Disturbance Grassmann Kernels for Subspace-Based Learning

Junyuan Hong
University of Science and Technology of China
Hefei, China
jyhong@mail.ustc.edu.cn

Huanhuan Chen[*]
University of Science and Technology of China
Hefei, China
hchen@ustc.edu.cn

Feng Lin
University of Science and Technology of China
Hefei, China
lf1995@mail.ustc.edu.cn

ABSTRACT

In this paper, we focus on subspace-based learning problems, where data elements are linear subspaces instead of vectors. To handle this kind of data, Grassmann kernels were proposed to measure the space structure and used with classifiers, e.g., Support Vector Machines (SVMs). However, the existing discriminative algorithms mostly ignore the instability of subspaces, which would cause the classifiers to be misled by disturbed instances. Thus we propose considering all potential disturbances of subspaces in learning processes to obtain more robust classifiers. Firstly, we derive the dual optimization of linear classifiers with disturbances subject to a known distribution, resulting in a new kernel, Disturbance Grassmann (DG) kernel. Secondly, we research into two kinds of disturbance, relevant to the subspace matrix and singular values of bases, with which we extend the Projection kernel on Grassmann manifolds to two new kernels. Experiments on action data indicate that the proposed kernels perform better compared to state-of-the-art subspace-based methods, even in a worse environment.

CCS CONCEPTS

• **Computing methodologies → Supervised learning by classification**; **Kernel methods**;

KEYWORDS

Subspace; Grassmann manifolds; Kernel; Noise; Classification; Supervised learning

1 INTRODUCTION

Approximately representing data by linear subspaces are widely applied in computer vision, such as, dynamic texture classification [10], video-based face recognition [2], shape sequence analysis [34],

---

[*]Corresponding author

action recognition [29], etc. The popularity could be attributed to the excellence of the subspace in representing a datum consisting of numerous vectors rather than a single one. For instance, a subspace can represent an action video by approximating the principal postures in bases [29]. For comparing such high-dimensional data, subspace representation is also efficient, due to the small size of subspace matrixes. Moreover, the annoying minor perturbation in data can be mitigated by constructing subspaces [31, 33].

We are interested in subspace-based learning which is an approach to represent the data as a set of subspaces instead of vectors. In order to study the similarity/dissimilarity between subspaces, the learning problems are formulated on the Grassmann manifold, a set of fixed-dimensional subspaces embedded in a Euclidean space [14]. Unfortunately, because of its nonlinear structure, traditional learning techniques for Euclidean space are not directly feasible for the Grassmann manifold. To tackle this issue, Grassmann kernels, e.g., the Binet-Cauchy kernel [30] and the Projection kernel [11], are proposed and have achieved promising performance in discriminative learning [14]. Particularly, the Projection kernel maps subspaces to projection matrixes lying in an isometric flattened Euclidean space. Its Euclidean properties make traditional learning methods easily adapt to subspaces, e.g., discriminant analysis [14, 18], dictionary learning [17] and metric learning [20].

Though discriminative learning on Grassmann manifolds has been widely studied, the stability of subspace representation is seldom discussed in the literature. In implementations, subspaces are approximately constructed by Singular-Value Decomposition (SVD) of designed data matrixes or sets of vectors. When noise appears in data, subspaces are prone to be rotated in unexpected directions. Plus, the bases of subspace representation are practically selected by singular values, which may alter when signal amplitude fluctuates. As a result, informative bases might be changed or missed from representation, if noise signals are overly significant. For small intra-class differences between subspace instances, such disturbances will mislead classifiers to a great extent.

Inspired by implicit data noising scheme [27], we investigate to take the disturbances into account in subspace-based discriminative learning. We depict the disturbances on the manifold in Fig. 1. For each instance, a large number of disturbed duplicates are generated subject to specific distributions on the nonlinear manifold and the learning objective function is modified by averaging over these duplicates. Through dual optimization of the objective, the expectation approximation of the average yields a new kernel, i.e., Disturbance Grassmann kernel.

Specifically, two types of disturbance are discussed. First, we consider a pseudo-Gaussian distribution on the Grassmann manifold. The variance varies on bases since bases of larger singular values tend to be stabler than others with respect to noise. Second,
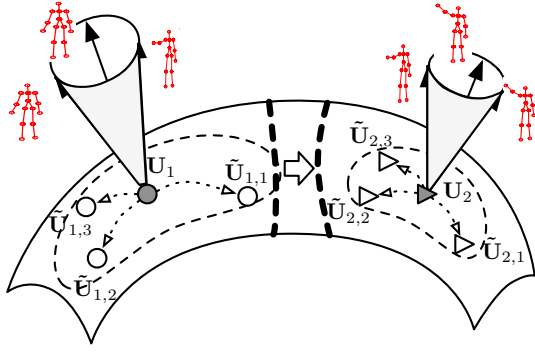
Figure 1: Illustration of learning with disturbances on the Grassmann manifold. Two gray points are subspace representations of action videos in which bases approximate the principal postures of an action. Disturbed subspaces (white points) are generated from them on the nonlinear manifold. The classification boundary (the thick dash line) is 'pushed' away from U_1 because the disturbance of U_1 is prone to be more significant.

a Dirichlet distribution is introduced to characterize the probability of missing bases. The scaled Projection kernels [14] turns out to be a special case of the Dirichlet disturbance. Furthermore, we analyze the noise modeled by subspaces, bridging the connection between subspace disturbances and data noise. It indicates that our proposed kernels properly encode the prior knowledge of real noise.

We demonstrate the disturbance Grassmann kernels with the Support Vector Machines on action data. The classifiers with proposed kernels suggest a better performance in comparison to state-of-the-art methods. Also, advanced subspace-based discriminative algorithms are compared. In challenging environments, e.g., recognition involving noise action or in low latency, the proposed kernels show their superior robustness.

2 RELATED WORK

In these decades, there has been a large body of work concerning Grassmann kernels and learning with noise. This section reviews related work in the context of the Projection kernel and implicitly noising, which are the main base of the DG kernel.

Thanks to the success of the Projection kernel in subspaces, researchers have paid attention to its extension to adapt to more complicated cases. From the probabilistic interpretation of the kernel, Hamm and Lee extend it to affine subspaces and scaled ones [14], which yields new kernels. Ways assembling multiple Projection kernels are proposed, to handle tensor data [24, 26] or data with multiple varying factors [37]. But these kernels do not exactly consider the disturbances of subspaces or the noise of data. One related work is learning a low-rank representation on Grassmann manifolds [36, 37], which is robust to minor non-informative disturbances. However, it fails to take account of the specific characteristics of subspace disturbances caused by data, which could be pretty large. Another try is perturbing data by affine transformations to help to explore the geometric structure of manifolds [25]. It is accompanied by drawbacks that the artificial perturbation relies

on specific data while the subspace representation is capable for a wider range of data, and the sampling of perturbation brings high computation burdens. A better way is to consider a subspace-level noise and learn models by implicitly noising.

Different from methods which improves the quality of data [39], noising scheme can be regarded as a kind of regularization [35] and has proven to be effective for training robust classifiers [8, 23] or extracting better features [7, 22]. Maaten et al. firstly propose the implicitly noising scheme for general learning problems [27]. The method, learning with Marginalized Corrupted Features (MCF), utilizes the expectation to approximate the average of the objective function over corrupted features. But it applies a uniform configuration of noise parameters for each instance neglecting the side information of data. Researchers, therefore, exploit instance-specific distributions to improve the method [16, 41]. The technique is also introduced in metric learning, for that noise may alter relationships of points [16, 28]. Nevertheless, their attempts focus on linear spaces. Different from them, we leverage the idea of the linear MCF to improve Grassmann classifiers with the help of implicit disturbances on subspaces. Rather than learning individual distributions from data, we determine the disturbance parameter according to the connection between subspace representation and data. It is more efficient and simpler and encloses information of subspace representation meanwhile. Inspired by the model-based kernel for efficient time series classification [5, 6], we treat a subspace as a model of data, then implement noising scheme through extending the kernel of models, i.e., the Projection kernel.

3 PRELIMINARIES

Before presenting the Disturbance Grassmann kernel, we would like to give a brief summary of basic knowledge of the Grassmann manifold.

A Grassmann manifold $\mathcal{G}(m, D)$ is a set of m-dimensional linear subspace of $\mathbb{R}^D$ for $0 \le m \le D$ and is a compact Riemannian manifold of $m(D-m)$ dimension [1]. An element of Grassmann manifolds can be expressed by a basis matrix $U = [u_1, u_2, \ldots, u_m] \in \mathbb{R}^{D \times m}$ which spans a subspace as $span(U) = \{x | x = Uv, \forall v \in \mathbb{R}^m\}$. The basis matrix satisfies that $U^T U = I_m$, where I_m is the identity matrix of size $m \times m$. The equivalence relation between two elements, U and U', are defined such that there exists a $m \times m$ orthogonal matrix Q_m such that

$$U' = UQ_m. \tag{1}$$

As a result, the basis matrix is rotation-invariant to any Q_m. For discriminative learning, some distance measurements between elements on the Grassmann manifold have been discussed [14]. Here, we specifically consider the metric resulting from projection mapping, which embeds the Grassmann manifold into the space of symmetric matrices.

The projection mapping is defined as Eq. (2) and the corresponding Projection kernel function is Eq. (3).

$$\phi : \mathcal{G}(m, D) \to Sym(D), \ \phi(U) = UU^T \tag{2}$$

$$\kappa_P(U_1, U_2) = tr[(U_1 U_1^T)(U_2 U_2^T)] \tag{3}$$

where $\text{Sym}(D)$ is a space consisting of of D-by-D symmetric matrixes and $\text{tr}(\cdot)$ denotes the trace operation. Each point on the Grassmann manifold is uniquely mapped to one projection matrix [19] and the notion of distances between them are well-preserved meanwhile [9]. The resultant projection metric is simply Euclidean in $\mathbb{R}^{D \times D}$ which approximate the true Grassmann geodesic distance up to a scale of $\sqrt{2}$ [17]. Thus it is easy for extending the Euclidean methods to the Grassmann manifold and gets popular in the research community. For more information about the geometry of the Grassmann manifold, please refer to [11].

4 DISTURBANCE GRASSMANN KERNELS

In this section, we first introduce learning problems with disturbed data, and then derive the Disturbance Grassmann kernels from the dual optimization of the learning objective function. Based on the formulation, we extend the Projection kernel to subspaces with potential disturbances.

4.1 Learning with Disturbed Subspaces

Suppose a set of observation-label pair set $\mathcal{D} = \{(\mathbf{U}_n, y_n)\}_{n=1}^N$, where $\mathbf{U}$ is an observation on the Grassmann manifolds instead of data space. Our goal is to predict class label y given a new value of $\mathbf{U}$. Following the derivation of the MCF methods [27], we augment the data set with disturbances as $\tilde{\mathcal{D}} = \{\{(\tilde{\mathbf{U}}_{nk}, y_n)\}_{k=1}^K\}_{n=1}^N$ and average their loss over disturbed samples. Each disturbed subspace, $\tilde{\mathbf{U}}_{nk}$, is drawn from a conditional distribution $P(\tilde{\mathbf{U}}|\mathbf{U}_n)$ independently. Let $\ell(\cdot)$ be the instance-specific loss function. The learning objective function can be written as a regularized empirical loss function, i.e.,

$$L(\mathbf{w}) = C \sum_{n=1}^N \frac{1}{K} \sum_{k=1}^K \ell(\mathbf{w}; (\tilde{\mathbf{U}}_{nk}, y_n)) + \frac{1}{2} \|\mathbf{w}\|^2 \qquad (4)$$

where $\mathbf{w}$ is the model parameter vector and C is regularization parameter. Consider ℓ to be logistic $(1 + \exp(-y\mathbf{w}^T\phi(\mathbf{U})))$ or hinge loss $\max(0, 1 - y\mathbf{w}^T\phi(\mathbf{U}))$. Minimizing the objective will lead to maximizing margins between classes.

Here, we do not optimize the objective function directly, but its dual problems instead. There are some reasons for doing so. First, the noising mechanism on the primal problems will be ineffective for nonlinear manifolds. On the primal problems, data points are treated as vectors in the Euclidean space. However, for nonlinear manifolds, points are not distributed uniformly in the embedded space. Feature-based corruption may generate meaningless points away from the manifold or the corruption distribution is improper for the manifold. For example, randomly altering entries of subspace matrixes would violate orthogonal constraint.

The second consideration is about the efficiency. Subspace data are usually of high dimension, whose projection matrix will be larger as a result. Computing the mapping in Eq. (2) will be of $O(mD^2)$ computational complexity. In contrast, the kernel function in Eq. (3) has as low computation cost as $O(m^2D)$ if $\mathbf{U}_1^T\mathbf{U}_2$ is calculated at first[1]. It is notable that the benefit comes up only when the number of training points is less than the data dimension.

---

[1]In many cases, the subspace dimension m is much smaller than the data dimension D.

4.2 Dual Optimization and New Kernels

By introducing a new variable, $\boldsymbol{\alpha}$, the dual optimization of Eq. (4) can be formulated as minimizing

$$L^*(\boldsymbol{\alpha}) = \frac{1}{2} \sum_{n, n'=1}^N \frac{1}{K^2} \sum_{k, k'=1}^K \alpha_{nk} \alpha_{n'k'} \kappa(\tilde{\mathbf{U}}_{nk}, \tilde{\mathbf{U}}_{n'k'})$$

$$- \sum_{n=1}^N \frac{1}{K} \sum_{k=1}^K h(\alpha_{nk}) \qquad (5)$$

$$\text{s.t. } g(\boldsymbol{\alpha}) = 0; \ 0 \leq \alpha_{nk} \leq C, \ \forall n, k$$

where $\kappa(\cdot)$ denotes a Grassmann kernel, $h(\alpha)$ the penalty term and $g(\boldsymbol{\alpha}) = 0$ the equation constraint. The penalty is $h(\alpha) = \alpha$ for SVMs, while $h(\alpha) = \alpha \log(\alpha) + (C - \alpha) \log(C - \alpha)$ for logistic regressions [40]. The equation constraint is $\sum_{n=1}^N \frac{1}{M} \sum_{k=1}^K y_n \alpha_{nk} = 0$ for SVMs and can be ignored for logistic regressions.

To avoid the number of coefficients α growing explosively when more disturbed samples are added, we let $\tilde{\mathbf{U}}_{nk}$ for any k share the same coefficient α_n. Subsequently, the number of disturbed samples can be increased to infinity, and the objective function is, therefore, approximated by the expectation form as

$$\mathcal{L}^*(\boldsymbol{\alpha}) = \frac{1}{2} \sum_{n, n'=1}^N \alpha_n \alpha_{n'} \mathbb{E}_{\tilde{\mathbf{U}}_n, \tilde{\mathbf{U}}_{n'}}[\kappa(\tilde{\mathbf{U}}_n, \tilde{\mathbf{U}}_{n'})] - \sum_{n=1}^N h(\alpha_n) \qquad (6)$$

$$\text{s.t. } g(\boldsymbol{\alpha}) = 0; \ 0 \leq \alpha_n \leq C, \ \forall n$$

where we omit the conditional dependence of expectation on $\mathbf{U}_n, \mathbf{U}_{n'}$ for brief. It is remarkable that we distinguish $\tilde{\mathbf{U}}_n$ and $\tilde{\mathbf{U}}_{n'}$ as two different variables even if n equals n' when computing the expectation. It can be proved from Eq. (5) where they are averaged separately.

We notice the objective function, Eq. (6), is barely different from the non-noised version except for the kernel. Hence, we extract the resultant kernel in Eq. (6) as a new kernel, *Disturbance Grassmann kernel*, i.e.,

$$\kappa_{\text{DG}}(\mathbf{U}_n, \mathbf{U}_{n'}) = \mathbb{E}_{\tilde{\mathbf{U}}_n, \tilde{\mathbf{U}}_{n'} | \mathbf{U}_n, \mathbf{U}_{n'}}[\kappa(\tilde{\mathbf{U}}_n, \tilde{\mathbf{U}}_{n'})]. \qquad (7)$$

The tractability of Eq. (7) depends on the form of the Grassmann kernel and the distribution. With the DG kernels, each new subspace $\mathbf{U}$ is classified using SVM or logistic regressions.

4.3 Extensions to the Projection Kernel

Because of the simple form of the Projection kernel, the DG kernel is tractable and can be computed in an efficient way. Assume disturbed subspaces subject to a distribution $P(\tilde{\mathbf{U}}|\mathbf{U}_n)$ defined on the Grassmann manifold. Substitute the Projection kernel, Eq. (3), into the DG kernel formula in Eq. (7), approaching

$$\text{tr}\left(\mathbb{E}_{\tilde{\mathbf{U}}_n | \mathbf{U}_n}[\tilde{\mathbf{U}}_n \tilde{\mathbf{U}}_n^T] \mathbb{E}_{\tilde{\mathbf{U}}_{n'} | \mathbf{U}_{n'}}[\tilde{\mathbf{U}}_{n'} \tilde{\mathbf{U}}_{n'}^T]\right).$$

Particularly, the definition also holds for $n' = n$, because $\tilde{\mathbf{U}}_{n'}$ and $\tilde{\mathbf{U}}_n$ are distinguished as two variables conditionally independent when $\mathbf{U}_n$ is determined. With the decomposition $\tilde{\mathbf{U}}\tilde{\mathbf{U}}^T = \sum_{l=1}^m \tilde{\mathbf{u}}_l \tilde{\mathbf{u}}_l^T$, the DG kernel extending the Projection kernel is given by

$$\kappa_{\text{DG}}(\mathbf{U}_n, \mathbf{U}_{n'}) = \sum_{l=1}^m \sum_{s=1}^m \text{tr}\left(\mathbb{E}[\tilde{\mathbf{u}}_l^n \tilde{\mathbf{u}}_l^{nT}] \mathbb{E}[\tilde{\mathbf{u}}_s^{n'} \tilde{\mathbf{u}}_s^{n'T}]\right). \qquad (8)$$

In the next section, we will manifest that such expectation operation will not increase computation cost when proper distributions,

$P(\tilde{\mathbf{U}}|\mathbf{U}_n)$, are introduced. For conciseness, we will refer the DG kernel as the extension of the Projection kernel, if not specified.

5 SUBSPACE DISTURBANCE

In practical implementations, a truncated subspace is used for representation rather than a full-rank one. Assuming the full-rank subspace to be $\mathbf{U}_f = [\mathbf{u}_1, \cdots, \mathbf{u}_r]$ with normalized singular values[2] λ_l for $l = 1, \cdots, r$, the truncated version lying on $\mathcal{G}(m, D)$ can be written as

$$\mathbf{U} = [\beta_1 \mathbf{u}_1, \cdots, \beta_r \mathbf{u}_r] \tag{9}$$

$$\beta_l = \mathbb{I}(\lambda_l > \lambda_m), \ \forall l \in \{1, \cdots, r\} \tag{10}$$

where $\mathbb{I}$ is the indicating function and λ_m is the threshold singular value. Specifically, we consider two kinds of disturbance:

(1) The disturbance of basis matrix $[\mathbf{u}_1, \ldots, \mathbf{u}_r]$, which should follow a proper distribution on the Grassmann manifold;
(2) The fluctuation of λ_l, which leads to dropout noise of β_l.

We will show the disturbance can imply analytical formula of the DG kernel.

5.1 Pseudo-Gaussian Disturbance

There are two ways to define parametric Probability Density Functions (PDFs) on Grassmann manifolds. One is to define models in the embedded high-dimensional Euclidean space. However, the integral over a restricted nonlinear manifold is usually intractable. For example, the Grassmann manifolds will lead to a spherical integral which has no simple solution. The other is intrinsically restricting the models on the manifold without relying on any Euclidean embedding. The general idea is to define a PDF in the tangent space of the manifold and 'wrap' it back onto the manifold. Because the tangent space is a vector space, it is possible to draw upon the pool of classical statistic methods. For such potential, we pursue a model in this way.

Actually, people can define arbitrary PDFs in the tangent space and wrap it back to the Grassmann manifold via an exponential mapping [11]. To make the PDF effective, it is necessary to restrict the domain of density functions such that the mapping is a bijection. Previously, the idea has been applied for statistic estimation on the Grassmann manifold by truncating beyond a radius of π [32].

A straightforward thought is to extend the multivariate Gaussian distribution to tangent vectors. The tangent vector of a Grassmann point $\mathbf{U}$ is

$$\mathbf{H} = \mathbf{U}_\perp \mathbf{Z} \tag{11}$$

where $\mathbf{Z} \in \mathbb{R}^{(D-m) \times m}$ and $\mathbf{U}_\perp$ is a D-by-$(D-m)$ matrix spanning the null space of $\mathbf{U}$. The exponential mapping from the tangent space to the manifold is[3]

$$\tilde{\mathbf{U}} = \mathrm{Exp}(\mathbf{H}) = \begin{pmatrix} \mathbf{U}\mathbf{V}_H & \mathbf{U}_H \end{pmatrix} \begin{pmatrix} \cos \Sigma_H \\ \sin \Sigma_H \end{pmatrix} \mathbf{V}_H^T \tag{12}$$

$$\text{s.t. } \mathbf{H} = \mathbf{U}_H \Sigma_H \mathbf{V}_H^T \tag{13}$$

where Eq. (13) is the compact SVD of the tangent vector. Supposing the PDF of a truncated Gaussian distribution is $f(\mathbf{H})$, the expectation of a projection matrix is

$$\mathbb{E}[\tilde{\mathbf{U}}\tilde{\mathbf{U}}^T] = \int \mathrm{Exp}(\mathbf{H}) f(\mathbf{H}) d\mathbf{H}$$

which is unfortunately intractable because the matrix exponential mapping, Eq. (12), is not elementary.

Besides, the mapping has a fatal drawback that it cannot preserve the density as hoped. A simple trial is wrapping the Gaussian distribution onto the sphere in $\mathbb{R}^3$. Samples around the pole opposite to the mean point will be denser than expected because of the wrapping operation. In addition, the truncated Gaussian distribution has no analytical form because the normalization factor cannot be expressed in terms of simple functions.

As far as the DG kernel is concerned, the expectation merely involves individual bases in Eq. (8). Accordingly, it is unnecessary to apply a density explicitly. Instead, we utilize an implicit density resembling a spherical Gaussian distribution on the Grassmann manifold which is uniform in all directions. We observe that subspace matrixes can be equivalently expressed in a more natural way coordinated by direction vectors and radius in the tangent space. The conclusion is summarized in Lemma 5.3 on the base of Lemmas 5.1 and 5.2.

LEMMA 5.1. *(Tangent expression) Given two subspace denoted by* $\mathbf{U}$ *and* $\tilde{\mathbf{U}}$ *on* $\mathcal{G}(m, D)$, *there exists a m-by-m orthogonal matrix* $\tilde{\mathbf{Q}}_m$ *such that*

$$\tilde{\mathbf{U}}\tilde{\mathbf{Q}}_m = \mathbf{U}\cos\Theta + \hat{\mathbf{H}}\sin\Theta \tag{14}$$

where Θ *is a m-by-m positive diagonal matrix describing tangent distance in entries, and* $\hat{\mathbf{H}}$ *a D-by-m tangent vector with orthonormal columns.*

LEMMA 5.2. *Any tangent vector on the Grassmann manifold has an equivalent expression as* $\hat{\mathbf{H}}\Theta$.

LEMMA 5.3. *Unitary tangent vector and distance can uniquely and compactly determine any subspace on Grassmann manifolds.*

PROOF. The first two lemmas can be proved from the exponential mapping in Eqs. (12) and (13) with the equivalent relationship defined in Eq. (1). The third lemma is naturally deducted from Lemmas 5.1 and 5.2. The detailed proofs of lemmas in the paper can be found in supplementary materials[4]. □

Therefore, a basis of a randomly disturbed subspace can be formulated as

$$\tilde{\mathbf{u}} = \mathbf{u}\cos\theta + \mathbf{w}\sin\theta \tag{15}$$

where $\mathbf{w} = \mathbf{U}_\perp \mathbf{x}$ such that $\|\mathbf{x}\| = 1$, and $\mathbf{u}$ is a column of a subspace $\mathbf{U}$. The θ is one entry of Θ. In correspondence, the $\mathbf{w}$ presents the direction.

With the tangent expression, we introduce a *pseudo-Gaussian distribution* on the Grassmann manifold. The direction vector is subject to a uniform direction distribution, denoted as $f(\mathbf{w})$. And the θ subject to a proper distribution whose form is not specified though. Without loss of generalization, we write it as $f(\theta; \sigma)$ which

---

[2]The normalization is practically meaningful for avoiding the influence of overall amplitude of data. For example, to compare two image sets, the difference of lightness is non-discriminative and can be eliminated by the normalization.

[3]The mapping is derived from the exponential geodesics on the Grassmann [11], thus we use the name, exponential mapping, and denote the mapping as $\mathrm{Exp}(\cdot)$.

[4]The supplementary file is available at http://jyhong.com/files/sigkdd_supp.pdf.

is governed by the variance σ. In Lemma 5.4, we show how the expectation can be expressed in a simple form with a coefficient.

LEMMA 5.4. *Assuming a disturbed subspace $\tilde{U}$ subject to the pseudo-Gaussian distribution on Grassmann manifold, then the expectation involving an individual basis will be given by*

$$\mathbb{E}[\tilde{\mathbf{u}}\tilde{\mathbf{u}}^T] = c(\sigma)\mathbf{u}\mathbf{u}^T + \frac{1 - c(\sigma)}{D - m}\mathbf{U}_\perp\mathbf{U}_\perp^T \qquad (16)$$

where σ denotes the level of variance and $c(\sigma)$ is a function of σ dominated between 0 and 1.

PROOF. Substituting the integrals

$$\int \mathbf{w}\mathbf{w}^T f(\mathbf{w})d\mathbf{w} = \int_{\mathbf{x}\in\mathbb{R}^{D-m}, \|\mathbf{x}\|=1} \mathbf{U}_\perp\mathbf{x}\mathbf{x}^T\mathbf{U}_\perp^T f(\mathbf{x})d\mathbf{x}$$
$$= \frac{1}{D - m}\mathbf{U}_\perp\mathbf{U}_\perp^T$$
$$c(\sigma) \triangleq \int \cos^2\theta f(\theta; \sigma)d\theta \qquad (17)$$

and Eq. (15) into the expectation yields the expected result. □

The exact distribution on the Grassmann manifold and the expectation, Eq. (17), are nontrivial for precise calculation. However, a well-guess is that the coefficient is a monotonically decreasing differential function of variance, because the original basis should be less influential when the disturbance is more significant. Finding an approximate function is not tough by fitting the boundary cases, the smallest and largest variance. Because the form of Eq. (17) is not determined, it is free to define the range of σ. We let 0 and 1 be the lower and upper bound, respectively.
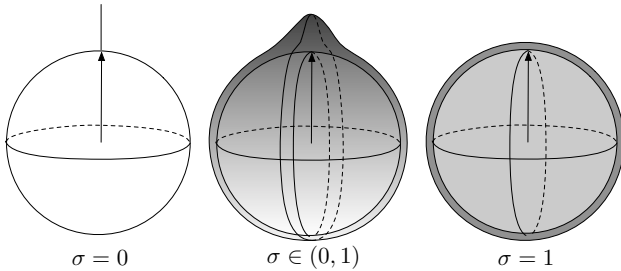

Figure 2: The pseudo-Gaussian distribution on a 2-sphere for different variance σ. The arrow represents the mean basis. The darker color represents the higher probability density.

In Fig. 2, we visualize the preferred distribution resembling a Gaussian distribution on a 2-dimensional sphere. When variance σ is zero, the coefficient $c(\sigma)$ should be 1. While σ is close to upper bound, the distribution should degenerate to an uniform distribution whose expectation should be

$$\mathbb{E}[\mathbf{u}\mathbf{u}^T] = \frac{1}{D - m + 1}\mathbf{u}\mathbf{u}^T + \frac{1}{D - m + 1}\mathbf{U}_\perp\mathbf{U}_\perp^T.$$

Hence, we can intuitively design the coefficient as

$$c(\sigma) = \frac{1}{\sigma^2(D - m) + 1}.$$

The following question is how the variance ought to be chosen. Of course, a naive solution is uniformly disturbing all bases, whereas we recommend a singular-value-related variance

$$\sigma^2 \propto \frac{\epsilon^2}{D}\frac{1}{\lambda^2} \qquad (18)$$

governed by a parameter ϵ. From the practical standpoint, bases tend to be disturbed more easily when their singular values are small. Theoretically, the variance expression rises up from the SVD perturbation theory under Gaussian noise [38] when the variance σ is close to zero. Because the approximation is a Euclidean assumption of the manifold corresponding to the tangent space and we have used precisely wrapping, the restriction for σ is no longer essential. To fit the variance in the range [0, 1], we let the square variance be a function of singular value as

$$\sigma_\lambda^2 = 1 - \exp\left\{-\frac{\epsilon}{D}\left(\frac{1}{\lambda} - 1\right)\right\} \qquad (19)$$

where $\lambda = 1$ is correspond to the case when non-normalized singular value is approaching infinity. Also the extremely small singular value λ can suggest a boundary case of variance. It is notable that in Eq. (19), we remove the square operation in ϵ and λ because it will be numerically easier for tuning ϵ without the square.

Let $\Delta = \frac{m - \text{tr}\Sigma}{D - m}$ and $\Sigma_{ii} = c(\sigma_{\lambda_i})$ and substitute the expectation in Eq. (16) into the DG kernel approaching

$$\kappa_{\text{PG}}(\mathbf{U}_n, \mathbf{U}_{n'}) = \text{tr}\left\{(\Sigma_n - \Delta_n\mathbf{I}_m)\mathbf{U}_n^T\mathbf{U}_{n'}(\Sigma_{n'} - \Delta_{n'}\mathbf{I}_m)\mathbf{U}_{n'}^T\mathbf{U}_n\right\}$$
$$+ \Delta_n\text{tr}\Sigma_{n'} + \Delta_{n'}\text{tr}\Sigma_n$$
$$+ \Delta_n\Delta_{n'}(D - 2m) \qquad (20)$$

which is named the DG Pseudo-Gaussian (DG-PG) kernel. Its computation cost is up to $O(m^2D)$.

5.2 Dirichlet Disturbance

We have assumed the singular values to be unchanged so far. However, they will fluctuate when the amplitude of signal changes. To incorporate such disturbances, we now temporarily let $\mathbf{u}_1, \ldots, \mathbf{u}_r$ fixed.

The normalized singular values imply that $\sum_{l=1}^r \lambda_r = 1$. Thus we stack the singular values in column vector $\boldsymbol{\lambda}$ and make the singular values following a Dirichlet distribution as

$$\tilde{\boldsymbol{\lambda}} \sim \text{Dir}(\boldsymbol{\lambda})$$

where the model parameter is set as the original singular values.

It is of our interest how probable a basis is retained in representation, or a singular value is larger than the threshold λ_m. For the Dirichlet disturbance, we consider the marginal distribution of an individual singular value is a beta distribution. Its cumulative distribution function is, therefore, a regularized incomplete beta function $I_x(\lambda_l, 1 - \lambda_l)$. Thus we can obtain

$$P(\beta_l = 1) = 1 - P(\lambda_l \leq \lambda_m)$$
$$= I_{1-\lambda_m}(1 - \lambda_l, \lambda_l) \triangleq p_l \qquad (21)$$

and the expectation of the projection matrix is equivalent to the projection matrix of a scaled subspace, i.e.,

$$\mathbb{E}[\tilde{\mathbf{U}}\tilde{\mathbf{U}}^T] = \sum_{l=1}^{m} \mathbb{E}_{\beta_l}[\beta_l \mathbf{u}_l \mathbf{u}_l^T] = \sum_{l=1}^{m} p_l \mathbf{u}_l \mathbf{u}_l^T. \quad (22)$$

Define a diagonal matrix Σ with p_l in entries. Then the disturbance kernel is a DG Dirichlet (DG-Dir) kernel, i.e.,

$$\kappa_{\text{Dir}}(\mathbf{U}_n, \mathbf{U}_{n'}) = \text{tr}(\Sigma_n \mathbf{U}_n^T \mathbf{U}_{n'} \Sigma_{n'} \mathbf{U}_{n'}^T \mathbf{U}_n) \quad (23)$$

whose time complexity is close to the original Projection kernel.

5.3 Summary of Disturbance Grassmann Kernels

In our derived DG kernels, the singular values play an important role in the coefficient of subspaces. They can be written uniformly as

$$\kappa_{\text{DG}}(\mathbf{U}_n, \mathbf{U}_{n'}) = \text{tr}(\mathbf{D}_n \mathbf{U}_n^T \mathbf{U}_{n'} \mathbf{D}_{n'} \mathbf{U}_{n'}^T \mathbf{U}_n) + R$$

where $\mathbf{D}$ denotes a positive diagonal matrix and R is the residual term. Interestingly, the scaled kernel proposed in [15] has the identical form with $R = 0$ and the entries of $\mathbf{D}$ are singular values. They study the probabilistic manner of constructing subspace which is similar to the SVD and deduct the probability distance leading to extended Projection kernels. Its success motivates us to think about the intrinsic statistic problems behind the Projection kernel. Different from them, we emphasize the probability of disturbances in the SVD operation, which has more tight links to the data.

6 DATA NOISE

Until now, our discussions are centered on the noise of Grassmann points, lacking necessary links to data noise. While in this section, we will analyze the origin of the discussed disturbances, to verify the firm connection between DG kernels and data noise.

6.1 Singular-Vector Perturbation

An asymptotic perturbation theory was previously proposed in [3]. It supposes that given any low-rank data matrix, as the perturbation converges to zero, a properly scaled version of its singular vectors corresponding to the nonzero singular values will converge in probability to a multivariate normal matrix. Notably, there is no exact constraint on the type of minor noise in data. As a consequence, approximated Gaussian noise in subspace matrixes can be a result of universal data noise, as long as it is fairly insignificant.

To reveal the form of approximated Gaussian noise, we start from the Gaussian perturbation in data. Suppose $\mathbf{X}$ is a $D \times N$ data matrix, and $\tilde{\mathbf{X}} = \mathbf{X} + \epsilon_X \mathbf{W}$ is its noisy version controlled by a small constant, where the entries of $\mathbf{W}$ are independently and identically distributed. For a given $m < D$, $\mathbf{X}$ has the singular-value decomposition $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T$, where Σ is a $m \times m$ singular-value diagonal matrix, and all singular values denoted by $\lambda_i, i = 1, \ldots, m$ are in descending order. The noise leads to a non-asymptotic result with a closed-form of approximation error. For the concern of this paper, we restate the results in Wang's paper [38] as Lemma 6.1.

LEMMA 6.1. *Given noised data matrix, i.e.* $\tilde{\mathbf{X}} = \mathbf{X} + \epsilon_X \mathbf{W}$ *and* $W_{ij} \sim \mathcal{N}(0, 1/D)$, *the resultant subspace involves a Gaussian noise*

on the null space of $\mathbf{U}$ *denoted by* $\mathbf{U}_\perp$. *When* ϵ_X *approaches zero, the disturbed subspace can be written as*

$$\tilde{\mathbf{U}} = \mathbf{U} + \epsilon_X \mathbf{U}_\perp \mathbf{W}_0 \Sigma^{-1}. \quad (24)$$

where $\mathbf{W}_0$ *is subject to the same Gaussian distribution.*

PROOF. The lemma can be proved according to the theorem of singular-vector perturbation under Gaussian noise [38]. For brevity, we enclose the detailed proof in supplementary materials. □

Roughly speaking, to ensure the approximation error has the lowest degree, we let the ϵ lower than any of $D^{-\beta-1/2}$, $D^{-3/4}$, and $\|\mathbf{U}\|_{\max}^{-1} D^{-1}$, where $0 < \beta < 1/2$ controls the probability of variance of Gaussian approximation increasingly. In our implementation, we approximately let $\epsilon \leq D^{-1/2}$. Therefore, we get the relation between variance and the singular values in Eq. (18).

The above theories strengthen the connection between the pseudo-Gaussian disturbance and the data noise, especially the Gaussian noise. Numerically, the analysis gives tips about the tendency of the variance taking account of the singular values.

6.2 Variant Subspace Representations

Given action videos, subspace representations vary mainly due to action variance. We define these subspace representations as points $\mathbf{U}_1, \ldots, \mathbf{U}_M$ on corresponding Grassmann manifolds $\mathcal{G}(m_1, D), \ldots, \mathcal{G}(m_M, D)$ where m_1 to m_M are natural numbers smaller than the maximum order m. These different subspaces reveal different variance of action videos, including adding and missing principal postures. Naive selection of one subspace for representation will lead to a poor generalization of classifiers.

To avoid missing useful features or incorporating unwanted features, we propose learning classifiers with all possible representations on variant Grassmann manifolds. Especially, these representations are obtained from training data, which do not require extra data. We first learn a subspace of full rank as $\mathbf{U}$ on $\mathcal{G}(m, D)$. For the learned subspace, a subset of bases is utilized. Specifically we define $\boldsymbol{\beta} \in \{0, 1\}^m$ where entry 0 denotes the deletion of a basis. Thus representation data could be augmented on variant Grassmann manifolds as $\{\phi(\mathbf{U}_n; \mathcal{G}_1)\}_{n=1}^N \rightarrow \{\{\phi(\mathbf{U}_n; \mathcal{G}_{\boldsymbol{\beta}_i})\}_{k=1}^K\}_{n=1}^N$ where K additive representations are assigned for each training instance.

Thus training with augmented subspaces is corresponding to dropout of bases with $\boldsymbol{\beta}$. We note that from Eq. (22), the distribution can also be viewed as a special case of Bernoulli distribution. It, therefore, corresponds to a dropout noise on bases where 0 entry denotes the deletion of basis. From this perspective, the Dirichlet disturbance is a reflection of learning with variant subspace representation.

7 EXPERIMENTS

On real data in Table 1, we evaluate the proposed kernels, DG-PG kernels and DG-Dir kernels with the renowned Support Vector Machines and compare them to different kernels. The parameter ranges for different models are listed in Table 2. Cross-validation over the range of parameters is applied for parameter selection.

The first two parameters, C and r are the parameters for all subspace-based SVMs. Typically, C is the soft margin parameter of SVMs. And, we tune the assumed uniform rank r for all subspaces,

Table 1: Summary of data sets.

Data set	Classes	Total number of actions	Experimental protocol
KARD [13]	Ten gestures (horizontal arm wave, high arm wave, two-hand wave, high throw, draw X, draw tick, forward kick, side kick, bend, and hand clap) and eight actions (catch cap, toss paper, take umbrella, walk, phone call, drink, sit down, and stand up)	10 subjects × 18 actions × 3 try = 540 actions	Experiment A: 1/3 training/ 2/3 testing per subject; Experiment B: 2/3 training/ 1/3 testing per subject; Experiment C: 1/2 training/ 1/2 testing.
UCFKinect [12]	balance, climb up, climb ladder, duck, hop, vault, leap, run, kick, punch, twist left, twist right, step forward, step back, step left, step right	16 subjects × 16 actions × 5 try = 1280 actions	4-fold cross-validation

Table 2: Parameter ranges for models.

Parameter	Range
C	$\{10^{-4}, 10^{-3}, \ldots, 10^5\}$
r	$\{1, 2, \cdots, 15\}$
ϵ	$\{10^{-6}, 10^{-2}, 0.05, 0.1, 0.2, \cdots, 1, 1.2, 1.7, 2, 5, 40\}$
λ_m	$\{0.001, 0.01, 0.1, 0.3, 0.5, 0.7, 0.9\}$

since we have no idea what is the real rank of the data. As a result, there will be two steps of truncation in Eq. (9), controlled by λ_m and r. For kernels other than the DG kernels, λ_m (or m) and r is same.

Selecting the parameter λ_m of DG-Dir kernel is quite straight. We scan from 0 to 1 for the best parameter. In real experiments, we find that a sparse parameter set as showed in Table 2 usually has a better performance with cross-validation. It might be ascribed to that the size of data sets is not large enough.

As for DG-PG kernel, the parameter ϵ governing the noise is not simple. Because the variance is a designed function of singular values, it has a non-trivial relation to the variance of the tangent noise. For this reason, we empirically choose some parameters to make the curves of the function of singular value as distinct as possible.

As a comparison, we run SVMs with other Grassmann kernels, including the original Projection (Proj) kernel, the extended (scaled/affine/affine scale) Projection (ScProj/AffProj/AffScProj) kernels, [14], and the Binet-Cauchy (BC) kernel [30]. Moreover, advanced discriminative algorithms on the Grassmann manifold, e.g., Projection Metric Learning (PML) [20], Grassmann Discriminant Analysis (GDA) [14], Graph embedding Grassmann Discriminant Analysis (GGDA) [18] are included. These methods are typical dimensionality reduction methods on Grassmann manifolds. Technically, PML and GGDA are exercised using public Matlab codes from the source papers. The refined features are classified using SVMs.

7.1 Activity Recognition

We perform activity recognition experiments with different difficulties on Kinect Activity Recognition (KARD) data set, to assess classifiers fully.

Data set Here it is of our interest to use the provided 15 joints of the skeleton in world coordinates. The skeleton features contain 45 continuous real values.

Table 3: The KARD data set is divided into three subsets presenting increasing difficulty for recognition.

Set 1	Set 2	Set 3
Horizontal arm wave	High arm wave	Draw tick
Two-hand wave	Side kick	Drink
Bend	Catch cap	Sit down
Phone call	Draw tick	Phone call
Stand up	Hand clap	Take umbrella
Forward kick	Forward kick	Toss paper
Draw X	Bend	High throw
Walk	Sit down	Horizontal arm wave

Setup In [13], Gaglio et al. proposed some evaluation experiments on the data set. We follow their configuration applying three different experiments (A/B/C). And according to the visual similarity between actions or gestures, these activities are separated into three sets of difficulties in ascending order. As shown in Table 3, activity set 1 is the simplest one since it is composed of quite different activities while the other two sets include more similar actions and gestures. And generalization ability of predictors is assessed according to classification error rates in test sets.

Result We report the classification error rates in Table 4. Each error rate is an averaged over ten random partitions of data set following different experimental settings.

7.2 Appended Noise Activities

Considering the heterogeneousness of action data, frames from unrelated source would lead to the activity videos represented by totally different subspaces. We specify the case in our experiment by artificially incorporating noise action or gestures in activity data, named Appended Noise Activities (ANA) experiment.

Setup For every experiment set defined in Table 3, we choose a noise class excluded from the set. Then an activity sequence randomly picked from the noise class is appended to a sample sequence in the experiment set. Such operation will significantly alter the singular vectors and values. Because the noise action sequence is from the same class which is not distinguishable, the discriminative ability of each basis will be influenced meanwhile. For illustration, we depict the similarity matrixes before and after adding noise activities in Fig. 3. The three bases are discriminative as showed initially (the first row), while the first and third bases are not discriminative for class one and two after appending noise.

Table 4: Classification error rates on KARD data set.

EXP	1A	1B	1C	2A	2B	2C	3A	3B	3C
DG-PG	0.75	0.50	**0.67**	1.13	0.87	1.33	2.69	1.00	1.17
DG-Dir	**0.69**	**0.12**	0.83	**1.06**	**0.25**	**0.25**	**2.31**	**0.37**	**1.00**
Proj	2.25	0.37	0.92	1.12	1.00	1.92	5.75	2.38	4.17
ScProj	2.50	0.75	1.67	1.31	0.50	1.08	4.62	1.00	2.75
AffProj	7.69	2.00	4.67	3.44	1.00	3.25	7.25	3.37	6.00
AffScProj	9.50	4.62	6.50	5.50	3.75	5.50	9.75	5.12	8.50
BC	6.25	3.50	4.83	5.50	2.88	3.92	9.88	4.00	6.33
PML	4.56	2.12	2.67	1.44	0.62	1.83	4.69	3.37	3.92
GDA	9.44	4.13	7.00	3.81	1.75	2.42	15.25	7.25	11.42
GGDA	6.44	4.87	5.67	1.75	0.50	2.17	12.00	6.50	10.75

Table 5: Classification error rates on KARD data set with appended noise activities.

| EXP	1A	1B	1C	2A	2B	2C	3A	3B	3C
DG-PG	**11.25**	8.25	**9.08**	**4.62**	**1.62**	**2.50**	**7.06**	**3.87**	**4.92**
DG-Dir	12.56	**7.38**	9.92	5.31	1.88	3.08	10.12	4.38	5.83
Proj	12.06	9.75	9.58	7.44	3.00	4.00	12.13	6.62	9.33
ScProj	12.94	7.50	12.33	6.94	2.37	5.67	17.25	8.00	12.83
AffProj	16.31	9.25	10.33	7.50	3.12	6.33	14.56	7.50	9.92
AffScProj	18.81	12.88	14.25	21.00	8.62	15.33	32.63	19.50	26.25
BC	19.19	13.88	18.17	15.31	10.62	15.25	23.38	16.12	20.42
PML	13.62	10.25	11.00	6.19	3.38	4.25	14.56	8.00	13.00
GDA	13.94	10.38	10.50	8.06	2.50	4.42	21.12	11.12	15.58
GGDA	12.25	9.12	10.92	5.94	2.25	4.08	17.69	10.00	13.67
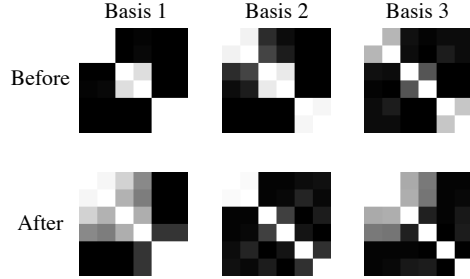
Figure 3: Similarity matrixes between actions measured by the Projection kernel of different bases before/after appending noise activities. Three class and two actions per class are tested. For each similarity matrix, instances are ordered by class and the darker color means the pair is less similar. .

Result Table 5 shows the classification error rates when noise activities are involved. Still, the error rate is averaged over ten training/test partitions.

7.3 Action Recognition in Low Latency

In this experiment, we consider the challenge of action recognition in low latency, using the UCFKinect data set. It is motivated by a daily requirement for action recognition, in which a rough estimation is required to be given before video recording is accomplished.

Except the technical necessity, a quick response of recognition system is in favor of better user experience.

Data set The skeletal joint locations over sequences of this data set are estimated using the Microsoft Kinect sensor and the PrimeSense NiTE. The coordinates of 15 joints are used for classification. During recording, subjects are advised to start from and end at an identical resting posture for all actions. The recording was manually stopped upon completion of the action.

Setup The latency of an action is defined as the duration between the time a user begins the action and the time the classifier classifies the action. To evaluate classifiers on different latency, the experiment in [12] is introduced here. The maximum video length K is set to force all sequence truncated at the K-th frames. Full videos are kept if shorter than K.

Result We summarize the average classification error rate versus maximum video length in Table 6. Some selected old results on the data sets are also reported in the table, where the Local Tangent Bundle (LTB) [29] kernel had the best performance on subspaces before and the Latency Aware Learning (LAL) [12] is the best non-Grassmann methods. The lengths of frames to show the results are selected according to previous works [12]. When the length is longer than 30, the difference of error rates will be less distinct. Therefore the step of length is smaller after 30.

7.4 Discussion

In the three data sets, our proposed kernels have shown superior performance compared to previous methods. Specifically, DG-Dir

Table 6: Classification error rates on varying maximum video length.

Frames	10	15	20	25	30	40	60
DG-PG	71.56	**42.50**	17.81	**6.87**	**2.81**	1.25	**0.62**
DG-Dir	**70.31**	43.44	**16.56**	**6.87**	**2.81**	0.62	**0.62**
Proj	75.94	46.88	20.94	9.38	3.12	1.56	1.88
ScProj	73.12	45.63	17.19	7.81	3.12	1.56	**0.62**
AffProj	73.75	51.88	25.94	13.75	5.63	2.19	0.94
AffScProj	72.19	50.94	24.38	15.31	6.25	3.75	1.94
BC	75.94	50.62	21.56	11.25	6.87	3.75	2.81
PML	73.44	48.12	19.06	10.31	3.44	1.88	1.25
GDA	80.94	56.88	22.50	13.44	6.87	2.19	2.81
GGDA	78.75	70.00	31.56	15.00	4.37	1.88	1.25
LTB	78.13	50.63	25.63	13.13	7.92	2.71	2.09
LAL	86.09	63.05	35.23	18.44	9.45	4.84	4.06

kernels perform best in KARD data set and in low latency, while DG-PG kernels do better in the ANA experiment.

For different difficulties of the KARD sets, the error rates of DG-Dir kernels is stably lower than 3% and better than those of the ScProj kernel, which use the fixed singular values as subspace coefficients. We attribute the superiority to the more flexible relation between the coefficients and the singular values.

The low-latency challenge can be treated as a basis-missing problem for subspace-based learning. As analyzed in Section 6.2, the case corresponds to a dropout noise of bases, which has been considered in the DG-Dir kernel. The similarity between DG-Dir kernels and ScProj kernels is verified again in the experimental results. The low-latency also has an impact on the singular vectors, which explains the good performance of DG-PG kernels.

The impact is more obvious in the ANA experiment. Consequently, we observe a superiority of DG-PG kernels in the case. The gaps between the DG-PG kernels and baseline methods are significant, especially in the set 2 and 3.

8 CONCLUSION

In this work, we novelly present the Disturbance Grassmann kernel concerning the potential disturbances for subspace-based learning. By investigating the disturbances in subspaces, we extend the Projection kernel to two new kernels corresponding to the Gaussian-like noise in subspace matrixes and the Dirichlet noise in singular values. We also prove the subspace disturbances are linked to data noise in specific cases. With action data, we demonstrate that the resultant kernels can outperform the original Projection kernel as well as previously used extended Projection kernels, Binet-Cauchy kernels for subspace-based learning problems. Results with noisy environment also highlight the superiorities of the DG kernel. The DG kernel is not confined to the Projection kernel and action data. Extensions to other kernels has not been explored yet, and we look forward to a follow-up study in the future. Besides, treating the subspace representation in model space, we expect to further exploit methods with metric co-learning [4] and apply them to other sequential data [21].

REFERENCES

[1] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. 2009. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press.

[2] Gaurav Aggarwal, AK Roy Chowdhury, and Rama Chellappa. 2004. A system identification approach for video-based face recognition. In *Proceedings of the 17th International Conference on Pattern Recognition*, Vol. 4. IEEE, 175–178.

[3] E Bura and R Pfeiffer. 2008. On the distribution of the left singular vectors of a random matrix and its applications. *Statistics & Probability Letters* 78, 15 (2008), 2275–2280.

[4] Huanhuan Chen, Fengzhen Tang, Peter Tino, Anthony G Cohn, and Xin Yao. 2015. Model Metric Co-Learning for Time Series Classification.. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence*. AAAI Press, 3387–3394.

[5] Huanhuan Chen, Fengzhen Tang, Peter Tino, and Xin Yao. 2013. Model-based Kernel for Efficient Time Series Analysis. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '13)*. ACM, New York, NY, USA, 392–400. https://doi.org/10.1145/2487575.2487700

[6] Huanhuan Chen, Peter Tino, Ali Rodan, and Xin Yao. 2014. Learning in the model space for cognitive fault diagnosis. *IEEE Transactions on Neural Networks and Learning Systems* 25, 1 (2014), 124–136.

[7] Minmin Chen, Kilian Q. Weinberger, Zhixiang (Eddie) Xu, and Fei Sha. 2015. Marginalizing Stacked Linear Denoising Autoencoders. *Journal of Machine Learning Research* 16 (2015), 3849–3875.

[8] Ning Chen, Jun Zhu, Jianfei Chen, and Bo Zhang. 2014. Dropout Training for Support Vector Machines. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*. 1752–1759.

[9] Yasuko Chikuse. 2012. *Statistics on Special Manifolds*. Vol. 174. Springer Science & Business Media.

[10] Gianfranco Doretto, Alessandro Chiuso, Ying Nian Wu, and Stefano Soatto. 2003. Dynamic textures. *International Journal of Computer Vision* 51, 2 (2003), 91–109.

[11] Alan Edelman, Tomás A Arias, and Steven T Smith. 1998. The geometry of algorithms with orthogonality constraints. *SIAM journal on Matrix Analysis and Applications* 20, 2 (1998), 303–353.

[12] Chris Ellis, Syed Zain Masood, Marshall F Tappen, Joseph J LaViola, and Rahul Sukthankar. 2013. Exploring the trade-off between accuracy and observational latency in action recognition. *International Journal of Computer Vision* 101, 3 (2013), 420–436.

[13] Salvatore Gaglio, Giuseppe Lo Re, and Marco Morana. 2015. Human activity recognition process using 3-D posture data. *IEEE Transactions on Human-Machine Systems* 45, 5 (2015), 586–597.

[14] Jihun Hamm and Daniel D Lee. 2008. Grassmann discriminant analysis: a unifying view on subspace-based learning. In *Proceedings of the 25th International Conference on Machine Learning*. ACM, 376–383.

[15] Jihun Hamm and Daniel D Lee. 2009. Extended Grassmann kernels for subspace-based learning. In *Advances in Neural Information Processing Systems*. 601–608.

[16] Xue-Min Si Yuan Jiang Han-Jia Ye, De-Chuan Zhan. 2017. Learning Mahalanobis distance metric: considering instance disturbance helps. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*. 3315–3321.

[17] Mehrtash Harandi, Conrad Sanderson, Chunhua Shen, and Brian C Lovell. 2013. Dictionary learning and sparse coding on Grassmann manifolds: an extrinsic solution. In *IEEE International Conference on Computer Vision*. 3120–3127.

[18] Mehrtash T Harandi, Conrad Sanderson, Sareh Shirazi, and Brian C Lovell. 2011. Graph embedding discriminant analysis on Grassmannian manifolds for improved image set matching. In *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2705–2712.

[19] Uwe Helmke, Knut Hüper, and Jochen Trumpf. 2007. Newton's method on Grassmann manifolds. *arXiv preprint arXiv:0709.2205* (2007).

[20] Zhiwu Huang, Ruiping Wang, Shiguang Shan, and Xilin Chen. 2015. Projection metric learning on Grassmann manifold with application to video based face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*. 140–149.

[21] Yang Li, Junyuan Hong, and Huanhuan Chen. 2016. Sequential Data Classification in the Space of Liquid State Machines. In *European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD 2016)*. Springer-Verlag, Berlin, Heidelberg, 313–328. https://doi.org/10.1007/978-3-319-46128-1_20

[22] Yingming Li, Ming Yang, Zenglin Xu, and Zhongfei Zhang. 2016. Learning with marginalized corrupted features and labels together.

[23] Jinliang Liu, Jie Cao, Zhiang Wu, and Qiong Qi. 2014. State estimation for complex systems with randomly occurring nonlinearities and randomly missing measurements. *International Journal of Systems Science* 45, 7 (2014), 1364–1374.

[24] Yui Man Lui. 2012. Tangent bundles on special manifolds for action recognition. *IEEE Transactions on Circuits and Systems for Video Technology* 22, 6 (2012), 930–942.

[25] Yui Man Lui and J. Ross Beveridge. 2008. Grassmann registration manifolds for face recognition. In *European Conference on Computer Vision*. Springer Berlin Heidelberg, Berlin, Heidelberg, 44–57.

[26] Yui Man Lui, J Ross Beveridge, and Michael Kirby. 2010. Action classification on product manifolds. In *IEEE Conference on Computer Vision and Pattern Recognition*. 833–839.

[27] Laurens Maaten, Minmin Chen, Stephen Tyree, and Kilian Q Weinberger. 2013. Learning with marginalized corrupted features. In *Proceedings of the 30th International Conference on Machine Learning*. 410–418.

[28] Qi Qian, Juhua Hu, Rong Jin, Jian Pei, and Shenghuo Zhu. 2014. Distance metric learning using dropout: a structured regularization approach. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '14)*. ACM, New York, NY, USA, 323–332. https://doi.org/10.1145/2623330.2623678

[29] Rim Slama, Hazem Wannous, Mohamed Daoudi, and Anuj Srivastava. 2015. Accurate 3D action recognition using learning on the Grassmann manifold. *Pattern Recognition* 48, 2 (2015), 556–567.

[30] Alex J Smola and Svn Vishwanathan. 2005. Binet-Cauchy kernels. In *Advances in Neural Information Processing Systems*. 1441–1448.

[31] Gilbert W. Stewart. 1990. *Perturbation Theory for the Singular Value Decomposition*. Technical Report. College Park, MD, USA.

[32] Pavan Turaga, Ashok Veeraraghavan, Anuj Srivastava, and Rama Chellappa. 2011. Statistical computations on Grassmann and Stiefel manifolds for image and video-based recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33, 11 (2011), 2273–2286.

[33] Matthew Turk and Alex Pentland. 1991. Eigenfaces for recognition. *Journal of Cognitive Neuroscience* 3, 1 (1991), 71–86.

[34] Ashok Veeraraghavan, Amit K Roy-Chowdhury, and Rama Chellappa. 2005. Matching shape sequences in video with applications in human movement analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27, 12 (2005), 1896–1909.

[35] Stefan Wager, Sida Wang, and Percy Liang. 2013. Dropout training as adaptive regularization. In *Proceedings of the 26th International Conference on Neural Information Processing Systems (NIPS'13)*. Curran Associates Inc., USA, 351–359.

[36] Boyue Wang, Yongli Hu, Junbin Gao, Yanfeng Sun, and Baocai Yin. 2014. Low rank representation on Grassmann manifolds. In *Asian Conference on Computer Vision*. Springer, 81–96.

[37] Boyue Wang, Yongli Hu, Junbin Gao, Yanfeng Sun, and Baocai Yin. 2016. Product Grassmann manifold representation and its LRR models.. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. 2122–2129.

[38] Rongrong Wang. 2015. Singular vector perturbation under Gaussian noise. *SIAM J. Matrix Anal. Appl.* 36, 1 (2015), 158–177.

[39] Zhiang Wu, Jie Cao, Haicheng Tao, and Yi Zhuang. 2013. A novel noise filter based on interesting pattern mining for bag-of-features images. *Expert Systems with Applications* 40, 18 (2013), 7555–7561.

[40] Hsiang-Fu Yu, Fang-Lan Huang, and Chih-Jen Lin. 2011. Dual coordinate descent methods for logistic regression and maximum entropy models. *Machine Learning* 85, 1 (1 Oct 2011), 41–75.

[41] Jingwei Zhuo, Jun Zhu, and Bo Zhang. 2015. Adaptive dropout rates for learning with corrupted features. In *Proceedings of the 24th International Conference on Artificial Intelligence*. AAAI Press, 4126–4132.