

# Two-View Motion Segmentation with Model Selection and Outlier Removal by RANSAC-Enhanced Dirichlet Process Mixture Models

Yong-Dian Jian · Chu-Song Chen

Received: 12 May 2009 / Accepted: 6 January 2010 / Published online: 16 January 2010  
© Springer Science+Business Media, LLC 2010

**Abstract** We propose a novel motion segmentation algorithm based on mixture of Dirichlet process (MDP) models. In contrast to previous approaches, we consider motion segmentation and its model selection regarding to the number of motion models as an inseparable problem. Our algorithm can simultaneously infer the number of motion models, estimate the cluster memberships of correspondences, and identify the outliers. The main idea is to use MDP models to fully exploit the geometric consistencies before making premature decisions about the number of motion models. To handle outliers, we incorporate RANSAC into the inference process of MDP models. In the experiments, we compare the proposed algorithm with naive RANSAC, GPCA and Schindler's method on both synthetic data and real image data. The experimental results show that we can handle more motions and have satisfactory performance in the presence of various levels of noise and outlier.

**Keywords** Motion segmentation · Mixture of Dirichlet process

## 1 Introduction

Motion segmentation is important in interpreting and reconstructing dynamic scenes. It is referred to as the problem of

labeling observed correspondences to multiple moving objects. The most challenging part of this problem is to simultaneously estimate the number of motion models and the labels of correspondences in the presence of noise and outliers. From a theoretical viewpoint, this problem can be interpreted as a coupled model selection and model fitting problem. To solve this problem, one has to select a suitable number of motion models, fit the correspondences to these models, and reject outliers at the same time.

Three major approaches were proposed to solve this problem in previous work. The first assumes the images are taken under affine projection model and formulates the motion segmentation problem based on a factorization or subspace separation framework (Costeira and Kanade 1998; Gruber and Weiss 2006; Kanatani 2002; Sugaya and Kanatani 2004; Vidal and Hartley 2004; Yan and Pollefeys 2006). The second relies on iteratively applying RANSAC (Fischler and Bolles 1981) to extract the motion model with the most correspondences (Schindler and Suter 2006; Torr 1998; Tuzel et al. 2005; Wills et al. 2006). The third accommodates multiple motions within a multi-body fundamental matrix by lifting the dimensionality of correspondences (Vidal et al. 2006; Wolf and Shashua 2001). Although all of the above three approaches (factorization, RANSAC-based and lifting approaches) have shown satisfactory results, each of the approaches still has its limitations. We will discuss them separately in the following three parts.

(1) Limitation regarding to the number of motion models in the scene:

In general factorization methods, they require at least  $2L$  views (Costeira and Kanade 1998) where  $L$  is the number of motion models in the scene, but this requirement is not always feasible. In the lifting approach, the number of motion models it can handle is also limited because the number

---

Y.-D. Jian  
College of Computing, Georgia Institute of Technology, Atlanta,  
GA 30332-0280, USA  
e-mail: [ydjian@gatech.edu](mailto:ydjian@gatech.edu)

C.-S. Chen (✉)  
Institute of Information Science, Academia Sinica, Taipei 11529,  
Taiwan  
e-mail: [song@iis.sinica.edu.tw](mailto:song@iis.sinica.edu.tw)

of required correspondences grows quartically to  $L$  (Vidal et al. 2006). In the RANSAC-based approaches, as the number of motion models increases, the number of outliers also increases for an individual motion model, because the correspondences belonging to the other motion models are also regarded as outliers (to be clear, we call them pseudo outliers hereafter). In this case, the performances of RANSAC-based approaches degrade significantly since they have to spend much more time sampling a set of consistent correspondences.

(2) Model selection regarding to the number of motion models:

Factorization and lifting approaches either assume the number of motion models is known in advance or use a complexity or rank measurement to estimate this number. The former makes a strong assumption that is unreasonable in practice; the latter might lead to premature decisions about the number of motion models in the presence of noise and outliers.

(3) Outlier handling:

The nature of factorization methods makes it difficult to handle outliers. Although using RANSAC to handle outliers is popular in the problem of motion segmentation, most of the previous work does not use it properly. As mentioned above, for an individual motion model, most of the correspondences are real outliers or pseudo outliers. In this case, it is inefficient to apply RANSAC to process all correspondences without firstly considering the geometric consistencies within the inliers or pseudo outliers.

In this paper, we suggest a new approach to alleviate the aforementioned problems. The problem of motion segmentation is considered within a nonparametric Bayesian framework, mixture of Dirichlet process (MDP) models (Antoniak 1974; Ferguson 1973). We assume the motion models in the scene are generated from a Dirichlet process, and the observed correspondences are samples from a mixture of these motion models. MDP model is suitable for performing inference of motion segmentation because it not only allows the number of motion models to be uncertain in advance, but also provides us the freedom to incorporate our prior knowledge about the motions in the inference process. In addition, considering that motion segmentation is essentially a labeling problem, we make our algorithm work on a discretized motion space which facilitates the MDP inference process and makes our algorithm robust to noise. In this way, the problem of inefficient sampling encountered in the RANSAC-based approaches is alleviated because we do not have to put much effort in sampling multiple correspondences on the same moving object; moreover, compared with the lifting approach, our method has a good property that the number of required correspondences only grows

linearly with the number of motions. After evaluating the likelihoods of the correspondences with respect to motion models, we then integrate the evaluated likelihoods within the MDP inference process which can simultaneously infer the number of motions, refine the model parameters and derive the cluster membership of correspondences.

Furthermore, we also incorporate RANSAC into the proposed algorithm to handle outliers. Compared with previous work, our algorithm exploits the epipolar constraints among the correspondences before applying RANSAC to remove outliers. From the perspective of RANSAC, using RANSAC in this way is more efficient because it only has to handle a smaller and proper set of the correspondences. From the perspective of MDP models, incorporating RANSAC is also beneficial, since it makes MDP models capable of handling outliers.

Three assumptions are made in this paper: (1) there are correspondences between images, (2) camera intrinsic parameters are known, and (3) objects are rigid and have full 3-D structure.

The remainder of this paper is organized as follows: Section 2 reviews previous work on motion segmentation. Section 3 gives an introduction to MDP models. Section 4 describes the proposed two-view motion segmentation algorithm. We conclude this paper with discussions and future work in Sect. 5.

## 2 Related Work

Motion segmentation has been studied in computer vision for many years. Earlier work on motion segmentation focuses on the specific scenarios in which objects either move perpendicularly to the optical axis of the camera or can be approximated by 2-D planes. For details, we refer the readers to Kumar et al. (2005), Wills et al. (2006) and Xiao and Shah (2005) for their recent developments. Below we only focus on the review of 3-D motion segmentation which deals with general rigid-body motions and structures.

The first research track originates from the celebrated factorization method (Tomasi and Kanade 1992) which assumed that there is only one moving object, and the images are taken under affine projection model. Hence the measurement matrix can be factorized into a product of camera matrix and shape matrix by singular value decomposition. Costeira and Kanade (1998) generalized this idea to multi-body factorization. This approach is mathematically elegant and can simultaneously process all correspondences across multiple images. However, calculating the optimal solution to their formulation is actually an NP-hard problem. To overcome this problem, they suggest approximating the optimal solution with a greedy algorithm, but the resulting algorithm is vulnerable to noise and outliers in practice. Kanatani

(2002) introduced the affine subspace constraint to improve the segmentation results, and uses a complexity measurement to balance the fitness and the number of motions. Gruber and Weiss (2006) incorporated spatial coherence of correspondences into a constrained factor analysis framework and solved it with EM algorithm. Yan and Pollefeys (2006) proposed a unified framework to handle general motions.

These methods report encouraging results, but owing to the nature of the factorization method, they have the following limitations:

- (1) they consider motion segmentation and its model selection regarding to the number of motion models as separate problems,
- (2) these methods seldom handle outliers explicitly, which means they may be infeasible in practice,
- (3) they generally require  $2L$  views where  $L$  is the number of motions.

The second track assumes that the images are taken under perspective projection model and uses the essential or fundamental matrix (Hartley and Zisserman 2004) as the measurement tool. Torr (1998) processed the correspondences with RANSAC (Fischler and Bolles 1981) and used a complexity measure to sequentially extract the dominant motion model in the scene. Schindler and Suter (2006) explicitly performed model selection over all parameterized models via maximum likelihood, and formulated the motion segmentation problem as a quadratic 0-1 integer programming problem. To handle multi-view cases, Schindler et al. (2008) generalized the method in Schindler and Suter (2006) by linking possible motions between two consecutive frames to form longer motion trajectories. These predetermined motion trajectories serve as possible solutions of motions presented in the images; then they formulate the multi-view motion segmentation problem as an integer programming problem similarly. The major limitation of this approach is that the dimension of the resulting integer programming problem increases with both the number of views and the number of motions presented in the scene. Moreover, it is known that integer programming problems are NP-hard, and hence they use heuristic algorithms to approximate the optimal solution.

Wolf and Shashua (2001) and Vidal et al. (2006) generalized the formulation of fundamental matrix to simultaneously accommodate multiple motion models by lifting the dimensionality of correspondences. In this way, the multi-body fundamental matrix, which describes the multilinear relationship between correspondences, can be linearized in a high-dimensional space. This framework provides an algebraic geometric approach to motion segmentation, but the required number of correspondences grows quartically to  $L$ , where  $L$  is the number of motion models; this property makes it only suitable for the scenes with limited motion

models. Following this research track, Yang et al. (2006) extended this idea to handle outliers.

The Hough Transform (HT) is a well-known technique to detect multiple models. Several previous studies applied this technique to motion segmentation but each of them has its limitation. Ballard and Kimball (1983) assumed depth information is known. Adiv (1985) assumed the objects can be approximated by piecewise planar surfaces. Bober and Kittler (1994) and Tian and Shah (1997) both combined HT with robust estimators to accommodate outliers; the former proposed a gradient-based algorithm to search in a discretized 3-D affine motion space while the latter adopted an efficient two-stage approach but is only applicable to motions with small rotations.

In addition to the above approaches, we are also aware of another developing strategy which attempts to directly formulate the higher order constraints presented in the motion segmentation problem, and solve it with non-negative tensor factorization (Shashua et al. 2006).

The incremental processing capability of motion segmentation is desirable in most real applications, but few of the previous studies discussed this problem. Although Morita and Kanade (1997) developed an incremental version of factorization method, it only accounts for a single motion.

Dirichlet process (DP) was first proposed in the seminal work of Ferguson (1973). The basic idea of DP is to induce a prior distribution over an arbitrary partition of the parameter space. The random samples drawn from a DP follow a Dirichlet distribution with the induced prior as its parameters. DP has many nice properties in handling nonparametric problems. One of the most important properties is that the posterior distribution of a DP given several samples is still a DP. Antoniak (1974) extended DP for finite or infinite mixture modeling, and likewise it also demonstrated that the posterior distribution of an MDP given samples is still an MDP. MDP models are developed as a practical tool for nonparametric Bayesian statistics by Escobar and West (1995), MacEachern and Muller (1998) and Neal (2000). Recently MDP models have successfully been applied to computer vision applications such as image segmentation (Orbanz and Buhmann 2006), object categorization (Sudderth et al. 2006), and motion segmentation (Jian and Chen 2007).

### 3 Mixture of Dirichlet Process (MDP) Models

#### 3.1 Notations

Let  $x$  be the observation data,  $\theta \in \Theta$  be its hidden model parameter,  $\Theta$  be the parameter space,  $f(x|\theta)$  be the likelihood of  $x$  given  $\theta$ ,  $B = \{B_1, \dots, B_r\}$  be a partition of  $\Theta$ , and  $(\Theta, B)$  be a measurable space.  $G(\theta)$  and  $G_0(\theta)$  are probability distributions over  $\Theta$ . In the motion segmentation problem, we can treat  $\Theta$  as the space of rigid-body transformation,  $B$  as an estimation of motions in the scene,  $x$  as the

image coordinates of correspondences, and  $\theta$  as the motion parameter. The notation  $A \sim B$  means that  $A$  is drawn from a distribution or random process defined by  $B$ .

### 3.2 Formulation of MDP Models

To understand MDP models, first we must know what is Dirichlet Process (DP). DP is a distribution of random probability measure  $G(\cdot)$  over a measurable space  $(\Theta, B)$ , and the random vector  $(G(B_1), \dots, G(B_r))$  is distributed as a finite-dimensional Dirichlet distribution:

$$(G(B_1), \dots, G(B_r)) \sim \text{Dirichlet}(\alpha G_0(B_1), \dots, \alpha G_0(B_r)), \tag{1}$$

where  $\alpha \in \mathfrak{R}^+$  is the concentration parameter and  $G_0(\cdot)$  is the base measure. Hereafter we abbreviate (1) as  $G \sim \text{DP}(\alpha G_0)$  for conciseness.

DP has some nice properties for nonparametric modeling. The first property (*flexible prior*) is that DP allows us to specify our prior knowledge and its strength over the parameter space. This property is common in Bayesian inference but not always available in nonparametric techniques. The second property (*closure*) is that the posterior of DP given observations is still a DP

$$G | \{\theta_i\}_{i=1}^N \sim \text{DP}\left(\alpha G_0 + \sum_{i=1}^N \delta_{\theta_i}\right), \tag{2}$$

where  $\delta_\theta$  denotes a delta function centered at  $\theta$ . To simplify (2), assuming there are  $L$  unique model parameters  $\{\phi_l\}_{l=1}^L$  among  $\{\theta_i\}_{i=1}^N$  ( $L \leq N$ ), we introduce a latent class variable  $c_i$  for each  $\theta_i$  such that  $\theta_i = \phi_{c_i}$  for  $1 \leq i \leq N$  and  $1 \leq c_i \leq L$ , and define  $N_l$  as the cardinality of the index set  $I_l := \{i | \theta_i = \phi_l, 1 \leq i \leq N\}$ . With the above definitions, we can simplify (2) as

$$G | \{\theta_i\}_{i=1}^N \sim \text{DP}\left(\alpha G_0 + \sum_{l=1}^L N_l \delta_{\phi_l}\right). \tag{3}$$

Equation (3) implies the following things:

- (1) the joint distribution of the observed samples is invariant to permutation (*exchangeability*),
- (2) the posterior probability of a certain model parameter is proportional to the number of times it has been observed (*aggregation*),
- (3) the weight of base measure decreases as the number of observations increases.

We will demonstrate the connection between the above properties and motion segmentation in Sect. 3.4.

Now we introduce the definition of MDP models. Here we only focus on its infinite mixture modeling. In MDP

models, a model parameter  $\theta$  is assumed to be drawn from a  $\text{DP}(\alpha G_0)$  where  $G_0$  is an induced prior over the parameter space. The mixture models is completed by introducing a likelihood function  $f(x|\theta)$ ; the observations are distributed as a mixture distribution over the model parameters drawn from a DP. Hence we can abbreviate MDP models as

$$x \sim f(x|\theta), \tag{4}$$

$$\theta \sim G, \tag{5}$$

$$G \sim \text{DP}(\alpha G_0). \tag{6}$$

### 3.3 Inference with MDP Models

The computation of posterior expectation of MDP models can be realized by Markov Chain Monte Carlo (MCMC) methods. We can sample from the posterior distribution of  $\{\theta_i\}_{i=1}^N$  by simulating a Markov chain that has this as its equilibrium distribution. In our method, the sampling method is based on Gibbs sampler (Escobar and West 1995; MacEachern and Muller 1998; Neal 2000). Although Gibbs sampler is originally used to sample a joint distribution, its usage in MDP inference has another meaning. It is used to make the motions parameters of the observed correspondences converge to the posterior distribution on partitions and parameters of MDP models.

The most direct approach is to imagine that  $x_i$  is the last observation, and then repeatedly resample the value of each  $\theta_i$  from its posterior distribution, given both the observation  $x_i$  and the current model parameters  $\theta_{-i} := \{\theta_j\}_{j=1}^N \setminus \{\theta_i\}$ . This resampling process is actually supported by the *closure* and *exchangeability* properties of DP. To find the posterior of  $\theta_i$  given  $x_i$  and  $\theta_{-i}$ , we can write it with the Bayes' law

$$p(\theta_i | \theta_{-i}, x_i) \propto p(x_i | \theta_i) p(\theta_i | \theta_{-i}). \tag{7}$$

By replacing  $p(x_i | \theta_i)$  with likelihood  $f(x_i | \theta_i)$  and  $p(\theta_i | \theta_{-i})$  with conditional DP prior, we have the posterior distribution

$$\theta_i | \theta_{-i}, x_i \sim q_{i0} G_0(\theta_i) + \sum_{l=1}^L q_{il} \delta_{\phi_l}(\theta_i), \tag{8}$$

$$q_{i0} = \frac{1}{Z_i} \alpha \int f(x_i | \theta) G_0(\theta) d\theta, \tag{9}$$

$$q_{il} = \frac{1}{Z_i} N_l^{-i} f(x_i | \phi_l) \quad \text{for } 1 \leq l \leq L, \tag{10}$$

where  $q_{i0}$  and  $q_{il}$  represent the weights of base measure and the  $l$ th model parameter respectively,  $L$  is the number of distinct model parameters in the current state,  $N_l^{-i}$  is the number of data in  $\phi_l$  without counting  $x_i$ , and  $Z_i$  is a normalization constant such that  $q_{i0} + \sum_{l=1}^L q_{il} = 1$ .

Hence we can resample  $c_i$ , the assignment of  $\theta_i$ , according to the discrete probability distribution defined by

$(q_{i0}, \dots, q_{iL})$ . If the base measure (i.e.  $q_{i0}$ ) is sampled, then we sample a new model parameter from  $G_0$ ; if one of the existing models (i.e.  $q_{i1}, \dots, q_{iL}$ ) is sampled, we assign the sampled model parameter to  $\theta_i$ . In the resampling process, a new cluster may be created if a new model is resampled from  $G_0$ ; existing clusters may be destroyed if a cluster has only one member and this member is resampled to another cluster. Iteratively executing (8) until  $\{\theta_i\}_{i=1}^N$  are convergent concludes the key steps of the MDP inference algorithm used in this paper.

### 3.4 Motivation of Using MDP Models in Motion Segmentation

MDP models have some desired properties for motion segmentation, and we demonstrate them as follows:

(1) General treatment:

Most of the previous work on motion segmentation treats the selection of motion numbers and the estimation of their motion parameters as separate problems. This point of view is unfeasible in practice because these two factors are highly correlated and could be seriously affected by noise and outlier. MDP models provide a nonparametric Bayesian framework to help us consider the above two problems as an indivisible one. Within the MDP framework, we can simultaneously estimate the number of motion models and their motion parameters.

(2) Flexible prior:

Most of the previous work on motion segmentation does not allow the incorporation of prior knowledge about the scene into their algorithms. In MDP models, it is quite easy to specify a prior distribution over the motion parameter space.

(3) Aggregation property:

In addition to the ability of inducing prior to guide the segmentation process, MDP models tend to favor the models with more supporting samples in the inference process. This aggregation property is intuitive and suitable for the problem of motion segmentation because motions with more supporting correspondences should have higher probability to become the target of other correspondences.

## 4 Two-View Motion Segmentation Based on RANSAC-Enhanced MDP Models

### 4.1 Preliminaries

We show how to use MDP models to deal with the two-view motion segmentation problem in this section. The general

configuration of this problem is to label the observed correspondences to different motions. Assume that we have  $N$  correspondences  $\{x_i\}_{i=1}^N$ . We define  $x_i \equiv (x_i^1, x_i^2)$  where  $x_i^k$  denotes the image coordinates of the  $i$ th correspondences in the  $k$ th frame. Given a pair of correspondences  $x_i$ , if their 3-D positions in two frames undergo a rigid-body transformation between two images, then  $x_i^1$  and  $x_i^2$  have the following relationship (Hartley and Zisserman 2004):

$$\hat{x}_i^{2t} F \hat{x}_i^1 = 0, \tag{11}$$

where  $\hat{x}_i^j$  is the homogeneous representation of  $x_i^j$ ,  $F = K^{-t} \hat{T} R K^{-1}$  is the fundamental matrix,  $K$  is the intrinsic parameter matrix,  $R$  is the rotation matrix, and  $\hat{T}$  is the skew-symmetric matrix of the translation vector  $T$ . Since the intrinsic parameters are assumed to be known in advance, we can define the likelihood of a pair of correspondences with respect to a rigid-body transformation as

$$r(x_i, F_j) = S(x_i^1, x_i^2, F_j) \tag{12}$$

$$= \frac{(\hat{x}_i^{2t} F_j \hat{x}_i^1)^2}{(F_j \hat{x}_i^1)_1^2 + (F_j \hat{x}_i^1)_2^2 + (F_j^t \hat{x}_i^2)_1^2 + (F_j^t \hat{x}_i^2)_2^2}, \tag{13}$$

where  $S(x_i^1, x_i^2, F_j)$  is the Sampson distance (Hartley and Zisserman 2004).

In contrast to previous approaches, our algorithm works on a discretized motion parameter space. We handle the problem of motion segmentation in this manner because it is essentially a problem of labeling correspondences. The accuracy of labeling is more important than that of the motion parameter estimation. If finer estimations of motion parameters are desired, we can always calculate them by using existing single-body structure-from-motion algorithms once the labels of correspondences are correctly recovered (Hartley and Zisserman 2004).

To reduce the dimension of the motion parameter space, we assume that the camera is calibrated and the correspondences are in the normalized image coordinate. Hence  $F = \hat{T} R$  becomes an essential matrix, which has only five degrees of freedom.

### 4.2 MDP-Based Two-View Motion Segmentation Algorithm

Here we describe how we design the main components for MDP models, and how we use it to design our basic MDP-based two-view motion segmentation algorithm. For the reasons of simplicity and clarity, we postpone presenting the detail of handling outliers to the next section.

First we define the base measure ( $G_0$ ) of our MDP model. Assume that there are  $M$  motion hypotheses  $\{m_j\}_{j=1}^M$ . We denote  $m_j \equiv (R_j, T_j)$  as a rigid-body transformation, where

$R_j$  is the rotation matrix and  $T_j$  is the translation vector respectively. Given the correspondences  $\{x_i\}_{i=1}^N$ , we first calculate the fundamental matrices  $\{F_j\}_{j=1}^M$  of the motion hypotheses  $\{m_j\}_{j=1}^M$  and then evaluate the pair-wise Sampson distances  $\{r_{ij} = r(x_i, F_j) | 1 \leq i \leq N, 1 \leq j \leq M\}$  with (13). Then we calculate the normalized scores  $\{s_j\}_{j=1}^M$  of the motion hypotheses  $\{m_j\}_{j=1}^M$  by

$$s_j = V(\mathbf{r}_j) \quad \text{for } 1 \leq j \leq M, \tag{14}$$

where  $V(\cdot)$  is a scoring function of a motion hypothesis, and  $\mathbf{r}_j \equiv [r_{1j}, r_{2j}, \dots, r_{Nj}]_{1 \times N}$  denotes the evaluated Sampson distances of  $j$ th motion. In our experiments, we simply define  $V(\cdot)$  as a normalized truncated linear function

$$V(\mathbf{r}_j) = \frac{\sum_{i=1}^N C_N(r_{ij}, v)}{\sum_{i=1}^N C_D(r_{ij}, v)}, \tag{15}$$

$$C_N(r, v) = \begin{cases} 0, & \text{if } r \geq v, \\ \frac{v-r}{v}, & \text{otherwise,} \end{cases} \tag{16}$$

$$C_D(r, v) = \begin{cases} 0, & \text{if } r \geq v, \\ 1, & \text{otherwise,} \end{cases} \tag{17}$$

where  $v$  is a cutoff threshold for Sampson distance. The numerator adopts a truncated linear function that excludes the correspondences with large residual. We put a normalization term in the denominator because MDP models already have the aggregation property for samples (3) and this term could prevent us from favoring the moving objects with more correspondences. At this stage, we merely collect the likelihoods of every motion hypothesis in  $\{m_j\}_{j=1}^M$ , and do not assert that  $L$ , the number of motion models presented in the scene, is already known. Therefore, the computational complexity of this step grows in  $O(MN)$  and is independent of  $L$ . With  $\{r_{ij}\}_{i=1}^N \{j=1}^M$  and  $\{s_j\}_{j=1}^M$ , we define the base measure  $G_0$  and the likelihood  $f$  as

$$G_0(\theta = m_j) = \frac{s_j}{\sum_{j=1}^M s_j}, \tag{18}$$

$$f(x_i | \theta = m_j) \propto \exp\left(-\frac{r_{ij}}{\sigma^2}\right), \tag{19}$$

where  $\sigma$  is the standard deviation of the Gaussian distribution. Substituting (18) and (19) into (8), (9) and (10) completes the construction of our MDP model.

The initial estimations of  $\{\theta_i\}_{i=1}^N$  assigned to the Gibbs sampler in (8) are

$$\theta_i = \left\{ m_{\hat{j}} | \hat{j} = \arg \min_j r_{ij} \right\}, \quad \text{for } 1 \leq i \leq N. \tag{20}$$

Once  $\{\theta_i\}_{i=1}^N$  converge, the labeling results of  $\{\theta_i\}_{i=1}^N$  are taken as the solution to our algorithm. The convergency criteria used here are (1)  $\{\theta_i\}_{i=1}^N$  no longer change or (2) a pre-defined number of iterations ( $K_1 = 20$  in our experiments) is

---

**Algorithm 1:** MDP-based two-view motion segmentation algorithm without handling outliers

---

**Input:**

- $\{x_i\}_{i=1}^N$ : the set of correspondences,
- $\{m_i\}_{i=1}^M$ : the set of motion hypotheses,
- $K_1$ : the maximum iteration of Gibbs sampler

**1 Main:**

- 2 Calculate  $\{r_{ij}\}_{i=1}^N \{j=1}^M$  with (13)
  - 3 Build  $G_0$  with (18), Initialize  $\{\theta_i\}_{i=1}^N$  with (20)
  - 4 **for**  $k_1 = 1$  **to**  $K_1$  **do** /\* Gibbs sampler \*/
  - 5     Resample  $\{\theta_i\}_{i=1}^N$  with (8)
  - 6     **if**  $\{\theta_i\}_{i=1}^N$  are convergent **then break**
  - 7 **end**
- 

achieved. Algorithm 1 summarizes our basic two-view motion segmentation algorithm. However, because of the stochastic nature of Gibbs sampler, the solutions output by Algorithm 1 may not be unique. Hence in practice, we will run Algorithm 1 several times and take the fittest labeling result as our solution. To measure the fitness of a segmentation result, we estimate the motion parameters according to the labeling results produced by our algorithm, and then calculate the averaged Sampson distances of the correspondences to their motion models.

4.3 Remove Outliers by Incorporating RANSAC into MDP Models

In computer vision applications, it is inevitable to have outliers in the correspondences. Ignoring outliers in the motion segmentation problem will deteriorate the overall segmentation performance. In the proposed method, although we already use Sampson distance to exclude apparent outlier motions for each correspondence, it is possible that a correspondence is assigned to an arbitrary motion in the following two cases. The first case is that a correspondence is resampled according to the motion prior, i.e.  $G_0(\theta)$  in (9). The second is that a correspondence point is attracted to a large cluster according to (10). In these cases, the correspondence point could be resampled to a motion where the Sampson distance is not small. Hence having an outlier removal stage after the regular MDP inference is necessary.

RANSAC is a popular tool to remove outliers, and it is known that the performance of RANSAC degrades as the ratio of outliers increases. However, despite this fact, most of the previous work still uses it to handle all correspondences at the same time, which implies that a high outlier rate will be encountered as the number of motions goes up. Our idea of handling outliers is also based on RANSAC, but we think it is more appropriate to apply RANSAC after

utilizing the geometric information within the groups of correspondences. To this end, we design two criteria to identify outliers when the Gibbs sampler converges to a stationary solution. The first criterion is in terms of the number of correspondences that belongs to a motion model; if the number of correspondences is insufficient to fit a model, then these correspondences are probably outliers. The second criterion is in terms of model fitness. If a model is big enough, we then apply RANSAC to remove outliers.

Our outlier removal mechanism is advantageous for both MDP models and RANSAC. From the perspective of MDP models, integrating RANSAC within MDP inference process as a post-processing step makes it possible for MDP models to handle outliers and then enhances their modeling capability. From the perspective of RANSAC, using RANSAC at this stage is more efficient than using it at the beginning, because we have utilized the epipolar constraints to form rudimentary clusters such that RANSAC only has to process a smaller and proper set of the correspondences.

Here we provide a rough analysis to give a picture of the reduction ratios of outlier rates relative to using merely RANSAC. Here we assume that the correspondences are uniformly distributed in the scene motions, and the outliers are also randomly distributed to each cluster. The reduction ratios of outlier rates are plotted in Fig. 1. We didn't plot the reduction for the cases with only one motion, because in those cases, using merely RANSAC instead of the proposed method will be more efficient. Yet in general configurations of motion segmentation, there are about four to six times of reduction, which results in a great saving for the number of required iterations for RANSAC.

Because of the effects of outliers and the stochastic nature of our algorithm, the above outlier removal procedure may mis-assign inliers to wrong models or misclassify inliers as outliers. To make it more robust, we can extend the above binary decision procedure to probabilistic decision. We rerun the Gibbs sampler several times ( $K_2 = 15$  in our experiments) and count the times that a certain pair of correspondences is labeled as an outlier. If a pair of correspondences is really an outlier, it will have a high probability to be labeled as outliers in terms of the above two criteria. After using a threshold ( $T_0$ ) to remove outliers, we begin another iteration by taking the remaining inliers as input to the above process. The above process stops when the inlier set does not change or a predefined number of iterations ( $K_3 = 4$  in our experiments) is achieved. These numbers of iterations are selected empirically. We observe that our algorithm is insensitive to the number of iterations if there are large enough. The effectiveness of this outlier removal procedure will be evaluated in the experiments. Algorithm 2 summarizes the proposed MDP-based two-view motion segmentation algorithm.

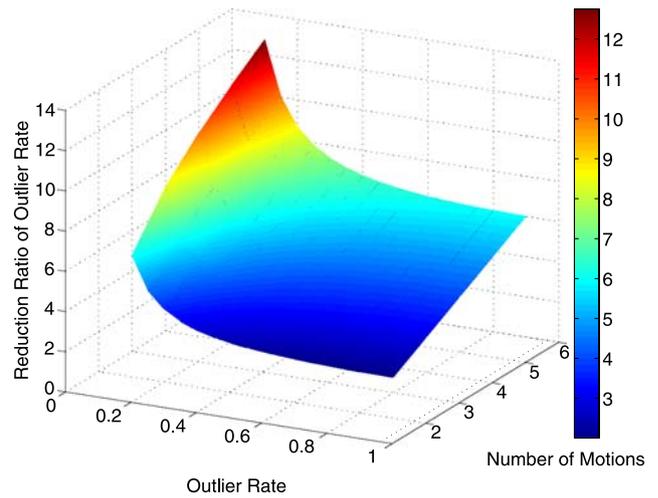


Fig. 1 Estimated reduction ratios of outlier rates

---

**Algorithm 2:** MDP-based two-view motion segmentation algorithm

---

```

Input:
 $\{x_i\}_{i=1}^N$ : the set of correspondences,
 $\{m_i\}_{i=1}^M$ : the set of motion hypotheses,
 $K_1$ : the maximum iteration of Gibbs sampler,
 $K_2$ : the times to check outlier criteria,
 $K_3$ : the maximum iteration of outlier removal,
 $T_0$ : a threshold for identifying outliers
1 Main:
2 Calculate  $\{r_{ij}\}_{i=1}^N \{j=1}^M$  with (13)
3  $I := \{1, \dots, N\}$  /* define an inlier set */
4 for  $k_3 = 1$  to  $K_3$  do
5    $\{O_i\}_{i=1}^N := 0$  /* reset counters */
6   for  $k_2 = 1$  to  $K_2$  do
7     Run lines 3–7 of Algorithm 1 with inliers
        $\{x_i\}_{i \in I}$ 
8     Check the outlier examination criteria in
       Sect. 4.3
9     foreach  $i \in I$  do Increment  $O_i$  if  $x_i$  is an
       outlier in this iteration
10    end
11    foreach  $i \in I$  do Remove  $i$  from  $I$  if
        $(O_i / K_3) > T_0$ 
12    if  $I$  no longer change then break
13 end

```

---

#### 4.4 Complexity Analysis

Here we analyze the computational and storage complexity of Algorithm 2. The computation cost of constructing base measure is  $O(MN)$  where  $M$  is the number of motion hypotheses and  $N$  is the number of correspondences.

The first loop (line 4) executes  $K_3$  times in the worst case. The second loop (line 6) runs  $K_2$  times. Line 7 executes for  $O(K_1N)$  times where  $K_1$  is the predefined maximum number of iterations for Gibbs sampler. The computational complexity of line 8 is proportional to the number of significant clusters in the current loop. (Here we denote  $L$  as the maximum number of clusters throughout the execution of the algorithm.) Lines 9 and 11–12 are simple operations which are negligible. In sum, the overall complexity would be  $O(MN + K_3K_2(K_1N + LR))$  where  $R$  is the upper bound complexity of applying RANSAC to estimate the motion parameter from correspondences. However, since  $K_1$ ,  $K_2$  and  $K_3$  are all constants in our algorithm, the complexity reduces to  $O(MN + LR)$ .

The storage complexity of our algorithm is  $O(MN)$ , which is infeasible when  $M$  is large. Yet this issue could be alleviated by using heuristics described in the next section, or some adaptive or hierarchical partition techniques studied in the literature of Hough transform (Illingworth and Kittler 1987; Li et al. 1986).

#### 4.5 Implementation Issues

Several implementation issues are addressed as follows.

First, how to efficiently generate a set of discrete motion hypotheses  $\{m_j\}_{j=1}^M$  is an application dependent problem. In the case where prior knowledge is available, the generation of motion hypotheses can focus on the high confidence regions in the motion parameter space. In the case without any prior knowledge, we can generate samples in the motion space with application dependent bounds and resolution. More specifically, we can sample the rotation angle  $\omega$  along a positive real value axis, and the translation direction  $T$  and rotation axis  $\Omega$  over a unit sphere. The former part should be straightforward. For the latter part, we adopt the iterative sphere subdivision technique described in Horn (1986). Once  $\Omega$  and  $\omega$  are selected, we can compose a rotation matrix with the quaternion notation. Although the quaternion notation is well-known for efficiently representing rotations, it is surprising that previous work usually adopts the Euler angles notation. It is known that uniformly partitioning the Euler angles would generate non-uniform samples in the motion space. This non-uniformity would make the vote accumulation step spend much more time processing the densely-sampled regions of the motion space which do not necessarily contain the scene motions. In our method, we adopt the quaternion notation to represent rotations and use the vertices of an iteratively sub-divided icosahedron as the rotation axes and translation vectors. This parametrization scheme would generate uniform samples in the motion space and facilitate the following process.

Second, the number of motion hypotheses should be as few as possible, because it directly affects the efficiency of our

algorithm. Each time we subdivide an icosahedron, the number of vertices would increase about four times. If we subdivide an icosahedron from zero to three times, there will be 12, 42, 162 and 642 vertices on them, and the corresponding angular resolutions of adjacent vectors on a sphere are 63.4, 31.7, 15.9 and 7.4 degrees respectively. Assuming that the translation vectors and the rotation axes are sampled from an icosahedron with three levels of subdivisions, and the rotation angles are quantized into five-degree intervals, we will have about  $10^6 \sim 10^8$  motion hypotheses which is infeasible in most cases. However, the number of motion hypotheses can be significantly reduced by the following heuristic observations: (1) the scale of  $T$  is redundant because it does not affect the corresponding epipolar lines and the Sampson distance, (2) the rotation angle is always non-negative because we have already sampled the rotation axes over the entire unit sphere, (3) for a pair of correspondences, the motion hypothesis corresponding to a solution that violates chirality (Hartley and Zisserman 2004) can be discarded, and (4) when building the base measure, we can set a threshold to remove the motions without sufficient supporting votes. With the above four heuristics, we can remove redundant or infeasible hypotheses in the stage of constructing the MDP base measure. In our experiments, the number of motion hypotheses is usually reduced from  $10^6 \sim 10^8$  to  $10^3 \sim 10^4$ . This reduction would make the following MDP inference stage more efficient.

Third, the discretization resolution of the motion space is crucial. When two motions are too close with respect to the discretization resolution, our approach may not be able to disambiguate them. Increasing the discretization resolution is not necessarily a good solution to this problem because there is a trade-off between segmentation performance and computational complexity. In the literature of Hough transform, this issue has been discussed thoroughly. A common practice is to partition the parameter space in an adaptive (Illingworth and Kittler 1987) or hierarchical (Li et al. 1986) manner instead of partition the entire parameter space uniformly at the beginning.

We can also reduce the memory complexity by utilizing other constraints such as the spatial coherence of correspondences or appearance of moving objects. Though these clues are not incorporated in the proposed method, they have been proved to be useful in the problem of motion segmentation (e.g. Gruber and Weiss 2006).

#### 4.6 Relation to the Hough Transform

The Hough Transform (HT) is a well-known technique to detect multiple models. Although the first step of our method is similar to the vote accumulation step in HT, the remaining steps of these two methods are different. Here we identify their key differences.

First, in HT, the accumulators are used as the only evidence for model detection. HT iteratively performs peak detection over the accumulators to extract its supports until no significant model remains. In our method, the accumulators do not play such a key role as that in HT. The accumulated votes are only used to construct the base measure ( $G_0$ ) of MDP models which provides a prior distribution over the motion space to guide the MDP inference process. Second, the model inference mechanisms of these two methods are also different. HT adopts a deterministic approach which iteratively extracts peaks in accumulators. HT may mislabel correspondences to wrong motions, and these incorrect decisions made in the early stages have no chance to be recovered in the later stages. In contrast, our method adopts a stochastic approach on the basis of MDP inference and RANSAC which lets significant models gradually emerge and makes outliers gradually removed. It also allows the correspondences mislabeled in the early iterations to be correctly clustered in the later stages.

#### 4.7 Experimental Results

In this section, we evaluate the performance of the proposed two-view motion segmentation algorithm under different concentration parameters, quantization levels, noise levels and outlier rates. Then we compare our performance with those of the other algorithms on synthetic data, Schindler's database,<sup>1</sup> and some real images. We choose the following algorithms for comparison:

- (1) Hough Transform (HT),
- (2) naive RANSAC (Torr 1998),
- (3) Vidal's GPCA-based two-view multibody structure from motion (Vidal et al. 2006) accompanied with its robust extension, RGPCA (Yang et al. 2006), and
- (4) Schindler's 2-View method (Schindler-2) (Schindler and Suter 2006).

These methods are chosen because they all require only two views.

For HT, the parametrization scheme is identical to that of our method (in Sect. 4.5). We iteratively detect the peaks and remove the votes of their supporting correspondences in the accumulators. Then we form a new motion with these points and perform RANSAC to remove outliers. This procedure is repeated until no significant model remains.

For RANSAC, since the intrinsic parameters are known, we implemented our RANSAC on the basis of Nister's five-point algorithm (Nister 2004; Stewenius et al. 2006). The verification technique proposed by Chum et al. (2003) (i.e. LORANSAC) is also incorporated.

For GPCA/RGPCA, the implementations are available on websites.<sup>2</sup> Since these packages do not estimate the number of motions, we have to provide this information as input to them. For Schindler's method, we adopt the scale estimator and the multi-branch search program released on his website<sup>1</sup> and re-implemented the other parts of the algorithm.

For the proposed algorithm, we generate the motion hypotheses  $\{m_j\}_{j=1}^M$  by subdividing an icosahedron three times and then take its vertices for both the translation direction  $T$  and rotation axis  $\Omega$ . We take the rotation angles between 10 to 45 degrees with 5-degree intervals in the experiments. The interval between 0 to 10 is not quantized because we think these motions can be approximated by translations. In the outlier removal process, we adopt identical settings as the naive RANSAC approach. All of the parameters in the experiments if not mentioned; we set the cutoff threshold  $\nu = 20$ , Gaussian standard deviation  $\sigma = 1$  (in (19)), RANSAC threshold to 0.01, the maximum number of iterations for Gibbs sampler  $K_1 = 20$ , the times to check outlier criteria  $K_2 = 15$ , the maximum number of iterations of outlier removal  $K_3 = 4$  and the outlier removal threshold  $T_0 = 0.5$ .

A pair of correspondences is identified as an error count if one of the following three cases happens:

- (1) false positive (labeling an outlier as an inlier),
- (2) false negative (labeling an inlier as an outlier),
- (3) misclassified (labeling an inlier to an incorrect motion).

All the segmentation error rates are calculated by averaging the results of 10 trials.

##### 4.7.1 Procedures of Generating Synthetic Datasets

The configuration used to generate the synthetic datasets is similar to that used in Nister (2004). Rigid-body objects are synthetically generated in the scene, and each of the objects has 20 3-D points on it. The 3-D points before and after applying a rigid-body transformation are projected onto the 2-D image planes under perspective projection model. These projected 2-D points are taken as the correspondences.

##### 4.7.2 Model Selection of the Concentration Parameter

We perform experiments to select a good concentration parameter ( $\alpha$ ) of MDP models for our algorithm. Following the procedure of generating synthetic data, we generate 15 datasets with three to five motions. We then inject samples of Gaussian noise ( $\sigma = 1$ ) and outliers (10%) to the correspondences, and evaluate the segmentation error rates on  $\alpha = 10^{-4}, 10^{-3}, \dots, 10^2$ . We observe that the choice of  $\alpha$  affects the overall segmentation error rates. If  $\alpha$  is set too

<sup>1</sup>Database and programs available at <http://www.vision.ee.ethz.ch/~konrads>.

<sup>2</sup>Software available at <http://perception.csl.uiuc.edu/gpca>.

low, MDP models are not able to generate new motion hypotheses; if  $\alpha$  is set too high, MDP models would generate too many new motion hypotheses and make the inference unstable. Our algorithm has the best performance when  $\alpha$  is around  $10^{-3}$  to  $10^{-2}$ . Hence we set  $\alpha = 10^{-2}$  in the following experiments.

#### 4.7.3 Sensitivity to Quantization Levels

Here we demonstrate the sensitivity of our algorithm to the quantization levels of motion parameters. Using the identical procedure, we generate 15 datasets with three motions. In Table 1, we show the error rates of different quantization levels and data configurations. The numbers below the R and T columns are the numbers of subdivisions applied on an icosahedron. The rotation axes and translation directions are drawn from the subdivided icosahedrons accordingly. The A column denotes the resolutions of rotation angles. The numbers under the other columns are the error rates of different data configurations where  $\sigma$  is the noise level and  $\gamma$  is the outlier rate. The baseline configuration is empirically selected. The following three variation sets have different quantization levels in R, T or A respectively.

As shown in the table, the error rates go higher as noise levels ( $\sigma$ ) or outlier rates ( $\gamma$ ) increase; for the variations of quantization resolutions, we observe that in the R and T sets, the error rates go lower as the quantization resolutions become finer. However, this conclusion is not always true for the A set; we observe that the error rates remain unchanged

**Table 1** Error rates (%) of different quantization and data configurations

	R	T	A	$\sigma = 1$ $\gamma = 10\%$	$\sigma = 3$ $\gamma = 10\%$	$\sigma = 1$ $\gamma = 25\%$	$\sigma = 3$ $\gamma = 25\%$
Baseline	3	3	5°	3.0	5.3	6.2	8.1
R Set	<b>1</b>	3	5°	5.6	6.5	9.3	13.0
	<b>2</b>	3	5°	4.1	5.4	7.7	10.9
	<b>4</b>	3	5°	2.8	4.9	5.8	9.0
T Set	3	<b>1</b>	5°	10.1	11.5	14.0	18.1
	3	<b>2</b>	5°	5.1	6.5	8.7	11.7
	3	<b>4</b>	5°	2.5	2.9	5.7	7.1
A Set	3	3	<b>15°</b>	5.3	5.9	9.8	12.9
	3	3	<b>10°</b>	4.4	5.4	8.2	9.4
	3	3	<b>2.5°</b>	3.1	5.0	6.2	8.5

**Table 2** Average CPU timing information of our method

Calculate pairwise Sampson distances	MDP Inference			Others	Total
	Build base measure	Gibbs sampler	RANSAC		
2.4s (21%)	2.1s (18%)	3.6s (32%)	1.9s (17%)	1.2s (11%)	11.2s (100%)

or become higher when the quantization resolutions of A increase from 5° to 2.5°. It means we could gain nothing from over-quantization of the parameter space. We also observe that the quantization resolutions of R, T and A have different influences on the segmentation performance. The quantization level of T has stronger influence to our method while those of R and A have comparable influences.

Although the numbers of the motion hypotheses are large ( $10^6 \sim 10^8$ ) at the beginning, in our experiments we observe that these numbers can be significantly reduced ( $10^3 \sim 10^4$ ) by applying the heuristics mentioned in Sect. 4.5. This reduction would make the following MDP inference stage more efficient.

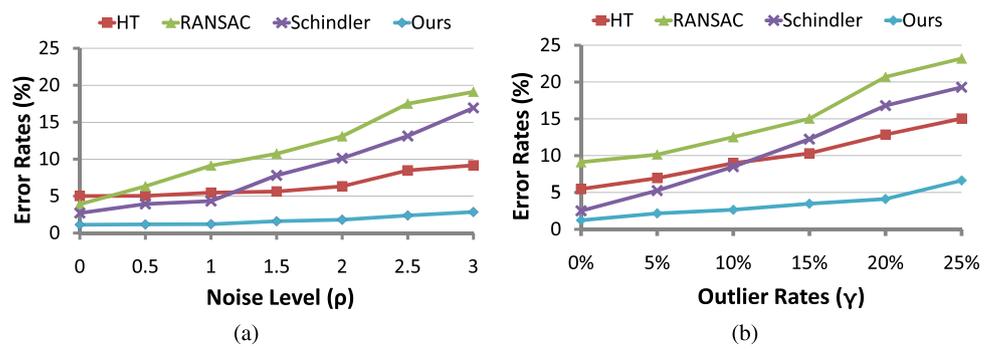
#### 4.7.4 Execution Time

In Table 2, we provide the average CPU timing information for our Matlab implementation of the proposed method on an Intel Core 2 Duo machine. The CPU times are measured on 15 datasets with three motions ( $\sigma = 1.5, \gamma = 15\%$ ). We also divide our algorithm into several components and identify their execution times and percentages. The first stage is to calculate pairwise Sampson distances. The MDP inference stage is divided into three parts: the construction of base measure, Gibbs sampler and RANSAC for outlier removal. Currently our implementation requires 11 seconds to process one dataset in average. Although this implementation is not suitable for real time applications, its speed can be improved by adopting C/C++ implementations or by parallelizing our algorithm.

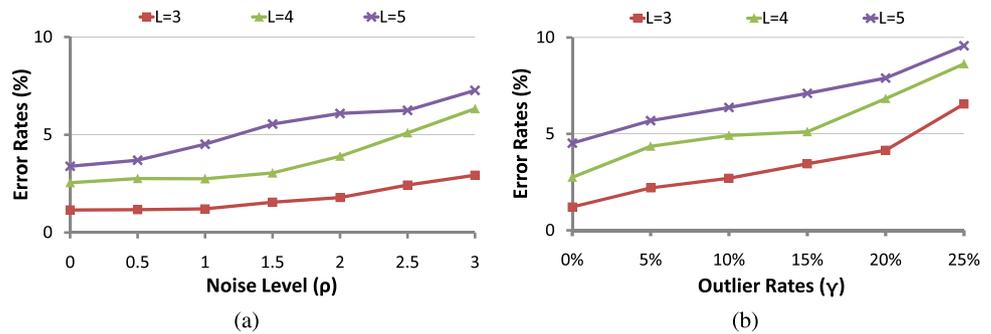
#### 4.7.5 Comparison with Other Methods on Synthetic Data

Here we conduct two experiments to compare the performance of our algorithm with those of the others. We randomly generate 15 synthetic scenes with three motions, and inject various levels of noise and outliers respectively. In the noise experiment, the noise levels ( $\rho$ ) range from 0 to 3 pixels; while in the outlier experiment, the outlier rates ( $\gamma$ ) range from 0 to 25 percents (with  $\rho$  set to 1). Then we apply all algorithms to segment the synthetic scenes, and the averaged error rates are shown in Fig. 2. GPCA and RGPCA break down in our experiments; their error rates are usually more than 40%, so we do not plot their performance in our figures. HT, Naive RANSAC and Schindler's method have moderate levels of error rates in both experiments. In both experiments, the error rates of the our method are consistently lower than those of the other methods.

**Fig. 2** Comparison of different algorithms on synthetic datasets. (a) The error rates of different noise levels. (b) The error rates of different outlier rates (with all noise levels set to 1)



**Fig. 3** The performance of the proposed algorithm in terms of different motions. (a) The error rates of different noise levels. (b) The error rates of different outlier rates (with all noise levels set to 1)



**Table 3** Error rates (%) of two-view motion segmentation algorithms

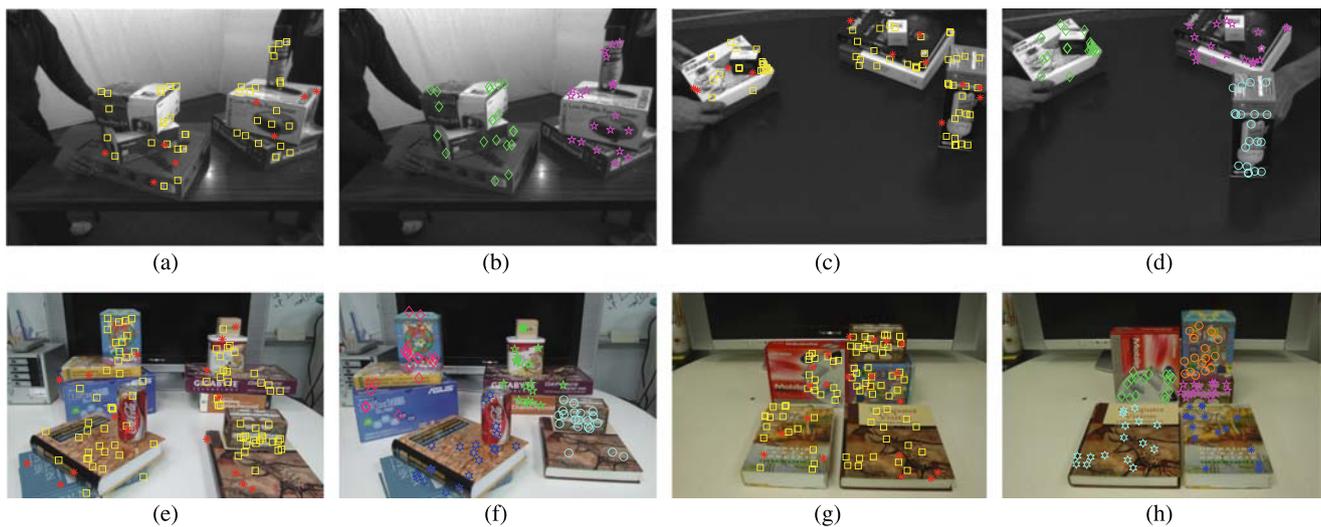
Method	Boxes (40 12) 2 motions		Bulb (60 15) 3 motions		Real1 (80 20) 4 motions		Real2 (100 25) 5 motions	
	MC	ER	MC	ER	MC	ER	MC	ER
HT	11.5	12.5	10.9	12.6	15.3	17.3	16.5	19.0
RANSAC	4.9	15.5	5.3	21.3	13.8	21.2	7.9	20.3
RGPCA	19.0	48.5	15.2	32.1	38.9	41.4	34.8	47.8
Schindler-2	3.8	5.9	10.7	13.5	12.4	15.1	12.1	14.5
Ours	2.1	2.7	2.6	4.0	4.7	6.3	5.2	8.5

To further investigate the performance of the proposed method, we generate synthetic scenes with five motions. The experimental results are shown in Fig. 3. The results show that the proposed method has modest error rates when the number of motions increases.

4.7.6 Experiments on Real Data

Four real image datasets are used in this experiment. The first two datasets are taken from Schindler’s database (“Boxes” and “Bulb” datasets) where the intrinsic parameters are available. There are two and three moving objects in these two datasets respectively. Since this database is originally used for multi-view motion segmentation, we pick four pairs of consecutive frames from each dataset as the input images to all algorithms and report the averaged error rates. In addition, we also picture two datasets (“Real1” and “Real2”) with four and five moving objects respectively. These two datasets are taken at indoor environment, and the

camera intrinsic parameters and radial distortion parameters are calibrated by using Zhang’s method (Zhang 2000). All of the correspondences are extracted by matching the SIFT features between two images (Lowe 2004). We manually identify whether a pair of correspondences is an inlier or outlier by inspecting the matching results. The averaged error rates are summarized in Table 3. For each dataset, the numbers next to its name indicate its configuration; the first number means the number of inliers, and the second one means the number of outliers. The error rates were separated into two parts. MC means the ratio between the number of misclassified inliers and the number of total inliers, and ER means the total error rate. The difference between MC and ER are caused by false positive and false negative. As in the experiments on synthetic data, GPCA and RGPCA break down because of the pixel noise, outliers and insufficient correspondences. The error rates of our algorithm are consistently lower than those of the other methods. Figure 4 il-



**Fig. 4** (Color online) Segmentation results of our method on four real datasets: (a), (b) Boxes, (c), (d) Bulb, (e), (f) Real1 and (g), (h) Real2. For each dataset, the presented two images are the inputs of our algorithm. The *left image* illustrates the memberships of inliers (yellow)

and outliers (red). The *right image* shows the segmentation results; the correspondences segmented to different motions are labeled with different colors

**Table 4** Error rates (%) of our algorithm on Tron’s benchmark

	2 motions		3 motions	
	Check	Traffic	Check	Traffic
Error rate	3.93	3.13	6.78	5.12

illustrates the segmentation results of our algorithm. We show the inlier/outlier labels and segmentation results with different colors. For each dataset, the labels of inliers and outliers are shown in the left image, and the segmentation results are shown in the right image.

#### 4.7.7 Experiments on Tron’s Benchmark

We also run our algorithm on Tron’s benchmark proposed in Tron and Vidal (2007). This benchmark contains a total of 155 motion sequences: 120 with two motions and 35 with three motions, and each sequence contains roughly 30 frames. We exclude the sequences in the “Articulated/Non-Rigid” category and run our algorithm on the “Checkboard” and “Traffic” categories. Since this benchmark provides trajectories to examine the performance of factorization-based methods, we uniformly subsample four points along the trajectories and report the average error rates over 20 trials in Table 4. We only report the misclassification error rates because the benchmark considers the cases without outliers. As shown in the table, our results are comparable or slightly worse than the best results reported in Tron and Vidal (2007), but the advantage of our method is that we can automatically determine the number of motions while most methods in Tron and Vidal (2007) need the users to specify the number of motions as an input parameter.

## 5 Conclusions and Future Work

We present an MDP-based two-view motion segmentation algorithm which treats motion segmentation and its model selection regarding to its number of motion models as an indivisible problem. To make it possible for MDP models to handle outliers, we use RANSAC as a post-processing step of MDP inference, and make RANSAC only have to process a smaller and proper set of correspondences.

In future work, we plan to use the method proposed in Makadia et al. (2005) or use Graphical Processing Unit (GPU) to speed up the computation of our method. In addition, we can also employ more advanced Markov chain Monte Carlo methods instead of Gibbs sampler to further boost the performance of MDP inference.

Handling dependent or degenerated motions are also important problems in motion segmentation. If two different objects move in the same way, it is impossible to separate them by using our method. This problem can be alleviated by making extra assumptions or utilizing other visual/geometric information (Gruber and Weiss 2006; Makadia et al. 2005; Schindler and Suter 2006). The problem of degenerated motions is not addressed in this paper.

In this paper, we only discuss the scenario with two images. It would be interesting to generalize the proposed method to handle the scenario with multiple images. We take the above directions as our future work.

**Acknowledgements** This work was supported in part under Grants 98-EC-17-A-02-S1-032 and NSC98-2221-E-001-012-MY3.

## References

- Adiv, G. (1985). Determining 3-dimensional motion and structure from optical-flow generated by several moving-objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(4), 384–401.
- Antoniak, C. E. (1974). Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *Annals of Statistics*, 2(6), 1152–1174.
- Ballard, D. H., & Kimball, O. A. (1983). Rigid body motion from depth and optical-flow. *Computer Vision Graphics and Image Processing*, 22(1), 95–115.
- Bober, M., & Kittler, J. (1994). Robust motion analysis. In *Proc. IEEE conf. on computer vision and pattern recognition* (pp. 947–952).
- Chum, O., Matas, J., & Kittler, J. (2003). Locally optimized ransac. In *Proc. 25th DAGM symposium* (Vol. 2781, pp. 236–243).
- Costeira, J. P., & Kanade, T. (1998). A multibody factorization method for independently moving objects. *International Journal of Computer Vision*, 29(3), 159–179.
- Escobar, M. D., & West, M. (1995). Bayesian density-estimation and inference using mixtures. *Journal of the American Statistical Association*, 90(430), 577–588.
- Ferguson, T. (1973). A Bayesian analysis of some nonparametric problems. *Annals of Statistics*, 1(2), 209–230.
- Fischler, M. A., & Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6), 381–395.
- Gruber, A., & Weiss, Y. (2006). Incorporating non-motion cues into 3D motion segmentation. In *Proc. European conference on computer vision*.
- Hartley, R. I., & Zisserman, A. (2004). *Multiple view geometry in computer vision* (2nd edn.). Cambridge: Cambridge University Press.
- Horn, B. (1986). *Robot vision*. New York: McGraw-Hill.
- Illingworth, J., & Kittler, J. (1987). The adaptive hough transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5), 690–698.
- Jian, Y. D., & Chen, C. S. (2007). Two-view motion segmentation by mixtures of Dirichlet process with model selection and outlier removal. In *Proc. international conference on computer vision*.
- Kanatani, K. (2002). Evaluation and selection of models for motion segmentation. In *Proc. European conference on computer vision* (Vol. 2352, pp. 335–349).
- Kumar, M. P., Torr, P. H. S., & Zisserman, A. (2005). Learning layered motion segmentations of video. In *Proc. international conference on computer vision* (Vol. 1, pp. 33–40).
- Li, H. W., Lavin, M. A., & Lemaster, R. J. (1986). Fast hough transform—a hierarchical approach. *Computer Vision Graphics and Image Processing*, 36(2–3), 139–161.
- Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 91–110.
- MacEachern, S. N., & Muller, P. (1998). Estimating mixture of Dirichlet process models. *Journal of Computational and Graphical Statistics*, 7(2), 223–238.
- Makadia, A., Geyer, C., Sastry, S., & Daniilidis, K. (2005). Radon-based structure from motion without correspondences. In *Proc. IEEE conf. on computer vision and pattern recognition*.
- Morita, T., & Kanade, T. (1997). A sequential factorization method for recovering shape and motion from image streams. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(8), 858–867.
- Neal, R. M. (2000). Markov chain sampling methods for Dirichlet process mixture. *Journal of Computational and Graphical Statistics*, 9(2), 249–265.
- Nister, D. (2004). An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6), 756–770.
- Orbanz, P., & Buhmann, J. M. (2006). Smooth image segmentation by nonparametric Bayesian inference. In *Proc. European conference on computer vision*.
- Schindler, K., & Suter, D. (2006). Two-view multibody structure-and-motion with outliers through model selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(6), 983–995.
- Schindler, K., Suter, D., & Wang, H. (2008). A model-selection framework for multibody structure-and-motion of image sequences. *International Journal of Computer Vision*, 79(2), 159–177.
- Shashua, A., Zass, R., & Hazan, T. (2006). Multi-way clustering using super-symmetric non-negative tensor factorization. In *Proc. European conference on computer vision*.
- Stewenius, H., Engels, C., & Nister, D. (2006). Recent developments on direct relative orientation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 60(4), 284–294.
- Sudderth, E., Torralba, A., Freeman, W., & Willsky, A. (2006). Depth from familiar objects: a hierarchical model for 3D scenes. In *Proc. IEEE conf. on computer vision and pattern recognition* (Vol. 2, pp. 2410–2417).
- Sugaya, Y., & Kanatani, K. (2004). Multi-stage unsupervised learning for multi-body motion segmentation. *IEICE Transactions on Information and Systems*, E87d(7), 1935–1942.
- Tian, T. Y., & Shah, M. (1997). Recovering 3D motion of multiple objects using adaptive hough transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(10), 1178–1183.
- Tomasi, C., & Kanade, T. (1992). Shape and motion from image streams under orthography—a factorization method. *International Journal of Computer Vision*, 9(2), 137–154.
- Torr, P. H. S. (1998). Geometric motion segmentation and model selection. *Philosophical Transactions of the Royal Society of London Series A*, 356(1740), 1321–1338.
- Tron, R., & Vidal, R. (2007). A benchmark for the comparison of 3D motion segmentation algorithms. In *Proc. IEEE conf. on computer vision and pattern recognition*.
- Tuzel, O., Subbarao, R., & Meer, P. (2005). Simultaneous multiple 3D motion estimation via mode finding on lie groups. In *Proc. international conference on computer vision* (Vol. 1, pp. 18–25).
- Vidal, R., & Hartley, R. (2004). Motion segmentation with missing data using powerfactorization and gpca. In *Proc. IEEE conf. on computer vision and pattern recognition*.
- Vidal, R., Ma, Y., Soatto, S., & Sastry, S. (2006). Two-view multibody structure from motion. *International Journal of Computer Vision*, 68(1), 7–25.
- Wills, J., Agarwal, S., & Belongie, S. (2006). A feature-based approach for dense segmentation and estimation of large disparity motion. *International Journal of Computer Vision*, 68(2), 125–143.
- Wolf, L., & Shashua, A. (2001). Two-body segmentation from two perspective views. In *Proc. IEEE conf. on computer vision and pattern recognition* (Vol. 1, pp. 263–270).
- Xiao, J. J., & Shah, M. (2005). Motion layer extraction in the presence of occlusion using graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10), 1644–1659.
- Yan, J. Y., & Pollefeys, M. (2006). A general framework for motion segmentation: independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In *Proc. European conference on computer vision*.
- Yang, A., Rao, S., & Ma, Y. (2006). Robust statistical estimation and segmentation of multiple subspaces. In *Proc. IEEE workshop on 25 years of RANSAC (joint with CVPR)*.
- Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11), 1330–1334.