SoS-RSC: A Sum-of-Squares Polynomial Approach to Robustifying Subspace Clustering Algorithms*

Mario Sznaier and Octavia Camps
Electrical and Computer Engineering
Northeastern University, Boston, MA, 02115
{msznaier,camps}@coe.neu.edu

Abstract

This paper addresses the problem of subspace clustering in the presence of outliers. Typically, this scenario is handled through a regularized optimization, whose computational complexity scales polynomially with the size of the data. Further, the regularization terms need to be manually tuned to achieve optimal performance. To circumvent these difficulties, in this paper we propose an outlier removal algorithm based on evaluating a suitable sum-of-squares polynomial, computed directly from the data. This algorithm only requires performing two singular value decompositions of fixed size, and provides certificates on the probability of misclassifying outliers as inliers.

1. Introduction

Many problems involve fitting piecewise models to a given set of points (Fig.1), often in the presence of outliers. Examples include motion segmentation [22], video segmentation [20], image compression [8], face recognition [1], and system identification [15]. Due to its importance, a substantial research effort has been devoted to this problem, leading to many algorithms, that can be roughly classified into statistical, algebraic and self-representation based.

RANdom SAmple Consensus (RANSAC) [5] type methods use a sampling based approach to find, at each iteration, a subspace that fits as many data points as possible. These points are then removed from the dataset and the process is repeated, until a given threshold on the percentage of inliers is exceeded. While in principle these methods can correctly cluster noisy data, even in the presence of outliers, they may require a large number of iterations to do so.

Algebraic methods such as Generalized Principal Component Analysis (GPCA) [16] and its variants exploit subspace arrangements properties, reducing the problem to estimating a multivariate polynomial from noisy measurements of its zeros. Once this polynomial has been found, the parameters of the subspaces can be recovered via polynomial differentiation. While GPCA works well with clean data, its performance degrades quickly with the noise level. This drawback has motivated the approaches in [19] and [3] where the original data is "cleaned" by solving a polynomial optimization. Although these methods can handle substantial noise levels and outlier percentages, their main drawback is their computational complexity.

Figure 1: Many practical applications involve fitting linear subspaces to sample data. Images from Coil-100, Extended Yale B, Caltech 256, Hopkins155, and MNIST datasets were used in our experiments.

An alternative to algebraic approaches are methods that exploit the self-expressiveness of subspaces. In most of these approaches, the goal becomes finding a clustering that renders a suitably defined affinity matrix, typically obtained by solving a sparsification or rank minimization problem, block-diagonal. Examples of these techniques include Sparse Subspace Clustering (SSC) [4], Robust PCA (RPCA) [2], Low Rank Representation (LRR) [13], Fixed Rank Representation (FRR) [14] and Robust Subspace Clustering (RSC) [20]. These methods work well in the presence of noise and offer theoretical recovery guarantees. On the other hand, handling outliers requires augmenting the objective function with additional regulariza-

*This work was supported in part by NSF grants IIS–1318145, ECCS–1404163, CMMI–1638234 and CNS–1646121; AFOSR grant FA9550-15-1-0392; and the Alert DHS Center of Excellence under Award Number 2013-ST-061-ED0001.

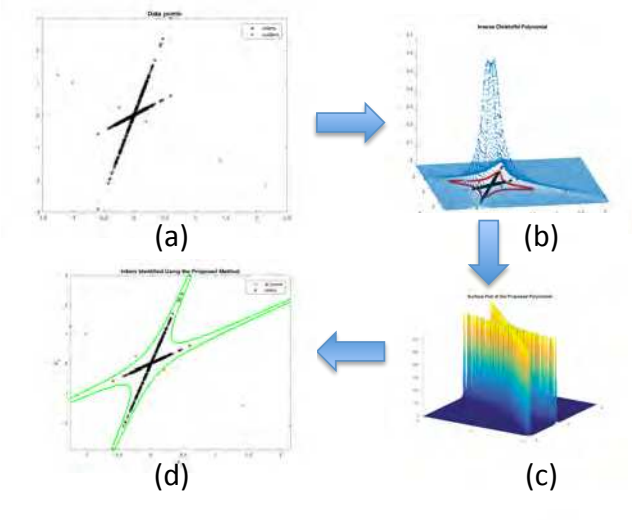


Figure 2: Proposed pipeline: (a) inliers (black) corrupted with outliers (red); (b) SoS approximation of the support of the inliers (blue) and its level set (red), (c) approximating the support of the inlier subspace using the inverse of the proposed polynomial; (d) inliers are the points inside a suitable level set of the polynomial.

tion terms, whose weights must be usually hand tuned to obtain good performance and typically recovery guarantees are lost. Alternatively, recently [25] proposed a method based on random walks over the representation graph of the data, which encodes the self-representation property of inliers. While this method works well for small to moderately large datasets, finding the representation graph entails solving an elastic net problem for each data point, a problem that, as in sparsification based approaches, scales polynomially with the size of the data and becomes prohibitively expensive for large data sets.

Finally, a remarkable exception to the need to solve regularized optimization problems when exploiting self-expressiveness is the Robust Shape Interaction Matrix (RSIM) method [9], which works directly with a shape interaction matrix (SIM) and handles noise and outliers indirectly by i) row normalization, ii) element-wise powering, and iii) carefully estimating the rank of the SIM matrix. Although RSIM has been empirically shown to outperform other spectral clustering methods when the data is corrupted with gross outliers, at the present time there are no theoretical results guaranteeing that this will be always the case. Additionally, this approach does not discard outliers, instead it assigns them to some of the inlier subspaces.

Overview of the proposed approach

To circumvent the difficulties noted above, we propose a computationally efficient algorithm to detect and remove outliers from (noisy) data that lies in an arrangement of subspaces. The main idea of the algorithm, illustrated in Fig. 2, is to find a polynomial whose level sets approximate the support of the inliers (that is, the polynomial is small in the inlier subspaces and has large values elsewhere). This polynomial is obtained by (i) first identifying "reliable" inliers, (ii) estimating the probability density function of these inliers, and (iii) finding a polynomial that minimizes the expected value of the quadratic fitting error to the (estimated) inlier subspaces.

The paper is motivated by the recent work in [10] showing that, given a cloud of points, there exists a sum-of-squares (SOS) polynomial Q, that can be calculated from the data and achieves high values at isolated points. Thus, in principle, points far from the cloud of inliers can be detected by simply evaluating Q. However, note that these points may not necessarily be outliers in the classical sense of the word (e.g. points far away from the subspaces). In fact, as illustrated in Fig. 2(b) isolated inliers will be classified as outliers even if they belong to the subspace arrangement. Conversely, outliers close to inliers may be incorrectly classified as inliers. In contrast, in this paper we explicitly exploit the underlying subspace structure to accurately identify outliers – i.e. points that do not belong to the subspace arrangement (see Fig. 2(d)), and, at the same time, reduce the computational complexity vis-a-vis [10].

Paper contributions:

• **Theoretical**. Our main theoretical contribution shows that, given a collection of data points lying in a subspace arrangement, corrupted by noise and gross outliers, the set of inliers can be approximated with high probability as the sub level set of a polynomial directly constructed from the data. This result is established by providing a stochastic reinterpretation of algebraic methods showing that these techniques seek to minimize the expected value, with respect to the empirical probability measure estimated from the data, of an homogeneous SOS polynomial. In fact, for noiseless data this polynomial coincides with the square of the hybrid decoupling polynomial used in GPCA.

• **Algorithmic:** The theoretical results outlined above lead to an efficient algorithm for outlier removal. Notably, this algorithm only requires performing two singular value decompositions whose size is *independent* of the number of data points, avoiding computationally expensive semi-definite optimizations. In addition, it provides bounds on the probability of misclassifying an inlier as an outlier.

These contributions are illustrated using data from the MNIST digits [12], Extended Yale B [6], Coil-100 [18], Caltech-256, [7] and Hopkins 155 datasets [22], where the proposed approach outperforms the state of the art, both in terms of performance (average Area Under the Curve (AUC) and average maximum F1 score), and in execution time, with a very pronounced ($\approx$ 10x-50x) gap for the larger data sets.

2. Preliminaries

2.1. Notation

$\mathbb{R}$	set of real numbers
$\sigma_i(\mathbf{M})$	the i-th largest singular value of $\mathbf{M}$
$\sigma_{min}(\mathbf{M})$	smallest singular value of $\mathbf{M}$
$(\mathbf{M})_i$	i^{th} row of $\mathbf{M}$.
$\mathbf{M} \succeq 0$	matrix $\mathbf{M}$ is positive semidefinite.
$\mathcal{P}_d^n$ ($\mathcal{P}_{d,h}^n$)	set of n^{th} degree multivariate (homogeneous) polynomials in d variables.
$s_{n,d} \doteq \binom{n+d}{d}$	number of monomials of degree up to n in d variables.
$\nu_n(\mathbf{x})$	Veronese map of degree n: $\nu_n\begin{pmatrix}x_1 & \dots & x_d\end{pmatrix} \doteq \begin{bmatrix}x_1^n & x_1^{n-1}x_2 & \dots & x_d^n\end{bmatrix}^T$ (*i.e.* all possible monomials of order n in d variables, in lexicographic order.)
$\mathcal{E}_\mu(.)$	Expected value w.r.t. the probability density function μ.

2.2. Moment matrices and orthogonal polynomials

Given a probability measure μ supported on $\mathbb{R}^d$, its corresponding moments sequence is given by

$$m_{\boldsymbol{\alpha}} = \mathcal{E}_\mu(\mathbf{x}^{\boldsymbol{\alpha}}) = \int_{\mathbb{R}^d} \mathbf{x}^{\boldsymbol{\alpha}} d\mu \tag{1}$$

where $\boldsymbol{\alpha} = \begin{bmatrix}\alpha_1 & \alpha_2 & \dots & \alpha_d\end{bmatrix}$ is a multi-index , $\mathbf{x} \doteq \begin{bmatrix}x_1 & x_2 & \dots & x_d\end{bmatrix}^T$ and $\mathbf{x}^{\boldsymbol{\alpha}} \doteq x_1^{\alpha_1}x_2^{\alpha_2}\cdots x_d^{\alpha_d}$.

In this paper, we are interested in using a matrix representation of a given sequence $\mathbf{m}$ that contains all the moments up to order $2n$. To this effect, we will arrange the moments according to a graded reverse lexicographic order (grevlex) of the corresponding monomials and form the matrix $\mathbf{M}_n \in \mathbb{R}^{s_{n,d}\times s_{n,d}}$ with entries given by $\mathbf{M}_n(\boldsymbol{\alpha},\boldsymbol{\beta}) = m_{\boldsymbol{\alpha}+\boldsymbol{\beta}}$. For example, for $d = 2$ variables, x_1 and x_2, there are $s_{n,d} = 6$ monomials of degree up to $n = 2$: $1, x_1, x_2, x_1^2, x_1x_2, x_2^2$. The elements of the corresponding moment matrix $\mathbf{M}_2$ with all moments of order up to $2n = 4$, are given by: $m_{(i,j)} = \mathcal{E}_\mu(x_1^i.x_2^j)$, with $0 \le i, j \le 3$.

Given a point $\mathbf{x} \in \mathbb{R}^d$, let

$$\boldsymbol{\phi}_n(\mathbf{x}) \doteq \begin{bmatrix}1 & x_1 & x_2 & \dots (x_1^{\alpha_1}x_2^{\alpha_2}\cdots x_d^{\alpha_d})\dots x_d^n\end{bmatrix}^T \tag{2}$$

where $\sum_{i=1}^d \alpha_i \le n$. It can be easily seen that

$$\mathbf{M}_n = \int_{\mathbb{R}^d} \boldsymbol{\phi}_n(\mathbf{x})\boldsymbol{\phi}_n^T(\mathbf{x}) d\mu \tag{3}$$

Thus, $\mathbf{M}_n \succeq 0$ (in fact $\mathbf{M}_n \succ 0$, except when $\mathbf{x}$ belongs to the zero set of a polynomial of degree at most n [11]).

The measure μ induces an inner product in $\mathcal{P}_d^n$, the space of polynomials in d variables of degree at most n, given by:

$$\langle P_1(.), P_2(.)\rangle_\mu \doteq \int_{\mathbb{R}^d} P_1(\mathbf{x})P_2(\mathbf{x}) d\mu \tag{4}$$

As shown next, this inner product can be computed directly from $\mathbf{M}_n$. Consider a generic polynomial $P(\mathbf{x}) \in \mathcal{P}_d^n$, $P(\mathbf{x}) = \sum_\alpha p_\alpha \mathbf{x}^\alpha$, where p_α denotes the coefficients of $P(.)$ in the canonical monomial basis. Collecting all the coefficients p_α in a vector $\mathbf{p}$ allows for compactly representing $P(.)$ as $P(\mathbf{x}) = \mathbf{p}^T\boldsymbol{\phi}_n(\mathbf{x})$. Then,

$$\langle P_1(.), P_2(.)\rangle_\mu = \int_{\mathbb{R}^d} \mathbf{p}_1^T\boldsymbol{\phi}_n(\mathbf{x})\boldsymbol{\phi}_n^T(\mathbf{x})\mathbf{p}_2 d\mu = \mathbf{p}_1^T\mathbf{M}_n\mathbf{p}_2 \tag{5}$$

Assume that $\mathbf{M}_n \succ 0$ and let $\mathbf{u}_i, \sigma_i$ denote its singular vectors and the corresponding singular values. From the derivation above, it follows that the polynomials associated with the coefficient vectors $\mathbf{c}_i \doteq \frac{1}{\sqrt{\sigma_i}}\mathbf{u}_i$ are an orthonormal basis, with respect to μ, of $\mathcal{P}_d^n$. As shown in [24] these orthonormal polynomials define a reproducing Kernel:

$$K_n(\mathbf{x},\mathbf{y}) \doteq \sum_{i=1}^{s_{n,d}} (\mathbf{c}_i^T\boldsymbol{\phi}_n(\mathbf{x}))(\mathbf{c}_i^T\boldsymbol{\phi}_n(\mathbf{y})) \tag{6}$$

Next, define the polynomial

$$Q_n(\mathbf{x}) \doteq K_n(\mathbf{x},\mathbf{x}) = \sum_{i=1}^{s_{n,d}} (\mathbf{c}_i^T\boldsymbol{\phi}_n(\mathbf{x}))^2 \tag{7}$$

Note that $Q_n(\mathbf{x})$ is a sum-of-squares polynomial, and hence non-negative (in fact, it can be shown [10] that $Q_n(\mathbf{x}) \ge 1$). The function $Q_n(\mathbf{x})^{-1}$ is known as the Christoffel function and it is related to the probability measure μ that induces orthorgonality of the set $\{\mathbf{c}_i\}$ through the following result (Theorem 2 in [10], see also section 7 in [24]):

$$Q_n(\mathbf{x})^{-1} = \min_{P\in\mathcal{P}_d^n} \int_{\mathbb{R}^p} P^2(\boldsymbol{\xi}) d\mu \text{ s. t. } P(\mathbf{x}) = 1 \tag{8}$$

a result that will play a key role in identifying outliers.

3. Robust subspace clustering

Our goal is to develop a computationally efficient algorithm for subspace clustering in the presence of gross outliers. Specifically, we address the following problem:

Problem 1. *[Robust Subspace Clustering] Given a (sufficiently dense) set of noisy i.i.d samples (with arbitrary, unknown distribution μ) $\mathbf{x}_i$ of points $\hat{\mathbf{x}}_i \in \mathbb{R}^d$ drawn from an arrangement of (linear) subspaces $\mathcal{A} \doteq S_1 \cup S_2 \cup \dots S_n$, corrupted with gross outliers, and a bound n on the number of subspaces: 1) Identify gross outliers in the data; and 2) Assign inliers to subspaces.*

Algebraic methods attempt to solve the problem above by finding a polynomial $P(\mathbf{x})$ in the vanishing ideal of $\mathcal{A}$[1] while spectral clustering methods work directly with an

[1] Recall that $I(\mathcal{A})$, the vanishing ideal of a subspace arrangement $\mathcal{A} \subseteq R^d$, is the set of all multivariate polynomials in d variables that vanish on all points in $\mathcal{A}$, that is $I(\mathcal{A}) \doteq \{P \in \mathcal{P}^d \colon P(\mathbf{z}) = 0\ \forall \mathbf{z} \in \mathcal{A}\}$.

affinity matrix obtained from the data. The advantage of the later methods is that they avoid the non-trivial step of estimating the vanishing ideal in the presence of noise and outliers. However, this is accomplished at the price of having to solve a regularized optimization problem [4, 13] or devising methods that are only experimentally shown to behave well in the presence of a moderate number of outliers [9]. In this paper we propose to use a combination of both approaches: (i) using a (perhaps crude) approximation to the vanishing ideal to find "reliable" inliers, (ii) use these reliable inliers to estimate a polynomial $P_s(\mathbf{x})$ in this ideal, (iii) identify the true inliers as those points where P_s is "small" (in a sense to be precisely defined later), and (iv) apply a spectral clustering based method (such as RSIM) only to these inliers. Notably, as discussed in the sequel, steps (i)-(iii) only involve performing two singular value decompositions on a matrix whose size is, in typical applications, much smaller than the data matrix, and thus are computationally far less costly than solving a regularized optimization problem.

3.1. Stochastic interpretation of algebraic methods

The first step towards developing efficient methods for outlier rejection is based on a stochastic re-interpretation of algebraic methods in the presence of noise. For the case of clean data, GPCA [23] provides an elegant algebraic solution to the clustering problem. For simplicity, consider the case where all subspaces have the same (known) dimension $k < d$ (the case where the subspaces have different dimensions will be considered later). Without loss of generality, we will assume that $k = d - 1$ (it that is not the case, one can project the data into a generic subspace of dimension K and consider hyperplanes of dimension $K - 1$ in this subspace). The key idea behind this method is the observation that, regardless of their labels, all points in the arrangement $\mathcal{A}$ satisfy the *hybrid decoupling constraint*:

$$P_{dc}(\mathbf{x}) = \prod_{i=1}^{n} (\mathbf{x}^T \mathbf{r}_i) \doteq \mathbf{c_n}^T \nu_n(\mathbf{x}) = 0 \tag{9}$$

where $\mathbf{r}_i$ denotes the normal to S_i, $P_{dc}(\mathbf{x}) \in \mathcal{P}^n_{d,h}$ is an n^{th} order homogeneous polynomial in d variables with coefficient vector $\mathbf{c}_n$, and $\nu_n(.)$ is the Veronese map of degree $n \doteq s_{n,d-1}$. Collecting all data into a matrix form leads to:

$$\mathbf{V}_n \mathbf{c}_n \doteq \left[\begin{array}{ccc} \boldsymbol{\nu}_n^T(\mathbf{x}_1) & \cdots & \boldsymbol{\nu}_n^T(\mathbf{x}_{N_p}) \end{array} \right]^T \mathbf{c}_n = \mathbf{0} \tag{10}$$

Thus, $\mathbf{c}_n$, can be computed by simply finding a vector in the right null space of the embedded data matrix $\mathbf{V}_n$. In the case of (moderately) noisy samples, the approach outlined above can still be used, by considering the polynomial defined by $\mathbf{c}_{min}$, the singular vector of $\mathbf{V}_n$ associated with its smallest singular value. However, this approach breaks down in the presence of outliers, since in this case the polynomial constructed from the singular vector of the corrupted embedded data matrix associated with its smallest singular value is a poor approximation to the actual polynomial $P_{dc}(\mathbf{x})$ in (9). As we show in the sequel, this difficulty can be avoided by considering a polynomial constructed by taking into account all singular vectors of $\mathbf{V}_n$.

Given $\mathbf{x} \in \mathbb{R}^p$, consider the lifting $\mathbf{x} \to \boldsymbol{\phi}_n(\mathbf{x}) \in \mathbb{R}^{s_{n,d}}$ defined by (2). Collect the liftings for all the N_p data points in a matrix $\mathbf{L}_n$ with rows $(\mathbf{L}_n)_i = \boldsymbol{\phi}_n^T(\mathbf{x_i})$ and define:

$$\mathbf{M}_n = \frac{1}{N_p} \mathbf{L}_n^T \mathbf{L}_n = \frac{1}{N_p} \sum_{i=1}^{N_p} \boldsymbol{\phi}_n(\mathbf{x_i}) \boldsymbol{\phi}_n^T(\mathbf{x_i}) \tag{11}$$

Comparing (11) with (3) it can be easily seen that $\mathbf{M}_n$ is an estimate of the moment matrix of the probability density function of the data points; where $m_{\boldsymbol{\alpha}} = \mathcal{E}_\mu(\mathbf{x}^{\boldsymbol{\alpha}})$ has been replaced by $\frac{1}{N_p} \sum_{i=1}^{N_p} \mathbf{x}_i^{\boldsymbol{\alpha}}$. Note that the lower right $s_{n,d-1} \times s_{n,d-1}$ submatrix of $\mathbf{M}_n$ is given by $\frac{1}{N_p} \mathbf{V}_n^T \mathbf{V}_n$.

Consider now an arbitrary homogeneous n^{th} order d-variate polynomial defined by a coefficient vector $\mathbf{p}$, $P(\mathbf{x}) = \mathbf{p}^T \boldsymbol{\phi}_n(\mathbf{x})$, where $\mathbf{p}^T = \left[\mathbf{0} \ \mathbf{c}^T \right]^T$, with $\mathbf{c} \in \mathbb{R}^{\binom{n+d-1}{n}}$. From (5) it follows that $\mathcal{E}(P^2(\mathbf{x}))$, where the expectation is taken with respect to the empirical probability density defined by the data points, is given by:

$$\begin{aligned} \mathcal{E}(P^2(\mathbf{x})) &= \left[0 \ \mathbf{c}^T \right] \mathbf{M}_n \begin{bmatrix} 0 \\ \mathbf{c} \end{bmatrix} = \left[0 \ \mathbf{c}^T \right] \begin{bmatrix} * & * \\ * & \frac{1}{N_p} \mathbf{V}_n^T \mathbf{V}_n \end{bmatrix} \begin{bmatrix} 0 \\ \mathbf{c} \end{bmatrix} \\ &= \frac{1}{N_p} \mathbf{c}^T \mathbf{V}_n^T \mathbf{V}_n \mathbf{c} \geq \frac{1}{N_p} \sigma_{min}(\mathbf{V}_n^T \mathbf{V}_n) \|\mathbf{c}\|^2 \end{aligned} \tag{12}$$

where equality holds when $\mathbf{c} = \mathbf{c}_{min}$. Thus, GPCA can be reinterpreted as finding an homogeneous polynomial $P_{n,h}(\mathbf{x}) = \mathbf{c}_n^T \boldsymbol{\phi}_n(\mathbf{x})$ that solves the problem

$$P_{n,h}(\mathbf{x}) = \left\{ \underset{P \in \mathcal{P}^n_{d,h}}{argmin} \, \mathcal{E}(P^2(\mathbf{x})) \colon P = \mathbf{c}^T \boldsymbol{\phi}_n(\mathbf{x}), \|\mathbf{c}\| = 1 \right\} \tag{13}$$

where the constraint $\|\mathbf{c}\| = 1$ is added to avoid trivial solutions. In the presence of outliers, the empirical probability distribution obtained from the data is a poor estimate of the inlier distribution. Thus the solution to (13) is typically a poor approximation to P_{dc} in (9), as shown in Fig 3. The main idea of this paper is to circumvent this difficulty by minimizing $\mathcal{E}(P^2(\mathbf{x}))$ with respect to the empirical distribution defined by a subset of the data consisting of "reliable" inliers. As illustrated in Fig 3, the level sets of this polynomial (green) approximate well the support of the true inliers, leading in this case to perfect inlier/outlier separation.

3.2. Finding reliable inliers

The starting point of the proposed method is to eliminate from the data potential outliers that skew the computation of $P_{n,h}$ in (13). To this effect, consider the matrix

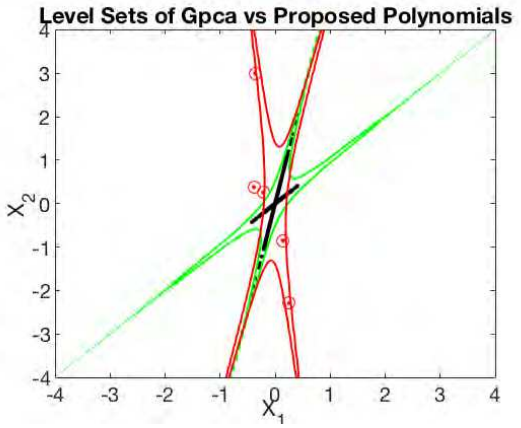


Figure 3: A few outliers (red points) can lead to a GPCA polynomial whose level sets (red) are a poor approximation of the inlier subspaces while the level sets of the proposed polynomial (green) successfully approximate the support of these subspaces

$\mathbf{M}_{n,h} \doteq \frac{1}{N_p}\mathbf{V}_n^T\mathbf{V}_n \in \mathbb{R}^{s_{n,d-1}\times s_{n,d-1}}$. As in (4), $\mathbf{M}_{n,h}$ induces an inner product in $\mathcal{P}_{d,h}^n$, the space of homogeneous polynomials, given by:

$$\langle P_1(.), P_2(.)\rangle_\mu \doteq \int_{\mathbb{R}^d} P_1(\mathbf{x})P_2(\mathbf{x})d\mu = \mathbf{p}_1^T\mathbf{M}_{n,h}\mathbf{p}_2 \tag{14}$$

where $P_i(\mathbf{x}) = \mathbf{p}_i^T\boldsymbol{\nu}_n(\mathbf{x})$. Next, let $\mathbf{U}\mathbf{diag}(\sigma_i)\mathbf{U}^T = \text{svd}(\mathbf{M}_{n,h})$ and consider the homogeneous polynomials $C_i(\mathbf{x}) \in \mathcal{P}_{n,h}$, $C_i(\mathbf{x}) \doteq \mathbf{c}_i^T\boldsymbol{\nu}_n(\mathbf{x})$ with coefficient vectors $\mathbf{c}_i \doteq \frac{1}{\sqrt{\sigma_i}}\mathbf{u}_i$, where $\mathbf{u}_i$ denotes the i^{th} column of $\mathbf{U}$. Note that these polynomials form an orthonormal basis of $\mathcal{P}_{n,h}$ with respect to the inner product defined by (14) since

$$\langle C_i(.), C_j(.)\rangle_\mu = \frac{1}{\sqrt{\sigma_i\sigma_j}}\mathbf{u}_i^T\mathbf{M}_{n,h}\mathbf{u}_j = \begin{cases} 1 & \text{if i=j} \\ 0 & \text{otherwise} \end{cases}$$

Thus,

$$K_{n,h}(\mathbf{x},\mathbf{y}) \doteq \sum_{i=1}^{s_{n,d-1}} (\mathbf{c}_i^T\boldsymbol{\nu}_n(\mathbf{x}))(\mathbf{c}_i^T\boldsymbol{\nu}_n(\mathbf{y})) \tag{15}$$

is a reproducing kernel in this space, with associated Christoffel polynomial given by

$$Q_{n,h}(\mathbf{x}) \doteq K_{n,h}(\mathbf{x},\mathbf{x}) = \sum_{i=1}^{s_{n,d-1}} (\mathbf{c}_i^T\boldsymbol{\nu}_n(\mathbf{x}))^2 \tag{16}$$

Due to orthonormality, $\mathcal{E}_\mu[C_i(\mathbf{x})] = \mathbf{c}_i^T\mathbf{M}_{n,h}\mathbf{c}_i = 1$. Thus, the expected value of $Q_{n,h}$ is given by

$$\mathcal{E}_\mu(Q_{n,h}) = \sum_{i=1}^{s_{n,d-1}} \mathcal{E}_\mu[c_i(\mathbf{x})] = s_{n,d-1} \tag{17}$$

Finally, given a threshold t, from Markov's inequality (chapter 8 in [21]) it follows that

$$\text{prob}\{Q_{n,h}(\mathbf{x}) \geq t\cdot s_{n,d}\} \leq \frac{1}{t} \tag{18}$$

Hence, a set of "reliable" inliers can be found by simply selecting those points where $Q_{n,h}(\mathbf{x}) < t\cdot s_{n,d}$. Since accurate estimation of a polynomial in the vanishing ideal $I(\mathcal{A})$ hinges on using a set without outliers, t in this step should be taken reasonably low (e.g. $t \sim 1-3$).

3.3. Estimating a polynomial that vanishes on all inlier subspaces

Since $K_{n,h}$ defined in (15) is a reproducing kernel in $\mathcal{P}_{n,h}$, from [24] it follows that, for each point $\mathbf{x} \in \mathbb{R}^d$

$$Q_{n,h}(\mathbf{x})^{-1} = \min_{P\in\mathcal{P}_{d,h}^n} \int_{\mathbb{R}^p} P^2(\boldsymbol{\xi})d\mu \text{ s. t. } P(\mathbf{x}) = 1 \tag{19}$$

For a given $\mathbf{x}^*$, denote by P^* the minimizer above. Markov's inequality [10] implies that, for any given threshold t, the mass of the set $\mathcal{S} = \{\mathbf{x}\colon (P^*(\mathbf{x}))^2 \leq t\}$ satisfies

$$\mu(\mathcal{S}) = \int_{P^*(\mathbf{x})^2\leq t} d\mu \geq 1 - \frac{\mathcal{E}_\mu[(P^*)^2)]}{t} = 1 - \frac{1}{tQ_{n,h}(\mathbf{x}^*)} \tag{20}$$

Consider first a scenario with noiseless data $\mathbf{x}_i \in \mathcal{I} \subset \mathcal{A}$, corrupted with outliers $\hat{\mathbf{x}}_i \in \mathcal{O}$ and let μ_{true} denote the probability density function of the inliers. Choose an arbitrary outlier $\mathbf{x}_o \in \mathcal{O}$ and define the polynomial $\tilde{P}(\mathbf{x}) \doteq \frac{P_{dc}(\mathbf{x})}{P_{dc}(\mathbf{x}_o)}$, where P_{dc} is the decoupling constraint polynomial defined in (9). Since by construction $P_{dc}(\mathbf{x}) = 0 \iff \mathbf{x} \in \mathcal{A}$, $\tilde{P}$ above is well defined and satisfies $\int_{\mathbb{R}^p}\tilde{P}(\boldsymbol{\xi})^2 d\mu_{true} = 0$ (the only homogeneous polynomial of degree n, up to a scaling factor, to do so). Thus, $\tilde{P}$ is the minimizer in (19). Since $\frac{1}{Q(\mathbf{x}_o)} = 0$, it follows from (20), that, for any $t > 0$,

$$\mathcal{I} \subseteq \mathcal{H} \doteq \{\mathbf{x}\colon \tilde{P}^2(\mathbf{x}) \leq t\} \tag{21}$$

In other words, the sub-level sets of $\tilde{P}^2$ provide a polynomial approximation to the support set of the measure μ_{true}.

Consider now the case of noisy data and let σ_{min} and $\mathbf{u}_{min}$ denote the smallest singular value and associated singular vector of the moments sub matrix $\mathbf{M}_{n,h}^{true}$, corresponding to the true inlier probability distribution. In this case, the GPCA polynomial associated with the data matrix is (up to a scale factor)

$$P_{gpca}(\mathbf{x}) \doteq \frac{\mathbf{u}_{min}^T\boldsymbol{\nu}(\mathbf{x})}{\mathbf{u}_{min}^T\boldsymbol{\nu}(\mathbf{x}_o)}$$

and yields a value

$$\frac{1}{Q_{gpca}(\mathbf{x}_o)} = \frac{\mathbf{u}_{min}^T\mathbf{M}_{n,h}^{true}\mathbf{u}_{min}}{[\mathbf{u}_{min}^T\boldsymbol{\nu}(\mathbf{x}_o)]^2} = \frac{\sigma_{min}}{[\mathbf{u}_{min}^T\boldsymbol{\nu}(\mathbf{x}_o)]^2}$$

Since by construction P_{gpca} satisfies the constraint in (19), it follows that the minimizer P^* of (19) satisfies $\frac{1}{Q^*} \leq$

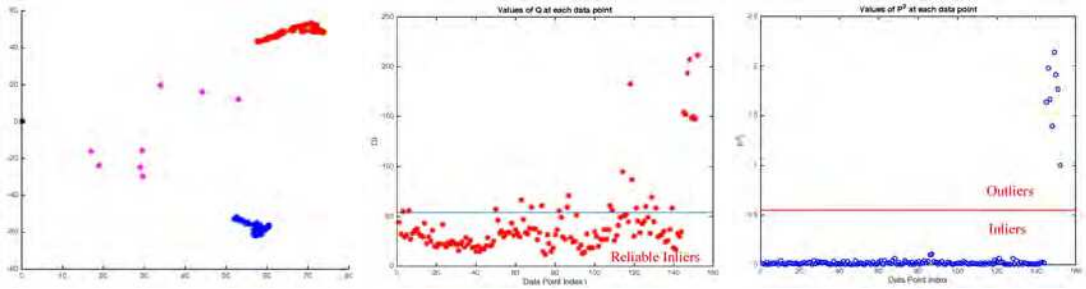


Figure 4: Left: 2D projections of two Coil-100 clusters (red and blue) corrupted with eight outliers (magenta). Center: The Q_i values (red). The horizontal line (blue) shows the threshold used to select "reliable" inliers to estimate the empirical distribution. Right: The $P^2(\mathbf{x})$ values. The horizontal line (red) shows the threshold used to classify the outliers. In this example, all eight outliers are correctly identified.

$\frac{1}{Q_{gpca}}$. Hence the set $H^* \doteq \{\mathbf{x}\colon P^*(\mathbf{x})^2 \leq t\}$ has a mass $\mu(H^*) \geq 1 - \frac{\sigma_{min}}{t[\mathbf{u}_{min}^T \boldsymbol{\nu}(\mathbf{x}_o)]^2}$. It follows that, when $[\mathbf{u}_{min}^T \boldsymbol{\nu}(\mathbf{x}_o)]^2 \gg \sigma_{min}$ (roughly speaking the outlier is well outside the noise level) then the set H^* contains most of the inliers. This observation is the basis of our algorithm.

3.4. Estimating the set of inliers

From the previous discussion, estimating the set of inliers requires finding the polynomial that solves (19) and the associated $Q_{n,h}(\mathbf{x})$. As we show next, both can be written in terms of the singular value decomposition of $\mathbf{M}_{n,h}$.

Theorem 1. *Let $\mathbf{u}_i, \sigma_i$ denote the singular vectors of $\mathbf{M}_{n,h}$ and the corresponding singular values. Define the vectors $\mathbf{c}_i = \frac{\mathbf{u}_i}{\sqrt{\sigma_i}}$. Given a point $\mathbf{x}_o$, the polynomial $P^*(\mathbf{x})$ that minimizes* (19) *and the corresponding value of $Q_{n,h}$ are given by $P(\mathbf{x}) = \boldsymbol{\nu}(\mathbf{x})\mathbf{p}^*$ and $Q_{n,h}(\mathbf{x}_o) = \sum_{i=1}^{s_{n,d}} (\mathbf{c}_i^T \boldsymbol{\nu}(\mathbf{x}_o))^2$ with:*

$$\mathbf{p}^* = \frac{1}{\sum_{i=1}^{s_{n,d}} (\mathbf{c}_i^T \boldsymbol{\nu}(\mathbf{x}_o))^2} \sum_{i=1}^{s_{n,d}} \mathbf{c}_i^T \boldsymbol{\nu}(\mathbf{x}_o) \mathbf{c}_i \tag{22}$$

Proof. Since the polynomials $C_i(\mathbf{x}) \doteq \mathbf{c}_i^T \boldsymbol{\nu}(\mathbf{x})$ form an orthonormal basis of $\mathcal{P}_{d,h}^n$, any solution to (19) can be written as $P(\mathbf{x}) = \sum_{i=1}^{s_{n,d}} \alpha_i C_i(\mathbf{x})$, Thus the objective in (19) can be rewritten as

$$\int_{\mathbb{R}^p} P^2(\boldsymbol{\xi}) d\mu = \int_{\mathbb{R}^p} \sum (\alpha_i C_i(\mathbf{x}))^2 d\mu = \sum_{i=1}^{s_{n,d}} \alpha_i^2 \tag{23}$$

where the last equality follows from orthonormality of the $C_i(.)$. Hence (19) is equivalent to:

$$\min_{\alpha_i} \sum_{i=1}^{s_{n,d}} \alpha_i^2 \text{ s. t. } \sum_{i=1}^{s_{n,d}} \alpha_i \mathbf{c}_i^T \boldsymbol{\nu}(\mathbf{x}_o) = 1 \tag{24}$$

It can be easily seen that the solution to the (convex) optimization above is given by $\alpha_i = \frac{\mathbf{c}_i^T \boldsymbol{\nu}(\mathbf{x}_o)}{\sum_{i=1}^{s_{n,d}} (\mathbf{c}_i^T \boldsymbol{\nu}(\mathbf{x}_o))^2}$ which proves (22). The proof is completed by noting that:

$$Q_{n,h}(\mathbf{x}_o) = \frac{1}{\int_{\mathbb{R}^p} (P^*(\boldsymbol{\xi}))^2 d\mu} = \frac{1}{\sum \alpha_i^2} = \sum_{i=1}^{s_{n,d}} (\mathbf{c}_i^T \boldsymbol{\nu}(\mathbf{x}_o))^2 \tag{25}$$

□

Once P^* has been found, the set of inliers is estimated as the intersection of the original data with the sub level set of $P^*(.)^2$ corresponding to a given threshold t, that is

$$\mathcal{H} = \{\mathbf{x}_i\colon [\boldsymbol{\nu}^T(\mathbf{x}_i)\mathbf{p}^*]^2 \leq t\} \tag{26}$$

In principle the ideas outlined above can be directly applied to find outliers. However, as it was discussed in the introduction and illustrated in Fig. 2(b) the procedure may identify isolated inlier points as outliers. To avoid this difficulty, inspired by [9] we propose to project all the data points onto the unit sphere, that is $\mathbf{x}_i \leftarrow \frac{\mathbf{x}_i}{\|\mathbf{x}_i\|_2}$. This projection will "concentrate" all points belonging to the same subspace in the region defined by the intersection of the subspace and the sphere, achieving higher density. Note in passing that this normalization still preserves the subspace structure and thus the proposed algorithm is still applicable. On the other hand, since this normalization puts all points (including the outliers) on the surface of the hyperphere, and hence as roots of the quadratic, non-homogenous polynomial $P(x) = \mathbf{x}^T\mathbf{x} - 1$, the approach in [10] will fail to detect outliers. The effectiveness of this approach is illustrated in Fig. 2(d). As shown in the figure, normalizing the data prior to applying the proposed algorithm allows for correctly detecting all the outliers.

3.5. Robust Subspace Clustering Algorithm

Collecting the ideas from sections 3.2-3.4 leads to the following robust inlier selection algorithm:

In the algorithm above, steps 4-11 implement the ideas described in Section 3.2, by first finding the polynomial $Q_{n,h}$ (steps 4-7), selecting as reliable inliers those points

where $Q_{n,h}(\boldsymbol{\nu}(\mathbf{x}_i)) \leq s_{n,d}t$ (step 8) and computing the corresponding moment matrix and its associated orthonormal basis (steps 9-11). Steps 12-16 select the set of inliers from the original data proceeding as outlined in sections 3.3 and 3.4: step 12 selects as outlier the point corresponding to the highest value of $Q_{n,h}$; steps 13 and 14 use this point to compute $\mathbf{p}^*$ the coefficients of the polynomial that solves (19) and steps 15 and 16 select as inlier set all the points where $P^*(\mathbf{x})^2 \leq t_2$. Finally, once the set of inliers is available, robust subspace segmentation proceeds by using an existing clustering method such as RSIM using these inliers.

These ideas are illustrated in Fig. 4, where the center plot shows the values of $Q_{n,h}$ computed using the moment matrix associated with the entire data-set (including outliers), and the "reliable" inliers left after thresholding. The rightmost plot shows the value of $P^2(\mathbf{x})$ calculated using these reliable inliers. Note that, while the plot of $Q_{n,h}$ is "noisy" (e.g. many inliers are above the threshold), $P^2(\mathbf{x})$ provides perfect inlier/outlier separation.

Algorithm 1 Robust Inlier Selection

1: Data: the data matrix $\mathbf{X} \in \mathbb{R}^{d \times N_p}$; n: upper bound on the number of subspaces; t_1, t_2: thresholds
2: $s_{n,d} \leftarrow \binom{n+d-1}{n}$
3: $\mathbf{x}_i \leftarrow \frac{\mathbf{x}_i}{\|\mathbf{x}_i\|_2}$ ▷ normalize the data.
4: Create matrix $\mathbf{V}_n \in \mathbb{R}^{N_p \times s_{n,d}}$ with rows $\boldsymbol{\nu}^T(\mathbf{x}_i)$
5: $\mathbf{M}_n \leftarrow \frac{1}{N_p}\mathbf{V}_n^T\mathbf{V}_n$
6: $\mathbf{U}, \boldsymbol{\Sigma} \leftarrow \mathbf{svd}(\mathbf{M}_n)$
7: $\mathbf{c} \leftarrow \mathbf{U}(\boldsymbol{\Sigma})^{-\frac{1}{2}}$, $\mathbf{Q}_i \leftarrow \|(\mathbf{V}_n\mathbf{c})_i\|^2$
8: $i_{rel} \leftarrow \{i \colon \mathbf{Q}_i \leq s_{n.d}t_1\}$ ▷ select reliable inliers
9: $\mathbf{X}_{rel} \leftarrow \mathbf{X}(:, i_{rel})$, $\mathbf{V}_{n,rel} \leftarrow \mathbf{V}_n(i_{rel}, :)$
10: $\mathbf{M}_{n,rel} \leftarrow \frac{1}{\text{size}(r_{rel})}\mathbf{V}_{n,rel}^T\mathbf{V}_{n,rel}$
11: $\mathbf{U_{rel}}, \boldsymbol{\Sigma}_{rel} \leftarrow \mathbf{svd}(\mathbf{M}_{n,rel})$
12: $i_{max} \leftarrow argmax_i(\mathbf{Q}_i)$ ▷ : select the "worst" outlier
13: $\mathbf{x}_o \leftarrow \mathbf{X}(:, i_{max})$;
14: $\mathbf{c} \leftarrow \mathbf{U_{rel}}(\boldsymbol{\Sigma_{rel}})^{-\frac{1}{2}}$, $\mathbf{p}^* \leftarrow \frac{\mathbf{c}(\mathbf{c}^T\boldsymbol{\nu}(\mathbf{x}_o))}{\|\mathbf{c}^T\boldsymbol{\nu}(\mathbf{x}_o))\|_2^2}$ ▷ $P^*(.)$
15: $\mathbf{h}_i \leftarrow [\boldsymbol{\nu}^T(\mathbf{x}_i)\mathbf{p}^*]^2\}$ ▷ score for point $\mathbf{x}_i$
16: $\mathcal{H} \leftarrow \{\mathbf{x}_i \colon \mathbf{h}_i \leq t_2\}$ ▷ sublevel set of P^*

3.6. Subspaces of different dimensions

When the subspaces have different dimensions, the ideal $I(\mathcal{A})$ is spanned by the polynomials associated with the vectors in $\mathbf{M}^\perp$, the null space of $\mathbf{M}$, which generically has dimension $m > 1$. By continuity, if the inliers are corrupted by (small enough) noise, the ideal is approximately spanned by the m singular vectors associated with the m smallest singular values of the noisy $\mathbf{M}$. Consider now a generic point $\mathbf{x}$ and note that, from (16), $Q_{n,h} \geq \sum_{i=1}^{m} \frac{(\mathbf{u}_i^T\boldsymbol{\nu}(\mathbf{x}))^2}{\sigma_i}$. It follows that for points far from the inlier spaces $Q_{n,h}$ is large, since $(\mathbf{u}_i^T\boldsymbol{\nu}(\mathbf{x}))^2 \gg \sigma_i$ for at least one i. Thus, the technique proposed in Section 3.4 to detect outliers and estimate reliable inliers still holds.

4. Experimental Evaluation

Datasets. We used data from the MNIST digits [12], Extended Yale B [6] Coil-100 [17], Caltech-256 [7], and Hopkins155 datasets [22] (see Fig. 1). The MNIST dataset has 70,000 images of handwritten digits 0 to 9. Random pairs of Digits between 1 and 9 were used as inliers. Outliers were images randomly selected from Digit 0. The Extended Yale B dataset has 64 images for each of its 38 subjects. All the images of a subset of random individuals were selected as inliers while outliers were images randomly selected from all the remaining subjects. The Coil-100 dataset has 72 images for each of its 100 objects. Random pairs of objects were used as inliers, and outliers were images randomly selected from the remaining 98 objects. The Caltech-256 database contains VGG features for images from 256 categories, with an average of 80 images each, and a clutter category with 827 images, which was used to randomly select outliers. Because Yale, Coil-100 and Caltech 256 have relatively few images per class, they were augmented using 2 (Yale and Caltech 256) and 3 (Coil-100) duplicates for each sample. The Hopkins155 is a challenging dataset with 155 video sequences and automatically detected (noisy) point trajectories of multiple number and types of motions. We used as inliers the trajectories without any pre-processing, and outliers were generated using a random walk.

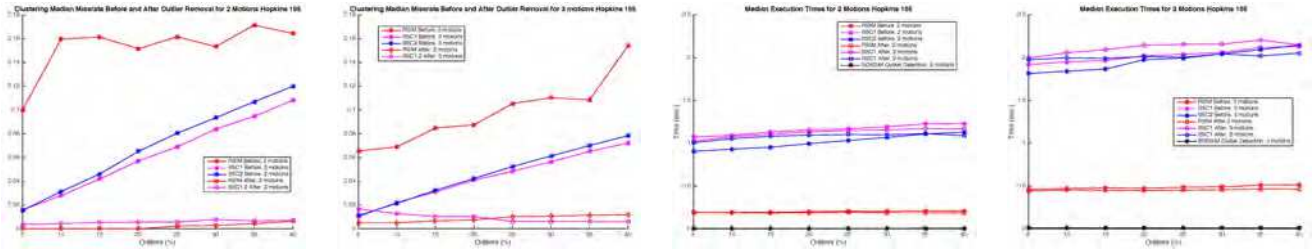
Evaluation and Comparison to state of the art[2]: We compared the performance of algorithm 1 for inlier selection against the state of the art, R-graph [25]. Following their experimental protocol, we measure performance using the average of the Area Under the Curve (AUC) of the receiving operating curves (roc) as t_2 is varied, the average of the maximum F1 score (i.e. geometric mean of Precision and Recall) and the average execution time in seconds. The AUC is always between 0 and 1, with a perfect model having AUC $= 1$ and a model that guesses randomly having AUC ≈ 0.5. The maximum F1 score is the best F1 as threshold t_2 is varied along the roc. Thus, a maximum F1 score of 1, signifies that there exists a threshold t_2 for which both precision and recall are 1.

Additionally, we evaluated the effect of cleaning outliers on the performance of linear subspace segmentation. To this effect, we compared the performance of two state of art clustering algorithms, SSC [4] and RSIM [9], when using the Hopkins155 corrupted data vs when using the cleaned data. Both algorithms rely on an affinity matrix and spectral clustering. However, while SSC uses an ADMM based implementation exploiting sparsity and self-representation

[2]We are grateful to the authors that shared their code for their algorithms R-graph, RSIM, and SSC

Table 1: Outlier Detection Performance

			Ours			R-graph		
Set	Clusters	Outlliers (%)	AUC (mean %)	F1 (mean %)	Time (sec)	AUC (mean %)	F1 (mean %)	Time (sec)
MNIST	2	5	78.89	**35.48**	**12.19**	**81.76**	32.33	150.60
Ext. Yale B	5	15	**99.67**	**99.31**	**1.28**	95.25	94.37	17.68
Ext. Yale B	10	15	**99.15**	**98.70**	**7.00**	91.86	91..12	98.48
Coil-100	2	25	**98.52**	**98.29**	**0.38**	96.20	97.13	6.56
Caltech 256	2	25	**93.79**	84.89	**2.67**	78.40	**86.77**	100.37
Hopkins 155	2	40	**99.81**	**99.60**	**0.014**	64.81	45.61	0.64
Hopkins 155	3	40	**100.00**	**100.00**	**0.023**	72.10	43.62	0.96
Hopkins 155	5	40	**100.00**	**100.00**	**0.029**	26.50	9.64	0.58



(a) (b) (c) (d)

Figure 5: Median misclassification rates RSIM, SSC1 and SSC2 for the Hopkins 155 dataset sequences with (a) 2 motions and (b) 3 motions before and after removing outliers and the corresponding average execution times for clustering and outlier detection (c) and (d).

to overcome outliers, RSIM does not do anything to explicitly address their presence.

Implementation details. The algorithm has only three parameters: an upper bound on the number of subspaces, a flag to indicate whether to use affine subspaces or not, and a threshold t_1. We report results using as upper bound on the number of subspaces, the number of inlier clusters. However, increasing this number did not have any significant effect. Threshold t_1 was fixed at 1. For R-graph, the algorithm has two parameters λ and α. We used the same values as provided by the authors in their code.

Results: The results of the experiments are summarized in Table 1 and Fig. 5. The outlier detection performance experiments provided in the table, show that the proposed algorithm not only has close to perfect performance in many cases, but also that it is 10 to 50 times faster than R-graph. As expected, the gap in execution time grows quickly with the dimension of the data, since the computational complexity of the proposed algorithm grows at most linearly with the number of data points (which only affect the number of sums performed to compute $\mathbf{M}_n$). The plots in Fig. 5 clearly show the degradation in classification (missrate) performance for both algorithms as the number of outliers increases. SSC does a better job than RSIM in the presence of outliers, but clearly does not remove all of them as its performance is far worse than when our algorithm is used to pre-process the data. Interestingly, RSIM performs significantly better than SSC1 and SSC2 after the outliers are removed. It also should be noted that the computational cost of performing the cleaning step is negligible compared to the time required for the segmentation.

5. Conclusions

In this paper we proposed a new approach to the problem of detecting outliers in a set of noisy points drawn from an arrangement of subspaces. The main idea is to first determine a set of reliable inliers and then use this set to find a robust estimate of a polynomial in the vanishing ideal of the arrangement. This is accomplished by using a stochastic reinterpretation of algebraic clustering methods to show that this estimate can be explicitly written in terms of the singular value decomposition of a moments matrix directly obtained from the data. As shown in the paper, the sublevel sets of this polynomial contain (with a probability that can be determined a-priori) the true inliers in the original data. Notably, the proposed method is virtually parameter free and only requires performing two singular value decompositions, avoiding the need to solve potentially costly regularized optimization problems. Finally, combining the proposed approach with existing clustering algorithms leads to substantial performance improvements.

References

[1] R. Basri and D. W. Jacobs. Lambertian reflectance and linear subspaces. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(2):218–233, 2003.

[2] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.

[3] Y. Cheng, Y. Wang, O. Camps, and M. Sznaier. Subspace clustering with priors via sparse quadratically constrained quadratic programming. *CVPR*, 2016.

[4] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2790–2797. IEEE, 2009.

[5] 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*, 24(6):381–395, 1981.

[6] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE transactions on pattern analysis and machine intelligence*, 23(6):643–660, 2001.

[7] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. 2007.

[8] W. Hong, J. Wright, K. Huang, and Y. Ma. Multiscale hybrid linear models for lossy image representation. *Image Processing, IEEE Transactions on*, 15(12):3655–3671, 2006.

[9] P. Ji, M. Salzmann, and H. Li. Shape interaction matrix revisited and robustified: Efficient subspace clustering with corrupted and incomplete data. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 4687–4695, 2015.

[10] J. Lasserre and E. Pauwels. Sorting out typicality with the inverse moment matrix SOS polynomial. *CoRR*, abs/1606.03858, 2016.

[11] J.-B. Lasserre. *Moments, positive polynomials and their applications*. Imperial College Press Optimization Series. Imperial College Press, London, 2010.

[12] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *IEEE*, volume 86, pages 2278–2324, Nov. 1998.

[13] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. 2012.

[14] R. Liu, Z. Lin, F. De la Torre, and Z. Su. Fixed-rank representation for unsupervised visual learning. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 598–605. IEEE, 2012.

[15] Y. Ma and R. Vidal. Identification of deterministic switched arx systems via identification of algebraic varieties. In *Hybrid Systems: Computation and Control*, pages 449–465. Springer, 2005.

[16] Y. Ma, A. Y. Yang, H. Derksen, and R. Fossum. Estimation of subspace arrangements with applications in modeling and segmenting mixed data. *SIAM review*, 50(3):413–458, 2008.

[17] S. Nayar, S. Nene, and H. Murase. Columbia object image library (coil 100). *Department of Comp. Science, Columbia University, Tech. Rep. CUCS-006-96*, 1996.

[18] S. A. Nene, S. K. Nayar, and H. Murase. Columbia object image library (coil-100). Cucs-006-96, Columbia University, February 1996.

[19] N. Ozay, M. Sznaier, C. Lagoa, and O. Camps. Gpca with denoising: A moments-based convex approach. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3209–3216. IEEE, 2010.

[20] M. Soltanolkotabi, E. Elhamifar, and E. Candes. Robust subspace clustering. *arXiv preprint arXiv:1301.2603*, 2013.

[21] R. Tempo, G. Calafiore, and F. Dabbene. *Randomized Algorithms for Analysis and Control of Uncertain Systems with Applications*. Springer-Verlag, London, 2013.

[22] R. Tron and R. Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.

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

[24] Y. Xu. Orthogonal polynomials of several variables. *Encyclopedia of Mathematics and its Applications*, 81, 2001.

[25] C. You, D. P. Robinson, and R. Vidal. Provable self-representation based outlier detection in a union of subspaces. In *IEEE CVPR*, pages 3395–3404, 2017.