IEEE Workshop on Vision and Motion Computing, pp. 44-49, Orlando FL, December 2002

Segmentation of Dynamic Scenes from Image Intensities*

René Vidal Shankar Sastry
Department of EECS, UC Berkeley
301 Cory Hall, Berkeley CA 94720
{rvidal,sastry}@eecs.berkeley.edu

Abstract

We present an algebraic geometric approach for segmenting both static and dynamic scenes from image intensities. We introduce the multibody affine constraint as a geometric relationship between the motion of multiple objects and the image intensities generated by them. This constraint is satisfied by all the pixels, regardless of the body to which they belong and regardless of depth discontinuities or perspective effects. We propose a polynomial factorization technique that estimates the number of affine motion models as well as their motion parameters in polynomial time. The factorization technique is used to initialize a nonlinear algorithm that minimizes the algebraic error defined by the multibody affine constraint. Our approach is based solely on image intensities, hence it does not require feature tracking or correspondences. It is therefore a natural generalization of the so-called direct methods in single-body structure from motion to multiple moving objects. We present simulation and experimental results that validate our approach.

1. Introduction

Segmentation of dynamic scenes is an important and challenging problem in visual motion analysis. Most of its difficulty is due to the need for simultaneous estimation of the number of moving objects and their motions, as well as the motion of the camera (egomotion).

When the sequence is static, *i.e.* when either the camera or the 3D world undergo a single 3D motion, one can model the scene as a mixture of affine motion models. Even though a single 3D motion is present, multiple 2D motion models arise because of perspective effects and/or depth discontinuities. In this case the task of *2D motion segmentation* is that of estimating these models from the image data. Classical approaches to 2D motion segmentation try to separate the image flow into different regions either by looking for flow discontinuities [9], while imposing some regularity conditions [1], or by fitting a mixture of probabilistic models [6, 14, 10]. The latter is usually done using an iterative process that alternates between segmentation and motion estimates using the EM algorithm. Alternative approaches are based on local features that incorporate spatial and temporal motion information. Similar features are grouped together using, for example, normalized cuts [8] or the eigenvectors of a similarity matrix [15]. Once a 2D motion segmentation of the scene has been obtained, the affine motion models are estimated separately using, for example, the so-called *direct methods* [5].

When the scene is dynamic, *i.e.* when both the camera and multiple objects move, one can still model the scene as a mixture of affine motion models. Some of these models are due to independent 3D motions, *e.g.* when the motion of an object relative to the camera can be well approximated by the affine motion model. Others are due to perspective effects and/or depth discontinuities, *e.g.* when some of the 3D motions are broken into different 2D motions. The task of *3D motion segmentation* is to obtain a collection of 3D motion models, in spite of perspective effects and/or depth discontinuities. This can be done by first extracting a collection of 2D motion models and then merging regions that have a consistent 3D motion. The estimation of the 2D motion models is usually done in an ad-hoc fashion that first extracts the dominant affine motion using direct methods and then subdivides the misaligned regions by computing the next dominant motion [4]. Alternative feature-based methods infer the 3D motion models directly from a collection of feature points. Previous work in this area includes [3, 7] for points moving linearly with constant speed, [2] for objects seen by an orthographic camera in multiple views, and [13] for objects seen by a perspective camera in two views.

In this paper we propose an algebraic geometric approach to affine motion segmentation from image intensities. We show that one can estimate the number of affine motion models and their parameters *analytically*, with no need for feature tracking or correspondences. We introduce

*The authors thank Yi Ma for his comments on the final preparation of this manuscript. This work is supported by ONR grant N00014-00-1-0621.

the *multibody affine constraint* as a geometric relationship between the multiple motions and the image intensities generated by them. This constraint is satisfied by all the pixels, regardless of the body to which they belong and regardless of depth discontinuities or perspective effects. We derive a rank constraint on the image measurements from which one can estimate the number of motions n. Given n, one can linearly solve for the *multibody affine motion* after embedding all the image measurements in a higher-dimensional space. Individual affine motions are recovered from the multibody one using a novel polynomial factorization technique that gives a unique global solution to the motion segmentation problem. This technique is then used to initialize a nonlinear algorithm that minimizes the algebraic error defined by the multibody affine constraint. We present simulation and experimental results that validate our approach.

2. Preliminaries

Consider an dynamic scene with an unknown number of independently and rigidly moving objects. We assume that the motion of the sequence can be modeled as a mixture of an *unknown* number n of *affine motion models*. Each motion model is described by the equations

$$\begin{aligned} \mathrm{u}(x,y) &= a_{11}x + a_{12}y + a_{13} && (1)\\ \mathrm{v}(x,y) &= a_{21}x + a_{22}y + a_{23} && (2)\end{aligned}$$

where $\boldsymbol{u} = [\mathrm{u}, \mathrm{v}, 1]^T \in \mathbb{P}^2$ is the optical flow of pixel $\boldsymbol{x} = [x, y, 1]^T \in \mathbb{P}^2$ and $a_{11}, \ldots, a_{23}$ are the affine motion parameters.

If we assume that the surface of each object is Lambertian, then the optical flow of pixel $\boldsymbol{x}$ is related to the partials of the image intensity at pixel $\boldsymbol{x}$ by the well-known *brightness constancy constraint*

$$I_x\mathrm{u} + I_y\mathrm{v} + I_t = 0. \quad (3)$$

Combining (1), (2) and (3) we obtain the *affine* constraint

$$I_x(a_{11}x+a_{12}y+a_{13})+I_y(a_{21}x+a_{22}y+a_{23})+I_t = 0, \quad (4)$$

which can be compactly written as

$$\boldsymbol{y}^T A\boldsymbol{x} = [I_x \quad I_y \quad I_t]\begin{bmatrix} a_{11} & a_{12} & a_{13}\\ a_{21} & a_{22} & a_{23}\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix} x\\ y\\ 1\end{bmatrix} = 0, \quad (5)$$

where $\boldsymbol{y} = [I_x, I_y, I_t]^T \in \mathbb{R}^3$ is the vector of spatial and temporal image derivatives, $A \in \mathbb{R}^{3\times3}$ is the affine motion, and $\boldsymbol{x} \in \mathbb{P}^2$ is the vector of pixel coordinates. For simplicity, we will represent $\boldsymbol{x}$ as an homogeneous vector $\boldsymbol{x} = [x, y, z] \in \mathbb{R}^3$ from now on, unless otherwise stated.

In the presence of $n = 1$ motion, the affine constraint $\boldsymbol{y}^T A\boldsymbol{x} = 0$ is bilinear on the image measurements $(\boldsymbol{x}, \boldsymbol{y})$ and linear on the affine motion A. Therefore, one can estimate A linearly from a collection of $N \geq 6$ image measurements $\{(\boldsymbol{x}^j, \boldsymbol{y}^j)\}_{j=1}^N$ using equation (5).

In the presence of n motions, $\{A_i\}_{i=1}^n$, we cannot solve the problem linearly because we do not know

1. The affine motion associated with each image measurement $(\boldsymbol{x}, \boldsymbol{y})$, *i.e.* the segmentation of the data.
2. The number of affine motion models n.

Therefore, we are faced with the following problem:

Problem 1 (Multibody affine motion segmentation) *Given a set of image measurements $\{(\boldsymbol{x}^j, \boldsymbol{y}^j)\}_{j=1}^N$ corresponding to an unknown number of affine motions, estimate the number of motions n, the motion parameters $\{A_i\}_{i=1}^n$, and the segmentation of the image measurements, i.e. the motion model associated with each image measurement.*

3. Multibody affine motion segmentation

3.1. The multibody affine constraint

Let $(\boldsymbol{x}, \boldsymbol{y})$ be an image measurement associated with any motion. Then, there exists a matrix of motion parameters A_i satisfying the affine constraint $\boldsymbol{y}^T A_i\boldsymbol{x} = 0$. Thus, regardless of the motion associated with the image measurement, the following *multibody affine constraint* must be satisfied by the number of affine motions n, the motion parameters $\{A_i\}_{i=1}^n$ and the image measurement $(\boldsymbol{x}, \boldsymbol{y})$

$$\boxed{\mathcal{A}(\boldsymbol{x}, \boldsymbol{y}) \doteq \prod_{i=1}^n(\boldsymbol{y}^T A_i\boldsymbol{x}) = 0.} \quad (6)$$

The multibody affine constraint converts Problem 1 into that of solving for the number of affine motions n and the motion parameters $\{A_i\}_{i=1}^n$ from the *nonlinear* equation (6). This nonlinear constraint defines a homogeneous polynomial of degree n in either $\boldsymbol{x}$ or $\boldsymbol{y}$. For example, if we let $\boldsymbol{x} = [x, y, z]^T$, then equation (6) viewed as a function of $\boldsymbol{x}$ can be written as a linear combination of the following monomials $\{x^n, x^{n-1}y, x^{n-1}z, \ldots, z^n\}$. It is readily seen that there are $M_n \doteq (n+1)(n+2)/2$ different monomials. Therefore, we can use the Veronese map of degree n, $\nu_n : \mathbb{R}^3 \to \mathbb{R}^{M_n}$, $[x, y, z]^T \mapsto [x^n, x^{n-1}y, x^{n-1}z, \ldots, z^n]^T$, to write the multibody affine constraint (6) in bilinear form as stated by the following Lemma (See [12] for the proof).

Lemma 1 (The bilinear multibody affine constraint) *The multibody affine constraint (6) can be written as*

$$\boxed{\nu_n(\boldsymbol{y})^T \mathcal{A}\nu_n(\boldsymbol{x}) = 0,} \quad (7)$$

where $\mathcal{A} \in \mathbb{R}^{M_n\times M_n}$ is a matrix representation of the symmetric tensor product of all the affine matrices $\{A_i\}_{i=1}^n$.

We call the matrix A the *multibody affine motion* since it is a natural generalization of the affine motion model to the case of multiple moving objects. Since equation (7) clearly resembles the bilinear form of the affine constraint for a single rigid body motion, we will refer to both equations (6) and (7) as the *multibody affine constraint*.

Example 1 (The two-body affine motion) *In the case of $n = 2$ affine motions $A_1 = [b_{ij}] \in \mathbb{R}^{3\times 3}$ and $A_2 = [c_{ij}] \in \mathbb{R}^{3\times 3}$, the multibody affine motion $A \in \mathbb{R}^{6\times 6}$ is given by:*

$$\begin{bmatrix} b_{11}c_{11} & A_{12} & b_{11}c_{13}+b_{13}c_{11} & b_{12}c_{12} & b_{12}c_{13}+b_{13}c_{12} & b_{13}c_{13} \\ A_{21} & A_{22} & A_{23} & A_{24} & A_{25} & A_{26} \\ 0 & 0 & b_{11}+c_{11} & 0 & b_{12}+c_{12} & b_{13}+c_{13} \\ b_{21}c_{21} & A_{42} & b_{21}c_{23}+b_{23}c_{21} & b_{22}c_{22} & b_{22}c_{23}+b_{23}c_{22} & b_{23}c_{23} \\ 0 & 0 & b_{21}+c_{21} & 0 & b_{22}+c_{22} & b_{23}+c_{23} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

where

$$\begin{aligned} A_{12} &= b_{11}c_{12}+b_{12}c_{11}, & A_{42} &= b_{21}c_{22}+b_{22}c_{21}, \\ A_{22} &= b_{11}c_{22}+b_{21}c_{12}+b_{12}c_{21}+b_{22}c_{11}, & A_{21} &= b_{11}c_{21}+b_{21}c_{11}, \\ A_{23} &= b_{11}c_{23}+b_{21}c_{13}+b_{13}c_{21}+b_{23}c_{11}, & A_{24} &= b_{12}c_{22}+b_{22}c_{12}, \\ A_{25} &= b_{12}c_{23}+b_{22}c_{13}+b_{13}c_{22}+b_{23}c_{12}, & A_{26} &= b_{13}c_{23}+b_{23}c_{13}. \end{aligned}$$

3.2. Estimation of the number of motions n and of the multibody affine motion A

Notice that, by definition, the multibody affine motion A depends explicitly on the number of affine motions n. Therefore, even though the multibody affine constraint (7) is *linear* in A, we cannot use it to estimate A without knowing n in advance. Fortunately, we can derive a rank constraint on the image measurements from which one can estimate n, hence A. We rewrite the multibody affine constraint (7) as $(\nu_n(\boldsymbol{y}) \otimes \nu_n(\boldsymbol{x}))^T \boldsymbol{a} = 0$, where $\boldsymbol{a} \in \mathbb{R}^{M_n^2}$ is the stack of the columns of A and $\otimes$ represents the Kronecker product. Given a collection of image measurements $\{(\boldsymbol{x}^j, \boldsymbol{y}^j)\}_{j=1}^N$, the vector $\boldsymbol{a}$ satisfies the system of linear equations

$$P_n \boldsymbol{a} = 0, \tag{8}$$

where the j^{th} row of $P_n \in \mathbb{R}^{N\times M_n^2}$ is $(\nu_n(\boldsymbol{y}^j) \otimes \nu_n(\boldsymbol{x}^j))^T$, for $j = 1, \ldots, N$.

In addition to equation (8), the matrix A has to satisfy other constraints due to the fact that the 3^{rd} row of each A_i equals $e_3^T = [0, 0, 1]$. We show in Appendix A (See also Example 1) that $Z_n \doteq n(n+1)(n+2)(3n+5)/24$ entries of A are equal to zero and that the entry (M_n, M_n) is equal to one. Therefore, in order to determine $\boldsymbol{a}$ we solve the homogeneous equation

$$\boxed{\tilde{P}_n \tilde{\boldsymbol{a}} = 0,} \tag{9}$$

where $\tilde{\boldsymbol{a}} \in \mathbb{R}^{M_n^2 - Z_n}$ is the same as $\boldsymbol{a}$ with the zero entries removed and $\tilde{P}_n \in \mathbb{R}^{N\times(M_n^2 - Z_n)}$ is the same as P_n with the columns associated to zero entries removed. The scale of $\boldsymbol{a}$ is obtained by enforcing the additional constraint $\boldsymbol{a}_{M_n^2} = 1$. In order for the solution of (9) to be unique, we must have

$$\text{rank}(\tilde{P}_n) = M_n^2 - Z_n - 1. \tag{10}$$

This rank constraint on $\tilde{P}_n$ provides an effective criterion to determine the number of affine motions n from the given image intensities, as stated by the following Theorem.

Theorem 1 (Number of affine motion models) *Let $\tilde{P}_i \in \mathbb{R}^{N\times(M_i^2 - Z_i)}$ be the matrix in (9), but computed with the Veronese map ν_i of degree $1 \leq i \leq n$. If rank$(A_i) \geq 2$ for $i = 1, \ldots, n$, $N \geq M_n^2 - Z_n - 1$ and at least 6 points correspond to each motion, then*

$$\text{rank}(\tilde{P}_i) \begin{cases} > M_i^2 - Z_i - 1, & \text{if } i < n, \\ = M_i^2 - Z_i - 1, & \text{if } i = n, \\ < M_i^2 - Z_i - 1, & \text{if } i > n. \end{cases} \tag{11}$$

Therefore, the number of affine motions n is given by

$$\boxed{n \doteq \min\{i : \text{rank}(\tilde{P}_i) = M_i^2 - Z_i - 1\}.} \tag{12}$$

Proof: The proof for the case of fundamental matrices can be found in [12]. Since the polynomial $\boldsymbol{y}A_i\boldsymbol{x}$ is irreducible when rank$(A_i) \geq 2$, the proof is also valid for affine matrices. ■

In summary, we can use Theorem 1 to estimate the number of affine motions n incrementally from equation (12). Given n, we can linearly solve for the multibody affine motion A from (9). Notice however that the minimum number of image pixels needed is $N \geq M_n^2 - Z_n - 1$, which grows in the order of $O(n^4)$ for large n. Since in practice the number of motions is small, say $n \leq 10$, this is not a limitation. For example, for $n = 10$ motions we need $N \geq 2430$ pixels, which is easily satisfied by a 100×100 image.

3.3. Estimation of individual affine motions $\{A_i\}_{i=1}^n$

Given the multibody affine motion $A \in \mathbb{R}^{M_n\times M_n}$, we now show how to compute the individual affine motions $\{A_i \in \mathbb{R}^{3\times 3}\}_{i=1}^n$. From a mathematical point of view, this problem is equivalent to factoring the bi-homogeneous polynomial $\mathcal{A}(\boldsymbol{x}, \boldsymbol{y})$ into n bilinear expressions of the form $\boldsymbol{x}^T A_i \boldsymbol{y}$. To the best of our knowledge, this is a hard problem in real algebra and we are not aware of an efficient solution to it[1]. Nevertheless, the multibody affine segmentation problem is a particular case of the general factorization problem in which the 3^{rd} row of each A_i is of the form $[0, 0, 1]$. This extra knowledge will allow us to convert the bi-homogeneous factorization problem into a factorization

[1] Of course it can be solved in double exponential time using Gröebner basis and quantifier elimination.

of homogeneous polynomials, problem for which we developed a polynomial time solution in [11].

The conversion of one problem into the other goes as follows. Let $\boldsymbol{x}$ be the coordinates of a pixel in the first image frame. If the pixel undergoes motion A_i, then its optical flow $\boldsymbol{u}_i$ is given by $\boldsymbol{u}_i \doteq A_i\boldsymbol{x} \in \mathbb{R}^3$, $i = 1, \ldots, n$. Since

$$\nu_n(\boldsymbol{y})^T A \nu_n(\boldsymbol{x}) = \prod_{i=1}^{n} \left(\boldsymbol{y}^T A_i \boldsymbol{x}\right) = \prod_{i=1}^{n} (\boldsymbol{y}^T \boldsymbol{u}_i), \qquad (13)$$

we conclude that the vector $\tilde{\boldsymbol{u}} \doteq A\nu_n(\boldsymbol{x}) \in \mathbb{R}^{M_n}$ represents the coefficients of the homogeneous polynomial in $\boldsymbol{y}$

$$\boxed{g_n(\boldsymbol{y}) \doteq (\boldsymbol{y}^T\boldsymbol{u}_1)(\boldsymbol{y}^T\boldsymbol{u}_2)\cdots(\boldsymbol{y}^T\boldsymbol{u}_n) = \nu_n(\boldsymbol{y})^T\tilde{\boldsymbol{u}}.} \qquad (14)$$

We call the vector $\tilde{\boldsymbol{u}} \doteq A\nu_n(\boldsymbol{x}) \in \mathbb{R}^{M_n}$ the *multibody optical flow* associated to pixel $\boldsymbol{x}$ since it is a combination of all the optical flows $\{\boldsymbol{u}_i\}_{i=1}^n$ that the pixel can undergo depending on the motion associated with it[2]. We can also interpret the vectors $\{\boldsymbol{u}_i\}_{i=1}^n$ as linear combinations of the columns of A_i. For example, $A_i e_3$ represents the 3^{rd} column of A_i, hence $\tilde{\boldsymbol{u}} = A\nu_n(e_3)$ is a "combination" of all the 3^{rd} columns of each A_i. Intuitively, this means that if we knew how to recover the individual optical flows $\{\boldsymbol{u}_i\}_{i=1}^n$ from the multibody optical flow $\tilde{\boldsymbol{u}} = A\nu_n(\boldsymbol{x})$, then we would be able to recover the individual affine motions $\{A_i\}_{i=1}^n$ from A in a "column by column" basis[3].

From equation (14), we observe that recovering the optical flows $\{\boldsymbol{u}_i\}_{i=1}^n$ associated with pixel $\boldsymbol{x}$ from the multibody optical flow $\tilde{\boldsymbol{u}} = A\nu_n(\boldsymbol{x})$ is equivalent to factoring the homogeneous polynomial of degree n, $g_n(\boldsymbol{y})$, into the n homogeneous polynomials of degree one $\{\boldsymbol{y}^T\boldsymbol{u}_i\}_{i=1}^n$. We showed in [11] that this polynomial factorization problem has a unique solution (up to a scale for each factor) and that it is algebraically equivalent to solving for the roots of a polynomial of degree n in *one* variable, plus solving a linear system in n variables.

Given such a solution for the polynomial factorization problem, we show now how to actually recover the affine motions $\{A_i\}_{i=1}^n$ from A. To this end, let $e_1, e_2, e_3 \in \mathbb{R}^3$ be the standard basis for $\mathbb{R}^3$. Then the 1^{st}, $(M_n - n)^{th}$ and M_n^{th} rows of A are given by $\nu_n(e_1)^T A$, $\nu_n(e_2)^T A$ and $\nu_n(e_3)^T A$, respectively. Since

$$\nu_n(e_j)^T A \nu_n(\boldsymbol{x}) = \prod_{i=1}^{n} \left(e_j^T A_i \boldsymbol{x}\right) \qquad (15)$$

we can compute the j^{th} row of A_i, $\tilde{\boldsymbol{a}}_{ij} = \lambda_{ij}^{-1} A_i^T e_j$, up to a scale $\lambda_{ij} \in \mathbb{R}$, where $i = 1, \ldots, n$, $j = 1, 2$, by applying polynomial factorization to the vector $A^T\nu_n(e_j)$, $j = 1, 2$.

In order to solve for the unknown scales, we apply polynomial factorization to the vectors $A^T\nu_n(e_j + e_3)$ to obtain vectors $\tilde{\boldsymbol{c}}_{ij} = \gamma_{ij}^{-1} A_i^T(e_j + e_3)$ up to an unknown scale $\gamma_{ij} \in \mathbb{R}$. Since $A_i^T e_3 = e_3$, we have that

$$\gamma_{ij}\tilde{\boldsymbol{c}}_{ij} = \lambda_{ij}\tilde{\boldsymbol{a}}_{ij} + e_3 \qquad (16)$$

for vectors $\tilde{\boldsymbol{a}}_{ij}$ and $\tilde{\boldsymbol{c}}_{ij}$ that correspond to the same matrix A_i. We can check which $\tilde{\boldsymbol{c}}_{ij}$ corresponds to which $\tilde{\boldsymbol{a}}_{ij}$ by checking that $\tilde{\boldsymbol{c}}_{ij}^T[e_3]_\times\tilde{\boldsymbol{a}}_{ij} = 0$. Then the unknown scales are given by:

$$\lambda_{ij} = -\frac{([\tilde{\boldsymbol{c}}_{ij}]_\times\tilde{\boldsymbol{a}}_{ij})^T[\tilde{\boldsymbol{c}}_{ij}]_\times e_3}{\|[\tilde{\boldsymbol{c}}_{ij}]_\times\tilde{\boldsymbol{a}}_{ij}\|^2}. \qquad (17)$$

We now have all the rows of each A_i fully computed, except that we do not know which row corresponds to which matrix A_i. The association of rows to matrices can be easily determined by looking at the 3^{rd} column of each A_i. Such columns can be computed up to scale by applying polynomial factorization to the last column of A, $A\nu_n(e_3)$. The scale of the 3^{rd} column of A_i is immediately determined, since $A_i(3,3) = 1$. Then, in order to know which row corresponds to which affine motion, we check that the 3^{rd} entry of the 1^{st} row matches the 1^{st} entry of the 3^{rd} column. Similarly, for the 2^{nd} row. We then obtain

$$A_i = \begin{bmatrix}\lambda_{i1}\tilde{\boldsymbol{a}}_{i1} & \lambda_{i2}\tilde{\boldsymbol{a}}_{i2} & e_3\end{bmatrix}^T \quad i = 1, \ldots, n. \qquad (18)$$

Notice that this procedure for determining which row corresponds to which column fails when the entries $(1,3)$ or $(2,3)$ of two affine matrices are equal. We leave this case as an open problem.

In summary, we obtain $\{A_i\}_{i=1}^n$ from A as follows:

1. Set the 3^{rd} row of each A_i to $[0, 0, 1]$.

2. Compute the 3^{rd} column of each A_i up to scale by applying polynomial factorization to the M_n^{th} column of A, $A\nu_n(e_3)$. Since the entry $(3,3)$ of each A_i is equal to 1, the scale of the 3^{rd} column of each A_i equals the inverse of the its 3^{rd} entry.

3. The 1^{st} and 2^{nd} rows of each A_i, $\tilde{\boldsymbol{a}}_{i1}$ and $\tilde{\boldsymbol{a}}_{i2}$, can be computed up to scale by applying polynomial factorization to the 1^{st} and $(M_n–n)^{th}$ rows of A_i, $A^T\nu_n(e_1)$ and $A^T\nu_n(e_2)$, respectively.

4. The scale of the 1^{st} and 2^{nd} row of each A_i, λ_{i1} and λ_{i2}, can be obtained from equation (17), where $\tilde{\boldsymbol{c}}_{i1}$ and $\tilde{\boldsymbol{c}}_{i2}$ are obtained by applying polynomial factorization to the vectors $A^T\nu_n(e_1 + e_3)$ and $A^T\nu_n(e_2 + e_3)$, respectively. In equation (17), the $\tilde{\boldsymbol{c}}_{ij}$ associated with $\tilde{\boldsymbol{a}}_{ij}$ is determined by checking that $\tilde{\boldsymbol{c}}_{ij}^T[e_3]_\times\tilde{\boldsymbol{a}}_{ij} = 0$.

5. In order to know which row corresponds to which affine motion, we check that the 3^{rd} entry of the 1^{st} row matches the 1^{st} entry of the 3^{rd} column. Similarly for the 2^{nd} row.

[2] Mathematically, the multibody optical flow $\tilde{\boldsymbol{u}}$ is the symmetric tensor product of the individual optical flows $\{\boldsymbol{u}_i\}_{i=1}^n$.

[3] In a similar fashion, $A^T\nu_n(\boldsymbol{y})$ is the symmetric tensor product of the vectors $A_i^T\boldsymbol{y}$, *i.e.* $A^T\nu_n(\boldsymbol{y})$ is a "combination" of the rows of each A_i.

3.4. Nonlinear motion segmentation

The segmentation algorithm described in the previous section provides a unique global solution to the multibody affine motion segmentation problem. Although the problem is nonlinear, the solution is based on linear algebraic techniques thanks to the embedding of the image data into a higher-dimensional space via the Veronese map. However, such a linear solution is obtained at the cost of neglecting the algebraic structure of the multibody affine motion A. Recall that we treat the $M_n^2 - Z_n \sim O(n^4)$ entries of A as unknowns, although there are only $6n$ unknowns in $\{A_i\}_{i=1}^n$. While it is algebraically correct to neglect the internal structure of A provided that the conditions of Theorem 1 are met, the estimation of A can become numerically unstable in the presence of noise.

In this section, we propose a simple nonlinear segmentation algorithm that directly recovers the affine motion models $\{A_i\}_{i=1}^n$, without previously computing A. We assume that the number of motion has been previously estimated from equation (12) and search for the $6n$ affine parameters that minimize the algebraic error defined by the multibody affine constraint. That is, we minimize the error:

$$E(A_1, \ldots, A_n) = \sum_{j=1}^{N} \left(\prod_{i=1}^{n} (\boldsymbol{y}^{jT} A_i \boldsymbol{x}^j) \right)^2 . \qquad (19)$$

Notice that if $n = 1$, the minimization of E reduces to the standard linear least squares estimation of a single affine motion model. For $n > 2$ we use standard optimization routines starting from the solution provided by the factorization algorithm.

4. Experiments

We first test the proposed approach on synthetically generated data. We randomly pick $n = 2, 3, 4$ collections of $N = 600$ pixel coordinates and apply a different (randomly chosen) affine motion model to each one of them to generate their optical flows. From the optical flow associated to each pixel, we randomly choose a vector $\boldsymbol{y}$ of spatial and temporal image derivatives satisfying the brightness constancy constraint (3). Uniform noise from 0% to 5% is added to the partial derivatives $\boldsymbol{y}$. We run 1000 trials for each noise level. For each trial the error between the true affine motions $\{A_i\}_{i=1}^n$ and the estimates $\{\hat{A}_i\}_{i=1}^n$ is computed as

$$\text{error} = \frac{1}{n} \sum_{i=1}^{n} \left\{ \frac{\|A_i - \hat{A}_i\|}{\|A_i\|} \right\} \%. \qquad (20)$$

Figure 1 plots the mean error as a function of the noise level. In all the trials, the number of affine motions was correctly estimated from equation (12) as $n = 2, 3, 4$[4]. Notice that the estimates of the factorization algorithm are within 6% of the true affine motions for $n = 2$, even for a noise level of 5% in the image derivatives. However the performance deteriorates for $n = 3$. This is expected, because the factorization algorithm uses an over-parameterized representation of the multibody affine matrix. On the other hand, the estimates of the nonlinear algorithm are within 2.1%, 8.2% and 13.3% of the ground truth for $n = 2, 3$ and 4, respectively. The nonlinear algorithm is less sensitive to noise, because it uses a minimal parameterization of the multibody affine motion.

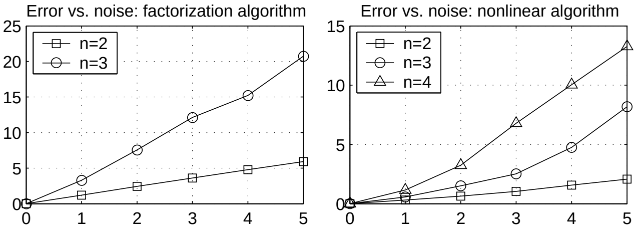


Figure 1. Error in the estimation of the affine models as a function of noise in the image partial derivatives.

We also tested the proposed approach by segmenting the real scene shown in Figure 2. The scene displays a leader-follower configuration with two robots moving in a circle and a static background. Even though the follower is tracking the leader, their 3D motions are not identical, yet they are similar enough to make their segmentation challenging.

We computed the image partial derivatives using standard derivative filters in one dimension and smoothing filters in the other two dimensions. Out of the 150×200 pixels in the image derivatives, we extracted $N = 1995$ pixels for which $|I_t| > \delta = 0.115$. We observed experimentally that it is important for the numerical stability of the algorithm not to consider pixels which are actually not moving.

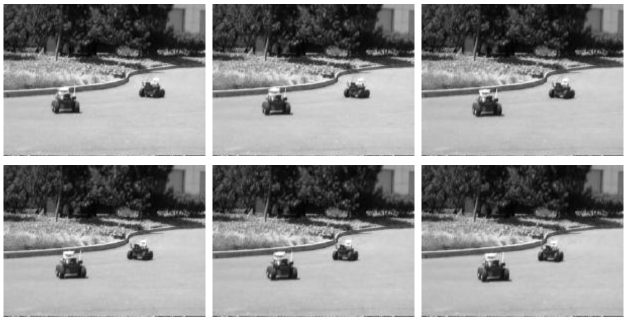
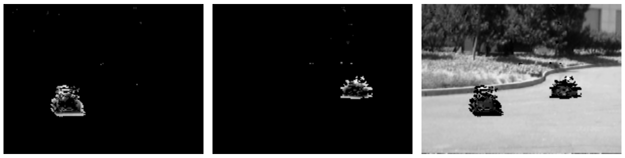


Figure 2. Two robot sequence

Figure 3 shows the results of applying our factorization technique to the image sequence. Notice that each one of the moving robots is correctly segmented.

[4] We declared the rank of $\tilde{P}_n$ to be r if $\sigma_{r+1}/(\sigma_1 + \cdots + \sigma_r) < \epsilon$, where σ_i is the i-th singular value of $\tilde{P}_n$ and $\epsilon = 3 \times 10^{-3}$.



(a) Robot 1 (b) Robot 2 (c) Background

Figure 3. Segmentation results

5. Conclusions and Future Work

We have proposed a geometric approach for the analysis of dynamic scenes from image intensities. We showed that one can determine the number of affine motions and the motion parameters directly from image intensities with no prior segmentation or correspondences and in spite of perspective effects or depth discontinuities. Our solution is based on a clear geometric interpretation the *multibody affine constraint* which transforms the affine motion segmentation problem into that of factoring homogeneous polynomials. Since the latter problem has a unique global solution, so does the motion segmentation problem. We used such a solution to initialize a nonlinear algorithm that minimizes the error defined by the multibody affine constraint. We presented simulation and experimental results validating the proposed approach. We conclude that the current implementation of the factorization algorithm is sensitive to noise due to an over-parameterized representation of the multibody affine motion. The nonlinear algorithm has an improved performance, because it uses a minimal representation. Future work will include the development of an optimal segmentation technique based on the multibody affine constraint. Since this constraint is segmentation independent, we expect to obtain an EM-type algorithm in which the expectation step is completely eliminated.

References

[1] M. Black and P. Anandan. Robust dynamic motion estimation over time. In *International Conference on Computer Vision and Pattern Recognition*, pages 296–302, 1991.

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

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

[4] M. Irani and P. Anandan. A unified approach to moving object detection in 2D and 3D scenes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(6):577–589, 1998.

[5] M. Irani and P. Anandan. About direct methods. In *Workshop on Vision Algorithms*, pages 267–277, 1999.

[6] A. Jepson and M. Black. Mixture models for optical flow computation. In *International Conference on Computer Vision and Pattern Recognition*, pages 760–761, 1993.

[7] 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.

[8] J. Shi and J. Malik. Motion segmentation and tracking using normalized cuts. In *IEEE International Conference on Computer Vision*, pages 1154–1160, 1998.

[9] A. Spoerri and S. Ullman. The early detection of motion boundaries. In *IEEE International Conference on Computer Vision*, pages 209–218, 1987.

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

[11] R. Vidal, Y. Ma, and S. Sastry. Generalized principal component analysis. Technical Report UCB/ERL 02/15, UC Berkeley, May 2002.

[12] R. Vidal, Y. Ma, S. Soatto, and S. Sastry. Two-view multibody structure from motion. *International Journal of Computer Vision*, 2002. Submitted.

[13] R. Vidal, S. Soatto, Y. Ma, and S. Sastry. Segmentation of dynamic scenes from the multibody fundamental matrix. In *Workshop on Visual Modeling of Dynamic Scenes*, 2002.

[14] Y. Weiss. Smoothness in layers: Motion segmentation using nonparametric mixture estimation. In *International Conference on Computer Vision and Pattern Recognition*, pages 520–526, 1997.

[15] Y. Weiss. Segmentation using eigenvectors: a unifying view. In *IEEE International Conference on Computer Vision*, pages 975–982, 1999.

Appendix A: Zeros of the multi-affine matrix

In addition to equation (8), the matrix A has to satisfy other constraints due to the fact that the 3^{rd} row of each A_i equals $e_3^T = [0, 0, 1]$. In order to determine these additional constraints, consider the polynomial $\mathcal{A}(\boldsymbol{x}, \boldsymbol{y})$ in (6), where $\boldsymbol{x} = [x, y, z]^T$ and $\boldsymbol{y} = [I_x, I_y, I_t]^T$. We observe that the monomials of $\boldsymbol{y} A_i \boldsymbol{x}$ involving I_t must also involve z. Therefore, the coefficients of monomials in $\mathcal{A}(\boldsymbol{x}, \boldsymbol{y})$ which are multiples of $I_t^i z^j$ with $0 \leq j < i \leq n$ are zero. Since the number of monomials which are multiples of $I_t^i z^j$ is the number of polynomials of degree $(n - i)$ in 2 variables (I_x and I_y) times the number of polynomials of degree $(n - j)$ in 2 variables (x and y), *i.e.* $(n - i + 1)(n - j + 1)$, the number of zeros in A is:

$$
\begin{aligned}
Z_n &= \sum_{i=1}^{n} \sum_{j=0}^{i-1} (n-i+1)(n-j+1) = \sum_{i=1}^{n} \frac{(n-i+1)(2n+3-i)i}{2} \\
&= \frac{(n+1)(2n+3)}{2} \sum_{i=1}^{n} i - \frac{3n+4}{2} \sum_{i=1}^{n} i^2 + \frac{1}{2} \sum_{i=1}^{n} i^3 \\
&= \frac{n(n+1)}{2} \left[\frac{(n+1)(2n+3)}{2} - \frac{(3n+4)(2n+1)}{2} + \frac{n(n+1)}{4} \right] \\
&= \frac{n(n+1)(n+2)(3n+5)}{24}
\end{aligned}
$$

In order to obtain the entries of the A that are zero, for each row of A associated to I_t^i, $i = 1, \ldots, n$, we look for the columns of A associated to z^j, for $j = 0, \ldots, i-1$. Finally, notice that the last monomial of $\mathcal{A}(\boldsymbol{x}, \boldsymbol{y})$ is exactly $(I_t z)^n$, hence the entry (M_n, M_n) of A is one.