ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-426

Randomized Distribution Feature for Image Classification

Hongming Shan and Junping Zhang*, [1]

Abstract.

Local image features can be assumed to be drawn from an unknown distribution. For image classification, such features are compared through the histogram-based model or the metric-based model. By quantizing these local features into a set of histograms, the histogram-based model is convenient and has vectorial representation of image but information could be lost in vector quantization. Unlike the histogram-based model, the metric-based model estimates the metrics over the underlying distribution of local features immediately, achieving better predictive performance. However, the model requires higher computational cost and loses the benefit of vectorial representation of image.

To retain the advantages of these two models, this paper proposes the (doubly) randomized distribution features that represent the underlying distribution of local features in each image as a vectorial feature by utilizing random Fourier feature. We prove the convergences of the similarity and distance based on the randomized distribution feature. Remarkable advantages of the randomized distribution feature are that it has vectorial representation and thus computes efficiently as the histogram-based model. Besides, it provides rigorous theory guarantee and competitive performance as the metric-based model. Compared with several state-of-the-art algorithms, experiments in three real-world datasets justify that our proposed approaches attain competitive classification accuracy with faster computational speed. Furthermore, we indicate that our proposed features can utilize the methods in learning based on vectors, which are broadly studied in traditional machine learning domain, to deal with the problems in learning based on distribution.

1 Introduction

Image representation plays a crucial role in computer vision domains. Generally, images could be represented by a set of high-dimensional, unordered and finite local features. For example, the shapes of object are characterized by a set of local descriptors at edges and corner points [11], and facial expressions are represented by a set of local image patches containing action units [8]. To some extent, these features in each image can be assumed to be drawn from an unknown distribution [25, 34], leading to a learning task based on distribution such as distribution regression with scalar response [32] and distribution to distribution regression [30].

Under this assumption, existing approaches to image classification are roughly categorized into two types: the histogram-based model and the metric-based one. The histogram-based model usually represents each image by the empirical, one-dimensional histogram that enumerates the occurrence probability of each point set in the bag of visual words. Here, the collection of these words is called a codebook or dictionary. The disadvantages of this method are that the size of codebook is difficult to select, and the computational cost of generating the codebook by the quantization algorithms is expensive. Besides, the information will be lost in the quantization process [34]. In contrast, the metric-based model estimates statistical metrics over the underlying distribution of images with higher accuracy. The advantage of this model is that it does not require quantization techniques and selecting the size of codebook, each of which could result in the loss of performance in image classification. However, these metrics suffer from high computational cost since they operate over pairwise samples. Another drawback of the model is that the matrices obtained by these metrics are only suitable for some specific learning algorithms, *e.g.*, kernel-based algorithms, but cannot be amenable for off-the-shelf use with any standard learning algorithm [22].

In this paper, we propose the (doubly) **R**andomized **D**istribution **F**eature (**RDF**) that could characterize the underlying distribution of local image features of each image as a vector. In this way, the proposed approaches achieve a vectorial representation of distribution, and thus inherit the property of high efficiency of the histogram-based model. Meanwhile, it can approximate the metrics defined on distribution as the metric-based model. Specifically, the distribution of local features is characterized as the mean of random Fourier features which are a low-dimensional embedding representation of kernel mapping function. As a result, the proposed approaches retain advantages from both the histogram-based model and the metric-based model. We also prove the convergences of the similarity and distance based on the randomized distribution feature in this paper. The experimental results show that the proposed methods could achieve competitive performance and reduce computational cost significantly.

The contributions of this paper are summarized here:

- We propose the (doubly) randomized distribution features that represent the distribution of local features extracted from images as a vector;
- We analyze the convergences of the similarity and distance based on the randomized distribution feature;
- Experimental results show that the (doubly) randomized distribution features work better than BoW in vectorial representation, and have competitive performance as the metrics defined over distributions directly. Most importantly, it is easy to implement and computes much faster;
- The proposed method could make learning problems on distribution where each input is a distribution become our traditional machine learning problems, where each input is a vector.

[1] Shanghai Key Laboratory of Intelligent Information Processing, School of Computer Science, Fudan University, Shanghai 200433, China.
Emails: {hmshan, jpzhang}@fudan.edu.cn.
* Corresponding author: Junping Zhang.

The paper is organized as follows. Section 2 briefly surveys the associated algorithms that learn from the distribution. Section 3 presents the preliminaries of the metric-based model, especially the similarity and distance between distributions. Section 4 introduces the proposed (doubly) randomized distribution features, and theoretically analyzes the convergences of the proposed approaches. Experiments in Section 5 demonstrate a comprehensible comparison between the performances of the proposed approaches and several recently published methods. Finally, Section 6 presents a conclusive summary.

2 Related works

Associated algorithms that deal with distributions could be roughly divided into two categories: the histogram-based model and the metric-based model. The most popular method in the histogram-based model is the bag of word (BoW) [9]. By quantizing each local feature into one of visual words by using *K*-means, BoW represents an image as one-dimensional histogram that enumerates the occurrence probability of each local feature of images in the bag of visual words. BoW suffers from high computational cost of generating codebook by *K*-means and the quantization process that the information could be lost. Therefore, some recent researches are devoted to accelerating quantization process, such as hierarchical *K*-means [28], KD-tree and random projection tree [7] and so on. To alleviate the loss of information in the quantization process, several researches attempt to learn more discriminant information from images by aggregating local descriptors [1, 16], learning a discriminant codebook [28], and keeping fisher information [31], etc.

Alternatively, the metric-based model defines various metrics such as similarity, distance and divergence between the distributions for avoiding information loss of the histogram-based model. Specifically, mean map kernel (set kernel) [12, 40] measures similarities among pairwise points. Distance metrics between two distributions such as maximum mean discrepancy (MMD) [13, 25] and nonparametric divergence [33, 34, 45] are commonly-used in machine learning and computer vision domains. Unlike the histogram-based model where features must be quantized and vectorized, the metric-based model can achieve better predictive performance since the comparison is done over the underlying distribution of local features. However, the metric-based model requires preserving the whole data sets and calculating metric between training sets and a new unseen set, which makes it infeasible even for a moderate-size problem. Moreover, the metric-based model is only suitable for some special learning algorithms that could use similarity/distance matrix between samples. Though condensing local features of each image could improve the speed and accuracy [45] , the aforementioned problem has not been addressed in essence.

To address these issues, we propose an alternative way that represents the underlying distribution of images by random distribution feature, more concretely, by averaging random Fourier feature [36, 37]. Currently, two related works in literature employ random Fourier feature to characterize the probability distribution in cause-effect inference [22], and to construct match kernel heuristically [2]. Compared to these previous studies, the major difference in this paper is that our work provides a theoretical analysis on the convergences of the similarity and distance between images when using random distribution feature. We also propose a doubly randomized distribution feature to represent images for further promoting performance.

3 Preliminary

In this section, we will introduce kernel embedding of the distribution and two metrics defined on distribution in details.

Following [22], the notations used in this paper are summarized in Table 1. Assume that two images are represented by unknown distributions P and Q separately, their local feature sets are $S = \{x_i\}_{i=1}^n \sim P$ and $T = \{z_j\}_{j=1}^m \sim Q$, respectively. Note that the local feature x and z reside in a d-dimensional space and $S \cup T \in \mathcal{X}$. In fact, S and T could construct their empirical distributions P_S and Q_T respectively, each of which is a set of local descriptors. For example, these features can be extracted from the local regions of images by histogram of gradient (HOG) [6] or scale-invariant feature transform (SIFT) [23].

Table 1. Notations used in this paper

$\mathbb{E}[\xi]$	Expected value of random variable ξ
P	True distribution
$S = \{x_i\}_{i=1}^n$	Point set randomly drawn from P
P_S	Empirical distribution of S
$\mathcal{X}$	Domain of random variable sampled from P and Q
κ	Kernel function from $\mathcal{X} \times \mathcal{X}$ to $\mathbb{R}$
$\mathcal{H}_\kappa$	RKHS induced by κ
$\mu_\kappa(P)$	Kernel embedding of the distribution P
$\mu_\kappa(P_S)$	Empirical kernel embedding of P_S
κ^{F}	Low-D representation of κ
$\mu_\kappa^{\mathrm{F}}(P)$	Low-D representation of $\mu_\kappa(P)$
$\mu_\kappa^{\mathrm{F}}(P_S)$	Low-D representation of $\mu_\kappa(P_S)$

3.1 Kernel embedding of the distribution

Let P denote the probability distribution of some random variable X taking value in a separable topological space $(\mathcal{X}, \tau_\mathcal{X})$. Then *kernel embedding of distribution* P associated with a continued, bounded, and positive-definite kernel function $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is described as follows:

$$\mu_\kappa(P) := \int_\mathcal{X} \kappa(x, \cdot)\, \mathrm{d}\, P(x), \tag{1}$$

where $\mu_\kappa(P)$ is an element in the reproducing kernel Hilbert space (RKHS) $\mathcal{H}_\kappa$ associated with kernel function κ [39].

Interestingly, a kernel function κ is said to be *characteristic* if the mapping μ_κ is injective [43], *i.e.*, $\|\mu_\kappa(P) - \mu_\kappa(Q)\|_{\mathcal{H}_\kappa} = 0$ iff $P = Q$. In other words, kernel embedding of the distribution does not lose any information about the distribution when equipped with a characteristic kernel. An example of this kernel is the Gaussian kernel. It will be used throughout this paper and is defined as follows:

$$\kappa(x, x') = \exp\Big(-\gamma \left\| x - x' \right\|_2^2 \Big), \gamma > 0. \tag{2}$$

Since it is unrealistic to get both the true distribution P and true embedding $\mu_\kappa(P)$ in practice, we utilize a sample set $S = \{x_i\}_{i=1}^n \sim P$ to construct the empirical distribution P_S instead. As a result, we approximate the empirical kernel embedding $\mu_\kappa(P_S)$ through P_S:

$$\mu_\kappa(P_S) := \frac{1}{n} \sum_{i=1}^n \kappa(x_i, \cdot) \in \mathcal{H}_\kappa. \tag{3}$$

As summarized in [26], the estimator in eq. (3) has several nice properties: 1) kernel embedding of distribution could preserve all the

information about distribution with characteristic kernel; 2) basic operation on distribution can be done by means of inner products in RKHSs; 3) no intermediate density estimation is required. Therefore, many algorithms benefit from eq. (3) such as maximum mean discrepancy [13], kernel dependency measure [14], Hilbert space embedding of HMMs [41] and kernel Bayes' rule [10]. Despite that the estimator in eq. (3) can be improved by utilizing Stein's phenomenon [26], this estimator is commonly used in practice. Furthermore, the convergence of empirical kernel embedding $\mu_\kappa(P_S)$ to the embedding of its population $\mu_\kappa(P)$ in RKHS norm has been proven in [22].

3.2 Mean map kernel

Note that kernel embedding of distribution does not result in any loss of information by using characteristic kernel. The similarity between distribution P and Q, called *mean map kernel* (MMK), is defined as inner product in RKHS [25]:

$$K^{\mathrm{MMK}}(P,Q) := \langle \mu_\kappa(P), \mu_\kappa(Q) \rangle_{\mathcal{H}_\kappa} = \mathbb{E}_{x,z}[\kappa(x,z)], \quad (4)$$

where $x \sim P$ and $z \sim Q$.

When we have the empirical distribution P_S and Q_T of images, similarly, the empirical mean map kernel is calculated as follows [25]:

$$K^{\mathrm{MMK}}(P_S,Q_T) = \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}\kappa(x_i,z_j). \quad (5)$$

It can be seen that MMK is a way of essentially aggregating the pairwise similarity over two local feature sets. It possesses many nice theoretical properties, *e.g.*, it is a positive-definite kernel [12, 25]. However, the computational complexity of this estimator in eq. (5) is $O(mnd)$ where d is the dimension of local feature.

3.3 Maximum mean discrepancy kernel

An alternative metric between two distributions, *maximum mean discrepancy* (MMD) [12], is to measure the distance between two distributions. Based on the same property of characteristic kernel, the distance between two distributions, referred as two-sample problem [13], is defined as a RKHS norm:

$$\begin{aligned} D(P,Q) &:= \|\mu_\kappa(P) - \mu_\kappa(Q)\|_{\mathcal{H}_k} \quad (6)\\ &= \left[\mathbb{E}_{x,x'}\kappa(x,x') + \mathbb{E}_{z,z'}\kappa(z,z') - 2\,\mathbb{E}_{x,z}\kappa(x,z)\right]^{1/2}, \end{aligned}$$

where two independent random variables x and x' are drawn from P, and the other two independent random variables z and z' are drawn from Q. Furthermore, x is independent of z.

When we have the empirical distribution P_S and Q_T, a biased (but asymptotically unbiased) estimator of MMD is obtained based on the law of large numbers:

$$\begin{aligned} D(P_S,Q_T) = \Bigg[&\frac{1}{n^2}\sum_{i,j=1}^{n}\kappa(x_i,x_j) + \frac{1}{m^2}\sum_{i,j=1}^{m}\kappa(z_i,z_j)\\ &- \frac{2}{mn}\sum_{i,j=1}^{n,m}\kappa(x_i,z_j)\Bigg]^{1/2}. \quad (7) \end{aligned}$$

If combining MMD with a level-2 kernel [25], an alternative similarity between two distributions, called MMD-based kernel (MMDK) or Gaussian-type RBF kernel [5] will be obtained when the Gaussian kernel defined in eq. (2) is used again. It is formulated as a universal kernel [5]:

$$\begin{aligned} K^{\mathrm{MMD}}(P_S,Q_T) &= \exp\left(-\gamma'\,\|\mu_\kappa(P_S) - \mu_k(Q_T)\|^2_{\mathcal{H}_\kappa}\right)\\ &= \exp\left(-\gamma' D^2(P_S,Q_T)\right), \quad (8) \end{aligned}$$

where γ' is a parameter for the level-2 kernel [25]. It is worth mentioning that although the combination of two level kernels makes MMD kernel more flexible on learning procedure, tuning these two bandwidths is very costly since the computational complexity of estimator in eq. (7) is $O((m+n)^2 d)$.

4 Randomized Distribution Feature

Since the kernel embeddings $\mu_\kappa(P_S) \in \mathcal{H}_\kappa$ are infinite dimensional for some characteristic kernel functions, kernel matrices are often used for dealing with the dual optimization problem. However, the construction of kernel matrices needs at least $O(n^2)$ computational and memory requirement, prohibitive for large n. Therefore, we employ the random Fourier feature to obtain a low-dimensional representation of $\mu_\kappa(P_S)$ [22] in order to avoid invoking the dual optimization. Easy to implement, the proposed method possesses a lot of additional advantages including vectorial representation, efficient computation, nice theory guarantee and competitive performance.

Assume that kernel function κ is real-valued and shift-invariant, Bochner's theorem [38] shows that for any $x, z \in \mathcal{X}$:

$$\kappa(x,z) = 2C_\kappa \mathbb{E}_{w,b}[\cos(\langle w,x\rangle + b)\cos(\langle w,z\rangle + b)], \quad (9)$$

where $w \sim \frac{1}{C_\kappa}p_\kappa$, $b \sim \mathcal{U}[0,2\pi]$, $p_\kappa : \mathcal{X} \to \mathbb{R}$ is the positive and integrable Fourier transform of κ, and $C_\kappa = \int_{\mathcal{X}} p_\kappa(w)\,\mathrm{d}w$ [22]. In this paper, Gaussian kernel in eq. (2) which is a shift-invariant kernel is approximated by eq. (9), if setting $p_\kappa(w) = \mathcal{N}(w|0, 2\gamma I)$ and $C_\kappa = 1$ [22].

Sampling t times from $p_\kappa(w)$ and $\mathcal{U}[0,2\pi]$, concretely, we have the parameters $\{(w_l, b_l)\}_{l=1}^t$. The kernel mapping $\kappa(x,\cdot)$ is then approximated by the following formula

$$\kappa^{\mathrm{F}}(x,\cdot) = \sqrt{\frac{2}{t}}\Big[\cos\left(\langle w_1,x\rangle + b_1\right), \ldots, \cos\left(\langle w_t,x\rangle + b_t\right)\Big]^T \in \mathbb{R}^t, \quad (10)$$

which is the low-dimensional representation of kernel mapping function $\kappa(x,\cdot)$ in a t-dimensional space through random Fourier feature [36, 37]. This random Fourier feature has been widely used to approximate kernel function in many applications [4, 20, 21] since its computation is more efficient than those of kernel methods.

By eq. (10), the empirical kernel embedding $\mu_\kappa(P_S)$ is further approximated by

$$\mu^{\mathrm{F}}_\kappa(P_S) = \frac{1}{n}\sum_{i=1}^{n}\kappa^{\mathrm{F}}(x_i,\cdot) \in \mathbb{R}^t. \quad (11)$$

This estimator has been studied in cause-effect inference [22] and heuristically used in match kernel [2]. Since it represents a distribution by random Fourier feature into a vector, we call it the *randomized distribution feature* (RDF) in this paper. It is noticeable that this estimator is efficient because its computational complexity is $O(ndt)$.

In the following two subsections, we will show how to use RDF to approximate the MMK and MMD between two distributions.

4.1 RDF based similarity

Given the two local feature sets S and T from two images and the sampled parameters $\{(w_l, b_l)\}_{l=1}^t$, vectorial feature I^{RDF} of image is represented by eq. (11). The similarity between two RDFs of images could be formulated as inner product:

$$K^{\mathrm{RDF}} = \langle \mu_\kappa^{\mathrm{F}}(P_S), \mu_\kappa^{\mathrm{F}}(Q_T) \rangle. \tag{12}$$

It is obvious that the similarity well approximates to MMK and is easy to implement. The computational complexity of this similarity is $O((m+n)dt)$ since it is linear with respect to the size of sample sets. The convergence of the similarity based on RDF to MMK is justified in the following theorem.

Theorem 1 *For any shift-invariant kernel κ, for the given two empirical distributions P_S of P and Q_T of Q on $\mathcal{X}$,respectively, and any $\delta > 0$, we have*

$$\begin{aligned} &\left| K^{\mathrm{MMK}}(P,Q) - K^{\mathrm{RDF}}(P_S, Q_T) \right| \\ &\qquad \leq 2\sqrt{2\log(\frac{2}{\delta})(\frac{1}{n}+\frac{1}{m}+\frac{1}{t})}, \end{aligned} \tag{13}$$

with the probability greater than $1-\delta$ over $\{x_i\}_{i=1}^n, \{z_j\}_{j=1}^m$, and $\{(w_l, b_l)\}_{l=1}^t$.

Furthermore, the expected absolute error is

$$\begin{aligned} &\mathbb{E}\left| K^{\mathrm{MMK}}(P,Q) - K^{\mathrm{RDF}}(P_S, Q_T) \right| \\ &\qquad \leq 2\sqrt{2\pi(\frac{1}{m}+\frac{1}{n}+\frac{1}{t})}. \end{aligned} \tag{14}$$

Proof to this theorem is attached in Appendix. Theorem 1 shows that the similarity based on RDF converges to MMK at a rate of $O(m^{-\frac{1}{2}})$ $(O(n^{-\frac{1}{2}}))$ with respect to the size of samples and $O(t^{-\frac{1}{2}})$ with respect to the dimension of low-dimensional embedding space.

4.2 Doubly RDF based similarity

This subsection introduces how to approximate the MMD by RDF. Similar to the introduction of MMD at Sec 3.3, the distance between two distributions represented by RDF is formulated as the Euclidean distance:

$$D^{\mathrm{RDF}}(P_S, Q_T) = \left\| \mu_\kappa^{\mathrm{F}}(P_S) - \mu_\kappa^{\mathrm{F}}(Q_T) \right\|. \tag{15}$$

Compared to MMD in eq. (7), this distance can be computed efficiently since the computational complexity of this distance is $O((m+n)dt)$, which is linear with respect to the size of sample sets. The convergence of $D^{\mathrm{RDF}}(P_S, Q_T)$ to the MMD $D(P,Q)$ is shown in the following theorem.

Theorem 2 *For any shift-invariant kernel κ, s.t., $\sup_{x\in\mathcal{X}} \kappa(x,x) \leq 1$, for the given two empirical distributions P_S of P and Q_T of Q on $\mathcal{X}$, respectively, and any $\delta > 0$, we have*

$$\begin{aligned} &{D^{\mathrm{RDF}}}^2(P_S, Q_T) - D^2(P,Q) \\ &\qquad \leq [\frac{1}{n}+\frac{1}{m}] + 4\sqrt{\log(\frac{1}{\delta})(\frac{9}{n}+\frac{9}{m}+\frac{16}{t})}, \end{aligned} \tag{16}$$

with the probability greater than $1-\delta$ over $\{x_i\}_{i=1}^n$, $\{z_j\}_{j=1}^m$ and $\{(w_l, b_l)\}_{l=1}^t$.

Furthermore, the expected error is

$$\begin{aligned} &\mathbb{E}\left[{D^{\mathrm{RDF}}}^2(P_S, Q_T) - D^2(P,Q) \right] \\ &\qquad \leq [\frac{1}{n}+\frac{1}{m}] + \sqrt{2\pi(\frac{9}{n}+\frac{9}{m}+\frac{16}{t})}. \end{aligned} \tag{17}$$

The proof for this can be seen in the Appendix. Theorem 2 implies that the distance based on RDF converges to MMD at a rate of $O(m^{-\frac{1}{2}})$ $(O(n^{-\frac{1}{2}}))$ with respect to the size of samples and $O(t^{-\frac{1}{2}})$ with respect to the dimension of low-dimensional embedding space.

Once RDFs of images are constructed, the similarity between two images is also formulated as

$$\kappa'(\mu_\kappa^{\mathrm{F}}(P_S), \mu_\kappa^{\mathrm{F}}(Q_T)) = \exp\big(-\lambda' {D^{\mathrm{RDF}}}^2(P_S, Q_T)\big), \tag{18}$$

which approximates to MMD kernel.

However, there is still a level-2 kernel κ' contained in this similarity. It is observed that eq. (18) can also be represented in a low-dimensional embedding space again by using the random Fourier feature, we thus propose an alternative way to represent the image as a vector by using the random Fourier feature twice, called *doubly randomized distribution feature* (DRDF) in this paper. It is defined as follows:

$$I^{\mathrm{DRDF}} = {\kappa'}^{\mathrm{F}}(\mu_\kappa^{\mathrm{F}}(P_S), \cdot) \in \mathbb{R}^{t'}, \tag{19}$$

where ${\kappa'}^{\mathrm{F}} : \mathbb{R}^t \to \mathbb{R}^{t'}$ and t' is the dimension of low-dimensional embedding space for approximating the RHKS $\mathcal{H}_{\kappa'}$ associated with kernel function κ'. In this way, the similarity in eq. (18) can be easily calculated by following inner product:

$$K^{\mathrm{DRDF}}(P_S, Q_T) = \langle I_{P_S}^{\mathrm{DRDF}}, I_{Q_T}^{\mathrm{DRDF}} \rangle. \tag{20}$$

where the computational complexity of this estimator is $O((m+n)dt + tt')$. Compared to eq. (8), there is no parameter to be tuned in eq. (20), resulting in high efficiency of computing DRDF.

4.3 Summary of the proposed methods

To facilitate the understanding of the proposed methods, a workflow is shown in Figure 1. As depicted in the figure, the local features of sample image are assumed to be drawn from an unknown mixture distribution which contains the scene features and human activity features to describe the event [19]. From bottom to top, the similarity would be more accurate as increasing dimension of low-dimensional embedding space into infinity, but the computational cost will become more expensive. From left to right, level of kernel increases in the methods with more flexibility. Technically, the level of kernel can be more than 2. Most importantly, each distribution is well represented by a vector.

5 Experiment

This section presents the application of proposed methods on image classification and distribution regression with scalar response.

5.1 Image Classification

In this section, we show the empirical performance of the proposed (doubly) randomized distribution feature in three real-world image classification tasks.

For image classification tasks, the images are represented as "sets of features"(SOF), *e.g.*, sets of unordered local feature vectors. The

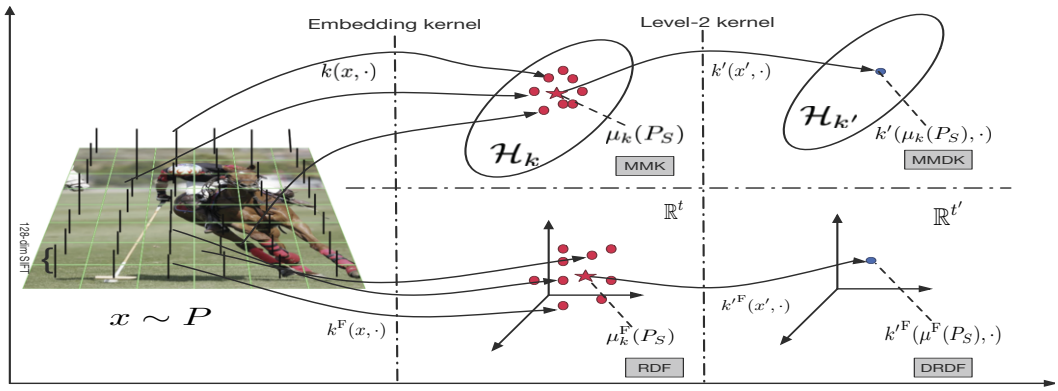

Figure 1. Workflow of the proposed methods. From bottom to top, the similarity would be more accurate and computational cost would be more expensive. From the left to right, level of kernel increases in the methods. Note that the red star denotes the mean point of red points.

proposed methods convert SOF into a vectorial representation like BoW. Therefore, it can be used off-the-shelf in conjunction with any learning algorithm for subsequent image classification. In this paper, we take multi-class SVM as the learning algorithm. For comparison, several algorithms are chosen from the histogram-based model and the metric-based model as follows:

The histogram-based model The BoW model is taken as the baseline algorithm. When we employ linear kernel and Gaussian kernel, the methods are called **BoW_L** and **BoW_G** respectively. For the fair comparison with other methods, the Euclidean distance is used in **BoW_G** method. The number of visual words is set as 1000 unless noted otherwise.

The metric-based model Three algorithms of the metric-based model, **MMK** [25], **MMDK** [25] and the state-of-the-art nonparametric divergence estimator **NPKL** [34], are employed for comparison. **MMK** has only one parameter λ to be decided, and **MMDK** has two parameters λ and λ' for embedding kernel and level-2 kernel respectively. As for **NPKL** [34], the nonparametric Rényi-α divergence between two distributions is used to approximate the KL divergence by setting $\alpha = 0.99$. Compared to **MMD**, the divergence estimated by **NPKL** is non-symmetric. Therefore, the kernel matrix based on nonparametric divergence should be projected to be a symmetric positive semi-definite matrix by symmetrizing the estimated Gram matrix and then projecting to the core of positive semi-definite matrices [15].

The RDF-based model The proposed **RDF** and **DRDF** are calculated based on our proposed (doubly) randomized distribution feature. Both dimensions of low-dimensional embedding space t for approximating embedding kernel and t' for approximating level-2 kernel are set as 1000 in this paper unless noted otherwise. γ and γ' in the Fourier transform p_κ and $p_{\kappa'}$ are calculated using the median trick separately [22]. For a given t (and t' when used), the similarity matrix we used in experiments is the average of 10 times repetition considering the random sample of w and b.

Parameter setting For **BoW_L**, **RDF** and **DRDF**, we use their similarity matrices directly. For other methods, γ in Gaussian kernel defined in eq. (2) is chosen from $\gamma_0 \times \{2^{-9}, 2^{-8}, \ldots, 2^9\}$, where γ_0 is estimated by median trick. The penalty to points within the margin C is chosen from $\{2^{-7}, 2^{-6}, \ldots, 2^4\}$. C and (when used) γ are chosen through joint 3-fold cross-validation on the training set. Note that there are two γ for different level Gaussian kernel in **MMDK**, it is pretty hard to tune these two γ by cross-validation because of the high computational cost of **MMDK**. According to the strategy used in [25], the best γ in **MMK** obtained by cross-validation is used for embedding kernel in **MMDK**, the γ' in level-2 kernel is then tuned by cross-validation. Finally, the 5th nearest neighbor in these estimators is used according to the suggestion in [34].

Feature extraction Local features are extracted as follows. The SOF representation of an image is based on the *dense* SIFT descriptors where step size 10 is used to sample image patches and the size of each patch is 12 in this paper unless noted otherwise. We only use the grayscale images to extract SIFT features and each image is represented by a set of 128-dimensional feature vectors. In order to reduce computational cost of the metric-based model, the dimension of SIFT is reduced by principal component analysis in our experiments, preserving 80% variance [34]. Note that each SOF may have different size, depending on the size of image.

Assessing running time For assessing the computational efficiency, each method was implemented in MATLAB® 2014b and executed on a server which has a total RAM of 512 GB and four AMD Opteron 6378 processors, each of which contains 16 cores. The running time of each algorithm for constructing the similarity/distance/divergence matrix is assessed in this paper.

Algorithm implementation Multi-class SVM in LibSVM package [3] is employed for image classification tasks in this paper. Besides, feature extraction of image and *K*-means use the *PHOW* and *kmeans* functions of the VLFEAT package [44] repectively. Furthermore, the code of **NPKL** is provided by [34] and the codes of **MMK**, **MMDK** and the proposed **(D)RDF** are implemented in *MEX C++* files which are invoked by MATLAB.

5.1.1 *Description of three benchmarked datasets*

In this subsection, we will describe three benchmarked datasets for different image classification tasks.

ETH-80 dataset [17] is widely used for *object classification*. This dataset contains 8 categories of objects. Each category has 10 different objects, and each object has 41 images from different view angles. Here we can suppose that the images with different view angles are drawn from the same distribution of that object. Moreover, all the images from one category could empirically describe the distribution of that category. Following [34], we extract dense SIFT descriptors in each patch of size 6 for the whole 3280 images in our experiments. The purpose of this experiment is to classify these objects into the 8 categories. In order to save the computational cost of the metric-based model, SIFT features are reduced to 29 dimensions by PCA in this experiment. Thus, each image is represented by a set of 576 29-dimensional features and this dataset produces $1,889,280$

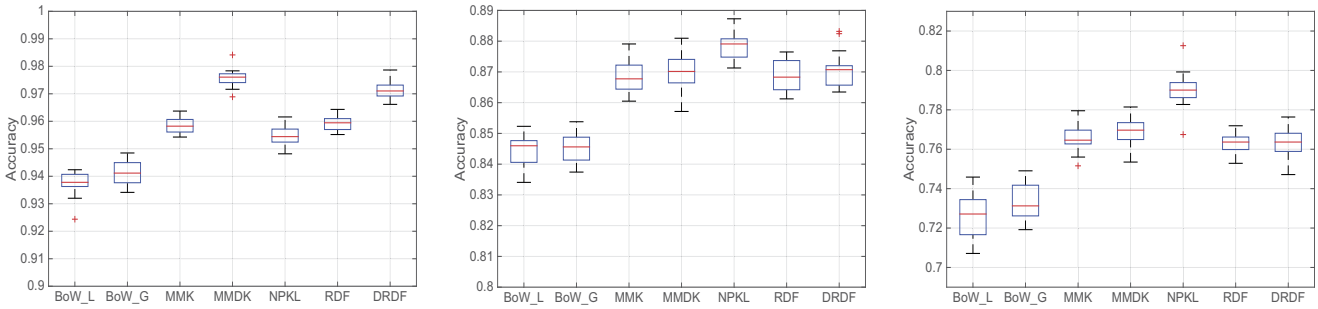

Figure 2. Accuracies on *(a)* ETH-80 dataset, *(b)* OT dataset and *(c)* SE dataset.

SIFT features in total.

OT data set [29] we consider here is a widely used benchmark for *scene classification*. In general, a scene image can be described by a distribution of local features, *e.g.*, the proportion of sky, water, tree, etc. OT dataset includes 8 outdoor scene categories: coast, forest, highway, inside city, mountain, open country, street and tall building. There are 2688 images in total, and each image is in 256×256 pixels. The purpose of this dataset is to classify test images into one of the categories. The original SIFT features are reduced to 30-dimension by using PCA. A typical image is thus represented by 484 30-dimensional local features, which means a total of $1,300,992$ SIFT features are extracted from this dataset.

UIUC Sport Event (SE) datasets [19] is considered in the third experiment since the various foreground activities of this dataset make it more difficult than other traditional scene classification, *e.g.*, the OT dataset we used above. This dataset contains Internet images of 8 sport event categories: badminton, bocce, croquet, polo, rock climbing, rowing, snowboarding, and sailing. Each image can be viewed as a *mixture* distribution of scene features and human activity feature to describe the event [19]. The number of images in each category varies from 137 to 250. We use all the 1574 images in experiments. As the size of images varies, the number of local features in each SOF varies from 88 to 484. As a result, there are totally $535,678$ SIFT features, each of which is reduced to 34 dimension.

5.1.2 *Classification accuracy*

For fair comparison and saving computational cost of metric based model, we employ 2-fold cross-validation to split data, which means 50% of data set for training and remaining 50% for testing. The average performance of 20 random runs is reported in Figure 2. From these three experimental results, it can be seen that the metric- and RDF-based methods outperform **BoW** model since the quantization in **BoW** results in the loss of information—potentially a lot of information. As the similarities based on **RDF** and **DRDF** are the approximators to **MMK** and **MMDK** respectively, it is not difficult to see that **DRDF** and **RDF** perform slightly worse compared with **MMK** and **MMDK**. These results justify that our proposed **(D)RDF** achieve competitive performance with the metric-based model and better performance than that of **BoW**.

In order to show whether the differences between the proposed methods and their corresponding versions in metric-based model are significant, a paired *t-test* at the significant level 5% is performed on these three real-world datasets. With this significant test, the result shows that **RDF** and **MMK** achieve statistically same performance on the ETH-80 and OT datasets. Meanwhile, **DRDF** and **MMDK** are statistically significant on these three datasets. A possible reason is that random Fourier feature is used twice in **DRDF**, leading to the loss of much more information for prediction when compared with **RDF**. Note that the *t-test* relies on the pre-specified dimension of vectorial representation in the proposed methods. Theorem 1 and2 indicate that **RDF** and **DRDF** converge to **MMK** and **MMDK** respectively as the dimension of vectorial representation increases.

Comparisons between algorithms in each model show that nonlinear feature, *i.e.* mapped into kernel feature space, achieves higher accuracy than original feature space. It can be noticed that **NPKL** achieves best performances on two of three datasets since its nonparameric estimation of divergence based on k-nearest neighbor. Remember that the metric-based model suffers from the expensively computational cost.

5.1.3 *Effect of parameters*

We examine the effect of parameters upon the performance of the proposed (doubly) randomized distribution feature on ETH-80 dataset.

Dimension of vectorial representation For fair comparison, the number of visual words in **BoW**, the dimension of embedding space t in **RDF** and another dimension t' in **DRDF** are set as the same value since this is the dimension of vectorial representation of each image. To analyze the influence of this value, we vary it from from 10 to 10000 and report the results in Figure 3(a). It can be seen that 1) the (doubly) randomized distribution feature work better than histogram representation of **BoW** and 2) all the algorithms converge when the dimension is greater than 1000. Note that **BoW** has not degenerated in performance as the dimension increases because 10000 is still small compared to the number of the all SIFT features extracted from this dataset.

Running time as the dimension increases Running time for constructing the similarity/distance matrices versus the dimension of image representation is reported in Figure 3(b). Since **BoW_L** and **BoW_G** spend almost the same time constructing similarity matrix and distance matrix, we combine them as one to show their running time. From Figure 3(b) it can be seen that the **BoW** needs higher computational cost than **RDF** and **DRDF**, especially when the number of vectorial representation gets large. **DRDF** needs more time than **RDF** slightly since the random Fourier feature is applied twice in **DRDF**.

Effect of two dimensions t and t' in DRDF To show the effect of **DRDF** caused by two dimensions t and t', a subset of 400 images is used to tune these two dimensions in order to reduce storage size. The effect of **DRDF** with various t and t' is shown in Figure 3(c). It

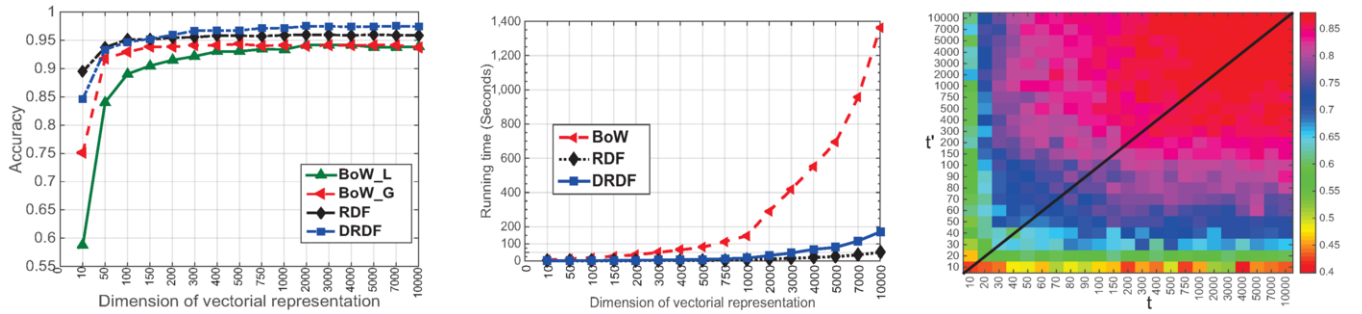

Figure 3. Figures are *(a)* Varying the dimension of vectorial feature of images; *(b)* Running time for constructing the similarity matrices; and *(c)* Sensitivity of **DRDF** with respect to t and t'.

can be seen that both dimension t and t' are important to the performance of **DRDF**, and dimension t tends to have more impact on the predictive performance compared to dimension t'. This conclusion coincides with effect of the two bandwidths in **MMDK** [25].

Influence of dimension reduction In order to investigate the influence of dimension reduction via PCA, we also perform experiments on raw SIFT features which are 128-D feature and show the experimental results in Table. 2. Due to expensively computational cost of the metric-based model, the performances of **MMK**, **MMDK** and **NPKL** are not included here.

Table 2. Classification accuracies and their standard deviations (in brackets) on three benchmarked datasets with raw SIFT features.

Datasets	**BoW_L**	**BoW_G**	**RDF**	**DRDF**
ETH-80	0.9464 (0.0165)	0.9508 (0.0127)	0.9612 (0.0091)	**0.9730** (0.0113)
OT	0.8506 (0.0117)	0.8569 (0.0103)	0.8701 (0.0145)	**0.8749** (0.0146)
SE	0.7444 (0.0245)	0.7553 (0.0261)	0.7807 (0.0229)	**0.7880** (0.0160)

By comparing classification accuracies on raw SIFT features as shown in Table. 2 and accuracies with pre-proceeding by keeping 80% variance as reported in Figure 2, we can see that each algorithm gains slightly improvement on its raw SIFT feature. On average, **BoW**-based algorithms improve about 1.5% while **RDF**-based ones improve only about 0.5%. We notice that **RDF**-based algorithms still achieve better performance than that of **BoW**-based algorithms on raw SIFT features. This means that although reducing the dimension of SIFT features is not a necessary step, it is worth doing this step so that the proposed algorithms can attain lower computational cost in the dimension-reduced space.

5.1.4 Running time over three datasets

In this subsection, we compare the running time of each algorithm for constructing the similarity/distance/divergence matrices over the aforementioned four datasets by using their whole samples.

Running time of each algorithm is reported in Table 3. We can see that the metric-based algorithms consume more computational time than other algorithms do for attaining good performance. Even though **BoW** saves more time than the metric-based models, it has the worst performance among these algorithms we used since a lot of information may be lost in the quantization process. To conclude, our proposed algorithms require less computational time yet achieve competitive predictive performance as the metric-based models do. More specifically, our proposed **RDF** and **DRDF** are at least 10 times faster than **BoW** in vectorial representation with achieving higher accuracies, and at least 500 times faster than the metric-based models with competitive performance on average.

Table 3. Running time among different algorithms (seconds).

Dateset	**BoW**	**NPKL**	**MMK**	**MMDK**	**RDF**	**DRDF**
ETH-80	146	10277	4066	11924	**5.9**	17.5
OT	114	5171	1812	5227	**3.8**	11.0
SE	70	1022	352	1045	**2.0**	4.7
Average	110	5490	2076	6065	**3.9**	11.0

5.2 Application on learning problems on distribution

Besides image classification, we also apply randomized distribution feature to learning problems on distribution. Distribution regression with scalar response [35] is considered here where each input is distribution and output is the scalar response. The setup of this experiment is to learn the skewness of Beta distribution when given their sample set. We generated 300 sample sets from Beta(a, b) distributions where a was varied between $[3, 20]$ randomly and b was fixed to be 3. We used 200 sample sets for training and 100 for testing. Each sample set consisted of 500 distributed i.i.d. points drawn from Beta$(a, 3)$. Note that the skewness of Beta(a, b) can be calculated as $\frac{2(b-a)}{(2+a+b)}\sqrt{\frac{1+a+b}{ab}}$.

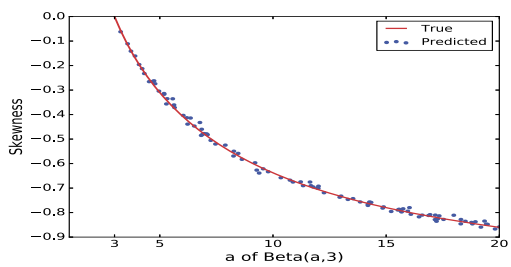

Figure 4. Skewness of Beta distribution

In this experiments, we used a 20-dimensional random distribution feature to represent a Beta distribution and regressed this vectorial representation to its skewness by least squared regression method. Figure 4 displays the predicted values for the 100 test sample sets.

Here we only report the result of RDF, because RDF has provided an accurate representation of distribution, and DRDF is to some extent a nonlinear regression with RDF. This experiment shows the proposed method could make learning problem on distribution become a traditional machine learning problem whose input is vector. More real-world applications can be concluded as the learning problems on distribution and benefited from our proposed features, such as counting the pedestrian or cells from a given image [18] and detecting anomaly group [27].

6 Conclusion and Discussion

In this paper, we introduce the randomized distribution feature to represent distribution. In this manner, the underlying distribution of local features extracted from images can be represented as a vector in image classification. Furthermore, we propose an alternative way to represent image by a doubly randomized distribution feature for further improving predictive performance. We also justify the convergences of the similarity and distance based on RDF. Our recommended feature representation of images inherits the advantages of both the histogram-based model and the metric-based model. It has vectorial representation and computes efficiently like BoW model, and has nice theory guarantee and competitive performance as the metric-based model. Experiments in three benchmark datasets justify these strengths of our proposed approaches. Furthermore, the proposed features could make learning problems on distribution become traditional machine learning problems where each input is a vector.

Compared with VLAD [1] / FV [31] that attempt to learn discriminant information for image classification task, our proposed method focuses on a general representation of distribution that could suit for not only image classification, but also other tasks such as distribution regression. To consider the data structure of distribution, a data-dependent random distribution feature based on Nyström method [46] deserves further studying. Theoretically, it is also of interest to derive tight error bound of convergence of similarity and distance based on RDF according to [42].

7 Acknowledgements

This work has been sponsored by the National Science Foundation of China (No. 61273299).

APPENDIX

Proof 1 (to Theorem 1) *The similarity based on RDF is calculated as follows:*

$$K^{\mathrm{RDF}}(P_S,Q_T)=\frac{2}{nmt}\sum_{i,j,l=1}^{n,m,t}\Big[\cos(\langle w_l,x_i\rangle+b_l)\cos(\langle w_l,z_j\rangle+b_l)\Big].$$

Taking expectation over $x_i, z_j, (w_l, b_l)$, we derive the following equality

$$\begin{aligned}&\mathbb{E}_{x_i,z_j,w_l,b_l}K^{\mathrm{RDF}}(P_S,Q_T)\\ =&\frac{1}{nmt}\sum_{i,j,l=1}^{n,m,t}\mathbb{E}_{x_i,z_j}\mathbb{E}_{w_l,b_l}2\Big[\cos(\langle w_l,x_i\rangle+b_l)\cos(\langle w_l,z_j\rangle+b_l)\Big]\\ =&\mathbb{E}_{x,z}\kappa(x,z)=K^{\mathrm{MMK}}(P,Q),\end{aligned}\tag{21}$$

where Bochner's theorem in eq. (9) is applied here. Eq. (21) indicates that $K^{\mathrm{RDF}}(P_S,Q_T)$ is an unbiased estimator of $K^{\mathrm{MMK}}(P,Q)$.

By introducing a variable Δ to measure the difference between $K^{\mathrm{RDF}}(P_S,Q_T)$ and $K^{\mathrm{MMK}}(P,Q)$, we have

$$\begin{aligned}\Delta&=K^{\mathrm{RDF}}(P_S,Q_T)-K^{\mathrm{MMK}}(P,Q)\\ &=\frac{1}{nmt}\sum_{i,j,l=1}^{n,m,t}\Big[2\cos(\langle w_l,x_i\rangle+b_l)\cos(\langle w_l,z_j\rangle+b_l)-\mathbb{E}_{x,z}\kappa(x,z)\Big].\end{aligned}\tag{22}$$

We first provide an upper bound on the difference between Δ and its expectation. Note that changing either of $x_i, z_j, (w_l, b_l)$ in eq. (22) results in changes in magnitude of at most $\frac{4}{n}$, $\frac{4}{m}$, or $\frac{4}{t}$, respectively. We can then apply McDiarmid's theorem [24], given a denominator in the exponent of $n(\frac{4}{n})^2+m(\frac{4}{m})^2+t(\frac{4}{t})^2=16\frac{mn+nt+mt}{mnt}$, to obtain

$$P\Big[\big|\Delta-\mathbb{E}_{x_i,z_j,w_l,b_l}\Delta\big|\geq\epsilon\Big]\leq 2\exp\Big(\frac{-mnt\epsilon^2}{8(mn+nt+mt)}\Big).$$

Let $\delta=2\exp(\frac{-mnt\epsilon^2}{8(mn+nt+mt)})>0$, we get $\epsilon=2\sqrt{2\log(\frac{2}{\delta})(\frac{1}{n}+\frac{1}{m}+\frac{1}{t})}$. Remember that $\mathbb{E}_{x_i,z_j,w_l,b_l}\Delta=0$ as shown in Eq. (21), thus at least $1-\delta$, we have

$$|\Delta|\leq 2\sqrt{2\log(\frac{2}{\delta})(\frac{1}{n}+\frac{1}{m}+\frac{1}{t})}.\tag{23}$$

So we derive the first inequality of theorem. Next we will derive the second inequality, i.e., *the expected absolute error between $K^{\mathrm{RDF}}(P_S,Q_T)$ and $K^{\mathrm{MMK}}(P,Q)$. The expected absolute error is*

$$\begin{aligned}\mathbb{E}|\Delta|&=\int_0^\infty P\Big[|\Delta|\geq\epsilon\Big]\,\mathrm{d}\epsilon\\ &\leq\int_0^\infty 2\exp\Big(\frac{-mnt\epsilon^2}{8(mn+nt+mt)}\Big)\,\mathrm{d}\epsilon=2\sqrt{2\pi(\frac{1}{m}+\frac{1}{n}+\frac{1}{t})}.\end{aligned}\tag{24}$$

Here eq. (24) is from the fact that expectation over non-negative probability distribution, i.e., *$\mathbb{E}[X]=\int_0^\infty xf_X(x)\,\mathrm{d}x=\int_0^\infty P[X\geq x]\,\mathrm{d}x,\forall x\geq 0$.* ■

Proof 2 (to Theorem 2) *This proof resembles Proof 1. We first bound the differerence between $D^{\mathrm{RDF}^2}(P_S,Q_T)$ and $D^2(P,Q)$ by introducing a variable Δ as follows*

$$\Delta=D^{\mathrm{RDF}^2}(P_S,Q_T)-D^2(P,Q)$$

Similarly, changing either of x_i, z_j or (w_l, b_l) results in changes in magnitude of at most $\frac{12}{n}$, $\frac{12}{m}$, or $\frac{16}{t}$, respectively. Applying McDiarmid's theorem [24] gives a denominator in the exponent of $n(\frac{12}{n})^2+m(\frac{12}{m})^2+t(\frac{16}{t})^2=\frac{256mn+144t(n+m)}{mnt}$, to obtain

$$P\Big[\Delta-\mathbb{E}_{x_i,x_{i'},z_j,z_{j'},w_l,b_l}\Delta\geq\epsilon\Big]\leq\exp\Big(-\frac{mnt\epsilon^2}{128mn+72t(n+m)}\Big).$$

Different from Proof 1, $D^{\mathrm{RDF}^2}(P_S,Q_T)$ is asymptotically unbiased estimator and the expectation over the difference is bounded by

$$\mathbb{E}_{x_i,x_{i'},z_j,z_{j'},w_l,b_l}\Delta\leq\frac{1}{n}+\frac{1}{m}.$$

Thus, we have the following inequality

$$P\Big[\Delta\geq[\frac{1}{n}+\frac{1}{m}]+\epsilon\Big]\leq\exp\Big(-\frac{mnt\epsilon^2}{128mn+72t(n+m)}\Big).$$

The two inequalities in Theorem 2 can be derived from above inequality similarly to Proof 1. ■

A detailed version of proof to theorems 1 and 2 can be found at http://www.iipl.fudan.edu.cn/~zhangjp/supp/rdf_sup.pdf.

REFERENCES

[1] Relja Arandjelovic and Andrew Zisserman, 'All about VLAD', in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1578–1585, (2013).

[2] Liefeng Bo and Cristian Sminchisescu, 'Efficient match kernel between sets of features for visual recognition', in *Advances in Neural Information Processing Systems*, pp. 135–143, (2009).

[3] Chih-Chung Chang and Chih-Jen Lin, 'Libsvm: A library for support vector machines', *ACM Transactions on Intelligent Systems and Technology*, **2**(3), 27, (2011).

[4] Radha Chitta, Rong Jin, and Anubhav K Jain, 'Efficient kernel clustering using random fourier features', in *IEEE 12th International Conference on Data Mining*, pp. 161–170, (2012).

[5] Andreas Christmann and Ingo Steinwart, 'Universal kernels on non-standard input spaces', in *Advances in Neural Information Processing Systems*, pp. 406–414, (2010).

[6] Navneet Dalal and Bill Triggs, 'Histograms of oriented gradients for human detection', in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pp. 886–893, (2005).

[7] Sanjoy Dasgupta and Yoav Freund, 'Random projection trees for vector quantization', *IEEE Transactions on Information Theory*, **55**(7), 3229–3242, (2009).

[8] Paul Ekman and Wallace V Friesen, *Facial Action Coding System*, Consulting Psychologists Press, 1978.

[9] Li Fei-Fei and Pietro Perona, 'A bayesian hierarchical model for learning natural scene categories', in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pp. 524–531, (2005).

[10] Kenji Fukumizu, Le Song, and Arthur Gretton, 'Kernel Bayes' rule', in *Advances in Neural Information Processing Systems*, pp. 1737–1745, (2011).

[11] Kristen Grauman and Trevor Darrell, 'The pyramid match kernel: Efficient learning with sets of features', *The Journal of Machine Learning Research*, **8**, 725–760, (2007).

[12] Arthur Gretton, Karsten M Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex J Smola, 'A kernel method for the two-sample-problem', in *Advances in Neural Information Processing Systems*, pp. 513–520, (2007).

[13] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola, 'A kernel two-sample test', *The Journal of Machine Learning Research*, **13**(1), 723–773, (2012).

[14] Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf, 'Measuring statistical dependence with Hilbert-Schmidt norms', in *Algorithmic Learning Theory*, pp. 63–77. Springer, (2005).

[15] Nicholas J Higham, 'Computing the nearest correlation matrix—a problem from finance', *IMA Journal of Numerical Analysis*, **22**(3), 329–343, (2002).

[16] Hervé Jégou, Matthijs Douze, Cordelia Schmid, and Patrick Pérez, 'Aggregating local descriptors into a compact image representation', in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3304–3311, (2010).

[17] Bastian Leibe and Bernt Schiele, 'Analyzing appearance and contour based methods for object categorization', in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pp. II–409, (2003).

[18] Victor Lempitsky and Andrew Zisserman, 'Learning to count objects in images', in *Advances in Neural Information Processing Systems*, pp. 1324–1332, (2010).

[19] Li-Jia Li and Li Fei-Fei, 'What, where and who? classifying events by scene and object recognition', in *Proceedings of IEEE International Conference on Computer Vision*, pp. 1–8, (2007).

[20] D Lopez-Paz, S Sra, A Smola, Z Ghahramani, and B Schölkopf, 'Randomized nonlinear component analysis', in *Proceedings of the 31st International Conference on Machine Learning*, pp. 1359–1367, (2014).

[21] David Lopez-Paz, Philipp Hennig, and Bernhard Schölkopf, 'The randomized dependence coefficient', in *Advances in Neural Information Processing Systems*, pp. 1–9, (2013).

[22] David Lopez-Paz, Krikamol Muandet, Bernhard Schölkopf, and Iliya Tolstikhin, 'Towards a learning theory of cause-effect inference', in *Proceedings of the 32rd International Conference on Machine Learning*, pp. 1452–1461, (2015).

[23] David G Lowe, 'Object recognition from local scale-invariant features', in *Proceedings of IEEE International Conference on Computer Vision*, volume 2, pp. 1150–1157, (1999).

[24] Colin McDiarmid, 'On the method of bounded differences', *Surveys in combinatorics*, **141**(1), 148–188, (1989).

[25] Krikamol Muandet, Kenji Fukumizu, Francesco Dinuzzo, and Bernhard Schölkopf, 'Learning from distributions via support measure machines', in *Advances in Neural Information Processing Systems*, pp. 10–18, (2012).

[26] Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, Arthur Gretton, and Bernhard Schoelkopf, 'Kernel mean estimation and Stein effect', in *Proceedings of the 31st International Conference on Machine Learning*, pp. 10–18, (2014).

[27] Krikamol Muandet and Bernhard Schölkopf, 'One-class support measure machines for group anomaly detection', in *Uncertainty in Artificial Intelligence*, pp. 449–458. Citeseer, (2013).

[28] David Nister and Henrik Stewenius, 'Scalable recognition with a vocabulary tree', in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pp. 2161–2168, (2006).

[29] Aude Oliva and Antonio Torralba, 'Modeling the shape of the scene: A holistic representation of the spatial envelope', *International Journal of Computer Vision*, **42**(3), 145–175, (2001).

[30] Junier Oliva, Barnabás Póczos, and Jeff Schneider, 'Distribution to distribution regression', in *Proceedings of the 30th International Conference on Machine Learning*, pp. 1049–1057, (2013).

[31] Florent Perronnin, Jorge Sánchez, and Thomas Mensink, 'Improving the fisher kernel for large-scale image classification', in *European Conference on Computer Vision*, 143–156, (2010).

[32] Barnabas Poczos, Aarti Singh, Alessandro Rinaldo, and Larry Wasserman, 'Distribution-free distribution regression', *International Conference on Artificial Intelligence and Statistics*, 507–515, (2013).

[33] Barnabás Póczos, Liang Xiong, and Jeff Schneider, 'Nonparametric divergence estimation with applications to machine learning on distributions', *arXiv:1202.3758*, (2012).

[34] Barnabás Póczos, Liang Xiong, Dougal J Sutherland, and Jeff Schneider, 'Nonparametric kernel estimators for image classification', in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2989–2996, (2012).

[35] Barnabás Póczos, Liang Xiong, Dougal J Sutherland, and Jeff Schneider, 'Support distribution machines', Technical report, Carnegie Mellon University, (2012).

[36] Ali Rahimi and Benjamin Recht, 'Random features for large-scale kernel machines', in *Advances in Neural Information Processing Systems*, pp. 1177–1184, (2007).

[37] Ali Rahimi and Benjamin Recht, 'Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning', in *Advances in Neural Information Processing Systems*, (2008).

[38] Walter Rudin, *Fourier analysis on groups*, number 12, John Wiley & Sons, 1990.

[39] Bernhard Schölkopf and Alexander J Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*, MIT press, 2002.

[40] Alex Smola, Arthur Gretton, Le Song, and Bernhard Schölkopf, 'A hilbert space embedding for distributions', in *Algorithmic learning theory*, pp. 13–31. Springer, (2007).

[41] Le Song, Byron Boots, Sajid M Siddiqi, Geoffrey J Gordon, and Alex J Smola, 'Hilbert space embeddings of hidden markov models', in *Proceedings of the 27th International Conference on Machine Learning*, pp. 991–998, (2010).

[42] Bharath Sriperumbudur and Zoltán Szabó, 'Optimal rates for random fourier features', in *Advances in Neural Information Processing Systems*, pp. 1144–1152, (2015).

[43] Bharath K Sriperumbudur, Arthur Gretton, Kenji Fukumizu, Bernhard Schölkopf, and Gert RG Lanckriet, 'Hilbert space embeddings and metrics on probability measures', *The Journal of Machine Learning Research*, **11**, 1517–1561, (2010).

[44] Andrea Vedaldi and Brian Fulkerson, 'VLFeat: An open and portable library of computer vision algorithms', in *Proceedings of ACM international conference on Multimedia*, pp. 1469–1472, (2010).

[45] Liang Xiong, Barnabás Póczos, and Jeff Schneider, 'Efficient learning on point sets', in *Proceeding of the International Conference on Data Mining*, pp. 847–856, (2013).

[46] Tianbao Yang, Yu-Feng Li, Mehrdad Mahdavi, Rong Jin, and Zhi-Hua Zhou, 'Nyström method vs random fourier features: A theoretical and empirical comparison', in *Advances in Neural Information Processing Systems*, pp. 476–484, (2012).