low dimensional subspace of a high dimensional space, hence motion segmentation is achieved by segmenting data living in multiple subspaces. We solve this problem in closed form using a polynomial fitting and differentiation technique called Generalized Principal Component Analysis. Unlike previous work, our method does not restrict the motion of the objects to be full dimensional or fully independent. Instead, our approach deals gracefully with all the spectrum of possible motions: from low dimensional and partially dependent to full dimensional and fully independent. We test our algorithm on various real and synthetic dynamic scenes with linear motions, planar and full motions, transparent motions, etc.**

Index Terms— **Motion segmentation, subspace clustering, Generalized Principal Component Analysis.**

I. Introduction

Computer vision is a very important sensor for many mobile robot applications, such as autonomous navigation, localization, formation control, pursuit evasion games, etc. A problem that is fundamental to most of these applications is multibody motion estimation and segmentation, which refers to the problem of estimating the number of independently moving objects in the scene; the motion of each one of the objects relative to the camera; the camera motion; and the segmentation of the image measurements according to their associated motion.

The case in which the camera is static has been traditionally tackled using background substraction techniques. However, such techniques are not suitable for the aforementioned applications, where the camera is mounted on a moving robot. The case in which the camera is moving is a very challenging problem in motion analysis, because it requires the simultaneous estimation of an unknown number of motion models, without knowing which pixels move according to the same model.

Because of theses difficulties, early studies concentrated on simplified motion models such as multiple points moving linearly with constant speed [4], [11], multiple points moving in a plane [12], and reconstruction of multiple translating planes [20]. The case of multiple moving objects seen by two perspective views was recently studied in [21], [17], [18], where a generalization of the 8-point algorithm based on the so-called multibody epipolar constraint and its associated multibody fundamental matrix was proposed. The method simultaneously recovers multiple fundamental matrices using multivariate polynomial factorization, and can be extended to most two-view motion models in computer vision, such as affine, translational and planar homographies, by fitting and differentiating complex polynomials [14]. Extensions of these two-view algorithms that deal with noisy data can be found in [18]. The case of multiple moving objects seen by three perspective views has also been recently solved by exploiting the algebraic and geometric properties of the multibody trifocal tensor [5]. To the best of our knowledge, there is no work on 3-D motion segmentation from multiple (more than 3) perspective views. The case of multiple views has only been studied in the case of affine cameras. For instance, the works of [1], [3] demonstrated that when the motion of the objects are independent and fully dimensional, motion segmentation can be achieved by thresholding the entries of a certain matrix built from point correspondences in multiple affine views. A similar technique can be applied in the case of optical flow in multiple frames, both in the case of perspective cameras [19] and central panoramic cameras [10]. Unfortunately, these methods are very sensitive to noise [7], [22] and fail with degenerate or partially dependent motions [23], [8].

In this paper we present a unifying theoretical framework for both infinitesimal and discrete 3-D motion segmentation from optical flow or point correspondences in *multiple* affine, perspective or central panoramic views. We exploit the fact that for these motion and camera models, the image measurements associated with a single object live in a low dimensional subspace of a high dimensional space, hence motion segmentation is achieved by segmenting data living in multiple subspaces. We solve this problem in closed form using a polynomial fitting and differentiation technique for subspace clustering called Generalized Principal Component Analysis. Unlike previous work, our

*This work has been supported with Johns Hopkins Whiting School of Engineering startup funds.

method does not restrict the motion of the objects to be full dimensional or fully independent. Instead, our approach deals gracefully with all the spectrum of possible motions: from low dimensional and partially dependent to full dimensional and fully independent. We test our algorithm on various real and synthetic dynamic scenes with linear, planar, and full motions, transparent motions, etc.

II. Motion Subspaces for Affine, Perspective and Central Panoramic Cameras

In this section we describe classes of camera models whose associated image measurements (discrete or differential) live in a low dimensional subspace of a high dimensional space.

A. Discrete Motion in Multiple Affine Views

Let $\boldsymbol{x}_{fp} \in \mathbb{R}^2$ be the image of a point $\boldsymbol{X}_p \in \mathbb{P}^3$, $p = 1, \ldots, P$, in frame $f = 1, \ldots, F$. Let $(R_f, T_f) \in SO(3)$ be the pose of one of the moving objects relative to the camera in frame f. Under the affine camera model the image point $\boldsymbol{x}_{fp}$ is obtained by projecting the 3-D point $[R_f\ T_f]\boldsymbol{X}_p$ orthographically. That is, the image point is given by $\boldsymbol{x}_{fp} = A_f \boldsymbol{X}_p$, where $A_f = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} [R_f\ T_f] \in \mathbb{R}^{2\times 4}$ is the so-called *affine camera matrix.*

Therefore, if we are given a set of P point correspondences $\{\boldsymbol{x}_{fp}\}$ in F frames, we can stack all the image measurements into a $2F \times P$ matrix W that satisfies

$$\begin{aligned} W \quad &= \quad MS^T \\ \begin{bmatrix} \boldsymbol{x}_{11} \cdots \boldsymbol{x}_{1P} \\ \vdots \qquad \vdots \\ \boldsymbol{x}_{F1} \cdots \boldsymbol{x}_{FP} \end{bmatrix}_{2F\times P} &= \begin{bmatrix} A_1 \\ \vdots \\ A_F \end{bmatrix}_{2F\times 4} \begin{bmatrix} \boldsymbol{X}_1 \cdots \boldsymbol{X}_P \end{bmatrix}_{4\times P}, \end{aligned} \tag{1}$$

where M is called the *motion matrix* and S is called the *structure matrix*. Since the first two rows of each A_f are rows of a rotation matrix, the above equation implies that $2 \leq \text{rank}(W) \leq 4$. Therefore, the image trajectories of a 3-D point associated with one of the moving objects live in a subspace of $\mathbb{R}^{2F}$ of dimension two, three or four. This rank constraint was derived in [13], and was used to propose the first multi-frame algorithm for estimating the motion of an affine camera observing a static scene.

B. Differential Motion in Multiple Perspective Views

Let $\Omega_f = (\omega_{1f}, \omega_{2f}, \omega_{3f})^T$ and $V_f = (v_{1f}, v_{2f}, v_{3f})^T$ be, respectively, the rotational and translational velocities of one of the moving objects relative to the camera at frame $f = 1, \ldots, F$. Under the perspective projection model, the projection of point $\boldsymbol{X}_p = (X_p, Y_p, Z_p, 1)^T \in \mathbb{P}^3$ on the zeroth frame is $(x_p, y_p)^T = (X_p, Y_p)^T / Z_p$, and its optical flow $\boldsymbol{u}_{fp} \in \mathbb{R}^2$ in the fth frame is:

$$\boldsymbol{u}_{fp} = \begin{bmatrix} x_p y_p & -(1+x_p^2) & -y_p & 1/Z_p & 0 & x_p/Z_p \\ (1+y_p^2) & -x_p y_p & x_p & 0 & 1/Z_p & y_p/Z_p \end{bmatrix} \begin{bmatrix} \Omega_f \\ V_f \end{bmatrix}.$$

Given measurements for the optical flow $\{\boldsymbol{u}_{fp}\}$ of P pixels in F frames, we can stack all the image measurements into a $2F \times P$ matrix W

$$W = \begin{bmatrix} \boldsymbol{u}_{11} & \cdots & \boldsymbol{u}_{1P} \\ \vdots & & \vdots \\ \boldsymbol{u}_{F1} & \cdots & \boldsymbol{u}_{FP} \end{bmatrix}_{2F\times P} \tag{2}$$

that can be factored into its motion and structure components as $W = MS^T$, where

$$M = \begin{bmatrix} \omega_{11} & \omega_{21} & -\omega_{31} & 0 & 0 & v_{11} & v_{31} & 0 \\ -\omega_{21} & 0 & 0 & \omega_{11} & \omega_{31} & v_{21} & 0 & v_{31} \\ \vdots & & & & & & & \vdots \\ \omega_{1F} & \omega_{2F} & -\omega_{3F} & 0 & 0 & v_{1F} & v_{3F} & 0 \\ -\omega_{2F} & 0 & 0 & \omega_{1F} & \omega_{3F} & v_{2F} & 0 & v_{3F} \end{bmatrix}_{2F\times 8}$$

$$S = \begin{bmatrix} x_1 y_1 & z_1 - x_1^2 & y_1 & y_1^2 - z_1 & x_1 & \frac{1}{\lambda_1} & \frac{x_1}{\lambda_1} & \frac{y_1}{\lambda_1} \\ \vdots & & & & & & & \vdots \\ x_P y_P & z_P - x_P^2 & y_P & y_P^2 - z_P & x_P & \frac{1}{\lambda_P} & \frac{x_P}{\lambda_P} & \frac{y_P}{\lambda_P} \end{bmatrix}_{P\times 8},$$

Therefore, $\text{rank}(W) \leq 8$, hence the vector containing the optical flow of a point between the zeroth and the fth frame for $f = 1, \ldots, F$ lives in a subspace of $\mathbb{R}^{2F}$ of dimension at most 8. This rank constraint, among others, was derived in [6] and was used to derive a multi-frame algorithm for the estimation of the optical flow of a moving camera observing a static scene.

C. Differential Motion and Central Panoramic Cameras

A central panoramic camera is a combination of a curved mirror and a lens with a unique focal point. For a calibrated camera with parameter $\xi \in [0, 1]$, the projection of a point $\boldsymbol{X}_p = (X_p, Y_p, Z_p, 1)^T \in \mathbb{P}^3$ is given by $(x_p, y_p)^T = (X_p, Y_p)^T / \lambda_p$, where $\lambda_p = -Z_p + \xi\sqrt{X_p^2 + Y_p^2 + Z_p^2}$.

As before, let Ω_f and V_f be, respectively, the rotational and translational velocities of one of the moving objects relative to the camera at frame f. We showed in [9] that for a central panoramic camera the optical flow $\boldsymbol{u}_{fp} \in \mathbb{R}^2$ of point $\boldsymbol{X}_p = (X_p, Y_p, Z_p, 1)^T \in \mathbb{P}^3$ is given by:

$$\boldsymbol{u}_{fp} = \begin{bmatrix} x_p y_p & z_p - x_p^2 & -y_p & \frac{1-\rho_p x_p^2}{\lambda_p} & -\frac{\rho_p x_p y_p}{\lambda_p} & \frac{(1-\rho_p z_p) x_p}{\lambda_p} \\ y_p^2 - z_p & -x_p y_p & x_p & -\frac{\rho_p x_p y_p}{\lambda_p} & \frac{1-\rho_p y_p^2}{\lambda_p} & \frac{(1-\rho_p z_p) y_p}{\lambda_p} \end{bmatrix} \begin{bmatrix} \Omega_f \\ V_f \end{bmatrix},$$

where $\rho_p = \xi^2/(1 + z_p)$ and

$$z_p = \frac{-1 + \xi^2(x_p^2 + y_p^2)}{1 + \xi\sqrt{1 + (1 - \xi^2)(x_p^2 + y_p^2)}}. \tag{3}$$

Given measurements for the optical flow $\{\boldsymbol{u}_{fp}\}$ of P pixels in F frames, we can stack all the image measurements into a $2F \times P$ matrix W

$$W = \begin{bmatrix} \boldsymbol{u}_{11} & \cdots & \boldsymbol{u}_{1P} \\ \vdots & & \vdots \\ \boldsymbol{u}_{F1} & \cdots & \boldsymbol{u}_{FP} \end{bmatrix}_{2F\times P} \tag{4}$$

that can be factored into its motion and structure components as $W = MS^T$, where

$$M = \begin{bmatrix} \omega_{11} & \omega_{21} & -\omega_{31} & 0 & 0 & v_{11} & -v_{21} & v_{31} & 0 & 0 \\ -\omega_{21} & 0 & 0 & \omega_{11} & \omega_{31} & 0 & -v_{11} & 0 & v_{21} & v_{31} \\ \vdots & & & & & & & & & \vdots \\ \omega_{1F} & \omega_{2F} & -\omega_{3F} & 0 & 0 & v_{1F} & -v_{2F} & v_{3F} & 0 & 0 \\ -\omega_{2F} & 0 & 0 & \omega_{1F} & \omega_{3F} & 0 & -v_{1F} & 0 & v_{2F} & v_{3F} \end{bmatrix}_{2F\times 10}$$

$$S = \begin{bmatrix} x_1y_1 & z_1-x_1^2 & y_1 & y_1^2-z_1 & x_1 & \frac{1-\rho_1 x^2}{\lambda_1} & \frac{\rho_1 x_1 y_1}{\lambda_1} & \frac{(1-\rho_1 z_1)x_1}{\lambda_1} & \frac{(1-\rho_1 y_1^2)}{\lambda_1} & \frac{(1-\rho_1 z_1)y_1}{\lambda_1} \\ \vdots & & & & & & & & & \vdots \\ x_Py_P & z_P-x_P^2 & y_P & y_P^2-z_P & x_P & \frac{1-\rho_P x_P^2}{\lambda_P} & \frac{\rho_P x_P y_P}{\lambda_P} & \frac{(1-\rho_P z_P)x_P}{\lambda_P} & \frac{(1-\rho_P y_P^2)}{\lambda_P} & \frac{(1-\rho_P z_P)y_P}{\lambda_P} \end{bmatrix}_{P\times 10}.$$

Therefore, rank$(W) \leq 10$, hence the vector containing the optical flow of a point between the zeroth and the fth frame for $f = 1, \ldots, F$ lives in a subspace of $\mathbb{R}^{2F}$ of dimension at most 10.

III. Segmentation of the Motion Subspaces

Assume we are given a matrix $W \in \mathbb{R}^{2F\times P}$ containing P image measurements (either point correspondences or optical flow) in F frames. If the image measurements are generated by a single moving object, then the columns of W span a subspace of $\mathbb{R}^{2F}$ of dimension at most 4, 8 or 10, depending on whether the camera is affine, perspective, or central panoramic.

Now, if the image measurements are generated by n independently moving objects, then the columns of W must live in a collection of n subspaces $\{S_i \subset \mathbb{R}^{2F}\}_{i=1}^n$. If the motion subspaces are fully dimensional, i.e. $\dim(S_i) = d = 4, 8$ or 10, and fully independent, i.e. $\dim(S_i \cup S_j) = \dim(S_i) + \dim(S_j)$ or equivalently $S_i \cap S_j = \{0\}$, then we must have rank$(W) = nd$, provided that $2F \geq nd$ and $P \geq nd$. This is because rank(W) = rank$([W_1 \cdots W_n])$, where W_i is the matrix of image measurements associated with the ith motion subspace and satisfies $W_i = M_i S_i^T$, so that

$$[W_1 \cdots W_n] = [M_1 \cdots M_n] \begin{bmatrix} S_1^T & & 0 \\ & \ddots & \\ 0 & & S_n^T \end{bmatrix}. \qquad (5)$$

The rank constraint rank$(W) = nd$ allows us to determine the number of independent motions directly from the measurements as

$$n = \frac{\text{rank}(W)}{d} \qquad (6)$$

In many applications, however, the motions need not be fully dimensional. In ground robot navigation, for example, the motion of each robot relative to the camera is constrained to be planar, which reduces the dimension of the motion subspaces to $d = 4$, $d = 4$ and $d = 5$ for affine, perspective and cental panoramic cameras, respectively. In addition, the motion subspaces may be partially dependent, i.e. $\max\{\dim(S_i), \dim(S_j)\} < \dim(S_i \cup S_j) < \dim(S_i) + \dim(S_j)$ or equivalently $S_i \cap S_j \neq \{0\}$, $S_i \cap S_j \neq S_i$ and $S_i \cap S_j \neq S_j$, which happens for instance when two objects move with the same rotation but different translation relative to the camera.

In order to deal both with fully independent and partially dependent motions, as well as full dimensional and degenerate motions, we need a subspace clustering algorithm that assumes only that the motion subspaces are different, i.e. $S_i \neq S_j$ for all $i \neq j = 1, \ldots, n$.[1] We solve this problem by fitting and differentiating polynomials using the Generalized Principal Component Analysis (GPCA) algorithm [16], [15], which consists of the following two steps:

1) Project the image measurements onto a $(d+1)$-dimensional subspace of $\mathbb{R}^{2F}$.
2) Estimate all the motion subspaces by fitting a homogeneous polynomial to the projected data. Segment the motion subspaces by taking the derivatives of this polynomial.

We discuss these steps in the following two sections.

A. Projection onto a Low Dimensional Subspace

The first step of the algorithm is dimensionality reduction, which we do via a linear projection onto a $(d+1)$-dimensional subspace of $\mathbb{R}^{2F}$. We use a linear projection, because the clustering of data lying on a collection of subspaces is preserved by a generic linear projection. For instance, if one is given data lying on two lines in $\mathbb{R}^3$ passing through the origin, then one can first project the two lines onto a plane in general position[2] and then cluster the data inside that plane. More generally the principle is [15]:

Theorem 1 (Cluster-Preserving Projections): If a set of vectors $\{\boldsymbol{x}^j\}$ all lie in n linear subspace of dimensions $\{d_i\}_{i=1}^n$ in $\mathbb{R}^D$, and if $\pi_{\mathcal{P}}$ represents a linear projection onto a subspace $\mathcal{P}$ of dimension D', then the points $\{\pi_{\mathcal{P}}(\boldsymbol{x}^j)\}$ lie in at most n linear subspaces of $\mathcal{P}$ of dimensions $\{d_i' \leq d_i\}_{i=1}^n$. Furthermore, if $D > D' > \max\{d_i\}$, then there is an open and dense set of projections that preserve the separation and dimensions of the subspaces.

Since in the case of motion subspaces the maximum dimensions are $d = 4$, 8 and 10, depending on the camera models, we can project the columns of W onto a subspace of dimension $(d+1) = 5, 9$ or 11. In choosing a projection, it makes sense to lose as little information as possible by

[1] Notice that this is equivalent to requiring that $\dim(S_i \cup S_j) > \max\{\dim(S_i), \dim(S_j)\}$.

[2] A plane perpendicular to any of the lines or perpendicular to the plane containing the lines would fail.

projecting into a dominant eigensubspace, which we can do simply by computing the SVD of $W = UV$ with $U \in \mathbb{R}^{2F\times(d+1)}$ and $V \in \mathbb{R}^{(d+1)\times P}$. The matrix V contains the desired projected data.

B. Estimation and Segmentation of Motion Subspaces

With an abuse of notation, from now we will denote the matrix of projected data as $W \in \mathbb{R}^{(d+1)\times P}$ and the projected motion subspaces as $\{S_i\}_{i=1}^n$. Let $\boldsymbol{w} \in \mathbb{R}^{d+1}$ be any of the columns of W. Since $\boldsymbol{w}$ must belong to one of the projected subspaces, say S_i, then there exists a vector $\boldsymbol{b}_i \in \mathbb{R}^{d+1}$ normal to subspace S_i such that $\boldsymbol{b}_i^T\boldsymbol{w} = 0$. Let $\{\boldsymbol{b}_i\}_{i=1}^n$ be a collection of n different vectors in $\mathbb{R}^{d+1}$ with the property that $\boldsymbol{b}_i$ is orthogonal to S_i, but not orthogonal to S_j for $j \neq i = 1,\ldots,n$. Then any column $\boldsymbol{w}$ of W must satisfy the following homogeneous polynomial of degree n in $d+1$ variables

$$p_n(\boldsymbol{w}) = (\boldsymbol{b}_1^T\boldsymbol{w})(\boldsymbol{b}_2^T\boldsymbol{w})\cdots(\boldsymbol{b}_n^T\boldsymbol{w}) = 0. \tag{7}$$

This polynomial can be expressed linearly in terms of its coefficients. For instance, if $n = 2$ and $d+1 = 2$ we have $p_n(\boldsymbol{w}) = c_1w_1^2 + c_2w_1w_2 + c_3w_2^2$, and we can solve for the coefficients from the linear system

$$[c_1, c_2, c_3]\begin{bmatrix} w_{11}^2 & \cdots & w_{1P}^2 \\ w_{11}w_{21} & \cdots & w_{1P}w_{2P} \\ w_{21}^2 & \cdots & w_{2P}^2 \end{bmatrix} = 0. \tag{8}$$

For arbitrary n and d, the above equation reads $\boldsymbol{c}^TL_n = 0$, where the jth column of L_n is formed by stacking all the monomials of degree n generated from the entries of the jth column of W. There are $M_n = \binom{n+d}{d}$ such monomials, hence $\boldsymbol{c} \in \mathbb{R}^{M_n}$ and $L_n \in \mathbb{R}^{M_n\times P}$. When all the motion subspaces are fully dimensional the choice of one normal vector per subspace is unique. This implies that $\boldsymbol{c}$ is uniquely defined, and so there is only one vector in the null space of L_n. In this case, we compute $\boldsymbol{c}$ in a least squares sense as the left singular vector of L_n associated with its smallest singular value. When one or more motion subspaces are not fully dimensional, the choice of one normal vector per subspace is not unique, hence there will be many vectors $\boldsymbol{c}$ in the null space of L_n. In the unlikely event that the eigenvector of L_n associated with its smallest eigenvalue happens to correspond to a choice of normal vectors in which one of the normal vectors is orthogonal to two or more subspaces at the same time, we can take any other vector $\boldsymbol{c}$ in the null space of L_n.

Given $\boldsymbol{c}$, a direct calculation shows that if $\boldsymbol{w}$ corresponds to motion subspace S_i, then the derivative of p_n at $\boldsymbol{w}$ gives the normal vector $\boldsymbol{b}_i$ up to scale factor, i.e.

$$\boldsymbol{b}_i = \frac{Dp_n(\boldsymbol{w})}{\|Dp_n(\boldsymbol{w})\|}. \tag{9}$$

Therefore, if we choose n columns of W, $\{\boldsymbol{w}_i\}_{i=1}^n$, each one belonging to each one of the n motion subspaces, then we can immediately obtain the normal vectors as $\boldsymbol{b}_i \sim Dp_n(\boldsymbol{w}_i)$. We refer the reader to [15] for a simple method for choosing such points.

Given the normal vectors $\{\boldsymbol{b}_i\}$, we can immediately cluster the columns of W by assigning $\boldsymbol{w}_j$ to motion subspace i if

$$i = \arg\min_{\ell=1,\ldots,n}\{(\boldsymbol{b}_\ell^T\boldsymbol{w}_j)^2\}. \tag{10}$$

Therefore, the motion segmentation problem is solved by fitting a polynomial p_n to the columns of W and computing the derivatives of this polynomials to assign each column to its corresponding motion subspace.

IV. Experimental results

We tested our 3-D motion segmentation algorithm for affine cameras on the motion sequences shown in Figure 1, which contain sequences with missing data (Boat), full motions (Can-Book), linear and planar motions (3-Cars), perspective effects[3] and transparent motions (Tea-Tins). The point correspondences were computed using the algorithm in [2]. Table I shows the segmentation results for each one of the sequences. Notice that the sequences have rather different number of feature points and number of frames. However, in all the cases the algorithm gives a misclassification error of less than 5%.

TABLE I

Percentage of misclassified correspondences for different motion sequences.

Sequence	Points	Frames	Motions	Error
Boat	686	11	2	2.19%
Can-Book	170	3	2	1.18%
Tea-Tins	84	3	2	1.19%
3-Cars	173	15	3	4.62%

We also tested our 3-D motion segmentation algorithm for perspective cameras on two video sequences. Figure 2 shows the *street* sequence available at http://www.cs.otago.ac.nz/research/vision/Research/OpticalFlow/opticalflow.html#Sequences, which contains two independent motions: the car translating to the right, and the camera panning to the right. Figure 2(a) shows frames 3, 8, 12 and 16 of the sequence with the corresponding optical flow superimposed. The optical flow measurements were computed using algorithm, which is available at http://www.cs.brown.edu/people/black/ignc.html. Figures 3(b)-(c) show the segmentation results. In frame 4 the car is partially occluded, thus only the frontal part of the car is segmented from the background. The door is incorrectly segmented because it is in a region with low texture. As time proceeds, motion information is integrated over time by incorporating optical flow from many frames in the optical flow matrix, thus the door is correctly segmented. In frame 16 the car is fully visible and correctly segmented from the moving background. Figure 3(a) shows the *two-robot* sequence with the corresponding optical flow superimposed. Figures 3(b)

[3] Most of the chosen sequences present noticeable perspective effects, due to large depth variations, or forward motions. The affine camera model does not take into account such effects.

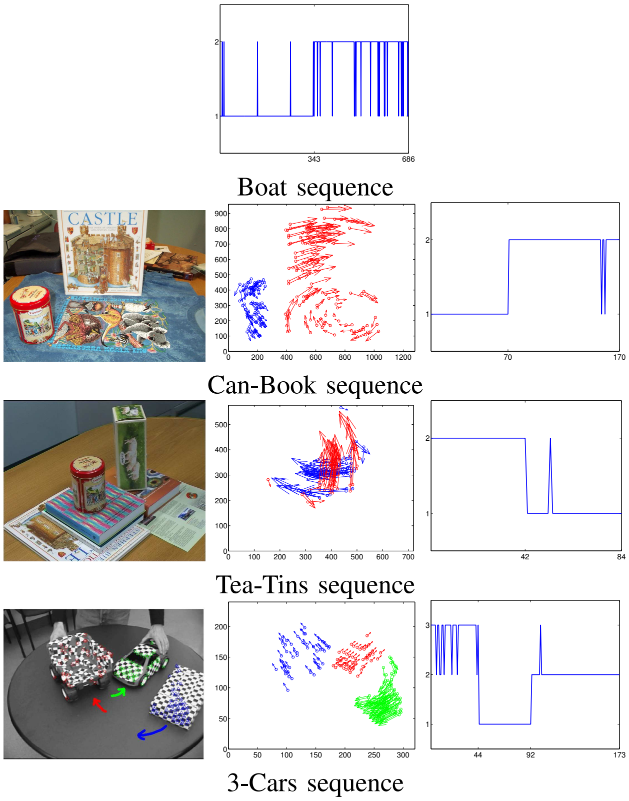


Fig. 1. Motion segmentation from point correspondences in multiple affine views for various sequences.

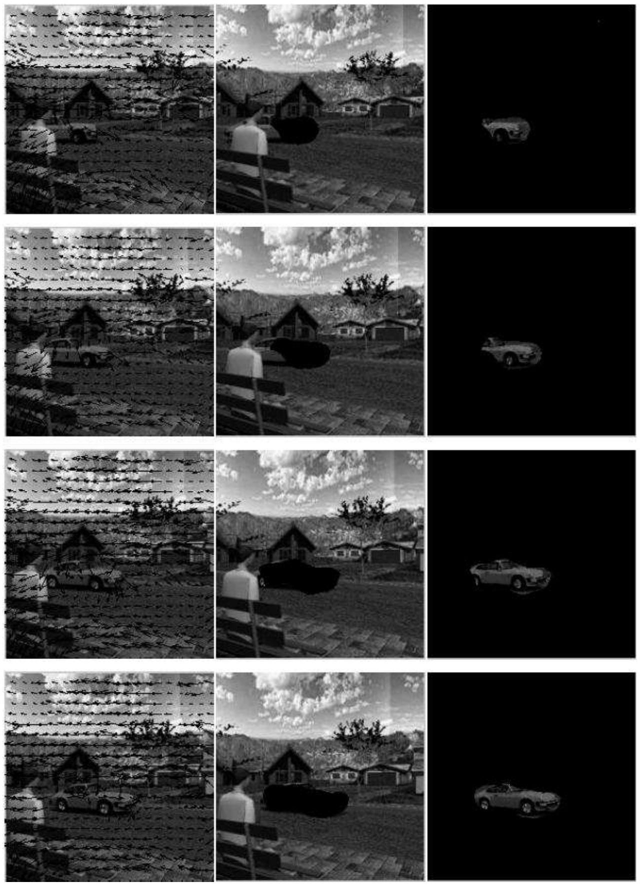


Fig. 2. Segmentation results for the *street* sequence. The sequence has 18 frames and 200 × 200 pixels. The camera is panning to the right while the car is also moving to the right. (a) Frames 3, 8 12 and 16 of the sequence with the corresponding optical flow superimposed. (b) Group 1: motion of the camera. (c) Group 2: motion of the car.

and 3(c) show the results of the segmentation. Groups 1 and 2 correspond to the each one of the moving objects, while group 3 corresponds to the background, which is the correct segmentation.

We also evaluated the performance of our algorithm in the case where two independently moving mobile robots are viewed by a static paracatadioptric camera ($\xi = 1$). We grabbed 18 images of size 240 × 240 pixels at a framerate of 5Hz. The optical flow was computed directly in the image plane using Black's algorithm available at http://www.cs.brown.edu/people/black/ignc.html. Figure 4 shows a sample of the motion segmentation based on the optical flow. On the left, the optical flow generated by the two moving robots is shown, and on the right is the segmentation of the pixels corresponding to the independent motions. The two moving robots are segmented very well from the static background.

V. Conclusions

This paper has presented a unifying theoretical framework for both infinitesimal and discrete 3-D motion segmentation from optical flow or point correspondences in multiple affine, perspective or central panoramic views. We exploited the fact that for various motion and camera models, the image measurements associated with a single object live in a low dimensional subspace of a high dimensional space, hence motion segmentation is achieved by segmenting data living in multiple subspaces. The segmentation of the motion subspaces was achieved by projecting the data onto a low dimensional subspace, fitting a polynomial to the projected data, and clustering the data points by evaluating the derivatives of this polynomials.

Open research avenues include making the proposed batch algorithm recursive, as well as extending it to projective reconstruction of multiple rigid-body motions from multiple perspective views.

Acknowledgment

The author thanks Professor Yi Ma and Professor Richard Hartley for insightful discussions about motion segmentation and subspace clustering.

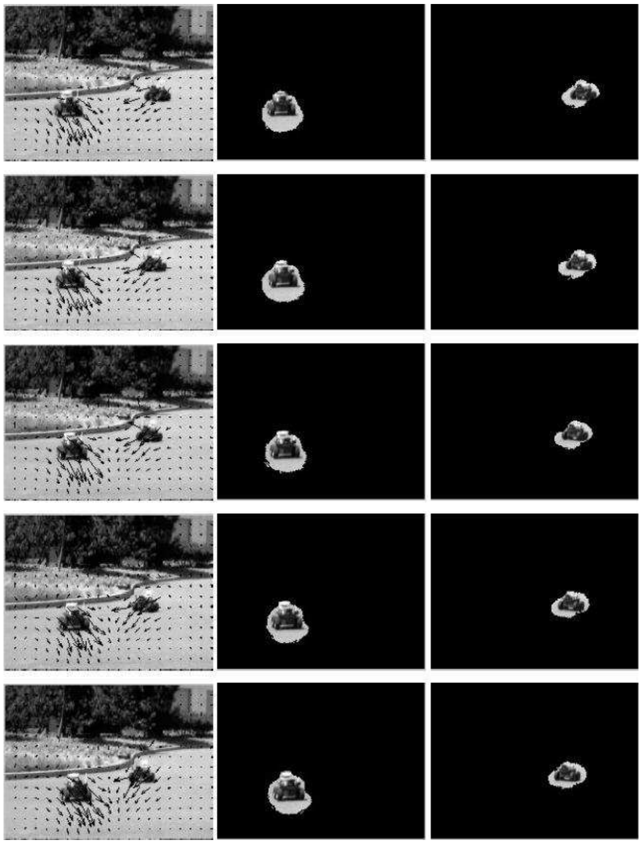
References

[1] T.E. Boult and L.G. Brown. Factorization-based segmentation of motions. In *Proc. of the IEEE Workshop on Motion Understanding*, pages 179–186, 1991.

[2] A. Chiuso, P. Favaro, H. Jin, and S. Soatto. Motion and structure causally integrated over time. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(4):523–535, 2002.

[3] J. Costeira and T. Kanade. Multi-body factorization methods for motion analysis. In *IEEE International Conference on Computer Vision*, pages 1071–1076, 1995.

[4] M. Han and T. Kanade. Reconstruction of a scene with multiple linearly moving objects. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 542–549, 2000.



(a) Optical flow (b) Group 1 (c) Group 2

Fig. 3. Segmentation results for the *two-robot* sequence. The sequence contains 6 frames and 200 × 150 pixels. (a) Frames 1-5 of the sequence with optical flow superimposed. (b) Group 1: one moving robot. (c) Group 2: the other moving robot.

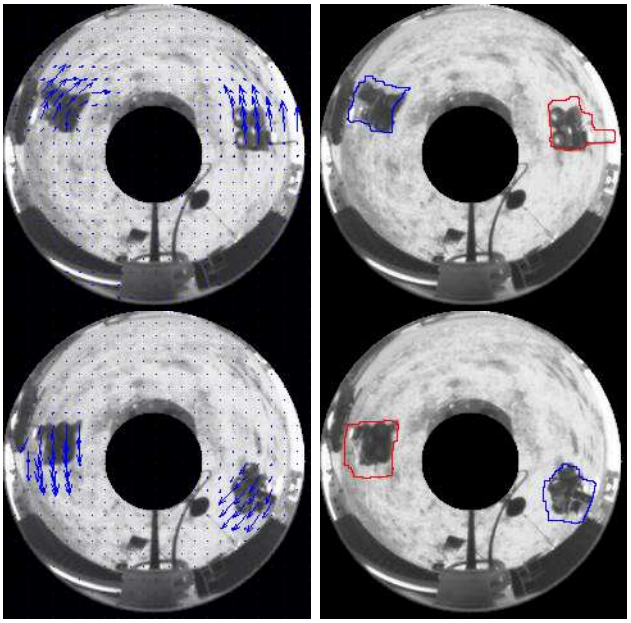


Fig. 4. Showing a an example of motion segmentation based on central-panoramic optical flow.

[5] R. Hartley and R. Vidal. The multibody trifocal tensor: Motion segmentation from 3 perspective views. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2004.

[6] M. Irani. Multi-frame optical flow estimation using subspace constraints. In *IEEE International Conference on Computer Vision*, pages 626–633, 1999.

[7] K. Kanatani. Motion segmentation by subspace separation and model selection. In *IEEE International Conference on Computer Vision*, volume 2, pages 586–591, 2001.

[8] K. Kanatani and Y. Sugaya. Multi-stage optimization for multi-body motion segmentation. In *Australia-Japan Advanced Workshop on Computer Vision*, pages 335–349, 2003.

[9] O. Shakernia, R. Vidal, and S. Sastry. Infinitesimal motion estimation from multiple central panoramic views. In *IEEE Workshop on Motion and Video Computing*, pages 229–234, 2002.

[10] O. Shakernia, R. Vidal, and S. Sastry. Multi-body motion estimation and segmentation from multiple central panoramic views. In *IEEE International Conference on Robotics and Automation*, 2003.

[11] A. Shashua and A. Levin. Multi-frame infinitesimal motion model for the reconstruction of (dynamic) scenes with multiple linearly moving objects. In *IEEE International Conference on Computer Vision*, volume 2, pages 592–599, 2001.

[12] P. Sturm. Structure and motion for dynamic scenes - the case of points moving in planes. In *European Conference on Computer Vision*, pages 867–882, 2002.

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

[14] R. Vidal and Y. Ma. A unified algebraic approach to 2-D and 3-D motion segmentation. In *European Conference on Computer Vision*, 2004.

[15] R. Vidal, Y. Ma, and J. Piazzi. A new GPCA algorithm for clustering subspaces by fitting, differentiating and dividing polynomials. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2004.

[16] R. Vidal, Y. Ma, and S. Sastry. Generalized principal component analysis (GPCA). In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 621–628, 2003.

[17] R. Vidal, Y. Ma, S. Soatto, and S. Sastry. Two-view multibody structure from motion. *International Journal of Computer Vision*, 2004.

[18] R. Vidal and S. Sastry. Optimal segmentation of dynamic scenes from two perspective views. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 281–286, 2003.

[19] R. Vidal, S. Soatto, and S. Sastry. A factorization method for multibody motion estimation and segmentation. In *Fortieth Annual Allerton Conference on Communication, Control and Computing*, pages 1625–1634, 2002.

[20] L. Wolf and A. Shashua. Affine 3-D reconstruction from two projective images of independently translating planes. In *IEEE International Conference on Computer Vision*, pages 238–244, 2001.

[21] L. Wolf and A. Shashua. Two-body segmentation from two perspective views. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 263–270, 2001.

[22] Y. Wu, Z. Zhang, T.S. Huang, and J.Y. Lin. Multibody grouping via orthogonal subspace decomposition. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 252–257, 2001.

[23] L. Zelnik-Manor and M. Irani. Degeneracies, dependencies and their implications in multi-body and multi-sequence factorization. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 287–293, 2003.