Nonparametric Kernel Estimators for Image Classification

Barnabás Póczos Liang Xiong Danica J. Sutherland Jeff Schneider*
School of Computer Science, Carnegie Mellon University, Pittsburgh, PA
www.autonlab.org

Abstract

We introduce a new discriminative learning method for image classification. We assume that the images are represented by unordered, multi-dimensional, finite sets of feature vectors, and that these sets might have different cardinality. This allows us to use consistent nonparametric divergence estimators to define new kernels over these sets, and then apply them in kernel classifiers. Our numerical results demonstrate that in many cases this approach can outperform state-of-the-art competitors on both simulated and challenging real-world datasets.

1. Introduction

There are numerous examples in computer vision where images are represented by unordered sets of features. For example, the shapes of objects can be represented by sets of local descriptors at edges and corner points [8]. Human faces can also be described by sets of local image patches containing certain facial parts. The SIFT [17], HOG [4], and PHOG [1] features extractors find stable image representations by detecting sets of local-affine invariant regions and other regions of interest.

To compare images represented by feature sets, a straightforward approach is to treat the sets as if they contained instances sampled from an unknown and possibly high-dimensional distribution. A common way to handle these distributions is to use (high-dimensional) histograms, and compare these histograms by some appropriate metric. "Bag-of-words" (BOW) based image processing algorithms use a similar approach, but include an additional clustering step: They treat each image as a bag of visual words, where the words are clustered feature vectors from local regions [5, 15]. Then each image is represented by the empirical, one-dimensional histogram of these words. The collection of these words is called a codebook or dictionary.

*This work was funded in part by the National Science Foundation under grant NSF-IIS0911032 and the Department of Energy under grant DESC0002607.

Histogram-based features been used in many state-of-the-art computer vision algorithms. However, they have some obvious limitations. When we quantize continuous distributions into bins, we lose information — potentially a lot of information. This problem is especially severe in high dimensions, where the curse of dimensionality makes histogram-based density estimators unreliable. Similarly, BOW algorithms quantize the distributions into clusters, which might lead to loss of information. Selecting the bin sizes for the histograms [27] and the size of dictionary for the BOW model are also difficult model selection problems.

In this paper we propose new image classification algorithms that operate directly on the set-of-features representation of the images. We assume that the elements of these sets are i.i.d. sample points from unknown distributions that characterize the images. In order to classify the images, we classify these distributions based on their i.i.d. sample set representations. The kernel-based approach is adopted: we introduce and estimate the kernel functions between these distributions. Having the estimated kernel matrix (also called a Gram matrix), we then apply kernel classifiers such as SVM for classification. The proposed kernels avoid the traditional clustering, quantization, or histogram building steps that could lead to loss of information.

These kernel functions on sets will be defined in terms of divergences/distances, just as the Euclidean distance is used to define Gaussian/RBF kernels on individual vectors. To this end, we will need to estimate the divergences between distributions. A straightforward approach would be to estimate the underlying densities and plug them into the corresponding divergence formulas. Histogram and BOW approaches follow this paradigm. Density estimation, however, is among the most difficult problems in statistics due to the curse of dimensionality. To avoid this problem, we develop our kernels based on a direct (no density estimation required) and nonparametric (minimal assumptions about the true distributions) approach. We show how to estimate a large family of divergences that includes the Rényi, Tsallis, Hellinger, Bhattacharyya, KL, L_2, and many other divergences. The estimator is provably consistent, nonparametric, and does not use histograms, kernel density estimators

(KDE), or any other density estimators. It depends on only simple k-nearest neighbor (k-NN) statistics.

We evaluate the empirical performance of the proposed kernels on both simulated and real-world datasets, and compare them to alternatives based on density estimation or parametric approximations. We show that our kernels achieve performances that match or beat the state of the art in several image classification tasks.

The paper is organized as follows. In the next section we review some related work. We formally introduce the distribution classification problem and show how to define kernels on distributions in Section 3. Section 4 describes how to evaluate kernels on distributions when the densities are unknown. Section 5 presents the results of numerical experiments. We conclude with a discussion in Section 6.

2. Related Work

Although several methods exist to measure the distance between sample sets, and kernels have also been defined on sets, all of these previous methods have their shortcomings. We will now review the most popular methods.

Nguyen et al. recently proposed a method for f-divergence estimation using its so-called "variational characterization properties" [20]. This approach involves an intractable optimization over an infinite-dimensional function space. When this function space is chosen to be a reproducing kernel Hilbert space (RKHS), this optimization problem reduces to an m-dimensional convex problem, where m is the sample size. This can be very demanding in practice for a only few thousand sample points, which is quite common in computer vision applications.

There are RKHS based approaches for defining kernels on unordered sets as well. The method proposed by Smola et al. [28] uses the interaction between pairs in the sample set, and hence its computation time is $\mathcal{O}(m^2)$. The divergence estimator we propose, by contrast, uses only k-NN distances in the sample set, a well-studied problem with efficient solutions such as k-d trees. Note also that choosing an appropriate kernel function for the RKHS can be a difficult model selection problem, a challenge not faced by our proposed divergence estimator.

Sricharan et al. [29] developed k-nearest-neighbor based methods similar to our method for estimating non-linear functionals of the density, of which divergences are a special case. In contrast to our approach, however, their method requires k to increase with the sample size m and diverge to infinity. k-NN computations for large k values can be very computationally demanding. In our approach we fix k on a small number (typically between 1 and 5), and are still able to prove that the divergence estimator is consistent.

Jebara and Kondor [11] have also studied the question of how to define kernels on distributions. Their approach fits a parametric family (*e.g.* exponential family) density to each set of points, and then using these fitted parameters estimates the inner products between the densities. Moreno et al. [19] also fit a parametric density to the data and use it to define a KL divergence-based kernel. Parametric approaches can work better than nonparametric methods when the sample size m is small, or if we know from prior knowledge that the true densities belong to these parametric families. When the assumptions do not hold, however, parametric methods introduce bias in estimating the inner products between densities. In contrast, our proposed method is completely nonparametric and provides provably asymptotically unbiased kernel estimations for certain kernels.

Kondor and Jebara [12] earlier introduced a kernel between distributions defined as Bhattacharyya's measure of affinity between finite dimensional Gaussians in a Hilbert space. This approach fits a Gaussian distribution to the features in a Hilbert space, but it can lead to a large bias when the data in the Hilbert spaces is not Gaussian. Furthermore, the approach is developed only for Bhattacharyya's measure. Our proposed method is asymptotically unbiased and can be used for many other divergences.

The Pyramid Matching Kernel [8], which also operates over unordered sets, has recently become popular in computer vision. In this approach each feature set is mapped to a multi-resolution histogram. These histogram pyramids are compared using a so-called "weighted histogram intersection computation." A shortcoming of this approach is that it needs to calculate d-dimensional histograms, which can become very inefficient for large d due to the curse of dimensionality. Selecting appropriate bin sizes is also a difficult problem for which only heuristics are known [27].

Póczos et al. [23] used a slightly less general version of our nonparametric divergence estimator similar to solve certain machine learning problems in the space of distributions. This paper studied only simple k-NN based classifiers, however. Here we use kernel methods that are more discriminative in classification tasks, and evaluate their performance on various image datasets.

3. Formal Problem Setting

In this section we formally define our image classification problem and show how kernel classifiers can be generalized to sample sets of distributions. Assume we have T inputs $X_1,\ldots,X_T$ each representing one image, where the tth input $X_t = \{X_{t,1},\ldots,X_{t,m_t}\}$ consists of m_t i.i.d. samples from density p_t. That is, X_t is a set of sample points, and $X_{t,j} \sim p_t$ for $j = 1,\ldots,m_t$. Let $\mathcal{X}$ denote the set of all such sample sets ($X_t \in \mathcal{X}$, $t = 1,\ldots,T$).

Further assume we are given T labels for these inputs $\{(X_t, Y_t)\}_{t=1}^T$. Here $Y_t \in \mathcal{Y} \doteq \{y_1,\ldots,y_c\}$ denotes the class label of the tth image. We seek a function $f : \mathcal{X} \to \mathcal{Y}$ such that for a new input and output pair $(X, Y) \in \mathcal{X} \times \mathcal{Y}$ we ideally have that $f(X) = Y$. For simplicity, we dis-

cuss only binary classification. The ideas below can be extended to c-class classification in the standard ways, *e.g.* voting among $c(c-1)/2$ pairwise classifiers or taking the most confident of c one-vs-all classifiers.

Let $\mathcal{K}$, which will serve as our feature space, denote a Hilbert space with inner product $\langle \cdot, \cdot \rangle_{\mathcal{K}}$. Let $\mathcal{P}$ stand for the set of density functions, and $\phi : \mathcal{P} \to \mathcal{K}$ be an operator that maps the density functions to the feature space $\mathcal{K}$. In what follows we will use the SVM kernel machine for the classification problem. The dual form of the "soft margin Support Vector Machine" can be described as follows [26]:

$$\hat{\alpha} = \arg\max_{\alpha \in \mathbb{R}^T} \sum_{i=1}^{T} \alpha_i - \frac{1}{2} \sum_{i,j}^{T} \alpha_i \alpha_j y_i y_j G_{ij}, \qquad (1)$$

subject to $\sum_i \alpha_i y_i = 0$, $0 \le \alpha_i \le C$, where $C > 0$ is a parameter, $y_i \in \{-1, 1\}$ are the class labels, and $G \in \mathbb{R}^{T \times T}$ is the Gram matrix: $G_{ij} \doteq \langle \phi(p_i), \phi(p_j) \rangle_{\mathcal{K}} = K(p_i, p_j)$. Now, the predicted class label of a test density p is simply $f(p) = \text{sign}(\sum_{i=1}^{T} \hat{\alpha}_i y_i K(p_i, p) + b)$, where the bias term b can be obtained by averaging $b = y_j - \sum_i y_i \alpha_i G_{ij}$ over all points with $\alpha_j > 0$.

There are many tools available to solve the quadratic programming task in (1). All that remains is to estimate $\{K(p_i, p)\}_i$ and $\{K(p_i, p_j)\}_{i,j}$ from the i.i.d. samples.

3.1. Constructing Kernels

Having two finite i.i.d. sample sets from densities p and q, we need to estimate $K(p, q)$, the kernel value between them. Many kernel functions, *i.e.* positive semi-definite functionals of p and q, can be constructed from

$$D_{\alpha,\beta}(p\|q) = \int p^{\alpha}(x)\, q^{\beta}(x)\, p(x)\, \mathrm{d}x, \qquad (2)$$

where $\alpha, \beta \in \mathbb{R}$, including linear $\int pq$, polynomial $(c + \int pq)^s$, and Gaussian kernels $\exp(-\frac{1}{2}\mu^2(p,q)/\sigma^2)$, $\mu^2(p,q) = \int p^2 + q^2 - 2pq$. In the Gaussian kernel, one can also try to use other "distances" $\mu(p,q)$, *e.g.* the Hellinger distance $(1 - \int p^{1/2} q^{1/2})^{1/2}$, the Bhattacharyya distance $(-\log \int p^{1/2} q^{1/2})^{1/2}$, the Tsallis-$\alpha$ divergence $\frac{1}{\alpha - 1}(\int p^{\alpha} q^{1-\alpha} - 1)$, or the Rényi-$\alpha$ divergence $\frac{1}{\alpha-1} \log \int p^{\alpha} q^{1-\alpha}$. Note that the $\alpha \to 1$ limit of the Rényi-α divergence is the KL divergence. These divergences are nonnegative and vanish iff $p = q$ almost surely. Nonetheless, the Rényi and Tsallis divergences are not symmetric, do not satisfy the triangle inequality, and do not lead to positive semi-definite Gram matrices. In Section 4.1 we will show how to address this problem.

4. Nonparametric Kernel Estimation

Let $X_{1:n} \doteq (X_1, \ldots, X_n)$ be an i.i.d. sample from a distribution with density p, and similarly let $Y_{1:m} \doteq (Y_1, \ldots, Y_m)$ be an i.i.d. sample from a distribution of density q. Let $\rho_k(i)$ denote the Euclidean distance of the kth nearest neighbor of X_i in the sample $X_{1:n}$, and similarly let $\nu_k(i)$ denote the distance of the kth nearest neighbor of X_i in the sample $Y_{1:m}$.

In order to estimate the values of the kernels in Section 3.1, we need to estimate $D_{\alpha,\beta}(p\|q)$ for some α, β. Borrowing the tools that have been applied for Rényi entropy [14], Shannon entropy [6], KL divergence [31], and Rényi divergence estimation [22], one can prove that the following estimator is L_2 consistent under certain conditions:

$$\widehat{D}_{\alpha,\beta} = \frac{B_{k,\alpha,\beta}}{n\,(n-1)^{\alpha}\, m^{\beta}} \sum_{i=1}^{n} \rho_k^{-d\alpha}(i)\, \nu_k^{-d\beta}(i), \qquad (3)$$

where $B_{k,\alpha,\beta} \doteq \bar{c}^{-\alpha-\beta} \frac{\Gamma(k)^2}{\Gamma(k-\alpha)\Gamma(k-\beta)}$, $\bar{c}$ denotes the volume of a d-dimensional unit ball, and Γ is the gamma function. Assume $supp(p)$ is a finite union of bounded convex sets. The following theorems claim the asymptotic unbiasedness and L_2 consistency of the estimator (3):

Theorem 1 (Asymptotic unbiasedness) *Let $-k < \alpha, \beta < k$. If $0 < \alpha < k$, then let p be bounded away from zero and uniformly continuous. If $-k < \alpha < 0$, then let p be bounded. Similarly, if $0 < \beta < k$, then let q be bounded away from zero and uniformly continuous. If $-k < \beta < 0$, then let q be bounded. Under these conditions we have that*

$$\lim_{n,m \to \infty} \mathbb{E}\left[\widehat{D}_{\alpha,\beta}(X_{1:n}\|Y_{1:m})\right] = D_{\alpha,\beta}(p\|q), \qquad (4)$$

i.e.*, the estimator is asymptotically unbiased.*

In the previous theorem we have stated conditions that lead to asymptotically unbiased divergence estimation. In the following theorem we will assume that the estimator is asymptotically unbiased for (α, β) as well as for $(2\alpha, 2\beta)$, and also assume that $D_{\alpha,\beta}(p\|q) < \infty, D_{2\alpha,2\beta}(p\|q) < \infty$.

Theorem 2 (L_2 consistency) *Let $k \ge 2$ and $-(k-1)/2 < \alpha, \beta < (k-1)/2$. If $0 < \alpha < (k-1)/2$, then let p be bounded away from zero and uniformly continuous. If $-(k-1)/2 < \alpha < 0$, then let p be bounded. Similarly, if $0 < \beta < (k-1)/2$, then let q be bounded away from zero and uniformly continuous. If $-(k-1)/2 < \beta < 0$, then let q be bounded. Under these conditions we have that*

$$\lim_{n,m \to \infty} \mathbb{E}\left[\left(\widehat{D}_{\alpha,\beta}(X_{1:n}\|Y_{1:m}) - D_{\alpha,\beta}(p\|q)\right)^2\right] = 0; \qquad (5)$$

that is, the estimator is L_2 consistent.

The supplementary material contains proof outlines.

4.1. Projecting to the Cone of PSD Matrices

Under certain conditions $\widehat{D}_{\alpha,\beta}$ is a consistent estimator of $D_{\alpha,\beta}$, and thus by plugging these estimators into the formulae in Section 3.1 we get consistent estimators for those kernels. Any particular estimated Gram matrix, however, might not be symmetric or positive semi-definite. We therefore symmetrize the estimated Gram matrix (by taking half the sum of it and its transpose), then project to the cone of positive semi-definite matrices by discarding any negative eigenvalues from its spectrum [9].

Rather than projecting the estimated kernel and then solving a dual SVM, one can actually combine these two steps into a single convex problem [18]. We do not pursue this approach in this paper, however.

5. Experiments

In this section, we show the empirical performance of the proposed kernels in both simulation studies and real-world image classification tasks. Code and datasets used here are available at `autonlab.org/autonweb/20680.html`.

In all these tasks, the objects of interest are represented as "bags of features" (BOF), *i.e.* unordered sets of feature vectors. The proposed kernel estimators as well as several other kernels between sets of points are used to calculate kernel matrices for these sets. The full kernel matrices are projected to be symmetric positive semi-definite and given to a multi-class SVM for classification.

Nonparametric divergence kernels These kernels are based on the proposed nonparametric Rényi-α divergence estimators (**NPR-α**) and Hellinger distance estimators (**NPH**). We use the $k = 5$th nearest neighbors in these estimators, except in Section 5.1, where small sample sizes necessitate $k = 1$. For NPR, we test the performance with $\alpha \in \{0.5, 0.8, 0.9, 0.99\}$. Note that when $\alpha = 0.99$ the Rényi-divergence approximates the KL divergence, and when $\alpha = 0.5$ it is twice the Bhattacharyya distance.

Parametric kernels These kernels are based on a Gaussian or Gaussian Mixture Model (GMM) assumption. We first fit the density to each group, and then compute the KL-divergence (**G-KL**, **GMM-KL**) [19] and *product probability kernels* (**G-PPK**, **GMM-PPK**) [11] with $\alpha = 0.5$ between the groups (therefore they are actually the *Bhattacharyya Coefficients* between Gaussians). Tuning the number of GMM components for each group is not feasible, so we always use 3 components. GMM-KL has no analytic form, so we use a *Monte Carlo* approximation with 500 samples.

BOW kernels To convert BOF to BOW, we quantize the feature to "words," and then compute the histogram of words for each group. The *chi-square distance* between these **BOW** histograms is used to construct the Gaussian kernel. The histograms can be further processed by **PLSA** [10] and then used in kernels based on Euclidean distance.

Pyramid matching kernel We also use the vocabulary-guided pyramid matching kernel (**PMK**) [7]; this variant performs better for high-dimensional data. We use the authors' implementation *libpmk*[1] with the suggested parameters.

We use *LibSVM* [3]'s multi-class SVM for classification. All kernel matrices are projected to be symmetric PSD as in Section 4.1 before use. The penalty to points within the margin C is chosen from $\{2^{-9}, 2^{-6}, \cdots, 2^{18}\}$. For PPK and PMK, we use their kernel values directly. For other kernels, we use Gaussian kernels $\exp\left(-\frac{1}{2}\mu^2/\sigma^2\right)$, where μ is the divergence/distance estimate. The kernel width σ is chosen from $\sigma_0 \times \{2^{-4}, 2^{-2}, \cdots, 2^{10}\}$, where σ_0 is the mean of the pairwise divergences. C and (when used) σ are chosen through joint 3-fold cross-validation on the training set.

For the image experiments, we extract features as follows unless indicated otherwise. The BOF representation we use is based on the *dense* SIFT descriptors. We put a regular 2D grid with step size 10 on each image, and compute SIFT descriptors on each grid node. These descriptors are 128-dimensional. In an attempt for scale invariance, we usually compute three SIFT descriptors with bin sizes of $\{6, 9, 12\}$ pixels at each point. After the feature extraction, each image is represented by a variable number of 128-dimensional feature vectors. Following [2], we can also include color information in the SIFT features by converting the images to HSV color space and separately extracting SIFT features from each color channel. Then SIFT features with the same location and bin size are concatenated together to construct the more descriptive "color SIFT" feature with dimensionality 384. Finally, we use PCA to reduce the feature vectors' dimensionality. Our implementation uses the PHOW function of the VLFEAT package [30] for feature extraction.

For BOW, these SIFT vectors are quantized by *K-means* into *visual words*, for which the vocabulary size (number of clusters) is 1000 for color images and 500 for grayscale images. The number of PLSA topics is 25, as in [2]. Following common practice in computer vision, the visual words are based on the original (uncompressed) feature vectors. Therefore the BOW methods do not compare to BOF kernels directly, as they are based on different features. In comparison, BOW loses information in the discretization step, while BOF kernels lose information when the feature dimension is reduced. We will show that our non-parametric kernels outperform BOW in most cases, perhaps indicating that less information is lost in PCA than in quantization.

We report kernel matrix construction times using 40 cores of a machine with four 12-core 2.3 GHz Opteron K10.5 processors. In this high-dimensional setting, k-d trees are ineffective, so we use simple brute-force search. Established techniques for approximate k-NN should result in significant speedups with limited loss of performance. In

[1] `people.csail.mit.edu/jjl/libpmk`

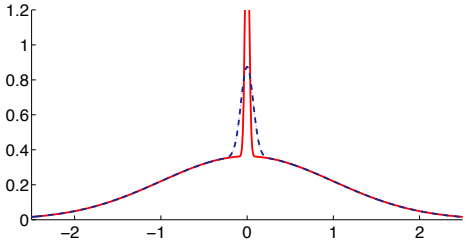

Figure 1: Densities of the two one-dimensional mixtures.

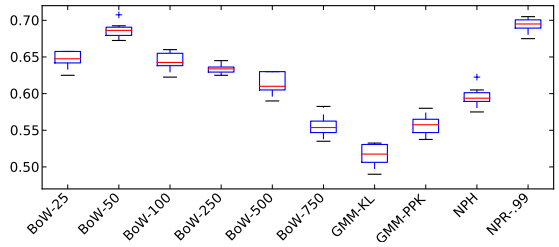

Figure 2: 1D mixture classification accuracies.

each case, we estimated divergences for the Hellinger distance and Rényi-α divergence with 20 values of α: -1, -.5, -.2, .1, .2, .3, . . . , .9, .99, 1.01, 1.1, 1.2, 1.3, 1.4, 1.5, and 2.

5.1. Artificial Gaussian Mixture Classification

We first compare the proposed kernels to others on artificial problems, to demonstrate two advantages of our kernel: its relatively few parameters requiring fine-tuning and its effectiveness in high-dimensional problems.

Consider the problem of distinguishing between the two Gaussian mixtures illustrated in Figure 1. The two mixtures each have a standard normal distribution with mixture coefficient $\frac{10}{11}$; the two classes are distinguished by the variance of the other component, which can be either .005 or .0005. Our task is to learn a classifier which can distinguish samples of size 30 from these two mixtures. (Although most feature sets will have substantially more than 30 data points for a real-world image, having a low number of sample points parallels having a moderate number of sample points in a high-dimensional space.) Note that this problem is quite difficult, as the expected number of samples from the distinguishing mixture is below 3.

Figure 2 shows accuracies from 8 runs of 10-fold cross-validation accuracies for several kernels on a dataset consisting of 200 samples from each mixture. The BOW method with codebook size K is denoted by BOW-K. The classification performance obtained by the Bayes-optimal classifier that chooses which mixture had a higher likelihood of generating the sample is 75%. The BOW kernel performs at its best only for codebook size 50; smaller and larger sizes both perform worse, some of them considerably so. In contrast, the proposed NPR and NPH methods perform well with minimal parameter selection, though it seems the Rényi divergence is better for this problem than the Hellinger.

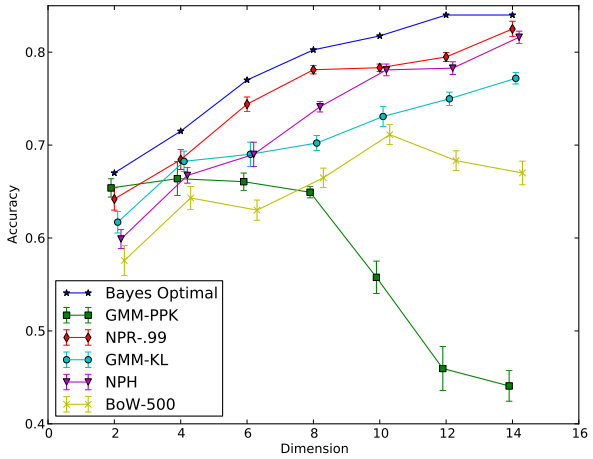

Figure 3: Mean and standard deviation accuracies on the high-dimensional artificial dataset.

We also show that our proposed kernel is capable of scaling up to higher-dimensional problems with small sample sets. This problem is similar, but the samples are of size 15 in $\mathbb{R}^d$. The common Gaussian has diagonal components 1 and off-diagonal components 0.2, while the distinguishing Gaussian has covariance matrix equal to either I_d or $I_d/2$, where I_d stands for the d-dimensional identity matrix. Each component has mean zero and mixture coefficient $1/2$. The distributions are more distinguishable in higher dimensions, as the components overlap less.

The results of 16 runs of 10-fold cross-validation for several kernels, as well as that of the Bayes-optimal classifier, are shown in Figure 3. The proposed NPR method outperformed its competitors in this experiment, and indeed achieved near-optimal results for all ds. BOW-500 is the only BOW method shown, but other codebook sizes performed similarly. The dimensionality at which performance peaked varied with the codebook size, so that *e.g.* BOW-100 peaked at dimension 8, and BOW-1000 at 14.

5.2. Object Classification

In the following sections we compare the performances of various kernels on real-world image datasets. We first examine object classification in the ETH-80 [13] dataset. This dataset contains 8 categories of objects; each category has 10 different objects, and each object has 41 images from different view angles. Following [8], we use a subset of 400 images for the experiment, selecting 5 images per object that capture its appearance from different angles. Sample images of two objects are shown in Figure 4. Our goal is to classify these objects into the 8 categories.

Figure 4: Images of two objects from the ETH-80 dataset. Each object has 5 different views.

For this dataset, we extract the color SIFT features with bin size fixed at 6 pixels, as scale invariance is not necessary for this problem. We then reduce the SIFT features to 18 dimensions using PCA, preserving 50% of variance. Each image is then represented by 576 18-dimensional points. Constructing our proposed kernels took 47 seconds.

We report the performance of 16 random runs of 2-fold cross-validation in Figure 5. We can see that our Rényi-divergence kernels perform better than BOW, and much better than the other methods. We note that BOW achieved impressive results only when properly tuned, as in the simulation study of Section 5.1. The improvement of NPR-0.9 (mean accuracy 90.9%) over BOW (88.3%) is statistically significant: a *paired t-test* shows a p-value below 10^{-3}. It is also interesting to see that GMM-based methods perform worse than simple Gaussian-based methods. This may be because it is harder to choose the parameters of a GMM, or because divergences between GMMs could not be obtained precisely; both of those problems are infeasible to remedy. PMK is not very accurate here, though fast to compute.

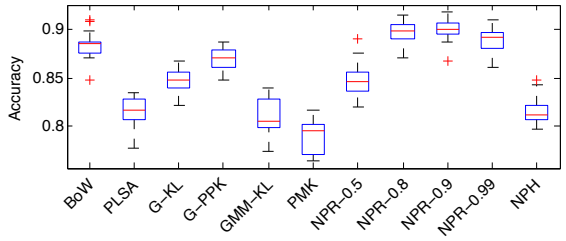

Figure 5: Classification accuracies on ETH-80.

Figure 6 shows the performance of the Rényi-α kernel for many values of α, along with the Hellinger performance for context. The best α values are clearly near 1, *i.e.* near the KL divergence, though performance seems to degrade faster when greater than 1 than when below.

5.3. Scene Classification

Scene classification using BOF/BOW representations is a well-studied problem for which many methods have been proposed (*e.g.* [5, 2, 24]). Here we test the performance of our non-parametric kernels against state-of-the-art methods.

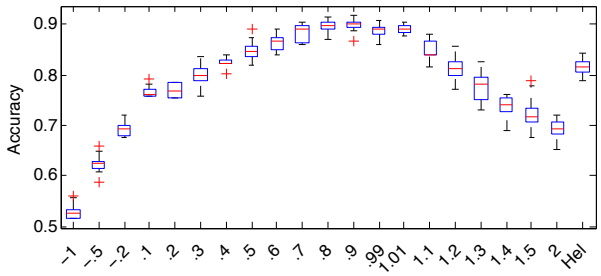

Figure 6: Classification accuracies on ETH-80 with Rényi-α for twenty αs, as well as Hellinger distance.

We use the OT dataset from [21], which contains 8 outdoor scene categories: *coast, forest, highway, inside city, mountain, open country, street*, and *tall building*. There are 2688 images in total, each about 256×256 pixels. Sample images are shown in Figure 7. The goal is to classify test images into one of the 8 categories.

Figure 7: Images from the 8 OT scene categories.

First, we use the grayscale versions of the OT images. The SIFT features are reduced to 19 dimensions using PCA preserving 70% of the variance, so that a typical image is represented by 1542 18-dimensional points. Constructing the proposed kernels took 14,195 seconds (just under 4 hours). The accuracies of 16 random 2-fold cross-validations are shown in Figure 8. The results here are very similar to those in the ETH-80 experiments. Our Rényi-divergence kernels still achieve the best overall accuracy, reaching 88.8% when $\alpha = 0.9$. It outperformed the BOW kernel, which is also very accurate (88.5%); a paired t-test shows a p-value below 0.03.

There are many methods for enhancing the BOF representation. For example, we can use color information as mentioned before. It is also possible to incorporate spatial information into the BOF representation. In the original BOF, an image is characterized by the distribution of its local features. By concatenating the x and y coordinates of each patch with the local feature vectors, these sets of new features allow us to cope with the joint distribution of local appearances and their locations in the image. Another

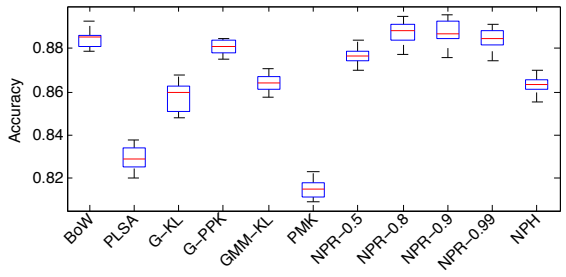

Figure 8: Accuracies on the OT dataset.

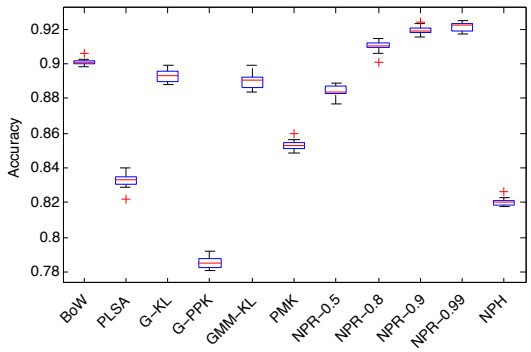

Figure 9: The OT dataset with color and spatial information.

Figure 10: Images from the 8 sports.

way is to include larger image regions into the feature set, so that co-occurrences of local objects can be captured at larger scales; for example, using a large enough SIFT descriptor, we are able to capture global concepts like "the top half of the image is mostly blank and the bottom half contains a lot of horizontal lines."

We combine the above approaches to increase the classification accuracy on the OT dataset. We extract color SIFT features with the bin sizes of $\{6, 12, 18, 24, 30\}$, to capture large-scale aspects of the images. A typical image then contains 1815 local features. The 384-dimensional color SIFT features are reduced by PCA to 53 dimensions, preserving 70% of the variance. Then the y coordinates of patches are appended to the feature vectors. The x coordinates are omitted, because in these scene images the horizontal location of objects usually carries little information. Finally, each feature dimension is normalized to zero mean and unit variance. Kernel construction on these larger, higher-dimensional features took 283,599 seconds (3.3 days).

The accuracies of 16 random runs are shown in Figure 9. Here we use 10-fold cross-validation, so we can directly compare to other published results. We can see adding the extra information greatly increased classification accuracies. NPR-0.99 achieved the best mean accuracy of 92.1%, much better than BOW's 90.1% (paired t-test $p < 10^{-13}$). Notably, this 92.1% accuracy (std dev .2%) surpasses the best previous result of which we are aware, 91.57% [25]. For comparison, the mean 2-fold cross-validation accuracies of NPR-0.99 and BOW are 90.7% and 88.8% respectively. GMM-PPK is not shown because it is too low.

5.4. Sport Event Classification

The BOF kernels can also be used for visual event classification [16] in the same manner as for scene classification. We use the dataset from [16], which contains Internet images of 8 sport event categories: *badminton, bocce, croquet, polo, rock climbing, rowing, sailing,* and *snowboarding*. This dataset is considered more difficult than traditional scene classification, as it involves much more widely varying foreground activity than does *e.g.* the OT dataset.

We use the first 130 images from each category, as in [16]. We use color SIFT features with dimensionality reduced to 57, and add spatial information in the form the patches' x and y coordinates. As image sizes vary, each BOF group contains 295 to 1542 vectors. Constructing our proposed kernels took 9,327 seconds (2.5 hours).

Figure 11 shows the accuracies of 16 random 2-fold cross-validations. We again see the kernel based on the Rényi-.9 divergence achieve the best accuracy of 87.1% (std dev .4%). This performance is at the same level as state-of-the-art methods such as [32], which attained 86.7%. It is worth noting that we used only PCA SIFT without further feature learning, as opposed to other methods which achieved significant performance increases by learning features. Compared to previous results, we can see that the performance of PPK methods decreased; we did not show GMM-PPK here because its accuracy is too low. The BOW method, though worse than Rényi-.9 with 83.5% (p below 10^{-8}), again performs well, showing its wide applicability.

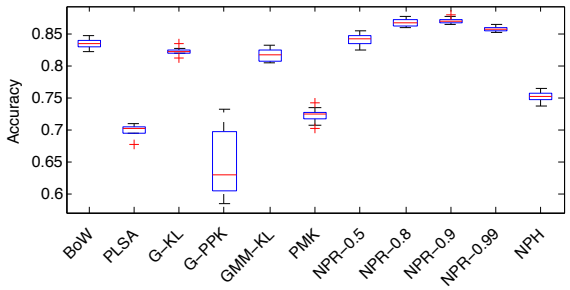

Figure 11: Classification accuracy on the sport dataset.

6. Discussion and Conclusion

In this paper we proposed a new method for image classification. We defined new kernels on sets of features and used consistent nonparametric divergence estimators for estimating the kernel values. Our goal was not to introduce new features; instead we were interested in improving the performance of traditional bag of features image representations through better dissimilarity measures.

Parametric methods for divergence estimation are usually biased, since the true distributions may not belong to assumed parametric families. Our nonparametric divergence estimator, however, is asymptotically unbiased. It is also easy to compute, requiring only certain k-NN distances.

For bag-of-words methods, setting the appropriate codebook size is a difficult model selection problem. It is similarly unknown how to choose the bin sizes for histogram-based methods. Our algorithm has comparably fewer parameters to tune, and avoids the inherent approximations of histograms, quantization, and clustering, which can lead to loss of information and decreased performance.

In our experiments, we demonstrated that the proposed method can outperform its state-of-the-art competitors on several challenging datasets, both artificial and real.

References

[1] Y. Bai, L. Guo, L. Jin, and Q. Huang. A novel feature extraction method using pyramid histogram of orientation gradients for smile recognition. In *ICIP*, 2009.

[2] A. Bosch, A. Zisserman, and X. Munoz. Scene classification using a hybrid generative/discriminative approach. *IEEE Trans. PAMI*, 30(4), 2008.

[3] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, pages 886–893, 2005.

[5] L. Fei-Fei and P. Perona. A Bayesian heirarcical model for learning natural scene categories. In *CVPR*, 2005.

[6] M. Goria, N. Leonenko, V. Mergel, and N. Inverardi. A new class of random vector entropy estimators and its applications in testing statistical hypotheses. *Journal of Nonparametric Statistics*, 17:277–297, 2005.

[7] K. Grauman and T. Darrell. Approximate correspondences in high dimensions. In *NIPS*, 2006.

[8] K. Grauman and T. Darrell. The pyramid match kernel: Efficient learning with sets of features. *Journal of Machine Learning Research*, 8:725–760, 2007.

[9] N. J. Higham. Computing the Nearest Correlation Matrix a Problem From Finance. *IMA Journal of Nummerical Analysis*, pages 329–343, 2002.

[10] T. Hofmann. Probabilistic latent semantic analysis. In *Uncertainty in Artificial Intelligence*, 1999.

[11] T. Jebara, R. Kondor, A. Howard, K. Bennett, and N. Cesabianchi. Probability product kernels. *Journal of Machine Learning Research*, 5:819–844, 2004.

[12] R. Kondor and T. Jebara. A kernel between sets of vectors. In *ICML*, 2003.

[13] B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. In *CVPR*, 2003.

[14] N. Leonenko, L. Pronzato, and V. Savani. A class of Rényi information estimators for multidimensional densities. *Annals of Statistics*, 36(5):2153–2182, 2008.

[15] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *IJCV*, 43:29–44, 2001.

[16] L.-J. Li and L. Fei-Fei. What, where and who? Classifying events by scene and object recognition. In *ICCV*, 2007.

[17] D. Lowe. Object Recognition from Local Scale-Invariant Features. In *ICCV*, volume 2, pages 1150–1157, 1999.

[18] R. Luss and A. d'Aspremont. Support vector machine classification with indefinite kernels. *Mathematical Programming Computation*, 1(2-3):97–118, 2009.

[19] P. J. Moreno, P. P. Ho, and N. Vasconcelos. A Kullback-Leibler divergence based kernel for SVM classification in multimedia applications. In *NIPS*, 2004.

[20] X. Nguyen, M. Wainwright, and M. Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Trans. Information Theory*, 2010.

[21] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3), 2001.

[22] B. Póczos and J. Schneider. On the estimation of α-divergences. In *AISTATS*, 2011.

[23] B. Póczos, L. Xiong, and J. Schneider. Nonparametric divergence estimation with applications to machine learning on distributions. In *UAI*, 2011.

[24] J. Qin and N. H. Yung. Scene categorization via contextual visual words. *Pattern Recognition*, 43, 2010.

[25] J. Qin and N. H. Yung. SIFT and color feature fusion using localized maximum-margin learning for scene classification. In *International Conference on Machine Vision*, 2010.

[26] B. Schölkopf and A. Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. The MIT Press, 2002.

[27] D. Scott. On optimal and data-based histograms. *Biometrika*, 66:605–610, 1979.

[28] A. Smola, A. Gretton, L. Song, and B. Schlkopf. A Hilbert space embedding for distributions. In *International Conference on Algorithmic Learning Theory (ALT)*, 2007.

[29] K. Sricharan, R. Raich, and A. Hero. Empirical estimation of entropy functionals with confidence. Technical Report, `arxiv.org/abs/1012.4188`, 2010.

[30] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. `www.vlfeat.org`, 2008.

[31] Q. Wang, S. R. Kulkarni, and S. Verdú. Divergence estimation for multidimensional densities via k-nearest-neighbor distances. *IEEE Trans. Information Theory*, 55(5), 2009.

[32] C. Zhang, J. Liu, Q. Tian, C. Xu, H. Lu, and S. Ma. Image classification by non-negative sparse coding, low-rank and sparse decomposition. In *CVPR*, 2011.

A. Supplementary Material

A.1. k-NN Based Density Estimators

In the consistency proofs we will use a few basic properties of k-NN density estimators; we review them here. [2] defines the k-NN based density estimators of p and q at X_i as

$$\hat{p}_k(X_i) = k/((n-1)\,\bar{c}\,\rho_k^d(i)) \tag{6}$$

$$\hat{q}_k(X_i) = k/(m\,\bar{c}\,\nu_k^d(i)). \tag{7}$$

These density estimators are consistent only when $k = k(n) \to \infty$, and $k = k(m) \to \infty$. We will use these estimators below in our divergence estimators; however, we will keep k fixed and will still be able to prove their consistency. The following theorems are well-known about the consistency of k-NN density estimators [2].

Theorem 3 (convergence in probability) *If $k(n)$ denotes the number of neighbors applied at sample size n, $\lim_{n\to\infty} k(n) = \infty$, and $\lim_{n\to\infty} n/k(n) = \infty$, then $\hat{p}_{k(n)}(x) \to_p p(x)$ for almost all x.*

Theorem 4 (convergence in sup norm) *Assume that $\lim_{n\to\infty} k(n)/\log(n) = \infty$ and $\lim_{n\to\infty} n/k(n) = \infty$. Then $\lim_{n\to\infty} \sup_x |\hat{p}_{k(n)}(x) - p(x)| = 0$ almost surely.*

A.2. Proof Outline of Theorems 1–2

To prove Theorems 1-2, we can repeat the argument of [3]. That paper was interested only in estimating the α-divergence, but we can use similar tools to prove the consistency of the more general $\widehat{D}_{\alpha,\beta}(X_{1:n}\|Y_{1:m})$ estimator. If we simply plugged $\hat{p}_k(X_i)$ and $\hat{q}_k(X_i)$ into (2), then we could estimate $D_{\alpha,\beta}(p\|q)$ with

$$\frac{1}{n}\sum_{i=1}^{n} \frac{k^{\alpha+\beta}}{\bar{c}^{\alpha+\beta}}(n-1)^{-\alpha} m^{-\beta} \rho_k^{-d\alpha}(i)\nu_k^{-d\beta}(i). \tag{8}$$

One can prove that this estimator is asymptotically biased for any fixed k. Let $\bar{c}$ denote the volume of a d-dimensional unit ball. Using the same tools as in [3], we will see that by introducing the multiplicative term $B_{k,\alpha,\beta} \doteq \bar{c}^{-\alpha-\beta}\frac{\Gamma(k)^2}{\Gamma(k-\alpha)\Gamma(k-\beta)}$, the following estimator is L_2 consistent under certain conditions:

$$\widehat{D}_{\alpha,\beta} = \frac{1}{n}\sum_{i=1}^{n}(n-1)^{-\alpha} m^{-\beta} \rho_k^{-d\alpha}(i)\nu_k^{-d\beta}(i) B_{k,\alpha,\beta}. \tag{9}$$

The Lebesgue lemma states that any function in $L_1(\mathbb{R}^d)$ restricted to a very small ball approximately looks like a constant function.

Lemma 5 (Lebesgue, 1910) *If $g \in L_1(\mathbb{R}^d)$, then for any sequence of open balls $\mathcal{B}(x, R_n)$ with radius $R_n \to 0$, and for almost all $x \in \mathbb{R}^d$,*

$$\lim_{n\to\infty} \frac{\int_{\mathcal{B}(x,R_n)} g(t)\mathrm{d}t}{\mathcal{V}\big(\mathcal{B}(x,R_n)\big)} = g(x). \tag{10}$$

Note that the estimation of $1/p(x)$ is simply $n\bar{c}\rho_k^d(x)/k$ with k-NN plug-in density estimators. Using Lemma 5, it is easy to prove that the distribution of $n\bar{c}\rho_k^d(x)$ converges weakly to an Erlang distribution with mean $k/p(x)$, and variance $k/p^2(x)$. For the details, see *e.g.* [1].

Now, if we divide $n\bar{c}\rho_k^d(x)$ by k, then asymptotically it has mean $1/p(x)$ and variance $1/(kp^2(x))$. Therefore (in accordance with Theorems 3–4) k should indeed diverge to infinity in order to get a consistent estimator; otherwise, the variance will not disappear. On the other hand, k cannot grow too fast: if, say, $k(n) = n$, then the estimator would be simply $\bar{c}\rho_k^d(x)$. This is a useless estimator since it is asymptotically zero whenever $x \in \mathrm{supp}(p)$.

Fortunately, in our case we do not need to apply consistent density estimators. The trick is that (2) has a special form: $\int p(x)p^\alpha(x)q^\beta(x)\mathrm{d}x$. In $\widehat{D}_{\alpha,\beta}$ this is estimated by

$$\frac{1}{n}\sum_{i=1}^{n} \left(\hat{p}_k(X_i)\right)^\alpha \left(\hat{q}_k(X_i)\right)^\beta B_{k,\alpha,\beta}, \tag{11}$$

where $B_{k,\alpha,\beta}$ is a correction factor that ensures asymptotic unbiasedness. Using Lemma 5 again, we can prove that the distributions of $\hat{p}_k(X_i)$ and $\hat{q}_k(X_i)$ converge weakly to the Erlang distribution with means $k/p(X_i)$, $k/q(X_i)$ and variances $k/p^2(X_i)$, $k/q^2(X_i)$, respectively [1]. Furthermore, they are conditionally independent for a given X_i. Therefore, "in the limit" (11) is simply the empirical average of the products of the $(-\alpha)$th (and $(-\beta)$th) powers of independent Erlang distributed variables. These moments can be calculated in closed form: the γth moments of an Erlang distribution is $\lambda^{-\gamma}\frac{\Gamma(k+\gamma)}{\Gamma(k)}$, where k, and $1/\lambda$ are the shape and scale parameters, respectively. For a fixed k, the k-NN density estimator is not consistent since its variance does not vanish. In our case, however, this variance will disappear thanks to the empirical average in (11) and the law of large numbers.

While the underlying ideas of this proof are simple, there are a couple of serious gaps in it. Most importantly, from Lemma 5 we can guarantee only the weak convergence of $\hat{p}_k(X_i)$, $\hat{q}_k(X_i)$ to the Erlang distribution. From this weak convergence we cannot imply that the moments of the random variables converge too. To handle this issue, we will need stronger tools such as the concept of asymptotically uniformly integrable random variables [4], and we also need the uniform generalization of Lemma 5. As a result, we need to put some extra conditions on the densities p and q

in Theorems 1–2. The details follow from a slight generalization of the derivations in [3].

References

[1] N. Leonenko, L. Pronzato, and V. Savani. A class of Rényi information estimators for multidimensional densities. *Annals of Statistics*, 36(5):2153–2182, 2008.

[2] D. Loftsgaarden and C. Quesenberry. A nonparametric estimate of a multivariate density function. *Annals of Mathematical Statistics*, 36:1049–1051, 1965.

[3] B. Póczos and J. Schneider. On the estimation of α-divergences. In *AISTATS*, 2011.

[4] A. van der Wart. *Asymptotic Statistics*. Cambridge University Press, 2007.