

## Review Article

# A Review of Subspace Segmentation: Problem, Nonlinear Approximations, and Applications to Motion Segmentation

**Akram Aldroubi**

*Department of Mathematics, Vanderbilt University, Nashville, TN 37212, USA*

Correspondence should be addressed to Akram Aldroubi; [akram.aldroubi@gmail.com](mailto:akram.aldroubi@gmail.com)

Received 4 November 2012; Accepted 20 December 2012

Academic Editors: M. A. Nappi and J.-G. Wang

Copyright © 2013 Akram Aldroubi. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The subspace segmentation problem is fundamental in many applications. The goal is to cluster data drawn from an unknown union of subspaces. In this paper we state the problem and describe its connection to other areas of mathematics and engineering. We then review the mathematical and algorithmic methods created to solve this problem and some of its particular cases. We also describe the problem of motion tracking in videos and its connection to the subspace segmentation problem and compare the various techniques for solving it.

## 1. Introduction

The subspace clustering problem is fundamental in many engineering and mathematics applications [1–11]. It can be described as follows: let  $\mathcal{U} = \bigcup_{i=1}^M S_i$  be the nonlinear set consisting of a union of  $M$  subspaces  $\{S_i \subset \mathcal{H}\}_{i=1}^M$  of a Hilbert or a Banach space  $\mathcal{H}$ . Let  $\mathbf{W} = \{w_j \in \mathcal{H}\}_{j=1}^N$  be a set of data points drawn from  $\mathcal{U}$ . The subspace segmentation (or clustering) problem is then to determine  $\mathcal{U}$  (equivalently determine  $S_i$  for  $i = 1, \dots, M$ ), from the data  $\mathbf{W} = \{w_j \in \mathcal{H}\}_{j=1}^N$ , that is, to

- (1) determine the number of subspaces  $M$ ;
- (2) find an orthonormal basis for each subspace  $S_i$ ,  $i = 1, \dots, M$ ;
- (3) group the data points belonging to the same subspace into the same cluster.

The data  $\mathbf{W}$  is often corrupted by noise; it may have outliers or some of the data vectors  $w \in \mathbf{W}$  may have missing entries. Therefore, any technique for solving the subspace segmentation problem above must be robust and stable for the aforementioned nonideal cases.

Depending on the application, the space  $\mathcal{H}$  can be finite or infinite dimensional. For example, the set of all two dimensional images of a given face  $i$ , obtained under different

illuminations and facial positions, can be modeled as a set of vectors belonging to a low dimensional subspace  $S_i$  living in a higher dimensional space  $\mathcal{H} = \mathbb{R}^D$  [12–14]. For this case, a set of such images from  $M$  different faces is a union  $\mathcal{U} = \bigcup_{i \in 1}^M S_i$ . Another application in which a union of subspaces provides a good model is the problem of motion tracking of rigid objects in videos. For this situation (further developed below), a 4-dimensional subspace is assigned to each moving object in a space  $\mathcal{H} = \mathbb{R}^{2F}$ , where  $F$  is the number of frames in the video. Examples where  $\mathcal{H}$  is infinite dimensional arise in sampling theory, and in learning theory [15–19]. For example, signals with finite rate of innovations are modeled by a union of subspaces that belongs to an infinite dimensional space such as  $L^2(\mathbb{R}^D)$  [2, 3, 20, 21].

*1.1. Known Number of Subspaces  $M$  and Dimensions  $d_i$ .* In some subspace segmentation problems, the number  $M$  of subspaces or the dimensions of the subspaces  $\{d_i\}_{i=1}^M$  are known or can be estimated [1, 8, 22, 23]. In these cases, the subspace segmentation problem, for both the finite and infinite dimensional space cases, can be formulated as follows.

Let  $\mathcal{H}$  be a Hilbert space,  $\mathbf{W} = \{f_1, \dots, f_m\}$  a finite set of vectors in  $\mathcal{H}$ ,  $\mathcal{C}$  a family of closed subspaces of  $\mathcal{H}$ , and  $\mathcal{S}$  the set of all sequences of elements in  $\mathcal{C}$  of length  $M$

(i.e.,  $\mathfrak{C} = \mathfrak{C}(M) = \{\{S_1, \dots, S_M\} : S_i \in \mathcal{C}, 1 \leq i \leq M\}$ ). The subspace segmentation problem formulation as a minimization problem is as follows.

*Problem 1* (optimization formulation of the subspace segmentation problem). (1) Given a finite set  $\mathbf{W} \subset \mathcal{H}$ , a number  $p$  with  $1 \leq p \leq \infty$ , and a fixed integer  $M \geq 1$ , find the infimum of the expression

$$e(\mathbf{W}, \mathbf{S}) := \sum_{f \in \mathbf{W}} \min_{1 \leq j \leq M} d^p(f, S_j), \quad (1)$$

over  $\mathbf{S} = \{S_1, \dots, S_M\} \in \mathfrak{C}$ , and  $d(x, y) := \|x - y\|_{\mathcal{H}}$ .

(2) Find a sequence of  $M$ -subspaces  $\mathbf{S}^o = \{S_1^o, \dots, S_M^o\} \in \mathfrak{C}$  (if it exists) such that

$$e(\mathbf{W}, \mathbf{S}^o) = \inf \{e(\mathbf{W}, \mathbf{S}) : \mathbf{S} \in \mathfrak{C}\}. \quad (2)$$

An example in finite dimensions is when  $\mathcal{H} = \mathbb{R}^D$  and  $\mathcal{C}$  is the family of all subspaces of  $\mathbb{R}^D$  of dimensions  $d$  no greater than  $r \leq D$ . For this case, when  $M = 1$ , and  $p = 2$ , this is a well-known least square problem that can be solved using the singular value decomposition technique [24]. An example in infinite dimensions is when  $\mathcal{H} = L^2(\mathbb{R}^D)$  and  $\mathcal{C}$  is a family of closed, shift-invariant subspaces of  $L^2(\mathbb{R}^D)$  that are generated by at most  $r < \infty$  generators [2]. Typical shift-invariant spaces with one generator are for example the space of bandlimited functions, generated by integer shifts of the generator function  $\text{sinc}(x) = \sin(x)/x$ . Other important shift invariant spaces are the spline spaces  $\mathcal{L}_n$  generated by the B-spline functions  $\beta^n$  of degree  $n$  [25, 26]. In these cases the subspaces in  $\mathcal{C}$  are also infinite dimensional subspaces of  $L^2$ . Thus, even in the case where  $M = 1$  and  $p = 2$ , this (least squares) problem is much more difficult than its finite dimensional counterpart. It should be noted that when  $M > 1$  and for any  $0 < p \leq \infty$  Problem 1 is neither linear nor convex [27, 28]. In the presence of outliers, it has been proven that the best value for  $p$  is  $p = 1$  [27, 28], and a good choice for light-tailed noise is  $p = 2$ . There are more general versions of Problem 1, for example, the Hilbert space  $\mathcal{H}$  can be replaced by a Banach space  $\mathcal{B}$ ; moreover, the family  $\mathcal{C}$  can be replaced by the more general type of family  $\mathcal{C}_1 \times \mathcal{C}_2, \dots, \mathcal{C}_M$  [22].

*1.2. Applications and Connection to Other Areas.* The subspace segmentation problem has connections to several active areas of research, including learning theory, compressed sampling, and signal processing in general [2, 3, 17, 21, 29–32]. Moreover, it is relevant to several computer vision applications including motion tracking of rigid objects in videos and facial recognition [1, 4, 14, 33–38].

*1.2.1. Connection to Compressed Sampling.* In compressed sampling, the goal is to find an unknown vector  $x \in \mathbb{R}^D$  from a small set of linear measurements  $\{y_i = a_i^T x\}_{i=1}^m$ ,  $m \ll D$ , where  $a_i$  are known sampling vectors. Clearly, this problem has a solution only if some extra information is known about  $x$  and if the sampling vectors  $a_i$ s are well chosen. In compressed sampling, the assumption is that, in a suitable basis, or frame, the unknown vector  $x$  is  $s$ -sparse

or nearly  $s$ -sparse (compressible), with  $s \ll D$  [30, 39–44]. This means that in a suitable basis or frame the vector  $x$  has at most  $s$  nonzero components, or, in the compressible assumption, that  $x$  has at most  $s$  large components. This sparsity assumption (or compressibility assumptions) implies that the vector  $x$  must belong (or must be close to) a union of subspaces of dimensions at most  $s$ . Thus, finding the sparse model for a class of signals can be obtained by solving the subspace segmentation problem in the special case where  $\mathcal{H} = \mathbb{R}^D$ , and where the  $\mathcal{C}$  is the class of subspaces of  $\mathbb{R}^D$  of dimensions at most  $s$ , and  $M = D \binom{D}{s}$  [45, 46].

*1.2.2. Connection to Learning Theory and Data Mining.* In many learning theory problems, a class of data may form a complex structure embedded in a high dimensional space  $\mathbb{R}^D$  [47–53]. In the neighborhood of each data point, the structure may be modeled by a local tangent space, or a union of tangent spaces whose dimensions are much smaller than the dimension of the ambient space  $\mathbb{R}^D$  [16]. The global shape of the data model can then be obtained from the observed data points by solving Problem 1.

*1.2.3. Connection to Signal Processing.* In signal processing, signals are often modeled by an infinite dimensional shift-invariant subspace of  $L^2(\mathbb{R}^d)$  [15, 54–61]. For example, the classical shift-invariant space is the space of bandlimited functions  $\text{PW}_{1/2}$ , also known as the Paley-Wiener space [62–67]. This is the space generated by the function  $\text{sinc} = \sin(\pi x)/\pi x$  and its integer shifts. Multiresolution and wavelet spaces are also shift-invariant spaces that are often used in signal processing applications. Choosing the model for a class of signals can be cast in terms of finding the solution of 1 from observed data. Unlike the compressed sampling or learning theory discussed earlier, in this situation the class  $\mathcal{C}$  consists of infinite dimensional subspaces of  $\mathcal{H} = L^2$  and therefore are more difficult to deal with even for a single shift-invariant subspace model ( $M = 1$ ) [68]. The case in which a signal model is not a single subspace but a union of several of such subspaces is natural as in the case of signals with finite rate of innovation [69–73].

*1.2.4. Application to Motion Tracking in Video.* The problem of tracking rigid moving objects in a video can be formulated as a subspace segmentation problem [33, 35, 74–77]. Let  $x_j(p)$ ,  $y_j(p)$  be the Cartesian coordinates of a point  $p$  of a moving object  $O$  in frame  $j$  of a video. By concatenating all the coordinates of  $p$  into a single vector

$$w_p = (x_1(p), y_1(p), x_2(p), y_2(p), \dots, x_F(p), y_F(p))^t, \quad (3)$$

we obtain the so-called *trajectory vector* of  $p$  whose length is  $D = 2F$  where  $F$  is the number of frames in the video. It can be shown that, for rigid bodies, the trajectories of any point of object  $O$  belong to a subspace  $S_O$  of  $\mathbb{R}^D$  of dimensions no greater than 4. Thus, if  $\mathbf{W} = \{w_1, \dots, w_N\}$  is a set of trajectory vectors from a set video of  $M$  moving objects (background is one such objects), then the set  $\mathbf{W}$  belongs to a union of  $M$

subspaces of dimensions at most 4. Thus, solving the subspace segmentation problem in this situation consists in using the data  $\mathbf{W}$  to find the subspaces, and then grouping together the trajectory vectors  $w_i$  that belong to the same objects according to the subspace they belong to. It can also be shown that human facial motion and other nonrigid motions can be approximated by linear subspaces [78].

*1.2.5. Application to Face Recognition.* It has been shown that the set of all two-dimensional images of a given face  $i$ , obtained under different illuminations and facial positions, can be modeled as a set of vectors belonging to a low dimensional subspace,  $S_i$ , living in a higher dimensional space  $\mathbb{R}^D$  [14]. A set of such images from different faces are then a union  $\{S_i \subset \mathcal{H}\}_{i=1}^M$ , where each face is associated with a give face.

*1.3. Dimensionality Reduction.* Since the data  $\mathbf{W}$  may live in a very high dimensional space  $\mathbb{R}^D$ , but  $\mathcal{U} = \bigcup_{i=1}^M S_i$  may consist of spaces  $S_i$  with dimension  $d_i \ll D$  and  $M \ll D$ , the subspace clustering problem can be solved in a smaller dimensional space  $D_e$ , the *effective dimension*. Specifically, if  $\sum_{i=1}^M d_i \ll D$ , then the data  $\mathbf{W}$  can be projected on a space of dimension  $D_e = \sum_{i=1}^M d_i$ , where the projection is not necessarily an orthogonal projection, but any “good” linear process that maps the data to another (low dimensional) space, for example, random projection [79–82]. As a result of projecting  $\mathcal{U} = \bigcup_{i=1}^M S_i \subset \mathbb{R}^D$  and the data  $\mathbf{W}$ , we get the set  $\tilde{\mathcal{U}} = \bigcup_{i=1}^M \tilde{S}_i \subset \mathbb{R}^{D_e}$  and the data  $\tilde{\mathbf{W}} \subset \mathbb{R}^{D_e}$ . It is now possible to solve the subspace segmentation problem with data  $\tilde{\mathbf{W}} \subset \mathbb{R}^{D_e}$  instead of  $\mathbf{W}$  and use the segmentation in the low dimensional space to solve the original problem. This dimensionality reduction technique can be very effective and is often used in conjunction with the subspace segmentation problem [12, 83].

## 2. Algebraic Methods for Finite Dimensional Noise Free Case

The general subspace segmentation problem described in Section 1 does not yet have a good approach for solving it. In the ideal case where no noise is present, there are several algebraic methods that can solve this problem as will be described below. However, in realistic situations when noise, outliers, and corrupted data are present, there are no satisfactory algorithms for finding the solution, even in the finite dimensional case when  $\mathcal{H} = \mathbb{R}^D$ . The difficulties are both theoretical as well as computational, as will be further described below.

In the ideal case, when  $\mathcal{H} = \mathbb{R}^D$ , and the data  $\mathbf{W} = \{w_1, \dots, w_N\}$  is drawn from a finite union of subspaces  $\{S_i \subset \mathcal{H}\}_{i=1}^M$ , the general problem can be solved using algebraic methods. Obviously, there must be enough data points. In particular, it is necessary that for each subspace  $S_i$  there is a subset of data points of  $\mathbf{W}$  that form a basis for  $S_i$ . However, this is not sufficient. Consider for example the very simple case in which the data is drawn from a union of two

subspaces  $S_1 \cup S_2$  of  $\mathbb{R}^3$  such that  $\dim S_1 = \dim S_2 = 1$ . If we are supplied with two points, one from each line  $S_1, S_2$ , we will not be able to decide whether the data is drawn from a single subspace  $S = S_1 + S_2$  or from the union  $S_1 \cup S_2$ . However, if we are supplied with enough points belonging to  $S_1$  and enough belonging to  $S_2$ , the structure  $S_1 \cup S_2$  becomes apparent.

*2.1. Reduced Row Echelon Form Method.* One of the recent algebraic methods for solving the noise free subspace segmentation problem under the independent subspace restriction is the reduced row echelon form (RREF) method [22]. This method is a generalization of the method of Gear who observed that, for four dimensional subspaces, the reduced echelon form can be used to segment motions in videos [84]. It turns out that in the noise free case the reduced row echelon form method can completely solve the subspace segmentation in almost its most general version.

The RREF is based on the familiar Gauss elimination techniques for solving linear systems of equation. However, for this method to work, certain assumptions on the data and the subspaces are needed. Specifically, there must be enough data to cover all the dimensions of the union of subspaces  $\mathcal{U} = \bigcup_{i \in I} S_i$  from which the data is drawn. Moreover, the subspaces  $\{S_i\}_{i=1}^M$  must be independent. To make these assumptions precise, we make the following definitions.

*Definition 1 (generic data).* Let  $S$  be a linear subspace of  $\mathbb{R}^D$  with dimension  $d$ . A set of data  $\mathbf{W}$  drawn from  $S \subset \mathbb{R}^D$  with dimension  $d$  are said to be *generic*, if (i)  $|\mathbf{W}| > d$ , and (ii) every  $d$  vector from  $\mathbf{W}$  forms a basis for  $S$ .

*Definition 2 (independent subspaces).* Subspaces  $\{S_i \subset \mathbb{R}^D\}_{i=1}^n$  are called independent if  $\dim(S_1 + \dots + S_n) = \dim(S_1) + \dots + \dim(S_n)$ .

Independent subspaces have the property that  $S_i \cap S_j = \{0\}$  for  $i \neq j$ . The converse, however, is false, for example, three subspaces  $S_1, S_2, S_3$  in  $\mathbb{R}^2$  with  $\dim S_i = 1$ ,  $i = 1, 2, 3$  can never be independent. More generally, if  $\{S_i \subset \mathbb{R}^D\}_{i=1}^n$  are independent, then  $\sum_{i=1}^n \dim(S_i) \leq D$  and  $S_i \cap S_j = \{0\}$  for  $i \neq j$ .

If we knew the subspaces  $S_i$ , it would be easy to partition the data  $\mathbf{W}$  into the partition  $P(\mathbf{W}) = \{\mathbf{W}_1, \dots, \mathbf{W}_M\}$  such that  $\mathbf{W}_i \subset S_i$ . Conversely, if we knew a partition  $P(\mathbf{W}) = \{\mathbf{W}_1, \dots, \mathbf{W}_M\}$  of the data  $\mathbf{W}$  such that the set  $\mathbf{W}_i$  comes from the same subspace  $S_i$ , then we would set  $S_i = \text{span } \mathbf{W}_i$  and our problem subspace segmentation would be solved.

However, all we are given is the data  $\mathbf{W}$ , and we do not know the partition  $P(\mathbf{W})$ . Thus, solving the subspace segmentation problem amounts to finding the partition  $P(\mathbf{W}) = \{\mathbf{W}_1, \dots, \mathbf{W}_M\}$  of  $\mathbf{W}$ . To do this, we construct a matrix  $\mathbf{W} = [w_1, \dots, w_N]$  whose columns are the data vectors  $w_i \in \mathbb{R}^D$ . The matrix  $\mathbf{W}$  is a  $D \times N$  matrix, where  $D$  may be large, while the rank of  $\mathbf{W}$  is often much smaller (noise free case). Using the three elementary row operation used

in Gaussian elimination, we transform  $\mathbf{W}$  to its reduced row echelon form  $\text{rref}(\mathbf{W})$

$$\text{rref}(\mathbf{W}) = \begin{bmatrix} R \\ 0 \end{bmatrix}, \quad (4)$$

where  $R$  is an  $r \times N$  matrix and where  $r$  is the rank of  $\mathbf{W}$ . By setting to the value 1 all nonzero coefficients in  $\text{rref}(\mathbf{W})$  we obtain the so-called *binary reduced echelon form* of  $\mathbf{W}$  denoted by  $\text{Brref}(\mathbf{W})$ . The Binary reduced row echelon form of  $\mathbf{W}$  has a structure that allows us to easily find the partition  $P(\mathbf{W}) = \{\mathbf{W}_1, \dots, \mathbf{W}_M\}$  and thereby solve the subspace segmentation problem as Theorem 3 below suggests [22].

**Theorem 3.** *Let  $\{S_i\}_{i=1}^M$  be a set of nontrivial linearly independent subspaces of  $\mathbb{R}^D$ . Let  $\mathbf{W} = [w_1, \dots, w_N] \in \mathbb{R}^{D \times N}$  be a matrix whose columns are drawn from  $\bigcup_{i=1}^M S_i$ . Assume the data is drawn from each subspace and that it is generic. Let  $\text{Brref}(\mathbf{W})$  be the binary reduced row echelon form of  $\mathbf{W}$ . Then*

- (1) *the inner product  $\langle e_i, b_j \rangle$  of a pivot column  $e_i$  and a nonpivot column  $b_j$  in  $\text{Brref}(\mathbf{W})$  is one, if and only if the corresponding column vectors  $\{w_i, w_j\}$  in  $\mathbf{W}$  belong to the same subspace  $S_l$  for some  $l = 1, \dots, M$ ;*
- (2) *moreover,  $\dim(S_l) = \|b_j\|_1$ , where  $\|b_j\|_1$  is the  $l_1$ -norm of  $b_j$ ;*
- (3) *finally,  $w_p \in S_l$  if and only if  $b_p = b_j$  or  $\langle b_p, b_j \rangle = 1$ .*

This theorem suggests a very simple yet effective approach to clustering the data points (Algorithm 1) and solves the subspace segmentation problem. This is done by finding a partition  $\{\mathbf{W}_1, \dots, \mathbf{W}_M\}$  of the data  $\mathbf{W}$  into  $M$  clusters such that  $\text{span } \mathbf{W}_i = S_i$ ,  $i = 1, \dots, M$ . The clusters can be formed as follows: pick a nonpivot element  $b_j$  in  $\text{Brref}(\mathbf{W})$ , and group together all columns  $b_p$  in  $\text{Brref}(\mathbf{W})$  such that  $\langle b_j, b_p \rangle > 0$ . Repeat the process with a different nonpivot column until all columns are exhausted. This is detailed in Algorithm 1 below.

Note that, we do not need to know the number of subspaces  $M$  nor do we need to know the dimensions of the subspaces  $S_i$  for solving the subspace segmentation problem in this case.  $M$  and  $\dim(S_i)$  are an output of the algorithm. The only assumption is that there are enough data points and that they are well distributed (they are generic), and that the subspaces  $S_i$  are independent.

For noisy data, the reduced row echelon form method does not work, and a thresholding must be applied. However, the effect of the noise on the reduced echelon form method depends on the noise level and the relative positions of the subspaces. This dependence has been analyzed in [22].

**2.2. The Generalized Principle Component Analysis GPCA.** Another algebraic method for solving the subspace segmentation problem is the so-called *generalized principle component analysis* (GPCA) [12, 85]. Although the most general form of this method solves the subspace segmentation problem in its entire generality for finite dimensions, we will only describe the idea behind the GPCA method in the simplified case where the number of subspaces  $M$  is known and when the subspaces are hyperplanes in  $\mathbb{R}^D$ , that is, their dimensions

is  $D-1$ . For this case, each subspace  $S_i$  can be described by its normal vector  $b_i = (b_{i1}, \dots, b_{iD})^T$ , and every data point  $x \in S_i$  satisfies the linear equation  $b_i^T x = 0$  where  $x = (x_1, \dots, x_D)^T$ . Thus, a data point  $x \in \mathbf{W}$  drawn from the union of subspaces  $\mathcal{U} = \bigcup_{i=1}^M S_i$  must satisfy the polynomial equations

$$\prod_{k=1}^M (b_k^T x) = 0. \quad (5)$$

The product  $\prod_{k=1}^M (b_k^T x)$  is in fact a homogeneous polynomial

$$p(x) = \sum_{\alpha} c_{\alpha} x_1^{\alpha_1} x_2^{\alpha_2} \dots x_D^{\alpha_D} \quad (6)$$

of degree  $M$ , where  $\alpha = (\alpha_1, \dots, \alpha_D)$ ,  $\sum_k \alpha_k = M$  ( $\alpha_k$  integers). Thus, if  $x \in \mathcal{U}$  it must satisfy the equation  $p(x) = 0$ . Hence, in order to solve the subspace segmentation problem for this case, we must

- (1) find the polynomial  $p$  by finding the values of its coefficients  $c_{\alpha}$ . This is done by creating a system of  $N$  linear equations in the unknown  $\{c_{\alpha}\}$  by setting  $p(w_i) = 0$ ,  $i = 1, \dots, N$  for each data  $w_i \in \mathbf{W}$ . If the number of data points is generic, then the solution of the system of equations determines the polynomial  $p$ ;
- (2) once the polynomial  $p$  is determined, it must be factored into its product  $p(x) = \prod_{k=1}^M (b_k^T x)$ . The vectors  $b_k$  can then be found by identification. The subspaces  $S_k$  in the unions  $\mathcal{U}$  are thus determined.

A modification of the GPCA method described in the previous section works for the general subspace segmentation in which neither the dimensions of the subspace nor their number is described [12, 85]. However, as in the case of RREF method, this method cannot work directly when noise is present and some modification is needed in the presence of noise and outliers as described in [12, 85].

### 3. Optimization Methods and Subspace Segmentation in the Presence of Noise

The algebraic methods discussed in the previous section do not work without modification for the case in which the data is corrupted by noise or outliers. Even with some of the adjustments to take care of noisy environment, the algebraic algorithms do not perform well when the noise is not small. Algorithms rated according to their simplicity, computational speed, and their performance in nonideal situations. Thus, algebraic methods or their modifications may be the algorithms of choice if the noise is small and computational speed is the main requirement. However, when noise is relatively large and accuracy is important, other methods are needed. In this section we discuss other methods that are robust to noise and other inaccuracies in the data.

One of the methods for the subspace segmentation problem when noise is present is typified by Problem 1. Minimizing the functional described in Problem 1 amounts to finding the union of subspaces that is nearest to the data.

```

Require:  $D \times N$  data matrix  $\mathbf{W}$ .
1: Find  $\text{rref}(\mathbf{W})$  of  $\mathbf{W}$ .
2: Find  $\text{Brref}(\mathbf{W})$  of  $\mathbf{W}$  by setting all non-zero entries of  $\text{rref}(\mathbf{W})$  to 1.
3: for all  $j$  from 1 to  $N$  do
4:   Pick the  $j$ th column  $b_j$  of  $\text{Brref}(\mathbf{W})$ .
5:   if  $b_j$  is pivot then
6:     continue
7:   end if
8:   for all  $i$  from 1 to  $j-1$  do
9:     if  $b_i$  is non-pivot and  $\langle b_i, b_j \rangle > 0$  then
10:       Place  $\{b_i, b_j\}$  in the same cluster  $C_i$ .
11:       break
12:     end if
13:   end for
14: end for
15: for all  $C_i$  do
16:   Pick any  $b \in C_i$ .
17:   Separate  $b$  into unit vectors  $u_i^1, \dots, u_i^{d_i}$ . {These vectors form a basis for a subspace  $S_i$  with dimension  $d_i$ .}
18:   for all  $k$  from 1 to  $N$  do
19:     if  $b_k \in \{u_i^1, \dots, u_i^{d_i}\}$  then
20:       Place  $b_k$  in the same cluster  $C_i$ . {This is for handling pivot columns.}
21:     end if
22:   end for
23:   Place the corresponding columns in  $\mathbf{W}$  into the same cluster  $\mathbf{W}_i$ .
24: end for
25: Renumber indices  $i$ 's of  $S_i$  starting from 1.

```

ALGORITHM 1: Subspace segmentation—row echelon form approach—no noise.

However, some a priori knowledge of the number of subspaces  $M$  and the dimensions of the subspaces  $S_i$  may be necessary. The cost function can be modified to incorporate a cost that depends on the number of subspaces  $M$  and their dimensions, if these quantities are unknown. But before getting into algorithms for solving Problem 1, the existence of a minimizer is a theoretical question of interest. Thus, we start by some of the results pertaining to this issue.

**3.1. Existence of a Minimizer to Problem 1.** Given a family  $\mathcal{C}$  of closed subspaces of  $\mathcal{H}$ , a solution to Problem 1 may not exist even in the linear case when  $M = 1$ . For example, assume that  $\mathcal{H} = \mathbb{R}^2$  and  $\mathcal{C}$  is the set of all lines through the origin except the line  $x = 0$ . For this case, a minimizer may exist for certain distribution of data points but not for others. The existence of a solution here means that a minimizer exists for any distribution of any finite number of data points. We will describe the existence results when  $\mathcal{H}$  is a Hilbert space. The case when  $\mathcal{H}$  is not a Hilbert space is very difficult and only partial results are known.

It turns out that the existence of a minimizing sequence of subspaces  $\mathbf{S}^o = \{S_1^o, \dots, S_r^o\}$  that solves Problem 1 is equivalent to the existence of a solution to the same problem but for  $M = 1$  [2].

**Theorem 4.** *Problem 1 has a minimizing set of subspaces for all finite sets of data and for any  $M \geq 1$  if and only if it has a minimizing subspace for all finite sets of data and for  $M = 1$ .*

Therefore, the following definition is useful.

*Definition 5.* A set of closed subspaces  $\mathcal{C}$  of a separable Hilbert space  $\mathcal{H}$  have the minimum subspace approximation property (MSAP) if for every finite subset  $\mathbf{W} \subset \mathcal{H}$  there exists an element  $S \in \mathcal{C}$  that minimizes the expression

$$e(\mathbf{W}, S) = \sum_{f \in \mathbf{W}} d^2(f, S), \quad S \in \mathcal{C}. \quad (7)$$

Using this terminology, Problem 1 has a minimizing sequence of subspaces if and only if  $\mathcal{C}$  satisfies the MSAP. If  $\mathcal{H} = \mathbb{C}^D$  and  $\mathcal{C} = \{S \subset \mathbb{C}^D : \dim S \leq s\}$ , then  $\mathcal{C}$  satisfies MSAP. This fact is easy to prove directly and is in fact a consequence of the Eckart-Young theorem [24]. Another situation is when  $\mathcal{H} = L^2(\mathbb{R}^d)$  and  $\mathcal{C} = \mathcal{L}_n = \{S : S = \text{span}\{\varphi_1(x-k), \dots, \varphi_n(x-k) : k \in \mathbb{Z}^d\}\}$  is the set of all shift-invariant spaces of length at most  $n$ . For this last case, a result in [68] implies that  $\mathcal{C} = \mathcal{L}_n$  satisfies the MSAP.

In order to understand the general case, we identify each subspace  $S \in \mathcal{C}$  with the orthogonal projector  $Q = Q_S$  whose kernel is exactly  $S$  (i.e.,  $Q = I - P_S$ , where  $P_S$  is the orthogonal projector on  $S$ ). Now we can think of  $\mathcal{C}$  as a set of projection operators and endow it with the induced weak operator topology. This setting allows us to give the necessary and sufficient conditions for a class  $\mathcal{C}$  to have the MSAP property for the case when  $p = 2$  in (1). Note that it is sufficient that  $\mathcal{C}$  is closed in order for  $\mathcal{C}$  to have the

MSAP. However, this condition is too strong as the following example shows: let  $\mathcal{H} = \mathbb{R}^3$  and consider the set  $\mathcal{C} = \mathcal{C}_1 \cup \mathcal{C}_2$  which is the union of the plane  $\mathcal{C}_1 = \text{span}\{e_1, e_2\}$  and the set of lines  $\mathcal{C}_2 = \cup_v \{\text{span}\{v\} : v = e_3 + ce_2, \text{ for some } c \in \mathbb{R}\}$ . Then  $\mathcal{C}$  (identified with a set of projectors as described earlier) is not closed (since  $Q_{\text{span}\{e_2\}} \notin \mathcal{C}$ ). However, it is easy to show that this set satisfies the MSAP, since if the infimum in (1) is achieved by the missing line given by  $\text{span}\{e_2\}$ , it is also achieved by the plane  $\mathcal{C}_1$ .

For finite dimensions, the weak operator and strong operator topologies are the same and the characterization of the MSAP can be obtained in terms of the convex hull of the family  $\mathcal{C}^+$  consisting of  $\mathcal{C}$  together with the positive semidefinite operators added to it. Recall that the convex hull  $\text{co}(A)$  of a set  $A$  is the smallest convex set containing  $A$ , that is,  $\text{co}(A)$  is the intersection of all convex sets containing  $A$ . For finite dimensions, the following theorem give the necessary and sufficient conditions for the MSAP property and hence the necessary and sufficient conditions for the existence of a solution to Problem 1, when  $p = 2$  in (1).

**Theorem 6.** *Suppose  $\mathcal{H}$  has dimension  $d < \infty$ . Then the following are equivalent*

- (i)  $\mathcal{C}$  satisfies MSAP;
- (ii)  $\mathcal{C}^+$  is closed;
- (iii)  $\text{co}(\mathcal{C}^+) = \text{co}(\overline{\mathcal{C}^+})$ .

The necessary and sufficient conditions in infinite dimensions for the existence of solutions when  $p = 2$  can be found in [20], but are much more complicated. However, no such results are known for the existence of solution to Problem 1 when  $p \neq 2$ .

**3.2. Search Algorithms for Problem 1.** Searching for a solution to Problem 1 is easier when  $M = 1$  since this problem is then a linear problem. Using an algorithm  $\mathcal{A}_1$  for solving this simpler problem, the more difficult problem when  $M > 1$  can be solved by using  $\mathcal{A}_1$  multiple times in an iterative algorithm as follows.

Let  $\mathcal{P}(\mathbf{W})$  be the set of all partitions of the data  $\mathbf{W}$ , that is,  $P \in \mathcal{P}(\mathbf{W})$  if  $P_0 = \{\mathbf{W}_1, \dots, \mathbf{W}_M\}$  is such that  $\mathbf{W}_i \cap \mathbf{W}_j = \emptyset$  when  $i \neq j$ ,  $\cup_{k=1}^M \mathbf{W}_k = \mathbf{W}$ .

- (1) Let  $P = \{\mathbf{W}_1, \dots, \mathbf{W}_M\}$  be a partition of the data  $\mathbf{W}$ . For each  $i = 1, \dots, M$ , use Algorithm  $\mathcal{A}_1$  to find the subspace  $S_i^o(P) \in \mathcal{C}$  that is nearest to  $F_i$  in the sense that it minimizes  $e(\mathbf{W}_i, S) = \sum_{w \in \mathbf{W}_i} d^p(w, S)$ . We obtain a sequence of subspaces  $\mathbf{S} = \{S_1^o(P), \dots, S_M^o(P)\}$ .
- (2) Construct a new partition  $Q(\mathbf{S})$  by reassigning each data point  $w \in \mathbf{W}$  to its nearest subspace from  $\{S_1^o, \dots, S_M^o\}$  and by grouping together those points that are assigned to the same subspace.
- (3) Iterate between the two steps as described in Algorithm 2.

It can be shown that this algorithm always converges. However, the convergence may be a local minima instead of

the global one. For this reason, a good initial partition is important. This initial partition can be supplied by some modified version of the algebraic methods described in the previous section, for example.

There are many iterative algorithms for finding a solution to the subspace segmentation problem or some of its special cases (see, e.g., [86, 87]). Most of them iterate between partitioning the data and finding the union of subspaces that is consistent with the partition. The general algorithm described below solves the subspace segmentation problem by searching for the minimizer of Problem 1.

Note that the cost functions  $e(\mathbf{W}_i, S_i^o(P))$  and  $e(\mathbf{W}, \mathbf{S}^o(P))$  in the while loop of Algorithm 2 are the one defined by (1) in Problem 1, but correspond to  $e(\mathbf{W}_i, S_i^o(P))$  for  $M = 1$ .

Step 2 in Algorithm 2 is problem dependent. For example, in the situation where  $\mathcal{H} = \mathbb{R}^D$  and  $\mathcal{C}$  is the set of subspaces of dimensions no greater than  $n$ , Step 2 can be solved using the singular value decomposition (SVD). A similar algorithm works in a much more general context as described in [2].

## 4. Motion Segmentation

The problem of motion segmentation has been described in Section 1.2.4. This problem is a special case of subspace segmentation in which  $\mathcal{H} = \mathbb{R}^D$  and  $\mathcal{C}$  is the family of subspaces of dimensions no bigger than 4. There are many algorithms that have been developed to solve this problem, such as the methods based on sparsity [10, 88–90], the algebraic methods, [1, 12, 91], the statistical methods [76, 92–95], and the iterative methods [22, 86]. The most successful methods however are all based on the spectral clustering or some related method [22, 34, 36, 96, 97]. The main idea is that a similarity matrix  $\Xi$  is used to describe the “connection” between the points. Once this similarity matrix is obtained a classical clustering technique (such as the  $k$ -means) is applied to a projection of the similarity matrix  $\Xi$  on a low dimensional space (here projection is used loosely and is not necessarily an orthogonal projection). These methods are often tested and compared to the state-of-the-art methods on the Hopkins 155 Dataset [8], which serves as benchmark database to evaluate motion segmentation algorithms. It contains two and three motion sequences. Cornerness features that are extracted and tracked across the frames are provided along with the dataset. The ground truth segmentations are also provided for comparison. Figure 1 shows two samples from the dataset with the extracted features.

**4.1. Nearness to Local Subspace Algorithm.** Since most spectral clustering algorithms use similar overall structure, we describe the Nearness to Local Subspace (NLS) algorithm, which is the most performant of the spectral clustering type methods as applied to the Hopkins 155 Dataset. Other spectral clustering based algorithms will be discussed in Section 4.2.

The NLS method works whenever the dimensions of the subspaces are equal and known. First, a local subspace is estimated for each data point (vector). Then, the distances between the local subspaces and points are computed and a distance matrix is generated. This is followed by construction

```

1: Pick any partition  $P \in \mathcal{P}(\mathbf{W})$ 
2: For each subset  $\mathbf{W}_i$  in the partition  $P$  find the subspace  $S_i^o(P) \in \mathcal{C}$ 
   that minimizes the expression  $e(\mathbf{W}_i, S) = \sum_{w \in \mathbf{W}_i} d^P(w, S)$ 
3: while  $\sum_{i=1}^M e(\mathbf{W}_i, S_i^o(P)) > e(\mathbf{W}, S^o(P))$  do
4:   for all  $i$  from 1 to  $M$  do
5:     Update  $\mathbf{W}_i = \{w \in \mathbf{W} : d(w, S_k^o(P)) \leq d(w, S_k^o(P)), k = 1, \dots, M\}$ 
6:     Update  $S_i^o(P) = \operatorname{argmin}_{S \in \mathcal{C}} e(\mathbf{W}_i, S)$ 
7:   end for
8:   Update  $P = \{\mathbf{W}_1, \dots, \mathbf{W}_M\}$ 
9: end while
10:  $S^o = \{S_1^o(P), \dots, S_M^o(P)\}$ 

```

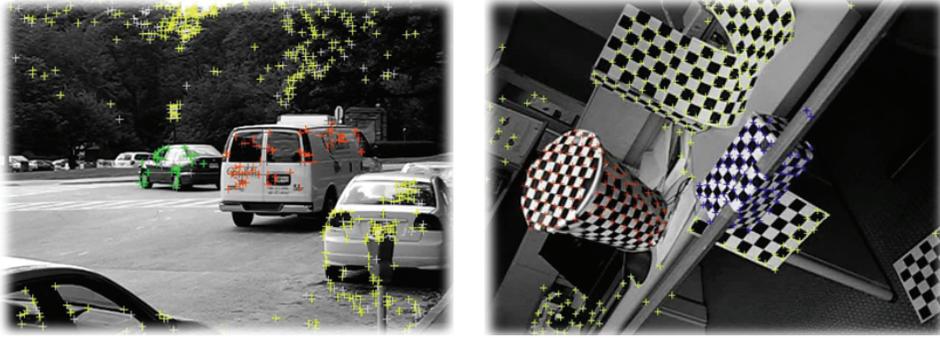
ALGORITHM 2: Optimal solution  $S^o$ .

FIGURE 1: Samples from the Hopkins 155 Dataset.

of a binary similarity matrix  $\Xi$  constructed by applying a data-driven threshold to the distance matrix. Finally, the segmentation problem is converted to a one-dimensional data clustering problem.

The algorithm for subspace segmentation is given in Algorithm 3. It assumes that the subspaces have dimension  $d$  (for motion segmentation,  $d = 4$ ). The details of the various steps are as follows.

*Dimensionality Reduction and Normalization.* A dimensionality reduction step is typical in any algorithm, including those using spectral clustering. Let  $W$  be a  $D \times N$  data matrix whose columns are drawn from a union of subspaces, where each subspace has dimensions at most  $d$ . The data  $W$  is possibly perturbed by noise and may have other imperfections. One way to reduce the dimensionality of the problem is to use SVD. Specifically, compute the SVD of  $W$

$$W = U\Sigma V^t, \quad (8)$$

where  $U = [u_1 \ u_2 \ \dots \ u_D]$  is an  $m \times D$  matrix,  $V = [v_1 \ v_2 \ \dots \ v_N]$  is an  $N \times N$  matrix, and  $\Sigma$  is a  $D \times N$  diagonal matrix with diagonal entries  $\sigma_1, \dots, \sigma_l$ , where  $l = \min\{m, N\}$ .

If the rank of the data  $W$  is not known, one can use the modal selection algorithm [34] to estimate its rank  $r$  by

$$r = \operatorname{argmin}_r \frac{\sigma_{r+1}^2}{\sum_{i=1}^r \sigma_i^2} + \kappa r, \quad (9)$$

where  $\sigma_j$  is the  $j$ th singular value and  $\kappa$  is a suitable constant. Another possible model selection algorithm can be found in [98].  $U_r \Sigma_r (V_r)^t$  is the best rank- $r$  approximation of  $W = U\Sigma V^t$ , where  $U_r$  refers to a matrix that has the first  $r$  columns of  $U$  as its columns and  $V_r$  refers to the first  $r$  rows of  $V^t$ . In the case of motion segmentation, if there are  $m$  independent motions across the frames captured by a moving camera, the rank of  $W$  is between  $2(m+1)$  and  $4(m+1)$ .

To reduce the dimensionality of the data, replace the data matrix  $W$  with the matrix  $(V_r)^t$  that consists of the first  $r$  rows of  $V^t$ . This step is justified by the following proposition in [22].

**Proposition 7.** *Let  $A$  and  $B$  be  $m \times n$  and  $n \times k$  matrices. Let  $C = AB$ . Assume  $J \subset \{1, 2, \dots, k\}$ .*

- (i) *If  $b_i \in \operatorname{span}\{b_j : j \in J\}$  then  $c_i \in \operatorname{span}\{c_j : j \in J\}$ .*
- (ii) *If  $A$  is full rank and  $m \geq n$  then  $b_i \in \operatorname{span}\{b_j : j \in J\} \Leftrightarrow c_i \in \operatorname{span}\{c_j : j \in J\}$ .*

It should also be noted that this step reduces additive noise as well, especially in the case of light-tailed noise, for example, Gaussian noise. The number of subspaces corresponds to the number of moving objects. Dimensionality reduction corresponds to Steps 1, 2, and 3 in Algorithm 3.

Another type of data reduction is normalization. Specifically, the columns of  $(V_r)^t$  are normalized to lie on the unit sphere  $\mathbb{S}^{r-1}$ . This is because by projecting the subspace on the unit sphere we effectively reduce the dimensionality of

**Require:** The  $D \times N$  data matrix  $W$  whose columns are drawn from subspaces of dimension  $d$

**Ensure:** Clustering of the feature points.

- 1: Compute the SVD of  $W$  as in (8).
- 2: Estimate the rank of  $W$  (denoted by  $r$ ) if it is not known. For example, using (9) or any other appropriate choice.
- 3: Compute  $(V_r)^t$  consisting of the first  $r$  rows of  $V^t$ .
- 4: Normalize the columns of  $(V_r)^t$ .
- 5: Replace the data matrix  $W$  with  $(V_r)^t$ .
- 6: Find the angle between the column vectors of  $W$  and represent it as a matrix.  
{i.e.,  $\arccos(W^t W)$ .}
- 7: Sort the angles and find the closest neighbors of column vector.
- 8: **for all** Column vector  $x_i$  of  $W$  **do**
- 9: Find the local subspace for the set consisting of  $x_i$  and  $k$  neighbors (see (10)).  
{Theoretically,  $k$  is at least  $d - 1$ . We can use the least square approximation for the subspace (see the section *Local Subspace Estimation*). Let  $A_i$  denote the matrix whose columns form an orthonormal bases for the local subspace associated with  $x_i$ .}
- 10: **end for**
- 11: **for**  $i = 1$  to  $N$  **do**
- 12: **for**  $j = 1$  to  $N$  **do**
- 13: define  $H = (d_{ij}) = (\|x_j - A_i^t x_j\|_p + \|x_i - A_j^t x_i\|_p) / 2$
- 14: **end for**
- 15: **end for** {Build the distance matrix}
- 16: Sort the entries of the  $N \times N$  matrix  $H$  from smallest to highest values into the vector  $h$  and set the threshold  $\eta$  to the value of the  $T$ th entry of the sorted and normalized vector  $h$ , where  $T$  is such that  $\|\chi_{[T, N^2]} - h\|_2$  is minimized, and where  $\chi_{[T, N^2]}$  is the characteristic function of the discrete set  $[T, N^2]$ .
- 17: Construct a similarity matrix  $\Xi$  by setting all entries of  $H$  less than threshold  $\eta$  to 1 and by setting all other entries to 0. {Build the binary similarity matrix}
- 18: Normalize the rows of  $\Xi$  using  $l_1$ -norm.
- 19: Perform SVD  $\Xi^t = U_n \Sigma_n (V_n)^t$ .
- 20: Cluster the columns of  $\Sigma_n (V_n)^t$  using  $k$ -means.  $\Sigma_n (V_n)^t$  is the projection on to the span of  $U_n$ .

ALGORITHM 3: Subspace segmentation.

the data by one. Moreover, the normalization gives equal contribution of the data matrix columns to the description of the subspaces. Note that the normalization can be done by using  $l_p$  norms of the columns of  $(V_r)^t$ . This normalization procedure corresponds to Steps 4 and 5 in Algorithm 3.

*Local Subspace Estimation.* The data points (i.e., each column vector of  $(V_r)^t$ ) that are close to each other are likely to belong to the same subspace. For this reason, a local subspace is estimated for each data point using its closest neighbors. This can be done by generating a distance matrix  $(a_{ij}) = (\|x_i - x_j\|_p)$  and then sorting each column of the distance matrix to find the neighbors of each  $x_i$ , which is the  $i$ th column of  $(V_r)^t$ .

Once the distance matrix between the points is generated, one can find, for each point  $x_i$ , a set of  $k + 1 \geq d$  points  $\{x_i, x_{i_1}, \dots, x_{i_k}\}$  consisting of  $x_i$  and its  $k$  closest neighbors. Then a  $d$ -dimensional subspace that is nearest (in the least square sense) to the data  $\{x_i, x_{i_1}, \dots, x_{i_k}\}$  is generated. This is accomplished using SVD

$$X = [x_i, x_{i_1}, \dots, x_{i_k}] = A \Sigma B^t. \quad (10)$$

Let  $A_i$  denote the matrix of the first  $d$  columns of  $A$  associated with  $x_i$ . Then, the column space  $C(A_i)$  is the  $d$ -dimensional subspace nearest to  $\{x_i, x_{i_1}, \dots, x_{i_k}\}$ . Local subspace estimation corresponds to Steps 6 to 10 in Algorithm 3.

*Construction of Binary Similarity Matrix.* So far, we have associated a local subspace  $S_i$  to each point  $x_i$ . Ideally, the points and only those points that belong to the same subspace as  $x_i$  should have zero distance from  $S_i$ . This suggests computing the distance of each point  $x_j$  to the local subspace  $S_i$  and forming a distance matrix  $H$ .

The distance matrix  $H$  is generated as  $H = (d_{ij}) = (\|x_j - A_i^t x_j\|_p + \|x_i - A_j^t x_i\|_p) / 2$ . A convenient choice of  $p$  is 2. Note that as  $d_{ij}$  decreases, the probability of having  $x_j$  on the same subspace as  $x_i$  increases. Moreover, for  $p = 2$ ,  $\|x_j - A_i^t x_j\|_2$  is the Euclidean distance of  $x_j$  to the subspace associated with  $x_i$ .

Since we are not in the ideal case, a point  $x_j$  that belongs to the same subspace as  $x_i$  may have nonzero distance to  $S_i$ . However, this distance is likely to be small compared to the distance between  $x_j$  and  $S_k$  if  $x_j$  and  $x_k$  do not belong to the same subspace. This suggests that we compute a threshold that will distinguish between these two cases and transform the distance matrix into a binary matrix in which a zero in the  $(i, j)$  entry means  $x_i$  and  $x_j$  are likely to belong to the same subspace, whereas  $(i, j)$  entry of one means  $x_i$  and  $x_j$  are not likely to belong to the same subspace.

To do this, we convert the distance matrix  $H = (d_{ij})_{N \times N}$  into a binary similarity matrix  $\Xi = (s_{ij})$ . This is done by applying a data-driven thresholding as follows.

TABLE I: Percentage of classification errors for sequences with two motions.

Checker (78)	GPCA	LSA	RANSAC	MSL	ALC	SSC-B	SSC-N	NLS
Average	6.09%	2.57%	6.52%	4.46%	1.55%	0.83%	1.12%	0.23%
Median	1.03%	0.27%	1.75%	0.00%	0.29%	0.00%	0.00%	0.00%
Traffic (31)								
Average	1.41%	5.43%	2.55%	2.23%	1.59%	0.23%	0.02%	1.40%
Median	0.00%	1.48%	0.21%	0.00%	1.17%	0.00%	0.00%	0.00%
Articulated (11)								
Average	2.88%	4.10%	7.25%	7.23%	10.70%	1.63%	0.62%	1.77%
Median	0.00%	1.22%	2.64%	0.00%	0.95%	0.00%	0.00%	0.88%
All (120 seq)								
Average	4.59%	3.45%	5.56%	4.14%	2.40%	0.75%	0.82%	0.57%
Median	0.38%	0.59%	1.18%	0.00%	0.43%	0.00%	0.00%	0.00%

- (1) Create a vector  $h$  that contains the sorted entries of  $H_{N \times N}$  from the smallest to the highest values. Scale  $h$  so that its smallest value is zero and its largest value is one.
- (2) Set the threshold  $\eta$  to the value of the  $T$ th entry of the sorted vector  $h$ , where  $T$  is such that  $\|\chi_{[T, N^2]} - h\|_2$  is minimized, and where  $\chi_{[T, N^2]}$  is the characteristic function of the discrete set  $[T, N^2]$ . If the number of points in each subspace is approximately equal, then we would expect  $N/n$  points in each subspace, and we would expect  $N^2/n^2$  small entries (zero entries ideally). However, this may not be the case in general. For this reason, we compute the data-driven threshold  $\eta$  that distinguishes the small entries from the large entries. The data-driven threshold  $\eta$  is chosen according to the method described in [1].
- (3) Create a similarity matrix  $\Xi$  from  $H$  such that all entries of  $H$  less than the threshold  $\eta$  are set to 1 and the others are set to 0.

*Segmentation.* The last step is to use the similarity matrix  $\Xi$  to segment the data. To do this, we first normalize the rows of  $\Xi$  using  $l_1$ -norm, that is,  $\tilde{\Xi} = D^{-1}\Xi$ , where  $D$  is a diagonal matrix  $(d_{ij}) = \sum_{j=1}^N s_{ij}$ .  $\tilde{\Xi}$  is related to the random walk Laplacian  $L_r(\tilde{\Xi} = I - L_r)$  [66]. Although other  $l_p$  normalizations are possible for  $p \geq 1$ , however, because of the geometry of the  $l_1$  ball,  $l_1$ -normalization brings outliers closer to the cluster clouds (distances of outliers decrease monotonically as  $p$  decreases to 1). Since SVD (which will be used next) is associated with  $l_2$  minimization, it is sensitive to outliers. Therefore  $l_1$  normalization works best when SVD is used.

Observe that the initial data segmentation problem has now been converted to segmentation of  $n$  1-dimensional subspaces from the rows of  $\tilde{\Xi}$ . This is because, in the ideal case, from the construction of  $\tilde{\Xi}$ , if  $x_i$  and  $x_j$  are in the same subspace, the  $i$ th and  $j$ th rows of  $\tilde{\Xi}$  are equal. Since there are  $n$  subspaces, then there will be  $n$  1-dimensional subspaces.

Now, the problem is again a subspace segmentation problem, but this time the data matrix is  $\tilde{\Xi}$  with each row as

a data point. Also, each subspace is 1-dimensional and there are  $n$  subspaces. Therefore, we can apply SVD again to obtain

$$\tilde{\Xi}^t = U_n \Sigma_n (V_n)^t. \quad (11)$$

Using Proposition 7, it can be shown that  $\Sigma_n (V_n)^t$  can replace  $\tilde{\Xi}^t$  and we cluster the columns of  $\Sigma_n (V_n)^t$ , which is the projection of  $\tilde{\Xi}$  onto the span of  $U_n$ . Since the problem is only segmentation of subspaces of dimension 1, we can use any traditional segmentation algorithm such as  $k$ -means to cluster the data points. The segmentation corresponds to Steps 18 to 20 in Algorithm 3.

*4.2. Other Spectral Clustering Methods.* Other subspace clustering methods use essentially the same general algorithm as above, but the main difference is the construction of the similarity  $\Xi$ . For example, Yan and Pollefeys' method estimates a subspace  $S_i$  for each point  $w_i$  and then uses the cordal distance between the local subspaces to construct a similarity matrix  $\Xi$ . The algorithm of Elhamifar and Vidal [88, 89] uses the sparsity method to compute a similarity matrix based on sparse representations of the data  $W$ . The sparse representations are found using the standard  $l_1$  minimization techniques in compressed sampling. We have tested these algorithms using  $l_2$  minimizations and found that both cases produce essentially the same results. Thus, it is our conclusion that it is the spectral clustering performed on the similarity matrix  $\Xi$  that is the main reason for the good performance of this and other related algorithms.

*4.3. Comparison of Motion Segmentation Algorithms.* Tables 1, 2, and 3 display some of the experimental results for the Hopkins 155 Dataset. Seven approaches are compared for the motion detection algorithms: (1) GPCA [12], (2) RANSAC [99], (3) local subspace affinity (LSA) [34], (4) MLS [93], (5) agglomerative lossy compression (ALC) [100], (6) sparse subspace clustering (SSC) [88], and (7) NLS. An evaluation of those algorithms is presented in [88] with a minor error in the tabulated results for articulated three-motion analysis of SSC-N. SSC-B and SSC-N correspond to Bernoulli and normal random projections, respectively [88]. Table I displays the misclassification rates for the two motions video sequences.

TABLE 2: Percentage of classification errors for sequences with three motions.

Checker (26)	GPCA	LSA	RANSAC	MSL	ALC	SSC-B	SSC-N	NLS
Average	31.95%	5.80%	25.78%	10.38%	5.20%	4.49%	2.97%	0.87%
Traffic (7)								
Average	19.83%	25.07%	12.83%	1.80%	7.75%	0.61%	0.58%	1.86%
Median	19.55%	23.79%	11.45%	0.00%	0.49%	0.00%	0.00%	1.53%
Articulated (2)								
Average	16.85%	7.25%	21.38%	2.71%	21.08%	1.60%	1.60%	5.12%
Median	16.85%	7.25%	21.38%	2.71%	21.08%	1.60%	1.60%	5.12%
All (35 seq)								
Average	28.66%	9.73%	22.94%	8.23%	6.69%	3.55%	2.45%	1.31%
Median	28.26%	2.33%	22.03%	1.76%	0.67%	0.25%	0.20%	0.45%

TABLE 3: Percentage of classification errors for all sequences.

All (155 seq)	GPCA	LSA	RANSAC	MSL	ALC	SSC-B	SSC-N	NLS
Average	10.34%	4.94%	9.76%	5.03%	3.56%	1.45%	1.24%	0.76%
Median	2.54%	0.90%	3.21%	0.00%	0.50%	0.00%	0.00%	0.20%

Table 2 shows the misclassification rates for the three motion sequences, and Table 3 presents the misclassification rates for all of the video sequences. It can be seen that the NLS algorithm outperforms all of the algorithms.

## Acknowledgment

This research is supported in part by NSF Grant DMS-110863.

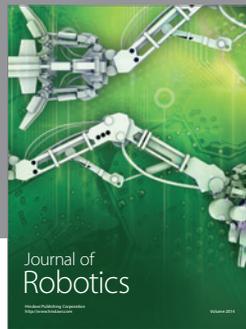
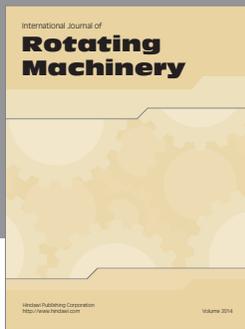
## References

- [1] A. Aldroubi and A. Sekmen, "Nearness to local subspace algorithm for subspace and motion segmentation," *IEEE Signal Processing Letters*, vol. 19, no. 10, Article ID 6275471, pp. 704–707, 2012.
- [2] A. Aldroubi, C. Cabrelli, and U. Molter, "Optimal non-linear models for sparsity and sampling," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5-6, pp. 793–812, 2008.
- [3] Y. Ma, A. Y. Yang, H. Derksen, and R. M. Fossum, "Estimation of subspace arrangements with applications in modeling and segmenting mixed data," *SIAM Review*, vol. 50, no. 3, pp. 413–458, 2008.
- [4] H.-P. Kriegel, P. Kroeger, and A. Zimek, "Subspace clustering," *WIREs Data Mining and Knowledge Discovery*, vol. 2, pp. 351–364, 2012.
- [5] S. Nitzan and A. Olevskii, "Revisiting Landau's density theorems for Paley-Wiener spaces," *Comptes Rendus Mathématique*, vol. 350, no. 9-10, pp. 509–512, 2012.
- [6] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, pp. 1–15, 2005.
- [7] G. Chen, A. V. Little, M. Maggioni, and L. Rosasco, "Some recent advances in multiscale geometric analysis of point clouds," in *Wavelets and Multiscale Analysis*, J. Cohen and A. I. Zayed, Eds., pp. 199–225, Birkhäuser, Boston, Mass, USA, 2011.
- [8] R. Vidal, "Subspace clustering," *IEEE Signal Processing Magazine*, vol. 28, no. 3, pp. 52–68, 2011.
- [9] Y. Lyubarskii and W. R. Madych, "Interpolation of functions from generalized Paley-Wiener spaces," *Journal of Approximation Theory*, vol. 133, no. 2, pp. 251–268, 2005.
- [10] Y. Sugaya and K. Kanatani, "Improved multistage learning for multibody segmentation, in," in *Proceedings of the 5th International Conference on Computer Vision Theory and Applications (VISAPP '10)*, pp. 199–206, 2010.
- [11] R. Hu, L. Fan, and L. Liu, "Co-segmentation of 3d shapes via subspace clustering," *Computer Graphics Forum*, vol. 31, pp. 1703–1713, 2012.
- [12] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis (GPCA)," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 1945–1959, 2005.
- [13] R. Basri and D. W. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, pp. 218–233, 2003.
- [14] J. Ho, M. H. Yang, J. Lim, K. C. Lee, and D. Kriegman, "Clustering appearances of objects under varying illumination conditions," in *Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1/11–1/18, June 2003.
- [15] A. Aldroubi and K. Gröchenig, "Nonuniform sampling and reconstruction in shift-invariant spaces," *SIAM Review*, vol. 43, no. 4, pp. 585–620, 2001.
- [16] W. K. Allard, G. Chen, and M. Maggioni, "Multi-scale geometric methods for data sets II: geometric multi-resolution analysis," *Applied and Computational Harmonic Analysis*, vol. 32, no. 3, pp. 435–462, 2012.
- [17] M. Soltanolkotabi and E. J. Candès, "A geometric analysis of subspace clustering with outliers," *Annals of Statistics*, vol. 40, no. 4, pp. 2195–2238, 2012.
- [18] Y. Sugaya and K. Kanatani, "Multi-stage optimization for multibody motion segmentation," in *Proceedings of IEICE Transactions on Information and Systems*, pp. 1935–1942, 2004.
- [19] A. D. Szlam, M. Maggioni, and R. R. Coifman, "Regularization on graphs with function-adapted diffusion processes," *Journal of Machine Learning Research*, vol. 9, pp. 1711–1739, 2008.

- [20] A. Aldroubi and R. Tessera, "On the existence of optimal unions of subspaces for data modeling and clustering," *Foundations of Computational Mathematics*, vol. 11, no. 3, pp. 363–379, 2011.
- [21] M. Petrik, "An analysis of laplacian methods for value function approximation in mdps," in *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pp. Morgan Kaufmann–2574, 2007.
- [22] A. Aldroubi and A. Sekmen, "Reduced row echelon form and non-linear approximation for subspace segmentation and high-dimensional data clustering," submitted to *Applied and Computational Harmonic Analysis*.
- [23] R. Tron and R. Vidal, "A benchmark for the comparison of 3-D motion segmentation algorithms," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07)*, June 2007.
- [24] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, no. 3, pp. 211–218, 1936.
- [25] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing. Part I. Theory," *IEEE Transactions on Signal Processing*, vol. 41, no. 2, pp. 821–833, 1993.
- [26] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing. Part II. Efficient design and applications," *IEEE Transactions on Signal Processing*, vol. 41, no. 2, pp. 834–848, 1993.
- [27] G. Chen and G. Lerman, "Foundations of a multi-way spectral clustering framework for Hybrid Linear Modeling," *Foundations of Computational Mathematics*, vol. 9, no. 5, pp. 517–558, 2009.
- [28] G. Liu, Z. Lin, and Y. Yu, "Robust subspace segmentation by low-rank representation," in *Proceedings of 27th International Conference on Machine Learning (ICML '10)*, pp. 663–670, June 2010.
- [29] F. De La Torre and M. J. Black, "A framework for robust subspace learning," *International Journal of Computer Vision*, vol. 54, no. 1–3, pp. 117–142, 2003.
- [30] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems*, vol. 23, no. 3, pp. 969–985, 2007.
- [31] G. Haro, G. Randall, and G. Sapiro, "Translated poisson mixture model for stratification learning," *International Journal of Computer Vision*, vol. 80, no. 3, pp. 358–374, 2008.
- [32] Y. C. Eldar and M. Mishali, "Robust recovery of signals from a structured union of subspaces," *IEEE Transactions on Information Theory*, vol. 55, no. 11, pp. 5302–5316, 2009.
- [33] S. Kang and K. H. Kwon, "Recovery of missing samples for oversampling in shift invariant spaces," *Journal of Mathematical Analysis and Applications*, vol. 391, no. 1, pp. 139–146, 2012.
- [34] J. Yan and M. Pollefeys, "A general framework for motion segmentation: independent, articulated, rigid, non-rigid, degenerate and nondegenerate," in *Proceedings of the 9th European Conference on Computer Vision*, pp. 94–106, 2006.
- [35] S. R. Rao, A. Y. Yang, S. S. Sastry, and Y. Ma, "Robust algebraic segmentation of mixed rigid-body and planar motions from two views," *International Journal of Computer Vision*, vol. 88, no. 3, pp. 425–446, 2010.
- [36] T. Lin and H. Zha, "Riemannian manifold learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, pp. 796–809, 2008.
- [37] C. Bregler, A. Hertzmann, and H. Biermann, "Recovering non-rigid 3D shape from image streams," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR '00)*, pp. 690–696, June 2000.
- [38] M. Brand, "Morphable 3D models from video," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. II456–II463, December 2001.
- [39] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Mathématique*, vol. 346, no. 9–10, pp. 589–592, 2008.
- [40] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.
- [41] S. Foucart, "A note on guaranteed sparse recovery via  $\ell_1$ -minimization," *Applied and Computational Harmonic Analysis*, vol. 29, no. 1, pp. 97–103, 2010.
- [42] P. Boufounos, G. Kutyniok, and H. Rauhut, "Sparse recovery from combined fusion frame measurements," *IEEE Transactions on Information Theory*, vol. 57, no. 6, pp. 3864–3876, 2011.
- [43] A. Aldroubi, X. Chen, and A. M. Powell, "Perturbations of measurement matrices and dictionaries in compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 33, no. 2, pp. 282–291, 2012.
- [44] Q. Sun, "Recovery of sparsest signals via  $l(q)$ -minimization," *Applied and Computational Harmonic Analysis*, vol. 32, no. 3, pp. 329–341, 2012.
- [45] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.
- [46] M. Aharon, M. Elad, and A. M. Bruckstein, "On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them," *Linear Algebra and Its Applications*, vol. 416, no. 1, pp. 48–67, 2006.
- [47] G. Lerman and T. Zhang, "Robust recovery of multiple subspaces by geometric  $l_p$  minimization," *The Annals of Statistics*, vol. 39, no. 5, pp. 2686–2715, 2011.
- [48] C. Archambeau, N. Delannay, and M. Verleysen, "Mixtures of robust probabilistic principal component analyzers," *Neurocomputing*, vol. 71, no. 7–9, pp. 1274–1282, 2008.
- [49] Y. M. Lu and M. N. Do, "A theory for sampling signals from a union of subspaces," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2334–2345, 2008.
- [50] P. W. Jones, M. Maggioni, and R. Schul, "Manifold parametrizations by eigenfunctions of the Laplacian and heat kernels," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 105, no. 6, pp. 1803–1808, 2008.
- [51] Q. Wu, J. Guinney, M. Maggioni, and S. Mukherjee, "Learning gradients: predictive models that infer geometry and statistical dependence," *Journal of Machine Learning Research*, vol. 11, pp. 2175–2198, 2010.
- [52] J. Zhang, H. Huang, and J. Wang, "Manifold learning for visualizing and analyzing high-dimensional data," *IEEE Intelligent Systems*, vol. 25, no. 4, Article ID 5401149, pp. 54–61, 2010.
- [53] F. Lauer and C. Schnorr, "Spectral clustering of linear subspaces for motion segmentation," in *Proceedings of IEEE International Conference on Computer Vision*, 2009.
- [54] A. Aldroubi and M. Unser, "Sampling procedures in function spaces and asymptotic equivalence with Shannon's sampling theory," *Numerical Functional Analysis and Optimization*, vol. 15, no. 1–2, pp. 1–21, 1994.
- [55] M. Unser, "Sampling—50 years after Shannon," *Proceedings of the IEEE*, vol. 88, no. 4, pp. 569–587, 2000.

- [56] A. Aldroubi, "Non-uniform weighted average sampling and reconstruction in shift-invariant and wavelet spaces," *Applied and Computational Harmonic Analysis*, vol. 13, no. 2, pp. 151–161, 2002.
- [57] M. Anastasio and C. Cabrelli, "Sampling in a union of frame generated subspaces," *Sampling Theory in Signal and Image Processing*, vol. 8, no. 3, pp. 261–286, 2009.
- [58] J. Xian and W. Sun, "Local sampling and reconstruction in shift-invariant spaces and their applications in spline subspaces," *Numerical Functional Analysis and Optimization*, vol. 31, no. 3, pp. 366–386, 2010.
- [59] A. Bhandari and A. I. Zayed, "Shift-invariant and sampling spaces associated with the fractional Fourier transform domain," *IEEE Transactions on Signal Processing*, vol. 60, no. 4, pp. 1627–1637, 2012.
- [60] D. Kushnir, M. Galun, and A. Brandt, "Fast multiscale clustering and manifold identification," *Pattern Recognition*, vol. 39, no. 10, pp. 1876–1891, 2006.
- [61] S. Ericsson, "Generalized sampling in shift invariant spaces with frames," *Acta Mathematica Sinica*, vol. 28, no. 9, pp. 1823–1844, 2012.
- [62] I. Maravić and M. Vetterli, "Sampling and reconstruction of signals with finite rate of innovation in the presence of noise," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 2788–2805, 2005.
- [63] J. A. Hogan, "Frame-based nonuniform sampling in Paley-Wiener spaces," *Journal of Applied Functional Analysis*, vol. 2, no. 4, pp. 361–400, 2007.
- [64] H. Boche and U. J. Mönich, "There exists no globally uniformly convergent reconstruction for the Paley-Wiener space  $PW_{1/\pi}$  of bandlimited functions sampled at Nyquist rate," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 3170–3179, 2008.
- [65] H. Boche and U. J. Mönich, "Unboundedness of thresholding and quantization for bandlimited signals," *Signal Processing*, vol. 92, no. 12, pp. 2821–2829, 2012.
- [66] S. Rao, R. Tron, R. Vidal, and Y. Ma, "Motion segmentation in the presence of outlying, incomplete, or corrupted trajectories," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 10, pp. 1832–1845, 2010.
- [67] B. A. Bailey, "Multivariate polynomial interpolation and sampling in Paley-Wiener spaces," *Journal of Approximation Theory*, vol. 164, no. 4, pp. 460–487, 2012.
- [68] A. Aldroubi, C. Cabrelli, D. Hardin, and U. Molter, "Optimal shift invariant spaces and their Parseval frame generators," *Applied and Computational Harmonic Analysis*, vol. 23, no. 2, pp. 273–283, 2007.
- [69] M. Vetterli, P. Marziliano, and T. Blu, "Sampling signals with finite rate of innovation," *IEEE Transactions on Signal Processing*, vol. 50, no. 6, pp. 1417–1428, 2002.
- [70] P. L. Dragotti, M. Vetterli, and T. Blu, "Sampling moments and reconstructing signals of finite rate of innovation: shannon meets strang-fix," *IEEE Transactions on Signal Processing*, vol. 55, no. 5, pp. 1741–1757, 2007.
- [71] V. Y. F. Tan and V. K. Goyal, "Estimating signals with finite rate of innovation from noisy samples: a stochastic algorithm," *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 5135–5146, 2008.
- [72] N. Bi, M. Z. Nashed, and Q. Sun, "Reconstructing signals with finite rate of innovation from noisy samples," *Acta Applicandae Mathematicae*, vol. 107, no. 1–3, pp. 339–372, 2009.
- [73] J. Berent, P. L. Dragotti, and T. Blu, "Sampling piecewise sinusoidal signals with finite rate of innovation methods," *IEEE Transactions on Signal Processing*, vol. 58, no. 2, pp. 613–625, 2010.
- [74] T. Kanade and D. D. Morris, "Factorization methods for structure from motion," *The Royal Society of London. Philosophical Transactions A*, vol. 356, no. 1740, pp. 1153–1173, 1998.
- [75] J. P. Costeira and T. Kanade, "A multibody factorization method for independently moving objects," *International Journal of Computer Vision*, vol. 29, no. 3, pp. 159–179, 1998.
- [76] P. H. S. Torr, "Geometric motion segmentation and model selection," *The Royal Society of London. Philosophical Transactions A*, vol. 356, no. 1740, pp. 1321–1340, 1998.
- [77] A. Goh and R. Vidal, "Segmenting motions of different types by unsupervised manifold clustering," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07)*, vol. 6, June 2007.
- [78] S. Smale and D. X. Zhou, "Shannon sampling II: connections to learning theory," *Applied and Computational Harmonic Analysis*, vol. 19, no. 3, pp. 285–302, 2005.
- [79] W. Johnson and J. Lindenstrauss, "Extensions of lipshitz mapping into hilbert space," *Contemporary Mathematics*, vol. 26, pp. 189–206, 1984.
- [80] D. Achlioptas, "Database-friendly random projections: Johnson-Lindenstrauss with binary coins," *Journal of Computer and System Sciences*, vol. 66, no. 4, pp. 671–687, 2003.
- [81] N. Silva and J. Costeira, "Subspace segmentation with outliers: a grassmannian approach to the maximum consensus subspace," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008.
- [82] R. I. Arriaga and S. Vempala, "Algorithm theory of learning: robust concepts and random projection," in *Proceedings of IEEE 40th Annual Conference on Foundations of Computer Science*, pp. 616–623, October 1999.
- [83] A. Aldroubi, M. Anastasio, C. Cabrelli, and U. Molter, "A dimension reduction scheme for the computation of optimal unions of subspaces," *Sampling Theory in Signal and Image Processing*, vol. 10, no. 1–2, pp. 135–150, 2011.
- [84] C. W. Gear, "Multibody grouping from motion images," *International Journal of Computer Vision*, vol. 29, no. 2, pp. 133–150, 1998.
- [85] R. Vidal, R. Tron, and R. Hartley, "Multiframe motion segmentation with missing data using PowerFactorization and GPCA," *International Journal of Computer Vision*, vol. 79, no. 1, pp. 85–105, 2008.
- [86] P. Tseng, "Nearest q-flat to m points," *Journal of Optimization Theory and Applications*, vol. 105, no. 1, pp. 249–252, 2000.
- [87] P. S. Bradley and O. L. Mangasarian, "k-plane clustering," *Journal of Global Optimization*, vol. 16, no. 1, pp. 23–32, 2000.
- [88] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR '09)*, pp. 2790–2797, June 2009.
- [89] E. Elhamifar and R. Vidal, "Clustering disjoint subspaces via sparse representation," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '10)*, pp. 1926–1929, March 2010.
- [90] E. Elhamifar and R. Vidal, "Sparse subspace clustering: algorithm, theory, and applications," <http://arxiv.org/abs/1203.1005>.
- [91] A. Sekmen, *Susbpace Segmentation*, Vanderbilt, 2012.

- [92] O. Faugeras, P. Torr, T. Kanade et al., “Geometric motion segmentation and model selection—discussion,” *Philosophical Transactions of the Royal Society A*, vol. 356, pp. 1338–1340, 1998.
- [93] A. Gruber and Y. Weiss, “Multibody factorization with uncertainty and missing data using the EM algorithm,” in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, pp. 1707–1714, July 2004.
- [94] L. Candillier, I. Tellier, F. Torre, and O. Bousquet, “SSC: statistical subspace clustering,” in *5mes Journées d'Extraction et Gestion des Connaissances (EGC '05)*, pp. 177–182, Paris, France, 2005.
- [95] S. Smale and D.-X. Zhou, “Learning theory estimates via integral operators and their approximations,” *Constructive Approximation*, vol. 26, no. 2, pp. 153–172, 2007.
- [96] G. Chen, S. Atev, and G. Lerman, “Kernel spectral curvature clustering (KSCC),” in *Proceedings of IEEE 12th International Conference on Computer Vision Workshops (ICCV '09)*, pp. 765–772, October 2009.
- [97] T. Zhang, A. Szlam, Y. Wang, and G. Lerman, “Randomized hybrid linear modeling by local best-fit flats,” in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 1927–1934, June 2010.
- [98] L. Zappella, X. Lladó, E. Provenzi, and J. Salvi, “Enhanced Local Subspace Affinity for feature-based motion segmentation,” *Pattern Recognition*, vol. 44, no. 2, pp. 454–470, 2011.
- [99] M. A. Fischler and R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [100] T. Sarlós, “Improved approximation algorithms for large matrices via random projections,” in *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS '06)*, pp. 143–152, October 2006.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

