Spectral clustering of linear subspaces for motion segmentation

Fabien Lauer
Heidelberg Collaboratory for Image processing,
University of Heidelberg, Germany
`fabien.lauer@iwr.uni-heidelberg.de`

Christoph Schnörr
Image and Pattern Analysis group,
Heidelberg Collaboratory for Image processing,
University of Heidelberg, Germany
`schnoerr@math.uni-heidelberg.de`

Abstract

This paper studies automatic segmentation of multiple motions from tracked feature points through spectral embedding and clustering of linear subspaces. We show that the dimension of the ambient space is crucial for separability, and that low dimensions chosen in prior work are not optimal. We suggest lower and upper bounds together with a data-driven procedure for choosing the optimal ambient dimension. Application of our approach to the Hopkins155 video benchmark database uniformly outperforms a range of state-of-the-art methods both in terms of segmentation accuracy and computational speed.

1. Introduction

Motion segmentation is an important initial step in the analysis of video sequences involving multiple objects. The basic idea, illustrated in Fig. 1, is to segment a set of tracked feature points into different groups corresponding to different objects on the basis of their motions. Note that other formulations, aiming at the segmentation of entire images rather than feature points, can be found for instance in [4].
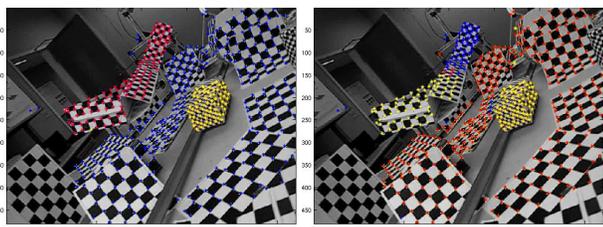

Figure 1. Sample frame from a video sequence of the Hopkins database [12] involving three motions (2 objects plus background). *Left:* the proposed algorithm correctly segment the feature points in 3 groups (different colors) on the basis of their trajectory. *Right:* in a similar setting, spectral curvature clustering [2] confuses the two moving objects.

Under the affine camera model, motion segmentation from tracked feature points amounts to a subspace separation problem [7], where each subspace corresponds to a different motion. The subspace separation problem itself can be thought of as extending standard clustering, which assumes a distribution of the data centered around multiple prototypes, to data distributed along planes. The increase in difficulty comes in part from the fact that multiple subspaces can intersect, so that distributions of points belonging to different groups are more likely to overlap, often leading to indistinguishability. Unlike earlier attempts at subspace separation from high-dimensional data, the paper proposes to keep a rather high dimension of the ambient space, where the probability of facing this indistinguishability is minimized. Based on this idea, the paper proposes a spectral clustering approach, in which the ambient space dimension is carefully selected to allow for the best segmentation.

Related work. Early attempts at motion segmentation include factorization-based methods [3, 6, 7, 18]. But except for [18], these methods are only suitable for fully independent motions and thus cannot deal with moving cameras, which imply a partial dependence between the motions. The Generalized Principal Component Analysis (GPCA) [14] is an algebraic method for subspace separation wich can deal with dependent motions, but which is not scalable in terms of the ambient space dimension and the number of motions. More recently, [11] presented an Agglomerative Lossy Compression (ALC) algorithm, which can additionally deal with outlying and corrupted trajectories. This method led to the best segmentation accuracy reported on the Hopkins155 benchmark database [12] so far.

Our approach is based on spectral clustering [9, 15], where the main idea is to find an embedding of the N data points, through a few eigenvectors of an N-by-N affinity matrix between all pairs of points, in which standard clustering algorithms, such as k-means, can be applied. In the context of motion segmentation, [10, 16] combine

factorization methods with spectral clustering by building the affinity matrix from the shape interaction matrix. For subspace separation, the Local Subspace Affinity (LSA) method [17] first estimates local subspaces in the neighborhood of the points and then uses spectral clustering with an affinity based on the principal angles between the local subspaces to segment the data. In Spectral Curvature Clustering (SCC) [1], the affinity is defined through the curvature of the spaces generated by all combinations of $d+1$ points for d-dimensional subspaces. Though a random sampling of the resulting affinity tensor is possible [2], the complexity of this procedure is still rather high for a large d.

Contributions. The contribution of the paper is twofold. First, we investigate the effect of the dimension of the ambient space on subspace separation. In particular, we show that low dimensions chosen in prior work do not offer sufficient separability between the subspaces and suggest lower and upper bounds on this dimension. Then, we show how spectral clustering can be adapted to the separation of subspaces of unknown dimensions by using an affinity matrix built from angular information between the points. In comparison to LSA [17], which also uses some angular information, the proposed method does not require to estimate local subspaces and principal angles, but instead merely amounts to computing dot products between the points. Thus, the complexity of the resulting algorithm remains low and does not depend on the dimension d of the subspaces as in SCC [2] (which additionally requires knowledge of d). Thanks to the speed of the algorithm, we devise an automated scheme to combine dimension reduction with spectral clustering, where the reduced dimension is specifically tuned for the clustering task. When applied to motion segmentation data from the Hopkins155 benchmark [12], the proposed method outperforms previously reported results both in terms of segmentation accuracy and computational speed.

Paper organization. The paper starts with the formulation of motion segmentation as a subspace separation problem in Sect. 2, including a discussion on the choice of the ambient space dimension. Spectral clustering is introduced in Sect. 3, while the proposed affinity measure is described in Sect. 3.2 and 3.3. The experiments on synthetic as well as real data are detailed in Sect. 5, before giving the conclusions in Sect. 6.

2. Motion segmentation with feature points

In this paper, we only consider the problem of motion segmentation from tracked feature points. The data take the form of N feature point trajectories $\{(x_{if}, y_{if})\}_{i=1,\ldots,N,f=1,\ldots,F}$ over F frames, where x_{if} and y_{if} are the coordinates of point i in frame f. The aim is to estimate the labels c_i for every point i, which classify the points into groups of different motions, and hence objects.

2.1. Motions as linear subspaces

Recently, many works on motion segmentation [8, 11, 12, 14] considered the affine camera model. Under this assumption, it can be shown that the trajectories of feature points belonging to the same rigid object (and same motion) lie in a linear subspace of dimension at most four. Indeed, the affine camera model, allows the matrix $\boldsymbol{W} \in \mathbb{R}^{2F \times P}$, containing the coordinates of all P points of a single object, to be written as

$$\begin{bmatrix} x_{11} & x_{21} & \ldots & x_{P1} \\ y_{11} & y_{21} & \ldots & y_{P1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1F} & x_{2F} & \ldots & x_{PF} \\ y_{1F} & y_{2F} & \ldots & y_{PF} \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \\ \vdots \\ \boldsymbol{A}_F \end{bmatrix} \begin{bmatrix} X_1 & \ldots & X_P \\ Y_1 & \ldots & Y_P \\ Z_1 & \ldots & Z_P \\ 1 & \ldots & 1 \end{bmatrix},$$
(1)

where X_{if}, Y_{if} and Z_{if} are the 3-D coordinates of point i and $\boldsymbol{A}_f \in \mathbb{R}^{2 \times 4}$ is the affine motion matrix at frame f. As the rank of the right hand side of (1) is bounded by 4, all columns of the matrix on the left hand side must lie in a 4-dimensional subspace of $\mathbb{R}^{2F}$. In this framework, segmenting multiple motions amounts to a linear subspace separation problem, where each subspace corresponds to a particular motion. In the remaining of the paper, the trajectory matrix $\boldsymbol{W} \in \mathbb{R}^{2F \times N}$ is built as in (1) but with N feature points from multiple objects.

2.2. Dimension of the ambient space

Methods such as GPCA [14] require to work in a low dimensional space. As the dimension D of this ambient space has an impact on the speed and accuracy of other algorithms as well, data, originally in $\mathbb{R}^{2F}$, are often preprocessed by a dimension reduction step. In [13], the authors suggest to use a 5-dimensional ambient space, which is the minimal dimension in which multiple 4-dimensional (4-D) subspaces can be embedded. Motivated by general work on compressive sensing [5], [11] suggests to use $D_{sp} = \min D$, subject to $D \geq 2d \log(2F/D)$, for d-dimensional subspaces (with $d = 4$ for motion segmentation). However, the ambient space dimension should be chosen to facilitate the separation of the subspaces. With respect to this aim, the choice $D = d + 1$ to separate d-dimensional subspaces is not optimal, as shown below.

Lower and upper bounds. Consider two linear subspaces S_1 and S_2 of dimensions d_1 and d_2, embedded in $\mathbb{R}^D$. Then the dimension of the intersection of S_1 and S_2 is bounded by

$$d_1 + d_2 - D \leq \dim(S_1 \cap S_2) \leq \min(d_1, d_2), \quad (2)$$

which, for assumed similar subspace dimensions $d_1 = d_2 = d$ as in motion segmentation, leads to

$$2d - D \leq \dim(S_1 \cap S_2) \leq d. \qquad (3)$$

By using $D = d + 1$, the dimension of the intersection is given by $d - 1 \leq \dim(S_1 \cap S_2) \leq d$ (where the upper bound is only obtained when $S_1 = S_2$), leading to the maximal dimension for the intersection. Due to the difficulty of correctly classifying points close to $S_1 \cap S_2$, minimizing $\dim(S_1 \cap S_2)$ should be considered.

We propose to use an ambient space dimension $D \geq 2d + 1$, which makes the lower bound vanish and thus allows the intersection $S_1 \cap S_2$ to have minimal dimension. Note that D has no influence on the upper bound in (2). By choosing D, we do not impose any restriction on the dimension of the intersection, but merely increase the probability of having a low dimension $\dim(S_1 \cap S_2)$.

Noisy data. For noisy data, the larger D is, the larger the effect of noise is. Thus, in practice, the minimal dimension should be chosen, *i.e.* $D = 2d + 1$.

Multiple subspaces. In practice, when dealing with n subspaces, we propose to use $D = 1 + \sum_{j=1}^{n} d_j$ to ensure enough separability between all the subspaces. This heuristics will be evaluated in the experiments of Sect. 5.1 for subspaces of similar dimension d, in which case $D = nd + 1$.

Degenerate subspaces. When separating degenerate subspaces with unknown dimensions, d and thus D are overestimated. In order to avoid giving to much weight to the noise in this case, in Sect. 4 we propose an automated selection of D within the bounds $nd_{min} + 1 \leq D \leq nd_{max} + 1$ for $d \in [d_{min}, d_{max}]$.

3. Spectral clustering of subspaces

In this section, the data are assumed to take the form of N points $\boldsymbol{x}_i \in \mathbb{R}^D$, $i = 1, \ldots, N$. The aim is to partition these points into n groups, where n is known. Three scenarios are considered: the points form tight clusters around n centers, the points are distributed along lines (1-D subspaces), or the points are sampled from subspaces of any dimension.

3.1. Spectral clustering of points

Spectral clustering can be seen as applying standard clustering to the data embedded in a particular space given by the first eigenvectors of an N-by-N affinity matrix $\boldsymbol{A}$, itself built from distances between pairs of points such that

$$A_{ij} \approx \begin{cases} 1, & \text{if } c_i = c_j \\ 0, & \text{otherwise,} \end{cases} \qquad (4)$$

where $c_i \in \{1, \ldots, n\}$ is the true label of point $\boldsymbol{x}_i$. Following [9], the spectral clustering algorithm below can be defined.

1. Build the affinity matrix $\boldsymbol{A}$ defined by $A_{ij} = \exp(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2 / 2\sigma^2)$, if $i \neq j$, $i = 1, \ldots, N$, $j = 1, \ldots, N$, and $A_{ii} = 0$.

2. Define $\boldsymbol{D}$ as a diagonal matrix, where $D_{ii} = \sum_{j=1}^{N} A_{ij}$, and the matrix $\boldsymbol{L} = \boldsymbol{D}^{-1/2} \boldsymbol{A} \boldsymbol{D}^{-1/2}$.

3. Find the leading n eigenvectors $\boldsymbol{u}_k$, $k = 1, \ldots, n$, of $\boldsymbol{L}$ to build the matrix $\boldsymbol{U} \in \mathbb{R}^{N \times n}$.

4. Normalize the rows of $\boldsymbol{U}$ by $U_{ij} \leftarrow U_{ij} / \sqrt{\sum_{j=1}^{n} U_{ij}^2}$.

5. Apply k-means or another algorithm to cluster the rows of $\boldsymbol{U}$ and then assign the samples $\boldsymbol{x}_i$ to groups accordingly.

3.2. Spectral clustering of lines

Consider now clustering points sampled from 1-D subspaces, *i.e.* lines. In this case, an affinity defined via a radial distance, $A_{ij} = g(\|\boldsymbol{x}_i - \boldsymbol{x}_j\|)$, is not suitable, since points belonging to different groups can be close to each other and have a large affinity. To deal with this issue, we propose to use an affinity based on angular information, *e.g.* defined by

$$A_{ij} = \left(\frac{\boldsymbol{x}_i^T \boldsymbol{x}_j}{\|\boldsymbol{x}_i\|_2 \|\boldsymbol{x}_j\|_2} \right)^2, \quad i \neq j. \qquad (5)$$

Figure 2 shows an example of line clustering in $\mathbb{R}^2$. Using the radial distance as in the algorithm of Sect. 3.1 yields a wrong partition of the data, whereas the affinity (5) allows the algorithm to recover the correct segmentation.
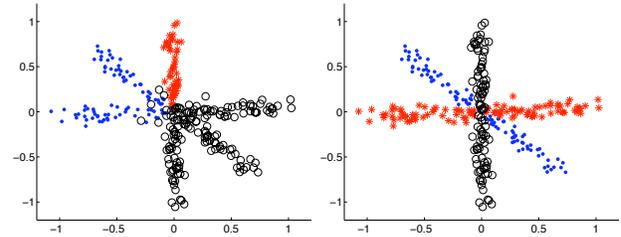
Figure 2. Spectral clustering of lines with a distance-based affinity confuses points from different subspaces (*left*), whereas the proposed angle-based affinity (5) enables to sharply discriminate them (*right*).

In the presence of noise, the values A_{ij} cannot be all 0 or 1. In this case, a better segmentation is obtained when the values A_{ij} for points of the same group are well separated from the values A_{ij} for points of different groups. To increase this separation, we propose to use

$$A_{ij} = \left(\frac{\boldsymbol{x}_i^T \boldsymbol{x}_j}{\|\boldsymbol{x}_i\|_2 \|\boldsymbol{x}_j\|_2} \right)^{2\alpha}, \quad i \neq j, \qquad (6)$$

where $\alpha \in \mathbb{N}^*$ is the parameter that tunes the sharpness of the affinity between two points. Figure 3 shows the affinity matrix corresponding to the data in Fig. 2 for different values of α. The values $\alpha > 1$ lead to a better separation of the groups, but when α is too large ($\alpha = 8$), affinities between points of the same group also vanish due to noise. In the experiments of Sect. 5, we use $\alpha = 4$. In practice, α should be tuned according to the noise level.

Figure 3. Affinity matrix of the data in Fig. 2 computed by (6) for $\alpha = 1$ (*top left*), $\alpha = 2$ (*top right*), $\alpha = 4$ (*bottom left*) or $\alpha = 8$ (*bottom right*). The darker, the larger the value of A_{ij}. Increasing α improves the intercluster separability, but can degrade the intracluster cohesiveness.

3.3. Subspaces of any dimension

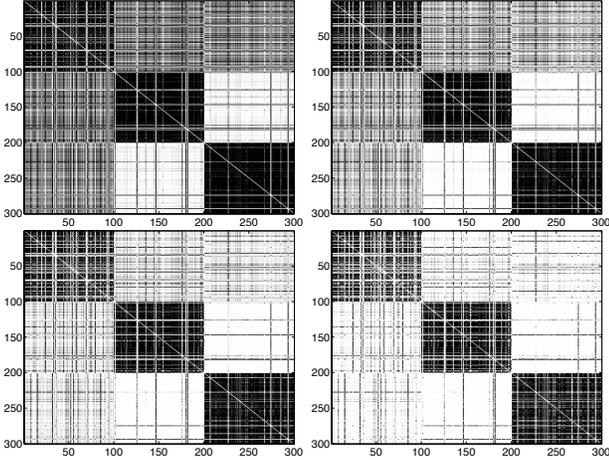Examples such as the ones in [9] show that spectral clustering has no difficulty in segmenting highly connected groups of points, *i.e.* groups for which there is a path between any pair of points in the corresponding similarity graph. The following studies under which conditions, the affinity (6) applied to samples drawn from subspaces fulfills this requirement.

Points of different groups. Assuming noiseless data and considering the subspace S_{c_i} that contains all the points of the same group as $\boldsymbol{x}_i$, we can express the affinity between $\boldsymbol{x}_i$ and any point not in S_{c_i} by

$$\forall \boldsymbol{x}_j \notin S_{c_i}, \quad \left(\frac{\boldsymbol{x}_i^T \boldsymbol{x}_j}{\|\boldsymbol{x}_i\|_2 \|\boldsymbol{x}_j\|_2} \right)^{2\alpha} = \cos^{2\alpha} \theta_{c_i c_j}, \quad (7)$$

where $\theta_{c_i c_j}$ is the first principal angle between the subspaces S_{c_i} and S_{c_j}, *i.e.* the one defined by $\cos \theta_{c_i c_j} = \max_{\boldsymbol{u} \in S_{c_i}} \max_{\boldsymbol{v} \in S_{c_j}} \boldsymbol{u}^T \boldsymbol{v} / \|\boldsymbol{u}\| \|\boldsymbol{v}\|$.

Points of the same group. Let $\mathcal{I}_c = \{i \mid \boldsymbol{x}_i \in S_c\}$ be the set of indexes of the points sampled from S_c. Then the angular density of the data for subspace S_c can be defined as

$$\cos^2 \delta_c = \min_{i \in \mathcal{I}_c} \left\{ \max_{j \in \mathcal{I}_c, j > i} (A_{ij})^{\frac{1}{\alpha}} \right\}, \quad (8)$$

where the range $j > i$ in the maximum operation removes the symmetry and ensures that the affinities are taken into account only once. This is required to detect angularly disconnected sets of points, which would yield a large value for $\max_{j \in \mathcal{I}_c} A_{ij}$, where $i \in \mathcal{I}_c$, but do not lead to a dense set when considered together.

The definition of the angular density (8) implies that for all $\boldsymbol{x}_i, \boldsymbol{x}_j \in S_{c_i}$, there is a sequence of points of indexes $j_k \in \mathcal{I}_{c_i}$, $k = 0, \ldots, K$, such that $j_0 = j$, $j_K = i$, and

$$\left(\frac{\boldsymbol{x}_{j_k}^T \boldsymbol{x}_{j_{k+1}}}{\|\boldsymbol{x}_{j_k}\|_2 \|\boldsymbol{x}_{j_{k+1}}\|_2} \right)^{2\alpha} \geq \cos^{2\alpha} \delta_{c_i}, \; k = 0, \ldots, K-1.$$
$$(9)$$

In other words, there is a path from $\boldsymbol{x}_j$ to $\boldsymbol{x}_i$ along the connected graph, that corresponds to the adjacency matrix $\boldsymbol{A}$ with all components below $\cos^{2\alpha} \delta_{c_i}$ set to zero.

Density vs. subspace angle condition. The affinity (6) cannot guarantee that a point of a subspace S_c has a higher affinity with other points of the same group than with points of other groups as in (4) (consider the extreme case, where each subspace is only sampled at two orthogonal points). However, it ensures that for $i = 1, \ldots, N$, there is a $j \in \mathcal{I}_{c_i}$, such that $A_{ij} > A_{ik}$ for all $k \notin \mathcal{I}_{c_i}$, under the following condition

$$\forall l \in \{1, \ldots, n\} \setminus c_i, \; \cos^2 \delta_{c_i} > \cos^2 \theta_{c_i l}, \quad (10)$$

or, as the single inequality,

$$\cos^2 \delta_{c_i} > \max_{l \in \{1, \ldots, n\} \setminus c_i} \cos^2 \theta_{c_i l}, \quad (11)$$

which links the angular density of the data of S_{c_i} with the subspace angles. The condition (11) guarantees that *each point $\boldsymbol{x}_i$ has a higher connection with its own group than with the others*. As will be seen in Sect. 5, in practice, this condition is easily satisfied for most motion segmentation data.

Example. Figure 4 shows the separation of two 2-D subspaces (planes) in $\mathbb{R}^D$ for $D = 3$ and 4. When $D = 3$, 47 points lying at the intersection of the two subspaces are misclassified, whereas only 6 points are misclassified when $D = 4$. In both cases, spectral clustering is applied by using the affinity (6) with $\alpha = 4$.

Comparison with Spectral Curvature Clustering (SCC). The main difference with SCC [2] is that we do not aim
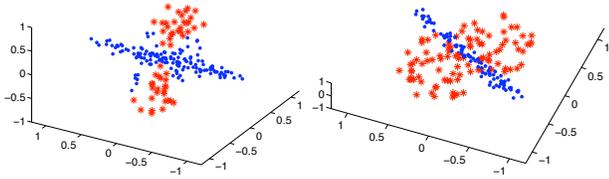
Figure 4. Clustering of planes in $\mathbb{R}^3$ leads to 47 errors around the intersection (*left*), whereas the separation in $\mathbb{R}^4$ misclassifies only 6 points (*right*, only the first 3 components are plotted).

at defining an optimal affinity measure between points belonging to subspaces. To do so, SCC has to consider $d + 1$ points simultaneously in order to determine if they belong to a d-dimensional linear subspace (there always exists a d-dimensional subspace containing an arbitrary set of d or less than d points). In the theoretical SCC, an affinity matrix $\boldsymbol{A} \in \mathbb{R}^{N \times N^d}$ is first computed from the polar curvature of the simplexes generated by all the combinations of $d + 1$ points as vertices. Spectral clustering is then applied to the N-by-N product matrix $\boldsymbol{A}\boldsymbol{A}^T$. To make this scheme practicable, one has to rely on a random sampling of $\boldsymbol{A}$ as proposed in [2].

At the contrary, the present paper proposes a suboptimal approach, in which only pairs of points are considered simultaneously, as in standard spectral clustering. This yields a low complexity of the affinity matrix computation, which only requires N^2 dot product operations between points. The price to pay for this gain in complexity is the requirement that each subspace must be sampled with a certain level of angular density in the data set. If this condition is met, the proposed affinity measure (6) is suitable for subspaces of any dimension, which makes the issue of estimating the correct dimension of the subspaces irrelevant.

4. Complete procedure

Dimension reduction. We consider the singular value decomposition (SVD) as our method of dimension reduction from $\boldsymbol{W} \in \mathbb{R}^{2F \times N}$ to $\boldsymbol{X} = [\boldsymbol{x}_1 \ \ldots \ \boldsymbol{x}_N]^T \in \mathbb{R}^{N \times D}$, which also appears as the standard procedure for motion segmentation in the literature [12, 13, 16, 17]. Note that this choice is also related to the factorization-based methods [3, 7], which use the SVD, $\boldsymbol{W} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$, to obtain a shape interaction matrix $\boldsymbol{Q} = \boldsymbol{V}\boldsymbol{V}^T$, that has a block-diagonal structure for independent motions and noiseless data. To deal with noise and dependencies, [18] uses the truncated SVD, $\boldsymbol{W} \approx \boldsymbol{U}_D\boldsymbol{\Sigma}_D\boldsymbol{V}_D^T$, and an interaction matrix defined through $Q_{ij} = \sum_{k=1}^{D} \exp((v_i^k - v_j^k)^2)$, where v_i^k is the kth component of the ith column of $\boldsymbol{V}_D$. More closely related to spectral clustering, [16] builds an affinity matrix from $\boldsymbol{Q}_D = \boldsymbol{V}_D\boldsymbol{V}_D^T$ as $A_{ij} = \exp(-1/|Q_{D_{ij}}|2\sigma^2)$ with an ad-

ditional parameter σ. In this framework, our approach with the affinity (6) can be seen as using the truncated SVD, but keeping the dot product structure of the shape interaction matrix as $\boldsymbol{Q} = (\tilde{\boldsymbol{V}}_D\tilde{\boldsymbol{V}}_D^T)^{2\alpha}$, where $\tilde{\boldsymbol{V}}_D$ is $\boldsymbol{V}_D$ with normalized rows. This normalization ensures that only angular information is taken into account.

Best ambient space dimension. According to Sect. 2.2, to ensure sufficient separability between n d-dimensional subspaces while maintaining the influence of noise as low as possible, we should choose the ambient space dimension $D = nd+1$. However, in the presence of degenerate motions, d may vary from a subspace to another and is unknown. Considering d_{min} and d_{max} as the minimal and maximal subspace dimensions (for motion segmentation, $d_{min} = 1$ and $d_{max} = 4$), we propose to tune D in the interval $nd_{min}+1 \leq D \leq nd_{max}+1$ by the following procedure.

Final algorithm.

1. Compute the SVD of the trajectory matrix, $\boldsymbol{W} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$, and set $D = nd_{min} + 1$.

2. Compute the affinity matrix $\boldsymbol{A}$ as in (6) from the N-by-D-data matrix $\boldsymbol{X} = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_D]$, built from the first columns $\boldsymbol{v}_i$ of $\boldsymbol{V}$.

3. Compute the matrix $\boldsymbol{L}$ as in the algorithm of Sect. 3.1 and extract its $n + 1$ leading eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{n+1}$.

4. Evaluate the relative gap $r_D = \frac{\lambda_n - \lambda_{n+1}}{\lambda_{n-1} - \lambda_n}$.

5. If $D = nd_{max} + 1$, go to step 6, otherwise $D = D+1$ and go to step 2.

6. Choose $D = \arg\max_k r_k$ and apply spectral clustering to $\boldsymbol{X} = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_D]$.

The intuition behind this procedure is that the best choice of D is the one that leads to the best estimation of the number of groups via r_D. Here, r_D acts as a model selection criterion, that determines the number of clusters n on the basis of the eigengap $|\lambda_n - \lambda_{n+1}|$ (this is justified in the ideal case, where $A_{ij} \in \{0, 1\}$, for which $\lambda_1 = \cdots = \lambda_n = 1$ and $\lambda_{n+1} < 1$ [9]). As n is known, r_D is used to track the values of D, that lead to the correct number of groups. As we use SVD, another approach would be to select D directly from the effective rank of $\boldsymbol{W}$ as in [17]. However, the criterion used in [17] to estimate the effective rank of $\boldsymbol{W}$ involves a parameter that depends on the noise level and seems to be difficult to tune in practice [12]. Instead, the proposed procedure does not involve additional parameters and specifically tunes D for a particular affinity measure in order to improve the subsequent clustering.

5. Experiments

In this section, some synthetic experiments are first provided to support some of the proposed heuristics. Then, the proposed algorithm is applied to real data to test its accuracy in motion segmentation tasks. In particular, we consider a benchmark, described in Sect. 5.2, for which results of various methods have been previously reported in the literature.

5.1. Synthetic experiments

To test the heuristics $D = nd+1$, we generate 100 random data sets of $N = 100n$ noiseless points uniformly sampled from d-dimensional subspaces of a D-dimensional ambient space (within the unit ball). In all these experiments, the random subspaces are generated with a minimum angle of $\pi/12$ between two subspaces and the data are mapped through SVD into $\mathbb{R}^D$. Table 1 shows the average error rates obtained by the proposed spectral clustering algorithm with $\alpha = 4$ and for $d = 3$ and different choices of n and D. These results support the ideas that: i) the separability of the subspaces depends on the ambient space dimension D, and ii) the number of subspaces also influences the optimal ambient space dimension (the dimension $D = nd + 1$ always yields a better separability than $D = 2d + 1$).

$n = 2$

D	$d+1$	$2d$	$2d+1$	nd	$nd+1$
Average	17.27	–	–	**0.00**	0.02
Median	17.75	–	–	0.00	0.00

$n = 3$

D	$d+1$	$2d$	$2d+1$	nd	$nd+1$
Average	44.78	2.69	0.29	1.33	**0.00**
Median	45.67	1.33	0.00	0.00	0.00

$n = 4$

D	$d+1$	$2d$	$2d+1$	nd	$nd+1$
Average	59.60	15.05	2.77	2.83	**0.01**
Median	60.00	13.38	1.63	0.00	0.00

Table 1. The ambient space dimension D influences the average and median error rates (in %) over 100 random trials, while its optimal value depends on the numbers of subspaces n.

5.2. Real data: the Hopkins155 database

The Hopkins database [12] includes 155 video sequences, each one involving either 2 or 3 motions (considering the background as a moving object), and is divided in three categories: *checkerboard*, *traffic* and *articulated*, of which representative samples appear in Fig. 5. Note that in the *articulated* sequences, some objects may be partially non-rigid, such as people walking. For each sequence, a set of N feature points were tracked using an automatic tracker, and errors in tracking were manually corrected. Thus, the

data correspond to the pixel coordinates of the N tracked feature points along F frames without outliers. Dimension reduction of these trajectories from $\mathbb{R}^{2F}$ to $\mathbb{R}^D$ is obtained by SVD as in [12].

Figure 5. Sample frames from the Hopkins database with ground truth segmentation (colors) for the 3 categories: checkerboard (left), traffic (middle) and articulated (right).

Reference model. The reference model included in the experiments is built from the full knowledge of the true segmentation. This model provides an approximate measure of the separability of the data under the affine camera model and for a given ambient space dimension. As in [12], the reference model fits a subspace to the data points in each group by using the SVD, resulting in least-squares estimates of the subspaces. Then, the data are re-segmented by assigning each point to the nearest subspace.

On the Hopkins database, similar experiments in [12] showed that, only for half of the sequences, a perfect segmentation could be obtained by separating linear subspaces in an ambient space of dimension $D = 5$. In this case, the reference model led to an average error rate of 2.35% on all sequences and of 1.66% and 4.73% on sequences of 2 and 3 motions, respectively. The following experiments will show that, by increasing the dimension of the ambient space, error rates twice as less can be obtained by the reference model and that, by estimating the best ambient space dimension for each data set, the proposed method can further improve the results without knowledge of the true segmentation.

Results. Tables 2 and 3 show the results obtained by the proposed spectral clustering (SC) approach for $\alpha = 4$ on sequences of 2 and 3 motions, respectively, whereas Table 4 provides the overall error rates on all sequences of the database. The columns of the Tables correspond to different choices of the ambient space dimension: $D = 5$ as in [12], $D = nd + 1$ (considering that all motions correspond to non-degenerate 4-D subspaces) and D tuned from the data by the procedure of Sect. 4 in the interval $[n+1, nd+1]$. The resulting error rates compare well with previously reported results from the literature, shown in Table 5. In particular, the proposed method with automatic tuning of the dimension D offers an overall error rate with 66% improvement over the best error rate obtained by Agglomerative Lossy Compression (ALC) in [11]. Moreover, Table 6 shows that

Checkerboard

	Ref_{nd+1}	SC_5	SC_{nd+1}	$\mathrm{SC}_{[n+1,nd+1]}$
Average	1.11	3.10	1.98	**0.85**
Median	0.23	0.00	0.24	0.00

Traffic

	Ref_{nd+1}	SC_5	SC_{nd+1}	$\mathrm{SC}_{[n+1,nd+1]}$
Average	0.15	1.05	10.42	**0.90**
Median	0.00	0.00	1.33	0.00

Articulated

	Ref_{nd+1}	SC_5	SC_{nd+1}	$\mathrm{SC}_{[n+1,nd+1]}$
Average	1.84	1.92	3.48	**1.71**
Median	0.00	0.00	0.88	0.00

All sequences

	Ref_{nd+1}	SC_5	SC_{nd+1}	$\mathrm{SC}_{[n+1,nd+1]}$
Average	0.93	2.46	4.30	**0.94**
Median	0.00	0.00	0.47	0.00

Table 2. Average and median error rates (in %) for sequences of **2 motions**. Subscripts indicate the ambient space dimension D.

Checkerboard

	Ref_{nd+1}	SC_5	SC_{nd+1}	$\mathrm{SC}_{[n+1,nd+1]}$
Average	2.79	10.29	**1.92**	2.15
Median	1.40	5.43	0.58	0.47

Traffic

	Ref_{nd+1}	SC_5	SC_{nd+1}	$\mathrm{SC}_{[n+1,nd+1]}$
Average	0.48	5.35	24.02	**1.35**
Median	0.34	0.00	26.60	0.19

Articulated

	Ref_{nd+1}	SC_5	SC_{nd+1}	$\mathrm{SC}_{[n+1,nd+1]}$
Average	4.26	**4.06**	**4.06**	4.26
Median	4.26	4.06	4.06	4.26

All sequences

	Ref_{nd+1}	SC_5	SC_{nd+1}	$\mathrm{SC}_{[n+1,nd+1]}$
Average	2.41	8.94	6.46	**2.11**
Median	0.99	4.93	1.06	0.37

Table 3. Average and median error rates (in %) for sequences of **3 motions**. Subscripts indicate the ambient space dimension D.

the corresponding average computing time (obtained on a 2.4GHz Core 2 Duo laptop with Matlab) is less than 1 second, whereas ALC needed 21 minutes (as reported in [11] on an unknown computer). In comparison to SCC [2] and LSA [17], the method is also faster, since it requires only N^2 dot product operations to build the affinity matrix.

Influence of α**.** Table 7 reports the error rates obtained for different values of α. Though this parameter appears to influence the segmentation accuracy, the overall error rate is still much less than the one obtained with ALC [11] for all $\alpha > 1$ in the set of tested values $\{1, 2, 4, 8\}$. Note that

Checkerboard

	Ref_{nd+1}	SC_5	SC_{nd+1}	$\mathrm{SC}_{[n+1,nd+1]}$
Average	1.53	4.90	1.96	**1.17**
Median	0.47	0.24	0.29	0.00

Traffic

	Ref_{nd+1}	SC_5	SC_{nd+1}	$\mathrm{SC}_{[n+1,nd+1]}$
Average	0.21	1.84	12.92	**0.98**
Median	0.00	0.00	1.53	0.00

Articulated

	Ref_{nd+1}	SC_5	SC_{nd+1}	$\mathrm{SC}_{[n+1,nd+1]}$
Average	2.21	2.25	3.57	**2.10**
Median	0.00	0.00	0.88	0.00

All sequences

	Ref_{nd+1}	SC_5	SC_{nd+1}	$\mathrm{SC}_{[n+1,nd+1]}$
Average	1.26	3.93	4.78	**1.20**
Median	0.18	0.00	0.56	0.00

Table 4. Average and median error rates (in %) for **all sequences** of 2 and 3 motions. The proposed method with D tuned in $[n+1, nd+1]$ outperforms others from the literature (see Table 5).

Overall error rate (in %)

	LSCC	SCC	LSA	ALC_5	ALC_{sp}
D	$nd+1$	$nd+1$	nd	5	D_{sp}
Average	10.39	8.66	4.87	3.83	**3.56**
Median	3.26	2.56	0.90	0.27	0.50

Computing time

	LSCC	SCC	LSA	ALC_5	ALC_{sp}
Average	2.24s	1.56s	9.47s	10min	21min

Table 5. Results of **methods from the literature** for **all sequences** of the Hopkins database. SCC [2] uses 3-D affine subspaces, while Linear SCC (LSCC) uses 4-D linear subspaces (results obtained with the Matlab code available at http://math.umn.edu/~glchen/scc). The results of LSA [17] are taken from [12] and those of ALC are taken from [11]. The smallest error, obtained by ALC in 21mn, is three times the one obtained by the proposed method in 0.74 seconds (see Tables 4 and 6).

Sequences of 2 motions

	$D=5$	$D=nd+1$	$D \in [n+1,nd+1]$
Average	0.25	0.30	0.56

Sequences of 3 motions

	$D=5$	$D=nd+1$	$D \in [n+1,nd+1]$
Average	0.51	0.59	1.35

All sequences

	$D=5$	$D=nd+1$	$D \in [n+1,nd+1]$
Average	0.31	0.36	0.74

Table 6. Computing time (in seconds) of the proposed method.

α also slightly influences the average computing time that ranges from 0.5 to 1.5 sec., respectively for $\alpha = 1$ and $\alpha = 8$.

Sequences of 2 motions

	$\alpha = 1$	$\alpha = 2$	$\alpha = 4$	$\alpha = 8$
Average	1.99	0.98	**0.94**	1.87
Median	0.00	0.00	0.00	0.00

Sequences of 3 motions

	$\alpha = 1$	$\alpha = 2$	$\alpha = 4$	$\alpha = 8$
Average	9.77	4.49	**2.11**	2.67
Median	1.80	1.02	0.37	0.20

All sequences

	$\alpha = 1$	$\alpha = 2$	$\alpha = 4$	$\alpha = 8$
Average	3.75	1.77	**1.20**	2.05
Median	0.24	0.00	0.00	0.00

Table 7. The average and median error rates (in %) with D automatically tuned in $[n+1, nd+1]$ remain low for a large range of the affinity parameter α.

6. Conclusions

This paper proposed a spectral clustering approach to the problem of separating subspaces of unknown dimensions (within approximate bounds). Beside first class results obtained on the Hopkins155 database for motion segmentation, the main contributions of the paper were to highlight the influence of the ambient space dimension in subspace separation applications and propose a heuristics to choose this dimension appropriately. However, when dealing with real data, involving possibly degenerate motions and subspaces of unknown dimensions, this heuristics may lead to an increased influence of the noise. Thanks to the speed of the proposed spectral clustering algorithm, a well-defined scheme has been proposed to select the best ambient space dimension for a particular data set. The final algorithm has been shown to be robust to different types of scenes and motions present in the Hopkins155 database, while remaining very efficient in terms of computing time.

Future work will study in a more formal way the relationships, empirically emphasized here, between the dimensions of the subspaces, the ambient space dimension and the density of the data. In addition, though good results were obtained for a large range of the affinity measure parameter α, we can expect further improvement by fine tuning α on each data set in a model selection framework.

References

[1] G. Chen and G. Lerman. Foundations of a multi-way spectral clustering framework for hybrid linear modeling. *Foundations of Computational Mathematics*, 2008.

[2] G. Chen and G. Lerman. Spectral curvature clustering (SCC). *Int. J. of Computer Vision*, 81(3):317–330, 2009.

[3] J. Costeira and T. Kanade. A multi-body factorization method for motion analysis. In *Proc. of the 5th Int. Conf. on Computer Vision (ICCV)*, pages 1071–1076, 1995.

[4] D. Cremers and S. Soatto. Motion competition: A variational approach to piecewise parametric motion segmentation. *Int. J. of Computer Vision*, 62(3):249–265, 2005.

[5] D. L. Donoho and J. Tanner. Counting faces of randomly-projected polytopes when the projection radically lowers dimension. Technical report, Stanford University, 2006.

[6] N. Ichimura. Motion segmentation based on factorization method and discriminant criterion. In *Proc. of the 7th IEEE Int. Conf. on Computer Vision (ICCV)*, pages 600–605, 1999.

[7] K. Kanatani. Motion segmentation by subspace separation and model selection. In *Proc. of the IEEE Int. Conf. on Computer Vision (ICCV)*, volume 2, pages 586–591, 2001.

[8] K. Kanatani and Y. Sugaya. Multi-stage optimization for multi-body motion segmentation. *IEICE Trans. on Information and Systems*, E87-D(7):1935–1942, 2004.

[9] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems (NIPS'02)*, volume 14, pages 849–856. MIT Press, 2002.

[10] J. Park, H. Zha, and R. Kasturi. Spectral clustering for robust motion segmentation. In *Proc. of the 8th Eur. Conf. on Computer Vision (ECCV)*, volume 3024 of *LNCS*, pages 390–401, 2004.

[11] S. R. Rao, R. Tron, R. Vidal, and Y. Ma. Motion segmentation via robust subspace separation in the presence of outlying, incomplete, or corrupted trajectories. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008.

[12] R. Tron and R. Vidal. A benchmark for the comparsion of 3-D motion segmentation algorithms. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 4, pages 1–8, 2007.

[13] R. Vidal and R. Hartley. Motion segmentation with missing data using powerfactorization and GPCA. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 310–316, 2004.

[14] R. Vidal, Y. Ma, and S. Sastry. Generalized principal component analysis (GPCA). *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(12):1945–1959, 2005.

[15] U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[16] H. Wang and P. F. Culverhouse. Robust motion segmentation by spectral clustering. In *Proc. of the British Machine Vision Conference*, pages 639–648, 2003.

[17] J. Yan and M. Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In *Proc. of the 9th Eur. Conf. on Computer Vision (ECCV)*, volume 3954 of *LNCS*, pages 94–106, 2006.

[18] L. Zelnik-Manor and M. Irani. Degeneracies, dependencies and their implications in multi-body and multi-sequence factorizations. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 287–293, 2003.