Robust GPCA Algorithm with Applications in Video Segmentation Via Hybrid System Identification

Kun Huang and Yi Ma

Coordinated Science Laboratory
Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801, USA
{kunhuang,yima}@uiuc.edu

Abstract

In this paper, we introduce the robust recursive GPCA algorithm that can segment a set of data points into multiple unknown linear models with different dimensions in the presence of noise and outliers. The algorithm is designed based on a robust model selection criterion for mixtures of subspaces called minimum effective dimension (MED). We apply the robust GPCA algorithm in a video segmentation problem via hybrid linear system identification, which has been successfully solved by the GPCA algorithm. Specifically, we model the hidden dynamics contained in the video sequence as a hybrid linear system without any input and develop two video segmentation schemes based on hybrid linear system identification, namely the direct segmentation and the segmentation for the embedded output data. Using the robust GPCA algorithms, both schemes generate satisfactory video segmentation results in a series of experiments. The direct segmentation requires less computational cost and the segmentation for the embedded output data captures the change in the hidden dynamics more faithfully.

1. Introduction

Generalized Principal Component Analysis (GPCA) is a novel algebraic geometric method for simultaneous data segmentation and multiple (linear) model estimation [15, 14, 8]. Although this method provides an elegant solution when the number of models is known and the data presents a moderate level of noise, in practical applications one must also deal with the following issues:

• Selectivity. For a given set of data, if the number of subspaces and their dimensions are not given a priori, then there might be more than one model that fits the data. For instance, sample points drawn from two intersecting lines in $\mathbb{R}^3$ can also be fit by the plane spanned by the two lines. In such cases, the purely algebraic method does not provide a guiding criterion for making a choice among the possible models.

• Robustness. As expected, the purely algebraic algorithm tolerates a moderate amount of noise in the data. Given that large amounts of noise and outliers are commonplace for many segmentation problems in vision, the robustness of the algorithm needs to be improved.

In a recent paper [8], we tackled these issues by introducing a model selection criterion that is specifically designed for GPCA-type models and developing a robust GPCA algorithm. In particular, we resolve the first difficulty by minimizing the “effective dimension” of the data, which only depends on the geometric configuration of the data and its model without assuming any probabilistic distributions for either the data or the models. As for the second issue, we propose to minimize the effective dimension with respect to a maximum error tolerance so as to improve the robustness of the resulting algorithm. We minimize the effective dimension over the subset of models that can be derived from the algebraic GPCA method hence restricting the possible solutions to those which are geometrically and algebraically correct. The combination of the maximum error tolerance with the algebraic GPCA method is also very natural and leads to a non-iterative algorithm, while other robust techniques typically require iterative optimization/minimization when searching for the best model. The result is a non-iterative and robust algorithm for estimating a mixture of subspaces of varying dimensions from given data.

This new robust GPCA algorithm has been applied to many difficult problems such as motion segmentation...
Therefore, it is likely that the segmentation of the video moving objects in the view, camera zooming in/out). The view or the action of the camera (e.g., number of moving objects in the view, camera zooming in/out). Therefore, it is likely that the segmentation of the video sequence is based on some hidden dynamics related to the observed motion and actions. This suggests that the video sequence can be modelled by some dynamical system(s).

Dynamical systems have been adopted to model video sequences in many studies. In [4], the image sequence is treated as the observations for a dynamical system and the reconstruction issue is converted into an observability problem. In [13], a linear dynamical system is used to model the image sequence. In [3], a linear dynamical system with noisy input is used to model the image sequence of dynamical texture. The dynamical system is characterized by its internal state and a Gaussian noise. The goal is then to characterize the random process for the Gaussian noise in order to synthesize new image frames. In [2], the image sequence is also modelled as a linear dynamical system and the goal is to use system identification approach to synthesize the sequence. In all the above approaches, the image sequence (video) is modelled by a single dynamical system. However, in practice, most video sequences contain segments with different motions and actions. For example, a video sequence of a traffic scene may contain a segment with only one car going across the view and another segment with multiple cars running at different speeds. It is therefore more appropriate to model the video sequence as a hybrid system switching among a pool of individual dynamical systems, each of which models one segment of video sequence. In the video sequence, the switching points usually correspond to some events relating to abrupt changes in the motion or actions.

One of the difficulties in applying the hybrid system model lies exactly in how to determine the switches, which, just like many other segmentation problems in computer vision, is traditionally considered as chicken-and-egg [10]. Fortunately, most of the video sequence modelling research adopt linear dynamical systems and we have shown in [16, 7] that the problem of identifying the hybrid linear system can be solved using the GPCA algorithm. Therefore, in this paper we deal with the video segmentation issue via hybrid linear system identification using GPCA. Specifically, since we usually only have the output (observation) of the system for the case of video sequence, we consider two simplified versions of the switch detection for the hybrid linear systems without input and apply them in video segmentation.

This paper is organized as following: in the next section we introduce the GPCA problem and our robust GPCA algorithm based on the new model selection criterion minimum effective dimension (MED). In Section 3, we introduce our results in applying the GPCA algorithm in hybrid linear system identification and present the two specific ways for handling the system without input. These methods will then be applied to the video segmentation problem in Section 4 with several experimental results. Finally we will discuss some further problems and future directions in Section 5.

2. Robust GPCA algorithm with minimal effective dimension criterion

2.1. Generalized principal component analysis (GPCA)

As mentioned above, GPCA is an algebraic method to fit a set of data points into an unknown number of linear models. The essence of the algebraic GPCA algorithm is to turn the multiple linear model fitting problem into a polynomial factorization problem and determine the number of models via a rank condition [15]. While the polynomial factorization problem is still difficult, in the algorithm it has been substituted by a polynomial differentiation step [14]. The details of the algorithm can be found in [14, 15, 8]. We just cite the main theorem based on which the algebraic GPCA algorithm was designed [14, 8]:

\[ f(X) = \prod_{j=1}^{n} (b_j^T X) = b_j^T v_n(X) = 0, \quad \forall X \in X, \quad (1) \]

where \( b_j \in \mathbb{R}^K \) is a normal vector to the \( j \)th subspace \( S_j \) and \( v_n(X) \in \mathbb{R}^{M_n} \), where \( M_n = \binom{n+K-1}{n} \)
and $\nu_n(X)$ is the vector of all monomials of degree $n$ in the entries of $X$, also known as the Veronese embedding of degree $n$. When $n$ is known, one can estimate the coefficients $b$ of all such polynomials from the null-space of the embedded data matrix $L_n = [\nu_n(X_1) \ldots \nu_n(X_N)]^T$, and the normal vectors $b_j$ to the $j$th subspace from the derivative of the polynomials $Df(X)$ at a point $X = Y_j$ in the $j$th subspace. A basis for the complement of $S_j$ can be obtained from $\text{span}_f Df(Y_j)$, hence the dimension of $S_j$ is given by $k_j = K - \text{rank}(\text{span}_f Df(Y_j))$.

![Recursive segmentation of the data points on one plane and two lines.](image)

Let us now consider the case in which $n$ is unknown. In this case, the main difficulty of directly applying Theorem 1 (possibly for multiple values of $n$) is that when the subspaces have different dimensions, there are polynomials of degree $d < n$ that also fit the data. In Figure 1, for instance, the interpretation of two planes instead of a plane and two lines leads to a polynomial of degree $2 < 3$. Therefore, we need to test if the data points on the two planes can be further segmented into lines. In other words, we can use a recursive scheme to search over all possible collections of subspaces that also respect the algebraic properties of the subspace structure, as stated by Theorem 1. However, a critical issue for the recursive approach is the stopping criteria for the recursion. For example, in Figure 1, how can the algorithm stop before “unreasonably” dividing the points on that plane into multiple strips? This issue is more significant when noise is present in the data set since the linear models usually can only approximate the data set and there can be many different ways to fit the noisy data points. The stopping criterion for the recursion in part determines if an accurate yet concise model can be found for the data set. Thus we are facing a general model selection issue. In the next section, we propose a minimal effective dimension criterion for selecting multiple linear models as in [8].

### 2.2. Minimal effective dimension (MED) criterion

**Definition 1 (Effective dimension)** Given $n$ subspaces $S_i = \bigcup_{i=1}^n S_i$ in $\mathbb{R}^K$ of dimension $k_i < K$, and $N_i$ sample points $X_i = \{X_i\}_{j=1}^{N_i}$ drawn from each subspace $S_i$, the effective dimension of the entire set of $N = \sum_{i=1}^n N_i$ sample points, $X = \bigcup_{i=1}^n X_i$, is defined to be:

$$\text{ED}(X, S) = \frac{1}{N} \sum_{i=1}^n k_i (K - k_i) + \frac{1}{N} \sum_{i=1}^n N_i k_i. \quad (2)$$

We contend that $\text{ED}(X, S)$ is the “average” number of (unquantized) real numbers that one needs to assign to $X$ per sample point in order to specify the configurations of the $n$ subspaces and the relative locations of the sample points in the subspaces. In the first term of equation (2), $k_i (K - k_i)$ is the total number of real numbers needed to specify a $k_i$-dimensional subspace $S_i$ in $\mathbb{R}^{K,2}$ in the second term of (2), $N_i k_i$ is the total number of real numbers needed to specify the $k_i$ coordinates of the $N_i$ sample points in the subspace $S_i$. In general $\text{ED}(X, S)$ can be a rational number, instead of an integer for a conventional “dimension.”

**Example 1 (One plane and two lines)** For the set of data points in Figure 1, suppose there are fifteen points in each line, and thirty points in the plane. When we use two planes to represent the data ($N = 2$), the effective dimension is: $\frac{1}{20} (2 \times 2 \times 3 - 2 \times 2^2 + 60 \times 2) = 2.07$; when we use one plane and two lines ($N = 3$), the effective dimension is reduced to: $\frac{1}{30} (2 \times 2 \times 3 - 2^2 - 2 \times 1 + 30 \times 1 + 30 \times 2) = 1.6$. In general, if the number of points $N$ is arbitrarily large (say approaching to infinity), depending on the distributions of points on the lines or the plane, the effective dimension may approach arbitrarily close to either 1 or 2, which reveals the true dimensions of the subspaces.

As suggested by this intuitive example, the subspace-structure that leads to the minimum effective dimension normally corresponds to a more “efficient” and hence more “natural” interpretation of the data in the sense that it achieves the best compression (or dimension reduction) among all permissible subspace-structures. This observation inspires us to pursue minimum effective dimension (MED) when fitting data points into

---

1. For affine subspaces (which do not necessarily pass the origin), we first make them subspaces using the homogeneous coordinates.
2. $k_i(K - k_i)$ is the dimension of the Grassmannian manifold of $k_i$-dimensional subspaces in $\mathbb{R}^K$. To specify a subspace, one can use the so-called Grassmannian coordinates which need exactly $k_i(K - k_i)$ entries: starting with a $K \times k_i$ matrix whose columns form a basis for the subspace, perform column-reduction so that the first $k_i \times k_i$ block is the identity matrix. Then, one only needs to give the rest $(K - k_i) \times k_i$ entries to specify the subspace.
multiple linear models. For noisy data, the resulting “minimum effective dimension” of the optimal model in general depends on the given error tolerance. In the extreme, if the error tolerance is arbitrarily large, the “optimal” subspace-model for any data set can simply be the zero-dimensional origin; if the error tolerance is zero instead, for data with random noise, most sample points need to be treated either as one-dimensional subspaces in \( \mathbb{R}^K \) or as points in the ambient space \( \mathbb{R}^K \) directly. Both ways bring the effective dimension up close to \( K \). Therefore, we can define the minimum effective dimension to allow the chosen model to tolerance certain error \( \tau \). We therefore define the minimum effective dimension with error tolerance as:

\[
\text{MED}(X, \tau) = \min_{S: \|X - X\|_\infty \leq \tau} \text{ED}(\hat{X}, S),
\]

where \( \hat{X} \) is the projection of \( X \) onto the subspaces \( S \), and the error norm \( \| \cdot \|_\infty \) indicates the maximum norm:

\[
\|X - \hat{X}\|_\infty = \max_{X \in \mathcal{X}} \|X - \hat{X}\|.
\]

The aim of robust GPCA model selection is then to find a multiple-subspace model which leads to the lowest effective dimension for a given error tolerance, hence the minimum effective dimension (MED) criterion. The relationship between the MED criterion and other model selection criteria such as minimum message length (MML) [17], minimum description length (MDL) [12], Bayesian information criterion (BIC), Akaike information criterion (AIC) [1], and Geometric AIC (G-AIC) [9] can be found in [8].

### 2.3. A robust recursive GPCA algorithm

As we have discussed in the previous sections, in order to implement a recursive GPCA algorithm, a stopping criterion is critical. The MED criterion just serves this purpose in two aspects: First, for every level of recursion, by checking if the effective dimension is reduced, the algorithm can automatically decide if the recursion should be carried on further. Second, since the MED is with respect to a specified error tolerance, with this error tolerance the algorithm can robustly assign the data points into multiple linear models. The robust recursive algorithm designed in such way can both respect the algebraic properties of the subspace structure, as stated by Theorem 1, and minimize the effective dimension [8, 15]. Another important feature of this algorithm is that it can automatically search for the number of groups \( n \) and their dimensions at each level recursion. It achieves this by testing if it can find \( n \) distinct linear models of different dimensions so that the data points can be assigned to these models within the error tolerance for different choice of \( n \). The details of the algorithm can be found in [8] and are omitted here.

**Comment 1 (Outliers)** To further improve the robustness of the algorithm, we can also assign a permissible percentage of outliers that may not be fit by the subspace-models with the given error tolerance \( \tau \). Setting aside the outliers will allow the GPCA search process to identify the dominant subspace-structure that the majority of the sample points admit, without being “side-tracked” by a few very bad data points. This is important especially when the permissible error tolerance has to be small in some problems. We typically allow about 20% of outliers in our experiments.

### 3. Hybrid LTI system identification and video segmentation

Both the algebraic and the robust GPCA algorithms have been successfully applied to hybrid linear time-invariant (LTI) system identification. In [16], the algebraic GPCA algorithm is adopted to identify the switches for hybrid linear systems with input/output models. In [7], we have shown that the robust recursive GPCA algorithm can detect the switches for hybrid linear time invariant (LTI) system with state space models. Here we briefly introduce the main techniques used for hybrid LTI system identification. We assume that the state-space LTI models involved have the same number of inputs and outputs but probably different orders.\(^4\)

\[
\text{Hybrid LTI: } \begin{cases}
x_{t+1} = A_{\lambda(t)}x_t + B_{\lambda(t)}u_t + v_t, \\
y_t = C_{\lambda(t)}x_t + D_{\lambda(t)}u_t + w_t,
\end{cases}
\]

where the discrete switching state \( \lambda(t) \) is a piecewise-constant function taking values in \( \{1, 2, \ldots, s\} \). \(^5\) \( A_i \in \mathbb{R}^{n_i \times n_i}, B_i \in \mathbb{R}^{n_i \times l}, C_i \in \mathbb{R}^{m \times n_i}, \) and \( D_i \in \mathbb{R}^{m \times l} \) are the system matrices for the \( i \)th LTI system, \( u_t \in \mathbb{R}^l, y_t \in \mathbb{R}^m, \) and \( x_t \in \mathbb{R}^{n_i} \) are the input, output, and state of the hybrid system, and \( v_t \) and \( w_t \) are the model and output noise, respectively. Regarding the above hybrid system, we are interested in the following problem:

**Problem 1 (Hybrid ID)** Given only the input \( u_t \) and the output \( y_t \) of system (4) over a period of time and an upper bound \( n \) on the order of all the LTI systems, determine:

\(^4\)One can always view the states of all of the systems, independent of order, as being embedded in the state-space of their highest order. For the method presented in this paper, it does not make any difference if continuity in the states is imposed or not.

\(^5\)We assume that the hybrid system is switching relatively slowly from one system to another.
1. the times where the discrete switching state changes value, i.e. the piecewise-constant function $\lambda(t)$

2. the number of discrete states $s$, i.e. the number of LTI systems involved in the hybrid system,

3. the unknown order $n_i$ of each LTI system,

4. ultimately, all of the system parameters $(A, B, C, D)$ and states $\{x_t\}$.

Notice that the last problem above is just the classic system identification problem if the first three problems are solved. Thus our study is focused on solving the first three problems.

As shown in [7], we can embed the input $u_t$ and output $y_t$ over a time interval $[t, t + k - 1]$ for a large enough $k$ into a high-dimensional space such that the embedded input/output data points reside on a low-dimensional subspace for each individual linear system. Therefore, the problem of determining which input/output data point is associated to which individual linear system becomes the problem of segmenting the embedded data points into multiple linear subspaces, which can just be solved by the GPCA algorithm. In [7], two ways of embedding the input/output data are presented. In this paper, we consider a simpler case.

3.1. Video segmentation and hybrid linear models

If we model the video sequence as a hybrid linear system, a major problem is that the input to the system is unknown. In [18], the input is modeled as a system feedback. In [3], the input was treated as part of the noise generated by a random process. In [2], the case of no input is also considered. Here we adopt the simplest assumption of no input to the system. Then the system equation for any individual linear system in (4) can be reduced to

Zero Input LTI: \[
\begin{align*}
  x_{t+1} &= Ax_t + v_t, \\
  y_t &= Cx_t + w_t, 
\end{align*}
\]

Note that for the video sequence, while the original output are the images which are points in a very high dimensional space, we can perform dimensionality reduction via PCA so that the final output $y_t$ has a very low dimension $m$.

For many video segments, the motion in the view and the action of the camera are simple and arguably can be described by a linear system with a very low order. Thus if the dimension of the output $m$ is larger than $n$, the output $y_t \in \mathbb{R}^m$ for that linear system resides on a subspace spanned by the columns of $C$. The subspace dimension is more than $n$. For a video sequence modeled by multiple linear systems with different matrices $C_i$’s, the output usually corresponds to multiple such linear subspaces associated with the different $C_i$’s. Therefore, if $m$ is large enough, with the robust GPCA algorithm we can segment the output $y_t$ into multiple subspaces spanned by the columns of the matrices $C_i$’s. We denote this video segmentation scheme as the direct segmentation.

There are two potential problems with this approach. First, it may not be able to capture all the changes in dynamics within the sequence. Second, over-splitting of the video segments is possible since the local linear structure of the system state $x_t$’s can be reflected in $y_t$ through the matrix $C$. Despite these possible drawbacks, we expect that many important events, especially the abrupt action of the camera such as change of view point and zooming are still be detected.

If we want to detect more detailed changes in the dynamics, we can embed the system output similar to the direct embedding method discussed in [7]. For a fixed $k$, let $Y_k = [y^T_t, y^T_{t+1}, \ldots, y^T_{t+k-1}]^T \in \mathbb{R}^{mk}$. We then have

$$Y_k = \Gamma_k x_t = \begin{bmatrix}
  C \\
  CA \\
  \vdots \\
  CA^{k-1}
\end{bmatrix} x_t. \quad (6)$$

The matrix $\Gamma_k$ is the observability matrix for this linear system. For $k \geq n$ with $n$ being the order of the system $(x_t \in \mathbb{R}^n)$, we have

$$\text{rank}(\Gamma_k) = q. \quad (7)$$

where $q$ is the dimension of the observable subspace for the linear system. For an observable system, $q = n$. Therefore, for a hybrid LTI system without input, most of the embedded output data points $Y_k$ belong to the different observable subspaces for the multiple linear systems. Hence, with the robust GPCA algorithm we can identify the multiple subspaces and further identify the switches among multiple systems. Notice that if the time interval $[t, t + k - 1]$ straddles a switch point between any two individual linear systems, the data point $Y_k$ does not belong to any of the observable subspaces and should be treated as outliers. This reflects the importance of handling outliers for the GPCA algorithm.

4. Experiments

In this section, we illustrate the above two schemes of video segmentation by testing them on a video sequence. These approaches have also been applied on several other video sequences with similar results.
The “table tennis” sequence with 150 image frames has been widely used in testing video segmentation and compression algorithms. This sequence contains the motions of two table tennis players and different camera motions. We use the same parameters for the robust GPCA algorithm for all the experiments in this paper. The error tolerance is set to be 0.05 radian and 20% of outlier points are allowed in each level of recursion.

For the sake of simplicity, we convert all the color image frames into gray scale. These images are of size $240 \times 352$. If we treat the images as the observed data points, our observation is of $240 \times 352 = 84480$ dimension. So our next step is dimensionality reduction. By performing PCA on the 150 image frames and retaining only the first $m$ ($m < 10$) principal components corresponding to every image frame, we obtain our observation $y_t \in \mathbb{R}^m$ of a very low dimension. Figure 2 displays the first three principal components for these data points. Interestingly, the distribution of these data points shows a clear structure.

![Figure 2: The first three principal components for the 150 image frames in the “table tennis” sequence with one point corresponding to one image frame.](image)

4.1. Direct segmentation of the video sequence

We begin with the scheme of direct segmentation on our observation $y_t$ ($1 \leq t \leq 150$). First, we choose $m = 2$. Since the value of $m$ is relatively small, we expect that it may not be able to capture the complicated dynamics in the scene. The segmentation result indeed confirms our guess. As shown in Figure 3, the image frames are segmented into different groups on some one dimensional linear subspace (lines) by the robust GPCA algorithm except the data points in group 1. The group 1 contains all the outliers, i.e. the data points that cannot be allocated into any linear models. A check on the original image frames corresponding to these outliers (from the 34th frame to the 60th frame) shows that this segment contains complicated motion of serving the ball and the zooming out action of the camera. It is not a surprise that the two-dimensional output cannot characterize the complicated dynamics. However, even at this low dimension, the segmentation still reveals some of the abrupt change of the scene. For example, the switch between the 74th and 75th frames reflects a change in the appearance where the image of the referee wearing a blue suit begins to appear in the 75th frames. The switch between the 89th and 90th frames corresponds to the sudden change of view point from one end of the table to the other end of the table (see Figure 4. Similar view point change occurs at the switch between the 148th and 149th frames.

![Figure 3: The direct segmentation of the 150 image frames into multiple groups for the case of $m = 2$. The $x$-axis is the index for the image frames. The $y$-axis is the group (model) number to which each image frame is assigned. The group 1 contains all the outliers.](image)

Next we increase $m$ to 3. The segmentation results by the robust GPCA algorithm is shown in Figure 5, in which there is no outlier. There is a big difference between Figure 5 and Figure 3. In Figure 5, the images between the 27th and 88th frames are segmented into eight segments. Notice that before the 27th frame, the camera is almost static and the motion in the view is just the simple bouncing of the ball. Between the 27th and 88th frames, the camera zooms out and the content includes a complicated ball serving motion. As shown in Figure 6, several of these short segments actually correspond to either the ball moving up and down, or different actions in serving the ball.

This can be explained by the over-splitting behavior of the direct segmentation as discussed in the previous section.

Another difference between the two segmentations corresponding to $m = 2$ and $m = 3$ is that the switch between the 74th and 75th frames for $m = 2$ does not
4.2. Segmentation for embedded output data

The above experiments demonstrate the direct segmentation of the video sequences. While the direct segmentation can capture many changes in scene contents, motions, and camera actions, it does not give a clear understanding of the change in the dynamics of video sequence. In order to really grasp the change in the system dynamics, we need to perform the segmentation using the embedded output data as described in Section 3.

We still choose $m = 3$. The $k$ in (6) is set as 2. The embedded output data $Y_t$ is then of dimension $mk = 3 \times 2 = 6$. Figure 7 shows the segmentation of the embedded data $Y_t$ into multiple linear models. No point is considered as outlier. Despite occasional misassignment, the segmentation result displays several advantages over the direct segmentation. First, there is a clear switch between the 23rd and 24th frames. It is very significant since the 24th frame is exactly the beginning of the zooming out action of the camera. Furthermore, a careful inspection of the image frames indicates that the zooming out process has two phases. Before the 32nd frame, the zooming action is slow. It then becomes rapid after the 32nd frame. This change is also detected by the algorithm. The other segments before the 89th frame all correspond to different phases for serving the ball with clear different motion dynamics. Such a segmentation preserves the action much better than that in direct segmentation. For example, the 6th group in Figure 7 is a combination of two groups in Figure 5, which avoids the over-splitting problem in the direct segmentation.

The segments also have different dimensions that correspond to different complexities in the dynamics. The group 2 and group 8 are both of dimension 1. This implies that they have relatively simple dynamics. In fact, the camera is static in both segments. While the motion in the group 2 is complex, it occupies only a small area of the image and hence the signal is small. All the other groups (except group 1 and group 3) are of dimension 2, which is consistent with the fact they contain motion of the player and zooming action of the camera. As we have discussed in Section 3, the dimensions of the segments relate to the dimensions of the observable subspaces for the linear dynamical systems and further correspond to the orders of these systems. Therefore, this video sequence can be modeled by a set of linear dynamical systems of order 1 or 2. This also
explains why \( m = 3 \) suffices for direct segmentation despite the over-splitting.

As can be conceived, the direct segmentation requires less computation due to the lower dimensionality of the data points comparing to the embedded output data. On a Compaq laptop with a Pentium IV 1.9GHz CPU and 512Mb memory, it takes 7 seconds for the direct segmentation of the testing sequence and 11 seconds for segmenting the embedded data.

5. Conclusion and discussion

In this paper, we introduce the robust recursive GPCA algorithm that can segment a set of data points into multiple unknown linear models with different dimensions in the presence of noise and outliers. To solve the model selection issue related to the recursive GPCA algorithm, we present the minimum effective dimension (MED) criterion as a model selection criterion specially designed for multiple linear models with a given error tolerance. This algorithm has been applied in many applications such as image representation and hybrid system identification.

Here we extend the technique of hybrid linear system identification in video segmentation by considering the special case of no input. Two segmentation schemes are developed under the assumption of no input. In the experiments, both schemes generate satisfactory video segmentation results despite the simplicity of the assumption of no input. The experiments reveal the advantages of each scheme. When the dimension of the output is large enough, the direct segmentation can detect the changes of the appearance in contents and abrupt motion and action of the camera with lower computational cost. The segmentation with the embedded output data, however, more faithfully capture the change in the system dynamics and reveal the complexity of the systems.

Our future work includes noise modelling, video synthesis based on the segmentation, and dynamic texture segmentation.

References