Hyperplane Clustering via Dual Principal Component Pursuit

Manolis C. Tsakiris [1] **René Vidal** [1]

Abstract

State-of-the-art methods for clustering data drawn from a union of subspaces are based on sparse and low-rank representation theory and convex optimization algorithms. Existing results guaranteeing the correctness of such methods require the dimension of the subspaces to be small relative to the dimension of the ambient space. When this assumption is violated, as is, e.g., in the case of hyperplanes, existing methods are either computationally too intensive (e.g., algebraic methods) or lack sufficient theoretical support (e.g., K-Hyperplanes or RANSAC). In this paper we provide theoretical and algorithmic contributions to the problem of clustering data from a union of hyperplanes, by extending a recent subspace learning method called Dual Principal Component Pursuit (DPCP) to the multi-hyperplane case. We give theoretical guarantees under which, the non-convex ℓ_1 problem associated with DPCP admits a unique global minimizer equal to the normal vector of the most dominant hyperplane. Inspired by this insight, we propose sequential (RANSAC-style) and iterative (K-Hyperplanes-style) hyperplane learning DPCP algorithms, which, via experiments on synthetic and real data, are shown to outperform or be competitive to the state-of-the-art.

1. Introduction

1.1. Hypeprlane clustering

Subspace clustering, the problem of clustering data drawn from a union of linear subspaces, is an important problem in machine learning, pattern recognition and computer vision (Vidal et al., 2016). A particular case of this problem is *hyperplane clustering*, which arises when the data lie in a union of hyperplanes, as in, e.g., projective motion segmentation (Vidal et al., 2006), 3D point cloud analysis (Sampath & Shan, 2010) and hybrid system identification (Vidal et al., 2003; Bako, 2011). Even though in some ways hyperplane clustering is simpler than general subspace clustering, since, e.g., the dimensions of the subspaces are equal and known a priori, modern *self-expressiveness-based* methods (Liu et al., 2013; Lu et al., 2012; Elhamifar & Vidal, 2013; Wang et al., 2013; You et al., 2016), in principle do not apply in this case, because they require small *relative subspace dimensions* d/D, where d, D are the dimensions of the subspace and ambient space, respectively.

From a theoretical point of view, one of the most appropriate methods for hyperplane clustering is Algebraic Subspace Clustering (ASC) which gives closed-form solutions by means of factorization or differentiation of polynomials (Vidal et al., 2005). However, the main drawback of ASC is its exponential complexity[1], which makes it impractical in many settings. Another method that is theoretically justifiable for clustering hyperplanes is Spectral Curvature Clustering (SCC) (Chen & Lerman, 2009), which computes a D-fold affinity between all D-tuples of points in the dataset. As in the case of ASC, SCC has combinatorial complexity and becomes cumbersome for large D. On the other hand, the intuitive classical method of K-Hyperplanes (KH) (Bradley & Mangasarian, 2000), which alternates between assigning clusters and fitting a new normal vector to each cluster with PCA, is perhaps the most practical method for hyperplane clustering, since it is simple to implement and it is robust to noise. However, KH is sensitive to outliers and is guaranteed to converge only to a local minimum; hence multiple restarts are in general required. Median K-Flats (MKF) (Zhang et al., 2009) shares a similar objective function as KH, but uses the ℓ_1-norm instead of the ℓ_2-norm, in an attempt to gain robustness to outliers. The minimization is done via a stochastic gradient descent scheme, and searches directly for a basis of each subspace, which makes it slower to converge for hyperplanes. Finally, any single robust subspace learning method suitable for high relative dimensions, such as RANSAC (Fischler & Bolles, 1981) or REAPER (Lerman et al., 2015), can be applied either i) in a sequential fashion

[1] Center for Imaging Science, Johns Hopkins University, Baltimore, MD, USA. Correspondence to: Manolis C. Tsakiris <m.tsakiris@jhu.edu>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017. Copyright 2017 by the author(s).

[1] The issue of robustness to noise for ASC has been recently addressed in Tsakiris & Vidal 2015b; 2017c.

by first learning the most dominant hyperplane, removing the points lying close to it, learning the second most dominant hyperplane, and so on, or ii) in an iterative fashion, by assigning points to clusters, fitting a hyperplane per cluster, reassigning the points to new clusters and so on.

1.2. Dual principal component pursuit

Dual Principal Component Pursuit (DPCP) (Tsakiris & Vidal, 2015a; 2017a) is an ℓ_1 single subspace learning method, which aims at recovering the orthogonal complement of the subspace in the presence of outliers, and as such it is particularly suited for hyperplanes. DPCP searches for the normal vector to a hyperplane by solving a non-convex ℓ_1 minimization problem on the sphere, or a recursion of linear programming relaxations, and under certain conditions, the normal to the hyperplane is the unique global solution to this non-convex ℓ_1 problem, as well as the limit point of the LP recursion. Motivated by the robustness of DPCP to outliers, one could naively use it for hyperplane clustering by recovering the normal vector to a hyperplane one at a time, while treating points from other hyperplanes as outliers. However, such a scheme is not a priori guaranteed to succeed because the assumptions in the theorems of correctness of DPCP assume that outliers are uniformly distributed on the sphere, an assumption which is violated when the data come from a union of hyperplanes.

1.3. Paper contributions

In this paper we provide a theoretical analysis of the non-convex ℓ_1 DPCP problem for data drawn from a union of hyperplanes. We show that as long as the hyperplanes are sufficiently separated, the dominant hyperplane is sufficiently dominant and the points are uniformly distributed (in a deterministic sense) inside their associated hyperplanes, the normal vector of the dominant hyperplane is the unique (up to sign) global minimizer of the DPCP problem. This suggests a DPCP-based sequential hyperplane learning algorithm, which uses DPCP to compute a dominant hyperplane, then a second dominant hyperplane and so on. Experiments on synthetic data show that this DPCP-based algorithm significantly improves over similar sequential algorithms, which are based on RANSAC or REAPER. Finally, 3D plane clustering experiments on real 3D point clouds show that an iterative (KH-style) DPCP scheme is very competitive to RANSAC, which is the predominant state-of-the-art method for such applications.

2. Preliminaries

2.1. Data model and the hyperplane clustering problem

Consider given a collection $\boldsymbol{\mathcal{X}} = [\boldsymbol{x}_1, \dots, \boldsymbol{x}_N] \in \mathbb{R}^{D \times N}$ of N points of the unit sphere $\mathbb{S}^{D-1}$ of $\mathbb{R}^D$, that lie in a union (arrangement) $\mathcal{A}$ of n hyperplanes $\mathcal{H}_1, \dots, \mathcal{H}_n$ of $\mathbb{R}^D$, i.e., $\boldsymbol{\mathcal{X}} \subset \mathcal{A} = \bigcup_{i=1}^n \mathcal{H}_i$, where each hyperplane $\mathcal{H}_i$ is the set of points of $\mathbb{R}^D$ that are orthogonal to a *normal vector* $\boldsymbol{b}_i \in \mathbb{S}^{D-1}$, i.e., $\mathcal{H}_i = \left\{ \boldsymbol{x} \in \mathbb{R}^D : \boldsymbol{x}^\top \boldsymbol{b}_i = 0 \right\}$, $i \in [n] := \{1, \dots, n\}$. We assume that the data $\boldsymbol{\mathcal{X}}$ lie in *general position* in $\mathcal{A}$, by which we mean two things. First, we mean that there are no linear relations among the points other than the ones induced by their membership to the hyperplanes. In particular, every $(D-1)$ points coming from $\mathcal{H}_i$ form a basis for $\mathcal{H}_i$ and any D points of $\boldsymbol{\mathcal{X}}$ that come from at least two distinct $\mathcal{H}_i, \mathcal{H}_{i'}$ are linearly independent. Second, we mean that the points $\boldsymbol{\mathcal{X}}$ uniquely define the hyperplane arrangement $\mathcal{A}$, in the sense that $\mathcal{A}$ is the only arrangement of n hyperplanes that contains $\boldsymbol{\mathcal{X}}$. This can be verified computationally by checking that there is only one up to scale homogeneous polynomial of degree n that fits the data, see Vidal et al. 2005; Tsakiris & Vidal 2017c for details. We assume that for every $i \in [n]$, precisely N_i points of $\boldsymbol{\mathcal{X}}$, denoted by $\boldsymbol{\mathcal{X}}_i = [\boldsymbol{x}_1^{(i)}, \dots, \boldsymbol{x}_{N_i}^{(i)}]$, belong to $\mathcal{H}_i$, with $\sum_{i=1}^n N_i = N$. With that notation, $\boldsymbol{\mathcal{X}} = [\boldsymbol{\mathcal{X}}_1, \dots, \boldsymbol{\mathcal{X}}_n]\boldsymbol{\Gamma}$, where $\boldsymbol{\Gamma}$ is an unknown permutation matrix, indicating that the hyperplane membership of the points is unknown. Moreover, we assume an ordering $N_1 \geq N_2 \geq \cdots \geq N_n$, and we refer to $\mathcal{H}_1$ as *the dominant hyperplane*. After these preparations, the problem of hyperplane clustering can be stated as follows: given the data $\boldsymbol{\mathcal{X}}$, find the number n of hyperplanes associated to $\boldsymbol{\mathcal{X}}$, a normal vector to each hyperplane, and a clustering of the data $\boldsymbol{\mathcal{X}} = \bigcup_{i=1}^n \boldsymbol{\mathcal{X}}_i$ according to hyperplane membership.

2.2. Review of dual principal component pursuit

Dual Principal Component Pursuit (DPCP) (Tsakiris & Vidal, 2017a) is a robust single subspace learning method. Given unlabeled data $\boldsymbol{\mathcal{X}}$, which consist of inliers in a single subspace $\mathcal{S}$ of $\mathbb{R}^D$ of dimension $d < D$, together with outliers to the subspace, DPCP computes a basis for the orthogonal complement of the inlier subspace $\mathcal{S}$. The key idea of DPCP is to identify a single hyperplane $\mathcal{H}$ with normal vector $\boldsymbol{b}$ that is *maximal* with respect to the data $\boldsymbol{\mathcal{X}}$. Such a *maximal hyperplane* is defined by the property that it must contain a maximal number of points $N_{\mathcal{H}}$ from the dataset, i.e., $N_{\mathcal{H}'} \leq N_{\mathcal{H}}$ for any other hyperplane $\mathcal{H}'$ of $\mathbb{R}^D$. Notice that such a maximal hyperplane can be characterized as a solution to the combinatorial problem

$$\min_{\boldsymbol{b}} \left\| \boldsymbol{\mathcal{X}}^\top \boldsymbol{b} \right\|_0 \ \text{ s.t. } \ \boldsymbol{b} \neq \boldsymbol{0}, \tag{1}$$

since $\left\| \boldsymbol{\mathcal{X}}^\top \boldsymbol{b} \right\|_0$ is the number of non-zero entries of $\boldsymbol{\mathcal{X}}^\top \boldsymbol{b}$, which is precisely the number of data points in $\boldsymbol{\mathcal{X}}$ that lie outside the hyperplane defined by $\boldsymbol{b}$. If $\mathcal{S}$ is a hyperplane, i.e., $\dim \mathcal{S} = D-1$, and if there are at least $\dim \mathcal{S} + 1 = D$ inliers, it is straightforward to show that (1) has a unique up to scale global minimizer, the normal vector to the inlier hyperplane. Since (1) is hard to solve, we relax it to

$$\min_{\boldsymbol{b}} \left\|\boldsymbol{\mathcal{X}}^\top \boldsymbol{b}\right\|_1 \ \text{s.t.} \ \|\boldsymbol{b}\|_2 = 1, \tag{2}$$

which is still challenging to solve, since it is a non-smooth non-convex optimization problem on the sphere. Problem (2) has appeared several times in the literature (Späth & Watson, 1987; Spielman et al., 2013; Qu et al., 2014; Sun et al., 2015). In fact, Späth & Watson 1987 proved the following fascinating result.

Proposition 1 *(Späth & Watson, 1987) Let $\boldsymbol{\mathcal{X}}$ be a $D \times N$ matrix of rank D. Then any global minimizer of (2) must be orthogonal to $D-1$ linearly independent columns of $\boldsymbol{\mathcal{X}}$.*

Proposition 1 establishes an encouraging property of problem (2) towards recovering the normal vector of the inlier hyperplane as its global minimizer. Indeed, it would suffice that the $D-1$ linearly independent points of $\boldsymbol{\mathcal{X}}$ that a global minimizer is orthogonal to, be points of the inlier hyperplane. Even though a priori it is not clear under what conditions this is the case, Tsakiris & Vidal 2015a provided an answer, informally stated as follows.

Proposition 2 *(Tsakiris & Vidal, 2015a) Suppose that the inliers are sufficiently uniformly distributed (in a deterministic sense defined in Grabner et al. 1997) inside the intersection of the inlier hyperplane and the (unit) sphere, and that the outliers are sufficiently uniformly distributed on the sphere. Then (2) has a unique up to sign global minimizer, equal to the normal vector of the inlier hyperplane.*

It was further shown in Tsakiris & Vidal 2015a that under the conditions of Proposition 2, and assuming that $\hat{\boldsymbol{n}}_0$ is a unit ℓ_2-norm vector sufficiently far from the inlier hyperplane, the recursion of linear programs

$$\boldsymbol{n}_{k+1} := \operatorname*{argmin}_{\boldsymbol{b}^\top \hat{\boldsymbol{n}}_k = 1} \left\|\boldsymbol{\mathcal{X}}^\top \boldsymbol{b}\right\|_1, \tag{3}$$

will converge in a finite number of iterations to the global minimizer of (2).[2]

2.3. Hyperplane clustering via DPCP?

Given the discussion in §2.2, the DPCP problem (2) seems a natural mechanism towards retrieving the normal vectors to the hyperplanes associated with a dataset $\boldsymbol{\mathcal{X}}$ lying in a hyperplane arrangement. Indeed, one may be tempted to solve problem (2) for such a dataset with the hope of obtaining a unique global minimizer, which is orthogonal to one of the underlying hyperplanes. In this case, points coming from the remaining hyperplanes are treated as outliers. Such an idea would give rise to *sequential* and *iterative* DPCP hyperplane clustering algorithms as described in §1.1. A sufficient condition for the correctness of this procedure would be that the global minimizer of the DPCP problem (2) be orthogonal to the inlier subspace. However, the conditions for this to be the case do not immediately follow from the work of Tsakiris & Vidal 2015a, since in the case of hyperplane clustering the *outliers* lie in a union of $n-1$ hyperplanes, and thus can not be uniformly distributed on the sphere, as the conditions of Tsakiris & Vidal 2015a require. The rest of the paper is devoted to providing such theoretical guarantees (§3), as well as introducing DPCP-based hyperplane clustering algorithms (§4).

3. Theoretical Contributions

3.1. Theoretical analysis of the continuous problem

As it turns out, one can gain important insights about the analysis of the DPCP problem (2) for data in a hyperplane arrangement, by first analyzing a certain *continuous* problem. To see what that problem is, let $\hat{\mathcal{H}}_i = \mathcal{H}_i \cap \mathbb{S}^{D-1}$, and note first that for any $\boldsymbol{b} \in \mathbb{S}^{D-1}$ we have

$$\frac{1}{N_i}\sum_{j=1}^{N_i} \left|\boldsymbol{b}^\top \boldsymbol{x}_j^{(i)}\right| \simeq \int_{\boldsymbol{x} \in \hat{\mathcal{H}}_i} \left|\boldsymbol{b}^\top \boldsymbol{x}\right| d\mu_{\hat{\mathcal{H}}_i}, \tag{4}$$

where the LHS of (4) is precisely $\frac{1}{N_i}\left\|\boldsymbol{\mathcal{X}}_i^\top \boldsymbol{b}\right\|_1$ and can be viewed as an approximation ($\simeq$) via the point set $\boldsymbol{\mathcal{X}}_i$ of the integral on the RHS of (4), with $\mu_{\hat{\mathcal{H}}_i}$ denoting the uniform measure on $\hat{\mathcal{H}}_i$. Letting θ_i be the principal angle between $\boldsymbol{b}$ and $\boldsymbol{b}_i$, i.e., the unique angle $\theta_i \in [0, \pi/2]$ such that $\cos(\theta_i) = |\boldsymbol{b}^\top \boldsymbol{b}_i|$, and $\pi_{\mathcal{H}_i} : \mathbb{R}^D \to \mathcal{H}_i$ the orthogonal projection onto $\mathcal{H}_i$, we have for any $\boldsymbol{x} \in \mathcal{H}_i$ that

$$\boldsymbol{b}^\top \boldsymbol{x} = \boldsymbol{b}^\top \pi_{\mathcal{H}_i}(\boldsymbol{x}) = \left(\pi_{\mathcal{H}_i}(\boldsymbol{b})\right)^\top \boldsymbol{x} \tag{5}$$

$$= \boldsymbol{h}_{i,\boldsymbol{b}}^\top \boldsymbol{x} = \sin(\theta_i)\hat{\boldsymbol{h}}_{i,\boldsymbol{b}}^\top \boldsymbol{x}, \tag{6}$$

with $\boldsymbol{h}_{i,\boldsymbol{b}} := \pi_{\mathcal{H}_i}(\boldsymbol{b})$ and $\hat{\boldsymbol{h}}_{i,\boldsymbol{b}} := \boldsymbol{h}_{i,\boldsymbol{b}} / \|\boldsymbol{h}_{i,\boldsymbol{b}}\|_2$. Hence,

$$\int_{\boldsymbol{x} \in \hat{\mathcal{H}}_i} \left|\boldsymbol{b}^\top \boldsymbol{x}\right| d\mu_{\hat{\mathcal{H}}_i} = \left[\int_{\boldsymbol{x} \in \hat{\mathcal{H}}_i} \left|\hat{\boldsymbol{h}}_{i,\boldsymbol{b}}^\top \boldsymbol{x}\right| d\mu_{\hat{\mathcal{H}}_i}\right] \sin(\theta_i) \tag{7}$$

$$= \left[\int_{\boldsymbol{x} \in \mathbb{S}^{D-2}} |x_1| d\mu_{\mathbb{S}^{D-2}}\right] \sin(\theta_i) = c \sin(\theta_i), \tag{8}$$

c being the average height of the unit hemisphere of $\mathbb{R}^{D-1}$. We can now view the objective function of (2), $\left\|\boldsymbol{\mathcal{X}}^\top \boldsymbol{b}\right\|_1 =$

$$\sum_{i=1}^n \left\|\boldsymbol{\mathcal{X}}_i^\top \boldsymbol{b}\right\|_1 = \sum_{i=1}^n N_i \left(\frac{1}{N_i}\sum_{j=1}^{N_i} \left|\boldsymbol{b}^\top \boldsymbol{x}_j^{(i)}\right|\right), \tag{9}$$

[2] Remarkably, (3) was first proposed in Späth & Watson 1987 as a means of solving (2), and it was established that it converges in a finite number of iterations to a critical point of (2).

as an approximation via $\boldsymbol{\mathcal{X}}$ of the function $\mathcal{J}(\boldsymbol{b}) :=$

$$\sum_{i=1}^{n} N_i \left(\int_{\boldsymbol{x} \in \hat{\mathcal{H}}_i} \left| \boldsymbol{b}^\top \boldsymbol{x} \right| d\mu_{\hat{\mathcal{H}}_i} \right) \overset{(8)}{=} \sum_{i=1}^{n} N_i \, c \, \sin(\theta_i). \quad (10)$$

In that sense, the continuous counterpart of problem (2) is

$$\min_{\boldsymbol{b} \in \mathbb{S}^{D-1}} \ \mathcal{J}(\boldsymbol{b}) := N_1 \, \sin(\theta_1) + \cdots + N_n \, \sin(\theta_n). \quad (11)$$

Note that $\sin(\theta_i)$ is the distance between the line spanned by $\boldsymbol{b}$ and the line spanned by $\boldsymbol{b}_i$.[3] Hence, every global minimizer $\boldsymbol{b}^*$ of problem (11) minimizes the sum of the weighted distances of $\operatorname{Span}(\boldsymbol{b}^*)$ from $\operatorname{Span}(\boldsymbol{b}_1), \dots, \operatorname{Span}(\boldsymbol{b}_n)$, and thus represents a weighted geometric median of these lines. Even though medians in Riemmannian/Grassmannian manifolds are an active subject of research (Draper et al., 2014; Ghalieh & Hajja, 1996), we are not aware of any literature that studies (11). The advantage of working with (11) instead of (2), is that the global minimizers of (11) depend solely on the *weights* N_i as well as on the geometry of the arrangement, captured by the principal angles θ_{ij} between $\boldsymbol{b}_i$ and $\boldsymbol{b}_j$. In contrast, the global minimizers of the *discrete* problem (2) will in principle also depend on the distribution of the points $\boldsymbol{\mathcal{X}}$. From that perspective, understanding when problem (11) has as unique solution $\pm\boldsymbol{b}_1$, is essential for understanding the potential of (2) for hyperplane clustering. Towards that end, we next provide a series of results pertaining to (11).[4] The first configuration that we examine is that of two hyperplanes. In that case the weighted geometric median of the two lines spanned by the normals to the hyperplanes always corresponds to one of the two normals:

Proposition 3 *Consider an arrangement of two hyperplanes in $\mathbb{R}^D$ with normal vectors $\boldsymbol{b}_1, \boldsymbol{b}_2$ and weights $N_1 \geq N_2$. Then the set $\mathfrak{B}^*$ of global minimizers of* (11) *satisfies:*

1. *If $N_1 = N_2$, then $\mathfrak{B}^* = \{\pm\boldsymbol{b}_1, \pm\boldsymbol{b}_2\}$.*
2. *If $N_1 > N_2$, then $\mathfrak{B}^* = \{\pm\boldsymbol{b}_1\}$.*

When $N_1 > N_2$, problem (11) recovers the normal $\boldsymbol{b}_1$ to the dominant hyperplane, irrespectively of how separated the two hyperplanes are, since, according to Proposition 3, the principal angle θ_{12} between $\boldsymbol{b}_1, \boldsymbol{b}_2$ does not play a role. The continuous problem (11) is equally favorable in recovering normal vectors as global minimizers in another extreme situation, where the arrangement consists of up to D perfectly separated (orthogonal) hyperplanes:

Proposition 4 *Consider $n \leq D$ hyperplanes in $\mathbb{R}^D$ with orthogonal normal vectors $\boldsymbol{b}_1, \dots, \boldsymbol{b}_n$, and weights $N_1 \geq N_2 \geq \cdots \geq N_n$. Then the set $\mathfrak{B}^*$ of global minimizers of* (11) *can be characterized as follows:*

1. *If $N_1 = \cdots = N_n$, then $\mathfrak{B}^* = \{\pm\boldsymbol{b}_1, \dots, \pm\boldsymbol{b}_n\}$.*
2. *If $N_1 = \cdots = N_\ell > N_{\ell+1} \geq \cdots N_n$, for some $\ell \in [n-1]$, then $\mathfrak{B}^* = \{\pm\boldsymbol{b}_1, \dots, \pm\boldsymbol{b}_\ell\}$.*

Propositions 3 and 4 are not hard to prove, since for two hyperplanes the objective function is strictly concave, while for orthogonal hyperplanes it is separable. In contrast, the problem becomes harder for $n > 2$ arbitrary hyperplanes. Even when $n = 3$, characterizing the global minimizers of (11) as a function of the geometry and the weights seems challenging. Nevertheless, when the three hyperplanes are equiangular and their weights are equal, the symmetry of the configuration allows us to analytically characterize the median as a function of the angle of the arrangement.

Proposition 5 *Consider three hyperplanes of $\mathbb{R}^D$, with normal vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3$ s.t. $\boldsymbol{b}_i^\top \boldsymbol{b}_j = cos(\theta) > 0$, $i \neq j$, and $N_1 = N_2 = N_3$. Then the set $\mathfrak{B}^*$ of global minimizers of* (11) *satisfies the following phase transition:*

1. *If $\theta > 60°$, then $\mathfrak{B}^* = \{\pm\boldsymbol{b}_1, \pm\boldsymbol{b}_2, \pm\boldsymbol{b}_3\}$.*
2. *If $\theta = 60°$, then $\mathfrak{B}^* = \{\pm\boldsymbol{b}_1, \pm\boldsymbol{b}_2, \pm\boldsymbol{b}_3, \pm\boldsymbol{\mu}\}$.*
3. *If $\theta < 60°$, then $\mathfrak{B}^* = \{\pm\boldsymbol{\mu}\}$,*

where $\boldsymbol{\mu} := (\boldsymbol{b}_1 + \boldsymbol{b}_2 + \boldsymbol{b}_3) / \left\| \boldsymbol{b}_1 + \boldsymbol{b}_2 + \boldsymbol{b}_3 \right\|_2$.

Proposition 5, whose proof uses nontrivial arguments from spherical and algebraic geometry, is particularly enlightening, since it suggests that the global minimizers of (11) are associated to the normals of the underlying arrangement when the hyperplanes are sufficiently separated, while otherwise they seem to be capturing the *median hyperplane* of the arrangement. This is in striking similarity with the results regarding the *Fermat point* of planar and spherical triangles (Ghalieh & Hajja, 1996). However, when the symmetry in Theorem 5 is removed, our proof technique no longer applies, and the problem seems even harder. Even so, one intuitively expects an interplay between the angles and the weights of the arrangement under which, if the hyperplanes are *sufficiently separated* and $\mathcal{H}_1$ is *sufficiently dominant*, then (11) should have a unique global minimizer equal to $\boldsymbol{b}_1$. Our next theorem formalizes this intuition.

Theorem 6 *Consider $n \geq 3$ hyperplanes in $\mathbb{R}^D$, with normals $\boldsymbol{b}_1, \dots, \boldsymbol{b}_n$ of pairwise principal angles θ_{ij} and weights N_i. Define an $(n-1) \times (n-1)$ matrix $\boldsymbol{\Theta}$ with $(i-1, j-1)$ entry given by $N_i N_j \cos(\theta_{ij})$, $1 < i, j \leq n$, and maximum eigenvalue $\sigma_{\max}(\boldsymbol{\Theta})$. If*

$$N_1 > \sqrt{\alpha^2 + \beta^2}, \ \textit{where} \quad (12)$$

$$\alpha := \sum_{i>1} N_i \sin(\theta_{1i}) - \sqrt{\sum_{i>1} N_i^2 - \sigma_{\max}(\boldsymbol{\Theta})} \geq 0, \quad (13)$$

[3] Recall that θ_i is a principal angle, i.e., $\theta_i \in [0, \pi/2]$.

[4] All proofs can be found at Tsakiris & Vidal 2017b.

$$\beta := \sqrt{\sum_{i>1} N_i^2 + 2 \sum_{i \neq j,\, i,j>1} \boldsymbol{\Theta}_{i-1,j-1}}, \quad \textit{and} \tag{14}$$

$$\gamma := \min_{j \neq 1} \sum_{i \neq j} N_i \sin(\theta_{ij}) - \sum_{i>1} N_i \sin(\theta_{1i}) > 0, \tag{15}$$

then problem (11) *admits a unique up to sign global minimizer* $\boldsymbol{b}^* = \pm \boldsymbol{b}_1$.

Let us provide some intuition about the meaning of the quantities α, β and γ in Theorem 6. To begin with, the first term in α is precisely equal to $\mathcal{J}(\boldsymbol{b}_1)$, while the second term in α can be shown to be a lower bound on the objective function $N_2 \sin(\theta_2) + \cdots + N_n \sin(\theta_n)$, if one discards hyperplane $\mathcal{H}_1$. Moving on, the quantity β / N_1 admits a nice geometric interpretation: $\cos^{-1}(\beta / N_1)$ is a lower bound on how small the principal angle of a critical point $\boldsymbol{b}^\dagger \neq \pm \boldsymbol{b}_1$ from $\boldsymbol{b}_1$ can be. Interestingly, the larger N_1, the larger this minimum angle is, which shows that critical hyperplanes $\mathcal{H}^\dagger$ that are distinct from $\mathcal{H}_1$, must be sufficiently separated from $\mathcal{H}_1$. Finally, the second term in γ is $\mathcal{J}(\boldsymbol{b}_1)$, while the first term is the smallest objective value that corresponds to $\boldsymbol{b} = \boldsymbol{b}_i$, $i > 1$, and so (15) simply guarantees that $\mathcal{J}(\boldsymbol{b}_1) < \mathcal{J}(\boldsymbol{b}_i)$, $\forall i > 1$. Next, condition $N_1 > \sqrt{\alpha^2 + \beta^2}$ of Theorem 6 is easier to satisfy when $\mathcal{H}_1$ is *close* to the rest of the hyperplanes (which leads to small α), while the rest of the hyperplanes are *sufficiently separated*[5] (which leads to small α and small β). Regardless, one can show that $\sqrt{2} \sum_{i>1} N_i \geq \sqrt{\alpha^2 + \beta^2}$ and so if $N_1 > \sqrt{2} \sum_{i>1} N_i$, then any global minimizer of (11) has to be one of the normals, irrespectively of what the angles θ_{ij} are. Finally, condition (15) is consistent with condition (12) in that it requires $\mathcal{H}_1$ to be close to $\mathcal{H}_i$, $\forall i > 1$, and $\mathcal{H}_i, \mathcal{H}_j$ to be sufficiently separated for $i, j > 1$. Once again, (15) can always be satisfied irrespectively of the θ_{ij}, by choosing N_1 sufficiently large, since only the positive term in the definition of γ depends on N_1.

3.2. Theoretical analysis of the discrete problem

We now study[6] the discrete formulation of DPCP, i.e., problem (2), for the case where $\boldsymbol{\mathcal{X}} = [\boldsymbol{\mathcal{X}}_1, \ldots, \boldsymbol{\mathcal{X}}_n]\boldsymbol{\Gamma}$, with $\boldsymbol{\mathcal{X}}_i$ being N_i points in $\mathcal{H}_i$, as described in §2.1. For any $i \in [n]$ and $\boldsymbol{b} \in \mathbb{S}^{D-1}$, we can write the quantity $\left\| \boldsymbol{\mathcal{X}}_i^\top \boldsymbol{b} \right\|_1$ as

$$\sum_{j=1}^{N_i} \left| \boldsymbol{b}^\top \boldsymbol{x}_j^{(i)} \right| = \boldsymbol{b}^\top \sum_{j=1}^{N_i} \mathrm{Sign}\left(\boldsymbol{b}^\top \boldsymbol{x}_j^{(i)} \right) \boldsymbol{x}_j^{(i)} \tag{16}$$

$$= N_i \, \boldsymbol{b}^\top \boldsymbol{x}_{i,\boldsymbol{b}}, \quad \boldsymbol{x}_{i,\boldsymbol{b}} := \frac{1}{N_i} \sum_{j=1}^{N_i} \mathrm{Sign}\left(\boldsymbol{b}^\top \boldsymbol{x}_j^{(i)} \right) \boldsymbol{x}_j^{(i)} \tag{17}$$

with $\boldsymbol{x}_{i,\boldsymbol{b}}$ being *the average point of* $\boldsymbol{\mathcal{X}}_i$ *with respect to the orthogonal projection* $\boldsymbol{h}_{i,\boldsymbol{b}} := \pi_{\mathcal{H}_i}(\boldsymbol{b})$ *of* $\boldsymbol{b}$ *onto* $\mathcal{H}_i$*.* $\boldsymbol{x}_{i,\boldsymbol{b}}$ can be viewed as an approximation to the vector integral

$$\int_{\boldsymbol{x} \in \hat{\mathcal{H}}_i} \mathrm{Sign}(\boldsymbol{b}^\top \boldsymbol{x}) \boldsymbol{x} \, d\mu_{\hat{\mathcal{H}}_i} = c \, \hat{\boldsymbol{h}}_{i,\boldsymbol{b}}. \tag{18}$$

This leads us to define the maximal *approximation error*

$$\epsilon_i := \max_{\boldsymbol{b} \in \mathbb{S}^{D-1}} \left\| \boldsymbol{x}_{i,\boldsymbol{b}} - c \, \hat{\boldsymbol{h}}_{i,\boldsymbol{b}} \right\|_2, \tag{19}$$

as $\boldsymbol{b}$ ranges over the entire unit sphere $\mathbb{S}^{D-1}$. Intuitively, the more uniformly distributed the points $\boldsymbol{\mathcal{X}}_i$ are inside $\hat{\mathcal{H}}_i$, the smaller ϵ_i is. This intuition can be formalized by means of the *spherical cap discrepancy* (Grabner et al., 1997; Grabner & Tichy, 1993) of $\boldsymbol{\mathcal{X}}_i$, given by

$$\mathfrak{S}_D(\boldsymbol{\mathcal{X}}_i) := \sup_{\mathcal{C}} \left| \frac{1}{N_i} \sum_{j=1}^{N_i} \mathbb{I}_{\mathcal{C}}\left(\boldsymbol{x}_j^{(i)} \right) - \mu_{\hat{\mathcal{H}}_i}(\mathcal{C}) \right|. \tag{20}$$

In (20) the supremum is taken over all spherical caps $\mathcal{C}$ of the sphere $\hat{\mathcal{H}}_i \cong \mathbb{S}^{D-2}$, where a spherical cap is the intersection of $\mathbb{S}^{D-2}$ with a half-space of $\mathbb{R}^{D-1}$, and $\mathbb{I}_{\mathcal{C}}(\cdot)$ is the indicator function of $\mathcal{C}$, which takes the value 1 inside $\mathcal{C}$ and zero otherwise. $\mathfrak{S}_D(\boldsymbol{\mathcal{X}}_i)$ is a deterministic measure of the uniformity of the point set $\boldsymbol{\mathcal{X}}_i$. By adjusting an argument of Harman 2010, one can show that

$$\epsilon_i \leq \sqrt{5} \mathfrak{S}_D(\boldsymbol{\mathcal{X}}_i), \quad \forall i \in [n], \tag{21}$$

which confirms that uniformly distributed points $\boldsymbol{\mathcal{X}}_i$ correspond to small ϵ_i. We note here that $\mathfrak{S}_D(\boldsymbol{\mathcal{X}}_i)$ decreases with a rate of (Dick, 2014; Beck, 1984)

$$\sqrt{\log(N_i)} N_i^{-\frac{1}{2} - \frac{1}{2(D-2)}}. \tag{22}$$

To state the main theorem of this section (Theorem 8) we need a definition.

Definition 7 *For a set* $\boldsymbol{\mathcal{Y}} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_L] \subset \mathbb{S}^{D-1}$ *and positive integer* $K \leq L$*, define* $\mathcal{R}_{\boldsymbol{\mathcal{Y}},K}$ *to be the maximum circumradius among the circumradii of all polytopes* $\left\{ \sum_{i=1}^K \alpha_{j_i} \boldsymbol{y}_{j_i} : \alpha_{j_i} \in [-1, 1] \right\}$*, where* $j_1, \ldots, j_K$ *are distinct integers in* $[L]$*, and the circumradius of a closed bounded set is the minimum radius among all spheres that contain the set. We now define the quantity of interest as*

$$\mathcal{R} := \max_{\substack{K_1 + \cdots + K_n = D-1 \\ 0 \leq K_i \leq D-2}} \sum_{i=1}^n \mathcal{R}_{\boldsymbol{\mathcal{X}}_i, K_i}. \tag{23}$$

We note that it is always the case that $\mathcal{R}_{\boldsymbol{\mathcal{X}}_i, K_i} \leq K_i$, with this upper bound achieved when $\boldsymbol{\mathcal{X}}_i$ contains K_i colinear points. Combining this fact with the constraint $\sum_i K_i = D - 1$ in (23), we get that $\mathcal{R} \leq D - 1$, and the more uniformly distributed are the points $\boldsymbol{\mathcal{X}}$ inside the hyperplanes, the smaller $\mathcal{R}$ is (even though $\mathcal{R}$ does not go to zero).

[5] We emphasize that the interpretation of *close* and *sufficiently separated* is relative to N_1 and $\theta_{12}, \ldots, \theta_{1n}$.

[6] More detailed arguments and proofs can be found in Tsakiris & Vidal 2017b.

Theorem 8 *Let $\boldsymbol{b}^*$ be a global minimizer of (2) with $\boldsymbol{\mathcal{X}} = [\boldsymbol{\mathcal{X}}_1, \dots, \boldsymbol{\mathcal{X}}_n]\boldsymbol{\Gamma}$, and suppose that $c > \sqrt{2}\epsilon_1$. If*

$$N_1 > \sqrt{\bar{\alpha}^2 + \bar{\beta}^2}, \text{ where} \tag{24}$$

$$\bar{\alpha} := \alpha + c^{-1}\left(\epsilon_1 N_1 + 2\sum_{i>1} \epsilon_i N_i\right), \text{ and} \tag{25}$$

$$\bar{\beta} := \beta + c^{-1}\left(\mathcal{R} + \sum \epsilon_i N_i\right), \tag{26}$$

with α, β as in Theorem 6, and if

$$\bar{\gamma} := \gamma - c^{-1}\left(\epsilon_1 N_1 + \epsilon_2 N_2 + 2\sum_{i>2} \epsilon_i N_i\right) > 0, \tag{27}$$

then problem (2) has a unique minimizer $\pm\boldsymbol{b}_1$.

Notice the similarity of conditions $N_1 > \sqrt{\bar{\alpha}^2 + \bar{\beta}^2}, \bar{\gamma} > 0$ of Theorem 8 with conditions $N_1 > \sqrt{\alpha^2 + \beta^2}, \gamma > 0$ of Theorem 6. In fact $\bar{\alpha} > \alpha, \bar{\beta} > \beta$ and $\bar{\gamma} < \gamma$, which implies that the conditions of Theorem 8 are strictly stronger than those of Theorem 6. This is no surprise since, as we have already remarked, the global minimizers of (2) depend not only on the geometry $\{\theta_{ij}\}$ and the weights $\{N_i\}$ of the hyperplane arrangement, but also on the distribution of the data points (parameters ϵ_i and $\mathcal{R}$). In contrast though to condition (12) of Theorem 6, N_1 now appears in both sides of condition (24) of Theorem 8, which is however harmless: under the assumption $c > \sqrt{2}\epsilon_1$, (24) is equivalent to the positivity of a quadratic polynomial in N_1, whose leading coefficient is positive, and hence (24) can always be satisfied for sufficiently large N_1. Another interesting connection of Theorem 6 to Theorem 8, is that assuming $\lim_{N_i \to \infty} \mathfrak{S}_D(\boldsymbol{\mathcal{X}}_i) = 0$, Theorem 6 can be seen as a limit version of Theorem 8: dividing (24) and (27) by N_1, letting $N_1, \dots, N_n$ go to infinity while keeping each ratio N_i/N_1 fixed, recalling that $\mathcal{R} \leq D - 1$, and noting that in view of (21) we have $\lim_{N_i \to \infty} \epsilon_i = 0$, we see that in the limit we recover the conditions of Theorem 6.

Finally, a theorem of the same flavor gives conditions under which (3) converges in a finite number of iterations to $\boldsymbol{b}_1$ or $-\boldsymbol{b}_1$; see Theorem 7 in Tsakiris & Vidal 2017b.

4. Algorithmic Contributions

4.1. DPCP via iteratively reweighted least squares

Späth & Watson 1987; Tsakiris & Vidal 2015a propose solving the non-convex problem (2) by means of the recursion of convex optimization problems (3), referred to as DPCP-r. This is computationally equivalent to a recursion of linear programs, which can be solved efficiently by an optimized LP solver such as GUROBI. However, these linear programs are in principle not sparse, which may render the running time of this approach prohibitive for big-data applications. To alleviate this issue, we solve (2) by standard *Iteratively Reweighted Least Squares* (IRLS) applied to ℓ_1 minimization problems (Candès et al., 2008; Chartrand & Yin, 2008; Daubechies et al., 2010; Lerman et al., 2015). The resulting algorithm, referred to as DPCP-IRLS, is dramatically faster than solving DPCP-r by GUROBI: a MATLAB implementation on a standard MacBook Pro with a dual core 2.5GHz processor and a total of 4GB cache memory is able to handle 6000 points of $\mathbb{R}^{1000}$ in about one minute, while in such a regime DPCP-r seems, as of now, inapplicable. Moreover, the performance of DPCP-IRLS, investigated in §5, suggests that DPCP-IRLS converges most of the time to a global minimizer of (2); the theoretical justification of this claim is ongoing research.

4.2. Hyperplane clustering algorithms via DPCP

Sequential Hyperplane Learning (SHL) via DPCP. Since at its core DPCP is a single subspace learning method (Tsakiris & Vidal, 2015a), we may as well use it to learn n hyperplanes in the same way that RANSAC (Fischler & Bolles, 1981) is used: learn one hyperplane from the entire dataset, remove the points close to it, then learn a second hyperplane, remove the points close to it, and so on. The main weakness of this approach is well known, and consists of its sensitivity to a thresholding parameter, which is necessary in order to be able to remove points.

To alleviate the need of knowing a good threshold, we propose to replace the process of removing points by a process of appropriately weighting the points. In particular, suppose we solve the DPCP problem (2) on the entire dataset $\boldsymbol{\mathcal{X}}$ and obtain a unit ℓ_2-norm vector $\boldsymbol{b}_1$. Now, instead of removing the points of $\boldsymbol{\mathcal{X}}$ that are close to the hyperplane with normal vector $\boldsymbol{b}_1$ (which would require a threshold parameter), we weight each and every point $\boldsymbol{x}_j$ of $\boldsymbol{\mathcal{X}}$ by its distance $\left|\boldsymbol{b}_1^\top \boldsymbol{x}_j\right|$ from that hyperplane. Then to compute a second hyperplane with normal $\boldsymbol{b}_2$ we apply DPCP on the weighted dataset $\left\{\left|\boldsymbol{b}_1^\top \boldsymbol{x}_j\right| \boldsymbol{x}_j\right\}$. To compute a third hyperplane, the weight of point $\boldsymbol{x}_j$ is chosen as the smallest distance of $\boldsymbol{x}_j$ from the already computed two hyperplanes, i.e., DPCP is now applied to $\left\{\left(\min_{i=1,2}\left|\boldsymbol{b}_i^\top \boldsymbol{x}_j\right|\right)\boldsymbol{x}_j\right\}$. After n hyperplanes have been computed, the clustering of the points is obtained based on their distances to the n hypeprlanes. We note here that the theoretical correctness of this weighted sequential scheme does not follow automatically from the theory presented in this paper, since the latter applies only to unit ℓ_2-norm points; studying DPCP for weighted points is ongoing research.

Iterative Hyperplane Learning (IHL) via DPCP. Another way to do hyperplane learning and clustering via DPCP is to modify the classic K-Hyperplanes, which we will be referring to as IHL-SVD (Bradley & Mangasarian, 2000; Tseng, 2000; Zhang et al., 2009) (see §1.1) by computing the normal vector of each cluster by DPCP, instead of, e.g., SVD; see §5.2 for more details. The resulting algorithm, IHL-DPCP, minimizes (up to a local minimum) the sum of the distances of the points to the estimated hyperplane arrangement, which corresponds to replacing the ℓ_2-norm in the objective of IHL-SVD with the ℓ_1-norm, precisely as in the case of MKF (Zhang et al., 2009).

5. Experimental Evaluation

5.1. Experiments using synthetic data

We evaluate SHL-DPCP (§4) using synthetic data, and compare it with similar algorithms, where instead of solving the DPCP problem (2) (either via DPCP-r or via DPCP-IRLS), one uses REAPER or RANSAC.[7] For fairness, RANSAC does not remove any points as it sequentially learns the hyperplanes, rather it selects them randomly using the probability distribution induced by weights defined in a similar way as in §4. Moreover, it is configured to run at least as long as DPCP-r, which uses a maximum of 20 iterations in (3), while REAPER and DPCP-IRLS use a maximum of 100 iterations and convergence accuracy 10^{-3}. The ambient dimension is set to $D = 4, 9, 30$, as inspired by major applications where hyperplane arrangements appear, e.g., $D = 4$ in 3D point cloud analysis (in homogeneous coordinates), and $D = 9$ in two-view geometry (Cheng et al., 2015). For each choice of D we randomly generate $n = 2, 3, 4$ hyperplanes and sample them as follows. Given n, we set $N = 300n$, with $N_i = \alpha^{i-1}N_{i-1}, i > 1$, where $\alpha \in (0, 1]$ is a parameter that controls the *balancing* of the clusters: $\alpha = 1$ means the clusters are perfectly balanced, while smaller values of α lead to less balanced clusters. We set $\alpha = 0.6$ (for $\alpha = 0.8, 1$ see Tsakiris & Vidal 2017b). Each cluster is sampled from a zero-mean unit-variance Gaussian distribution with support in the corresponding hyperplane. To make the experiment more realistic, we corrupt points from each hyperplane by adding white Gaussian noise of deviation $\sigma = 0.01$ with support in the direction orthogonal to the hyperplane. Moreover, we corrupt the dataset by adding $M/(M + N) = 10\%$ outliers sampled from a standard zero-mean unit-variance Gaussian distribution supported in the ambient space, where M is the number of outliers.

The left column of Figure 1 plots the clustering accuracy over 50 independent experiments as a function of the relative dimension $(D - 1)/D$ and the number of hyperplanes n. As expected, the performance degrades as either the relative dimension or the number of hyperplanes increases. There are at least two interesting things to notice. First, RANSAC is the best method when $D = 4$ irrespectively of the number of hyperplanes, since for such a low ambient dimension the probability that $D - 1 = 3$ randomly selected points lie in the same hyperplane is very high. Indeed, for $D = 4$ RANSAC's accuracy ranges from 99% $(n = 2)$ to 97% $(n = 4)$, as opposed to (for $n = 4$) REAPER (56%) or even DPCP-IRLS (89%) and DPCP-r (94%). On the other hand, DPCP-r is overall the best method with an 86% accuracy in the challenging scenario $(D-1)/D = 0.97, n = 4$, as opposed to 81% for DPCP-IRLS, 42% for REAPER and 28% for RANSAC. The right column of Figure 1 plots the clustering accuracy as a function of n and of the percentage of outliers, for $D = 9$ and additive noise as before. Evidently, DPCP-r and DPCP-IRLS are the best methods, with, e.g., a clustering accuracy of 97% and 94% respectively for $n = 2$ and 50% outliers, as opposed to 66% for RANSAC and 74% for REAPER.

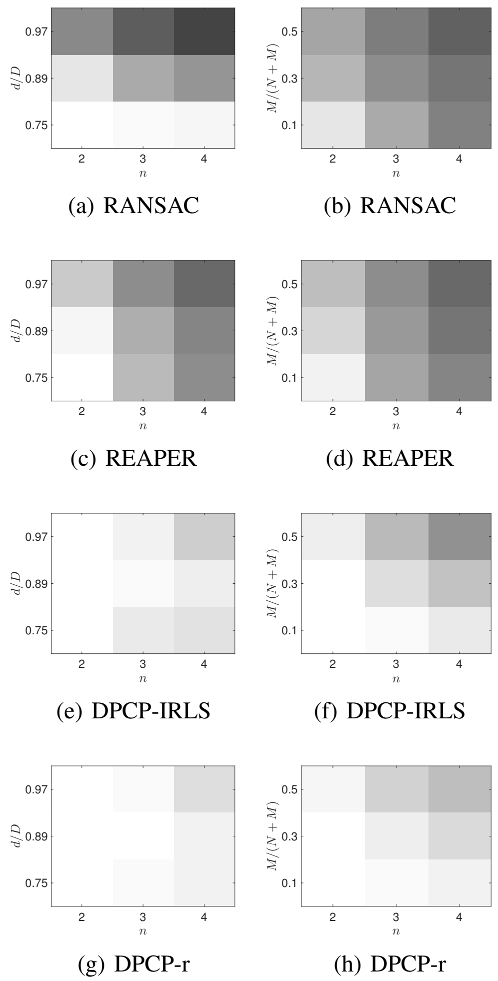
5.2. Experiments using real kinect data

In this section we explore various hyperplane clustering algorithms using the benchmark dataset NYUdepthV2 (Silberman et al., 2012). This dataset consists of 1449 RGBd data instances acquired using the Microsoft kinect sensor. Each instance corresponds to an indoor scene, and consists of the $480 \times 640 \times 3$ RGB data together with depth data for each pixel. The depth data can be used to reconstruct a 3D point cloud associated to the scene. In this experiment we use such 3D point clouds to learn plane arrangements and segment the pixels of the corresponding images based on their plane membership. This is an important problem in robotics, where estimating the geometry of a scene is essential for successful robot navigation.

In such 3D applications RANSAC is the predominant state-of-the-art method, since the probability of sampling three points from the dominant plane is very large. Thus we compare a sequential hyperplane learning RANSAC algorithm (SHL-RANSAC), which uses a threshold $(0.1, 0.01, 0.001)$ for removing points, to iterative K-Hyperplane-like algorithms based on SVD, REAPER, RANSAC and DPCP-IRLS, to be referred to as IHL-SVD, IHL-REAPER, and so on. These algorithms randomly initialize n hyperplanes, they cluster the points according to their distance to these hyperplanes, they refine the hyperplanes by fitting a new hyperplane at each cluster, re-assign points based on the new hyperplanes, and so on, until the objective function converges or 100 iterations are reached. We use 10 independent restarts, and we control the running time of SHL-RANSAC and IHL-RANSAC to be not less than that of IHL-DPCP-IRLS.

The algorithms do not operate on the raw 3D data, rather on standard superpixel representations of the data, where each

[7] We have compared with methods such as SCC or MKF, however we do not report on these methods since they perform significantly more inferior to RANSAC, REAPER or DPCP.



(a) RANSAC (b) RANSAC

(c) REAPER (d) REAPER

(e) DPCP-IRLS (f) DPCP-IRLS

(g) DPCP-r (h) DPCP-r

Figure 1. Sequential hyperplane learning: Left (right) column shows clustering accuracy (white corresponds to 1, black to 0) as a function of the number of hyperplanes n and the relative dimension d/D (percentage $M/(N+M)$ of outliers).

superpixel is represented by its median 3D point, weighted by the size of the superpixel. Moreover, since 3D planes in an indoor seen usually do not pass through a common origin, the algorithms work with homogeneous coordinates. Finally, the algorithms as described so far are purely geometric, in the sense that they do not take into account the spatial coherence of the 3D point cloud (nearby points are likely to lie in the same plane), and so we expect their output segmentation to be spatially incoherent. To associate a spatially smooth image segmentation to each algorithm, we use the normal vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ that the algorithm produced to minimize a Conditional-Random-Field (Sutton & McCallum, 2006) energy function $E(y_1, \ldots, y_N) :=$

$$\sum_{j=1}^{N} |\boldsymbol{b}_{y_j}^\top \boldsymbol{x}_j| + \lambda \sum_{k \in \mathcal{N}_j} \mathrm{CB}_{j,k} \exp\left(-\frac{\|\boldsymbol{x}_j - \boldsymbol{x}_k\|_2^2}{2\sigma_d^2}\right) \delta(y_j \neq y_k). \tag{28}$$

In (28) the first and second terms are known as *unary* and *pairwise potentials*, $y_j \in \{1, \ldots, n\}$ is the plane label of 3D point $\boldsymbol{x}_j$, which is the variable to optimize over, $\mathrm{CB}_{j,k}$ is the length of the common boundary between superpixels j, and k and $\mathcal{N}_j$ indexes the neighbors of $\boldsymbol{x}_j$. The parameter λ in (28) is set to the inverse of twice the maximal row-sum of the pairwise matrix, in order to achieve a balance between unary and pairwise terms. Minimization of (28) is done via Graph-Cuts (Boykov et al., 2001).

Table 1. Clustering error in % of 3D planes from Kinect data without (CRF(0)) and with (CRF(1)) spatial smoothing.

	$n=4$		$n=9$	
method	CRF(0)	CRF(1)	CRF(0)	CRF(1)
SHL-RANSAC	22.78	14.07	29.42	17.47
IHL-RANSAC	**16.80**	**10.71**	**22.78**	**14.24**
IHL-SVD	21.85	14.40	26.22	16.71
IHL-REAPER	20.94	13.71	25.52	16.27
IHL-DPCP-IRLS	20.77	13.64	25.38	16.10

Since NYUdpethV2 does not come with a ground truth annotation based on plane membership, we manually annotated 92 of the 1449 scenes in the dataset, in which dominant planes such as floors, walls, ceilings, tables and so on are present. Table 1 shows the clustering errors of various algorithms on these 92 annotated scenes for the identification of the first n dominant planes of each scene[8], where for SHL-RANSAC the error is averaged over the three different choices of a threshold. As expected, the clustering error increases for all methods as the number of planes to be identified increases. Again as expected, the performance of all algorithms improves significantly if one includes spatial smoothing. Notice that the best method is IHL-RANSAC, and not the sequential SHL-RANSAC, which seems a rather interesting finding. On the other hand, the rest of the methods seem to perform similarly to each other, with IHL-SVD being slightly inferior, since it is less robust to outliers, and IHL-DPCP-IRLS being overall the second best method.

6. Conclusions

In this paper we extended the framework of Dual Principal Component Pursuit (DPCP) to the case of data lying in a union of hyperplanes. We provided theoretical conditions under which the normal vector of the dominant hyperplane is the unique global minimizer of the non-convex ℓ_1 DPCP optimization problem. Moreover, we proposed a fast implementation of DPCP, as well as DPCP-based hyperplane clustering algorithms, which were shown to outperform or be competitive to state-of-the-art algorithms.

[8] If the scene has $m < n$ annotated planes, then the clustering error is computed only with respect to the first m dominant clusters identified by the algorithm.

Acknowledgements

This work was supported by NSF grants 1447822 and 1618637. The first author thanks Dr. Glyn Harman for his help in deriving equation (21), and Dr. Bijan Afsari for interesting conversations on geometric medians in nonlinear manifolds. The authors also thank two of the anonymous reviewers for providing constructive comments.

References

Bako, L. Identification of switched linear systems via sparse optimization. *Automatica*, 47(4):668–677, 2011.

Beck, J. Sums of distances between points on a sphere—an application of the theory of irregularities of distribution to discrete geometry. *Mathematika*, 31(01):33–41, 1984.

Boykov, Y., Veksler, O., and Zabih, R. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11): 1222–1239, 2001.

Bradley, P. S. and Mangasarian, O. L. k-plane clustering. *Journal of Global Optimization*, 16(1):23–32, 2000. ISSN 0925-5001.

Candès, E., Wakin, M., and Boyd, S. Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier Analysis and Applications*, 14(5):877–905, 2008.

Chartrand, R. and Yin, W. Iteratively reweighted algorithms for compressive sensing. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3869–3872. IEEE, 2008.

Chen, G. and Lerman, G. Spectral curvature clustering (SCC). *International Journal of Computer Vision*, 81 (3):317–330, 2009. ISSN 0920-5691.

Cheng, Y., Lopez, J. A., Camps, O., and Sznaier, M. A convex optimization approach to robust fundamental matrix estimation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2170–2178, 2015.

Daubechies, I., DeVore, R., Fornasier, M., and Güntürk, C. S. Iteratively reweighted least squares minimization for sparse recovery. *Communications on Pure and Applied Mathematics*, 63(1):1–38, 2010.

Dick, J. Applications of geometric discrepancy in numerical analysis and statistics. *Applied Algebra and Number Theory*, 2014.

Draper, B., Kirby, M., Marks, J., Marrinan, T., and Peterson, C. A flag representation for finite collections of subspaces of mixed dimensions. *Linear Algebra and its Applications*, 451:15–32, 2014.

Elhamifar, E. and Vidal, R. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11): 2765–2781, 2013.

Fischler, M. A. and Bolles, R. C. RANSAC random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 26:381–395, 1981.

Ghalieh, K. and Hajja, M. The fermat point of a spherical triangle. *The Mathematical Gazette*, 80(489):561–564, 1996.

Grabner, P. J. and Tichy, R.F. Spherical designs, discrepancy and numerical integration. *Math. Comp.*, 60 (201):327–336, 1993. ISSN 0025-5718. doi: 10.2307/2153170. URL http://dx.doi.org/10.2307/2153170.

Grabner, P. J., Klinger, B., and Tichy, R.F. Discrepancies of point sequences on the sphere and numerical integration. *Mathematical Research*, 101:95–112, 1997.

Harman, G. Variations on the koksma-hlawka inequality. *Uniform Distribution Theory*, 5(1):65–78, 2010.

Lerman, G., McCoy, M. B., Tropp, J. A., and Zhang, T. Robust computation of linear models by convex relaxation. *Foundations of Computational Mathematics*, 15 (2):363–410, 2015.

Liu, G., Lin, Z., Yan, S., Sun, J., and Ma, Y. Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):171–184, Jan 2013.

Lu, C-Y., Min, H., Zhao, Z-Q., Zhu, L., Huang, D-S., and Yan, S. Robust and efficient subspace segmentation via least squares regression. In *European Conference on Computer Vision*, 2012.

Qu, Q., Sun, J., and Wright, J. Finding a sparse vector in a subspace: Linear sparsity using alternating directions. In *Advances in Neural Information Processing Systems*, pp. 3401–3409, 2014.

Sampath, A. and Shan, J. Segmentation and reconstruction of polyhedral building roofs from aerial lidar point clouds. *Geoscience and Remote Sensing, IEEE Transactions on*, 48(3):1554–1567, 2010.

Silberman, N., Kohli, P., Hoiem, D., and Fergus, R. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision*, 2012.

Späth, H. and Watson, G.A. On orthogonal linear ℓ_1 approximation. *Numerische Mathematik*, 51(5):531–543, 1987.

Spielman, D.A., Wang, H., and Wright, J. Exact recovery of sparsely-used dictionaries. In *Proceedings of the 23d international joint conference on Artificial Intelligence*, pp. 3087–3090. AAAI Press, 2013.

Sun, J., Qu, Q., and Wright, J. Complete dictionary recovery using nonconvex optimization. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pp. 2351–2360, 2015.

Sutton, C. and McCallum, A. *An introduction to conditional random fields for relational learning*, volume 2. Introduction to statistical relational learning. MIT Press, 2006.

Tsakiris, M. C. and Vidal, R. Dual principal component pursuit. *arXiv:1510.04390v2 [cs.CV]*, 2017a.

Tsakiris, M. C. and Vidal, R. Hyperplane clustering via dual principal component pursuit. *arXiv:1706.01604 [cs.CV]*, 2017b.

Tsakiris, M. C. and Vidal, R. Filtrated algebraic subspace clustering. *SIAM Journal on Imaging Sciences*, 10(1): 372–415, 2017c.

Tsakiris, M.C. and Vidal, R. Dual principal component pursuit. In *ICCV Workshop on Robust Subspace Learning and Computer Vision*, pp. 10–18, 2015a.

Tsakiris, M.C. and Vidal, R. Filtrated spectral algebraic subspace clustering. In *ICCV Workshop on Robust Subspace Learning and Computer Vision*, pp. 28–36, 2015b.

Tseng, P. Nearest q-flat to m points. *Journal of Optimization Theory and Applications*, 105(1):249–252, 2000.

Vidal, R., Soatto, S., Ma, Y., and Sastry, S. An algebraic geometric approach to the identification of a class of linear hybrid systems. In *IEEE Conference on Decision and Control*, pp. 167–172, 2003.

Vidal, R., Ma, Y., and Sastry, S. Generalized Principal Component Analysis (GPCA). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1–15, 2005.

Vidal, R., Ma, Y., Soatto, S., and Sastry, S. Two-view multibody structure from motion. *International Journal of Computer Vision*, 68(1):7–25, 2006.

Vidal, R., Ma, Y., and Sastry, S. *Generalized Principal Component Analysis*. Springer Verlag, 2016.

Wang, Y-X., Xu, H., and Leng, C. Provable subspace clustering: When LRR meets SSC. In *Neural Information Processing Systems*, 2013.

You, C., Li, C.-G., Robinson, D., and Vidal, R. Oracle based active set algorithm for scalable elastic net subspace clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3928–3937, 2016.

Zhang, T., Szlam, A., and Lerman, G. Median k-flats for hybrid linear modeling with many outliers. In *Workshop on Subspace Methods*, pp. 234–241, 2009.