Optimizing Explicit Feature Maps on Intervals

Ondřej Chum

Visual Recognition Group
Faculty of Electrical Engineering
Czech Technical University in Prague
chum@cmp.felk.cvut.cz

Abstract

Approximating non-linear kernels by finite-dimensional feature maps is a popular approach for accelerating training and evaluation of support vector machines or to encode information into efficient match kernels. We propose a novel method of data independent construction of low-dimensional feature maps. The problem is formulated as a linear program that jointly considers two competing objectives: the quality of the approximation and the dimensionality of the feature map.

For both shift-invariant and homogeneous kernels the proposed method achieves better approximation at the same dimensionality or comparable approximations at lower dimensionality of the feature map compared with state-of-the-art methods.

Keywords: explicit feature maps, shift-invariant kernels, homogeneous kernels, linear programming

1. Introduction

Kernel machines, such as support vector machines (SVMs), can approximate any function or decision boundary arbitrarily well when provided with enough training data. However, such methods scale poorly with the size of the training set. On the other hand, it was shown [1] that linear SVMs can be trained in linear time with the number of training examples, which allows its application to very large datasets. Approximate embeddings, or feature maps, can preserve the accuracy of kernel methods and enable scaling to large datasets at the same time.

The demand for linear approximations of non-linear kernel functions is not limited to SVM classification. The idea of efficient match kernels [2] has been used in various areas of computer vision. Example applications relying on linear approximations of non-linear kernels include image matching and retrieval. Kernel description is used for interest points [3, 4], and for encoding dominant orientations of keypoints [5] into aggregated image descriptors, such as VLAD [6] or Fisher vectors [7].

Formally, for a positive definite kernel [8] $K : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ there exists a Hilbert space $\mathcal{H}$ and a mapping $\Psi : \mathbb{R}^n \rightarrow \mathcal{H}$ so that $K(\mathbf{x}, \mathbf{y}) = \langle \Psi(\mathbf{x}), \Psi(\mathbf{y}) \rangle_{\mathcal{H}}$, where $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ is a scalar product in $\mathcal{H}$. We propose a new method to estimate a low-dimensional mapping $\hat{\Psi} : \mathbb{R}^n \rightarrow \mathbb{R}^D$ so that $\hat{\Psi}(x)^\top \hat{\Psi}(y) \approx \langle \Psi(\mathbf{x}), \Psi(\mathbf{y}) \rangle_{\mathcal{H}}$. Naturally, a necessary property of the approximation is that minimal error is introduced. From the computational perspective, the dimensionality of the approximate feature should be minimal, too. These two criteria clearly compete. We propose an objective that trades off both criteria. The objective is relaxed into a linear program and can be optimized efficiently.

1.1. Related work

We briefly review the most relevant work on data independent (no training data needed) methods of kernel approximation. Random Fourier features were introduced by Rahimi and Recht in [9]. The feature map is a Monte Carlo approximation of the kernel where each dimension of the feature map is a cosine function drawn from the distribution given by the spectrum of the kernel signature. The Monte Carlo approach requires a relatively high number of samples to provide an accurate approximation, however, unlike most approaches, it is directly applicable to high-dimensional input data. The idea has been extended from shift-invariant kernels to skewed multiplicative histogram kernels in [10]. Maji and Berg [11] approximate the intersection kernel by a sparse feature map in closed-form. In [12] high dimensional sparse feature maps are derived and their relation to product quantisation is shown. Vedaldi and Zisserman [13] introduced a generalization of explicit feature maps to the family of additive homogeneous kernels. The proposed method achieves a considerably better approximation of feature maps of the same dimensionality or an equally good approximation of lower dimensionality of the feature maps compared to the results of [13], which are implemented in [14]. We also show that the proposed method allows for the optimization of meaningful errors measured on the homogeneous kernel output, rather than solely approximating the kernel signature. This article is an extended version of [15].

2. Problem formulation

In this section, the problem of shift-invariant kernel approximation is outlined, and then the proposed approach is described. For now, we will focus only on one dimensional kernels $K(x, y) : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$. Kernels in more dimensions are discussed in section 6.

Consider a family of shift-invariant (or stationary) kernels

$$K(x, y) = K(x + c, y + c) \qquad \forall x, y, c \in \mathbb{R}. \tag{1}$$

A signature $k(\lambda) : \mathbb{R} \to \mathbb{R}$ of a shift invariant kernel K is defined as $k(\lambda) = K(-\lambda/2, \lambda/2)$, which fully specifies the kernel, since $K(x, y) = k(x - y)$.

We will study approximations $\hat{K} : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ to shift-invariant kernels; in particular, those that can be written as the inner product of low-dimensional feature maps $\hat{\Psi} : \mathbb{R} \to \mathbb{R}^D$

$$\hat{K}(x, y) = \hat{\Psi}(x)^\top \hat{\Psi}(y) \approx K(x, y).$$

The kernel K will be approximated by approximating the kernel signature k by $\hat{k} : \mathbb{R} \to \mathbb{R}$ of the form

$$\hat{k}(\lambda) = \sum_{\omega \in \Omega} \alpha_\omega \cos(\omega\lambda), \tag{2}$$

where ω is a frequency, Ω is a finite set of frequencies $\Omega \subset [0, \omega^{\max}]$, and $\alpha_\omega \in \mathbb{R}_0^+$ are non-negative weights. Kernels of the form of (2) can be directly converted to feature maps

$$\hat{\Psi}_\omega(x) = \begin{pmatrix} \sqrt{\alpha_\omega} \cos(\omega x) \\ \sqrt{\alpha_\omega} \sin(\omega x) \end{pmatrix}. \tag{3}$$

From the identity

$$\cos(x - y) = \cos(x)\cos(y) + \sin(x)\sin(y)$$

it follows that

$$\hat{\Psi}_\omega(x)^\top \hat{\Psi}_\omega(y) = \alpha_\omega \cos(\omega(x - y)).$$

The feature map $\hat{\Psi}(x)$ defined by the signature $\hat{k}$ is a concatenation of $\hat{\Psi}_\omega$ for all $\omega \in \Omega$. The dimensionality $D(\hat{k})$ of the feature map $\hat{\Psi}(x)$ is

$$D(\hat{k}) = \sum_{\omega \in \Omega} \delta(\alpha_\omega) D_\omega, \tag{4}$$

where $\delta(\alpha_\omega) = 0$ for $\alpha_\omega = 0$, and $\delta(\alpha_\omega) = 1$ otherwise,

$$D_\omega = \begin{cases} 1 & \omega = 0 \\ 2 & \omega > 0. \end{cases}$$

Here, D_ω denotes the dimensionality of the feature map for a particular frequency ω. The value of $D_\omega = 1$ for $\omega = 0$ comes from the fact that $\hat{\Psi}_0 = \left(\sqrt{\alpha_\omega}, 0\right)^\top$, where the zero can be dropped from the embedding.

Input domain. The input x of the kernel function is typically some measurement, such as coordinates of a point in a canonical patch (of fixed size), the angle of dominant orientation, or an entry of a normalized histogram. We make the assumption that the measured features x come from a bounded interval $x \in [a, b]$. This assumption is natural for many practical problems. Given the properties of shift-invariant kernels, $x \in [a, b]$ implies that the kernel signature k needs to be approximated on the interval $[-M, M]$, $M = b - a$.

Error function. The similarity of the original signature function k and its approximation $\hat{k}$, is measured by an error function $C(k, \hat{k}) \in R_0^+$. In order to use discrete optimization methods, the error function used in the paper will only depend on a finite number of points z from an evaluation set Z, $z \in Z \subset [0, M]$. The points z are non-negative, as both k and $\hat{k}$ are symmetric. The discretization of the input domain is optimal for quantities that are discrete, such as for pixel coordinates. In many domains, a sufficiently fine discretization introduces negligible error compared to the error introduced by the measurement estimation, *e.g.*, the angle of the dominant orientation of a feature patch. If a continuous input domain is essential, then the number $|Z|$ of the points z has to be adjusted with respect to the maximal frequency $\omega^{\max}$ and the spectrum of the kernel signature k.

In the paper, the following two error functions will be used

$$C_1(k, \hat{k}) = \sum_{z \in Z} w(z) \cdot |k(z) - \hat{k}(z)|, \tag{5}$$

$$C_\infty(k, \hat{k}) = \max_{z \in Z} w(z) \cdot |k(z) - \hat{k}(z)|, \tag{6}$$

where $w(z) \in \mathbb{R}_0^+$ are weights that adjust the relative importance of the approximation error at point z. For all $w(z) = 1$ constant, (5) represents L_1 norm and (6) represents L_∞ norm.

2.1. Optimization

Two antagonistic objectives have to be considered in the approximation task: keeping the dimensionality $D(\hat{k})$ of the embedding low and obtaining the best possible approximation, as measured by $C(k, \hat{k})$, of the kernel as possible.

Since $D(\hat{k})$ is not convex and not continuous, we apply an LP relaxation [16] to make the optimization tractable. Instead of dealing with the dimensionality $D(\hat{k})$, which is a weighted L_0 norm (4), a weighted L_1 norm

$$\bar{D}(\hat{k}) = \sum_{\omega \in \Omega} D_\omega \alpha_\omega \tag{7}$$

is used. Recall that $\alpha_\omega \geq 0$.

The task of finding an approximation $\hat{k}$ that minimizes $\bar{D}(\hat{k})$ while preserving the closeness of the approximation is formulated as a linear program,

$$\min_{\hat{k}} \bar{D}(\hat{k}) \quad \text{subject to } C(k, \hat{k}) \leq C^{\max} \in \mathbb{R}^+.$$

The task of finding an approximation $\hat{k}$ of fixed dimensionality $D^{\max}$ of the feature map is sought while minimizing $C(k, \hat{k})$ is approximated by a linear program

$$\min_{\hat{k}} \bar{D}(\hat{k}) + \gamma C(k, \hat{k}),$$

where $\gamma \in \mathbb{R}^+$ is a constant controlling the trade-off between the quality of the approximation and the relaxed dimensionality $\bar{D}$ of the feature map. A version of binary search for the appropriate weight γ is used: the LP is executed with an initial value of γ. The output is checked for the value of D (not $\bar{D}$), if $D \leq D^{\max}$ the value of γ is increased (higher importance to the fit cost); otherwise, the value of γ is decreased (higher importance to the solution sparsity). The solution with the best fit is selected among the LP outputs with $D \geq D^{\max}$.

2.2. Alternative feature maps

Methods, such as [9], that adopt a Monte Carlo approximation to the kernel use a different feature map than (3). While (3) generates a two-dimensional feature map for each non-zero frequency $\omega \neq 0$, the alternative feature map adds only one-dimension with each frequency ω_i

$$\hat{\Psi}_i(x) = cos(\omega_i x + b_i), \tag{8}$$

where ω_i is drawn from the normalized spectrum of $k(\lambda)$, and b_i is drawn from a uniform distribution on the interval $[0, 2\pi]$. A dot product $\hat{\Psi}_i(x)\hat{\Psi}_i(y)$ of this feature map for x and y

$$cos(\omega_i x + b_i) \cdot cos(\omega_i y + b_i) = \frac{cos(\omega_i(x-y))}{2} + \frac{cos(2b_i + \omega_i(x+y))}{2} \tag{9}$$

is not shift-invariant, as it also depends on $x + y$ in addition to $x - y$. The feature map can be seen as a Monte-Carlo approximation: for a fixed ω_i, the second term of (9) is integrated out by a sufficient number of samples of b

$$\int_0^{2\pi} cos(2b + \omega_i(x+y)) \, \mathrm{d}b = 0.$$

In practice, with a finite number of samples, kernel approximations based on the feature map (8) are only approximately shift invariant. The feautre map (8) is appropriate for Monte-Carlo estimates, but not suitable for low-dimensional feature maps.

3. Periodic kernels

Let k be a kernel signature that is periodic with period $2M$. The task in this section is to approximate k on the interval $[-M, M]$, which is equivalent to approximating on $\mathbb{R}$ since k is periodic. The spectrum of k is restricted to harmonics of the base frequency π/M, and hence

$$\Omega_0 = \left\{ i\frac{\pi}{M} \;\middle|\; i \in \mathbb{N}_0 \right\}. \tag{10}$$

A standard approach to this problem is to project the function k to an orthogonal basis $\cos(i\pi/M\lambda)$. The function k is then approximated using basis functions with the highest values of the coefficients. Such an approach is efficient for one dimensional kernels, and the method proposed in this paper does not bring any contribution to this problem. Results for multiplicative kernels (Section 6) are applicable to periodic functions.

4. Aperiodic kernels

In this section, we derive an approximation of kernels on the interval $[-M, M]$ with signature that is aperiodic (or do not have period $2M$). Many shift invariant kernels, including the RBF kernel, are not periodic.

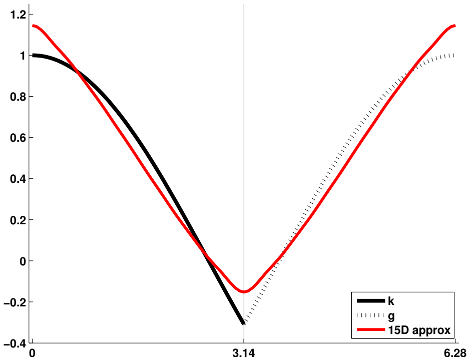


Figure 1: Kernel signature $k = \cos(0.6\lambda)$ (solid black curve) that is not periodic on the interval $[-\pi, \pi]$ is approximated via approximating periodic function g (solid black and dashed black) using only harmonic angular frequencies $\Omega = \mathbb{N}_0$. Frequencies with negative coefficients are truncated which leads to a poor approximation (red curve).

4.1. Discrete frequencies

Following [13], for an aperiodic kernel signature k, there is a function g with period $2M$ and $g(\lambda) = k(\lambda)$ for $\lambda \in [-M, M]$. Then approximating periodic g, as in the previous section, using harmonic frequencies Ω_0 (10) only, approximates k on $[-M, M]$.

This approach has two drawbacks: First, even though k has a non-negative spectrum due to Bochner's theorem, this does not hold for g. All frequencies with negative weights have to be left out [13]. As a consequence, the approximation of the signature function k cannot be arbitrarily precise, even for very high dimensional feature maps. Second, approximating g instead of k is not optimal with respect to the dimensionality of the feature map. To demonstrate these claims, consider the toy example in Figure 1. The kernel signature $k(\lambda) = \cos(0.6\lambda)$ approximated on $[-\pi, \pi]$ is not periodic with the period of 2π. The approximation of periodic g by harmonic frequencies $\omega \in \mathbb{N}_0$ with non-negative coefficients is not satisfactory. The exact feature map, originating from $\hat{k} = \cos(0.6\lambda)$, is two-dimensional, but the optimal frequency $\omega^* = 0.6 \notin \Omega = \mathbb{N}_0$[1].

A simple generalization of the above approach increases the number of possi-

[1] The problem can be alleviated by approximating the signature $k = \cos(0.6\lambda)$ on interval $[-5\pi, 5\pi]$. This toy example was selected as an extreme case to demonstrate the drawbacks of the standard approach.

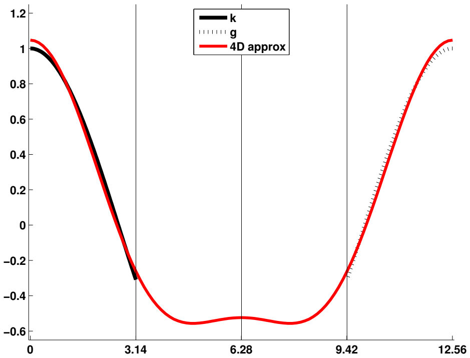


Figure 2: Approximating kernel signature $k = \cos(0.6\lambda)$ (solid black curve) on $[-\pi, \pi]$ via approximation of periodic function g (solid black and dashed black) with period 4π using harmonic frequencies $F = \{i/2 \mid i \in \mathbb{N}_0\}$. There are no constraints imposed on g on $(\pi, 3\pi)$. Approximation by 4D feature map drawn in red.

ble frequencies with increasing $j \in \mathbb{N}$

$$\Omega_j = \left\{ i\frac{\pi}{2^j M} \;\middle|\; i \in \mathbb{N}_0 \right\}. \tag{11}$$

Since $\Omega_j \subset \Omega_{j+1}$, approximation with frequencies Ω_{j+1} will not be worse than with Ω_j. With j approaching infinity, the set Ω_j will contain frequencies arbitrarily close to any real-valued frequency. However, sets Ω_j with large j are impractical in real problems. Sets Ω_j of practical use lead to better approximations than Ω_0, but still can only reach a discrete subset of possible frequencies.

Using the set of frequencies in (11) can be interpreted as approximating a periodic function $g(\lambda)$ with period of $2^{j+1}M$, where $g(\lambda) = k(\lambda)$ for $\lambda \in [-M, M]$, and no constraints imposed on $g(\lambda)$ in interval $\lambda \in (M, 2^j M)$. The situation is depicted in Figure 2 for $j = 1$.

4.2. Continuous frequencies

In the framework of discrete optimization used in this paper, the pool of frequencies Ω is required to be finite and, for practical reasons, not extremely large. To access any real frequency while preserving finite Ω, we will slightly modify the form of the approximation of the kernel signature $\hat{k}$ to

$$\hat{k}(\lambda) = \sum_{\omega \in \Omega} \alpha_\omega \cos((\omega + d_\omega)\lambda), \tag{12}$$

where

$$|d_\omega| \leq d^{\max} \tag{13}$$

is a small difference of the frequency. The differences d_ω are estimated jointly with the weights α_ω by the linear program. That is, instead of using exactly frequency ω in the approximation, any frequency within the interval $[\omega - d^{\max}, \omega + d^{\max}]$ can be used. The first order Taylor expansion of the cosine function in the frequency variable ω (not in λ) reads

$$\cos((\omega + d_\omega)\lambda) \approx \cos(\omega\lambda) - d_\omega \lambda \sin(\omega\lambda). \tag{14}$$

Such an approximation is good only in a small neighbourhood of ω, which is controlled by the size $d^{\max}$ of the“trust region” (13). By substituting (14) into (12), we obtain

$$\hat{k}(\lambda) = \sum_{\omega \in \Omega} \alpha_\omega \cos(\omega\lambda) - \sum_{\omega \in \Omega} d_\omega \alpha_\omega \lambda \sin(\omega\lambda). \tag{15}$$

By introducing an auxiliary variable $\beta_\omega = d_\omega \alpha_\omega$, equation (13) transforms to

$$|\beta_\omega| \leq \alpha_\omega d^{\max}. \tag{16}$$

Both (15) and (16) in variables (α_ω, β_ω) are in a form that can be written as a linear program. Compared to the original formulation, $|\Omega|$ variables β_ω, and $2|\Omega|$ constraints (16) were introduced to the linear program.

The advantages of the proposed approach are illustrated in Figure 3 on the 1D RBF kernel approximation. Compared with the classical approach of orthogonal projection onto the harmonic-frequencies cosine basis, the proposed method approximates the kernel signature exactly on the domain of interest, the interval $[-M, M]$, using no additional constraints. Further comparisons on 1D RBF kernel approximation can be found in Figure 4.

Implementation details. In our experiments, we first apply an approximation with a discrete set Ω of frequencies equally spaced in $[0, \omega^{\max}]$, with spacing at most $d^{\max} = 0.1$ as described in section 2.1. Then, an iterative process is performed. Each iteration alternates between the LP approximation, which uses the first order Taylor expansion formulation (12) and the frequency update

$$\begin{array}{rcl} d_\omega & = & \beta_\omega / \alpha_\omega \text{ ... compute } d\text{'s} \\ \omega & \leftarrow & \omega + d_\omega \text{ ... update frequencies} \\ d^{\max} & \leftarrow & d^{max}/2 \text{ ... reduce the max step.} \end{array}$$

The iteration is used to eliminate the approximation error introduced by the Taylor expansion. In each step, the allowed difference in frequency $d^{\max}$ is halved, which guarantees convergence.

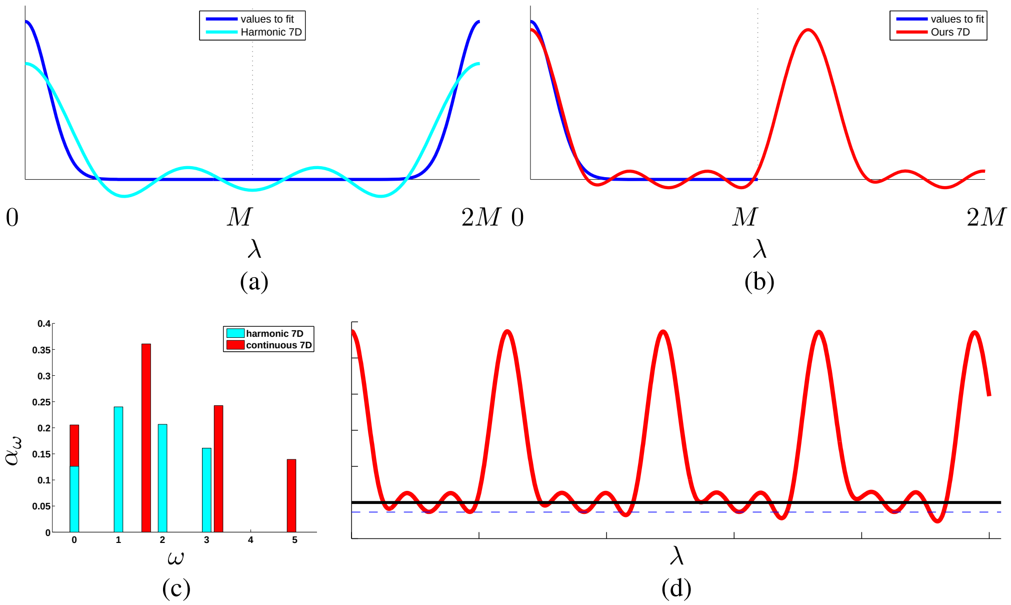


Figure 3: Comparison of the standard approximation using harmonic frequencies and the proposed method on a 1D RBF kernel: (a) harmonic frequencies result in a periodic function and thus implicitly enforce additional constraint on $[M, 2M]$, (b) result of the proposed method, (c) frequencies and weights used by the two methods, (d) the proposed method in general produces non-periodic function (neglecting the finite precision for the frequencies).

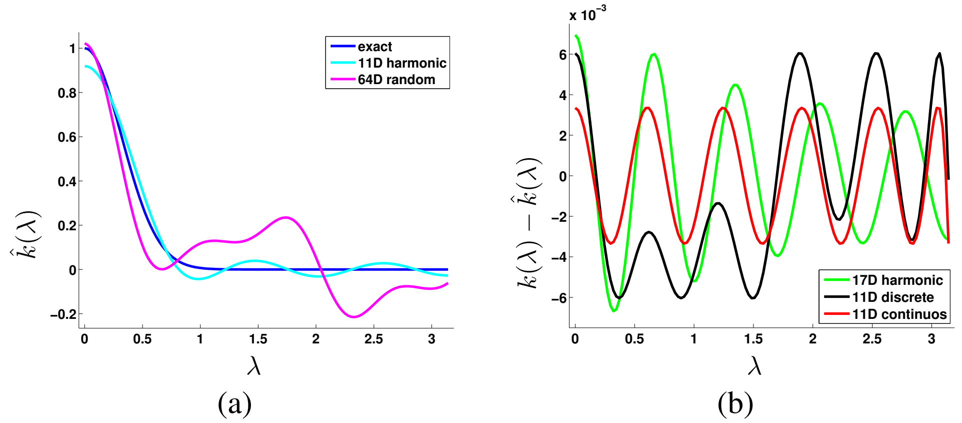


Figure 4: Approximation of a one-dimensional RBF kernel: (a) the shape of the approximate kernel signature for an 11D feature map via orthogonal projection onto angular harmonics and $\hat{\Psi}_{\mathrm{RF}}(0)^\top \hat{\Psi}_{\mathrm{RF}}(\lambda)$ for random feature maps [9], (b) approximation error for 17 dimensional feature map via orthogonal projection, and the proposed 11 dimensional feature map for the discrete and continuous methods respectively.

5. Homogeneous kernels

A homogeneous kernel is a positive definite kernel $K : \mathbb{R}_0^+ \times \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$ satisfying

$$K_h(cx, cy) = cK_h(x, y) \qquad \forall x, y, c \geq 0.$$

Following [13], by setting $c = \sqrt{xy}$, any homogeneous kernel can be decomposed as

$$\begin{aligned} K_h(x, y) &= \sqrt{xy} \cdot K_h\left(\sqrt{x/y}, \sqrt{y/x}\right) = \\ &= \sqrt{xy} \cdot k_h(\log y - \log x), \end{aligned} \tag{17}$$

where $k_h(\lambda)$ is a signature of K_h

$$k_h(\lambda) = K_h(e^{-\lambda/2}, e^{\lambda/2}).$$

The signature of the homogeneous kernel (17) resembles the signature of the shift-invariant kernel after transforming the input domain into log-space. The homogeneous kernel can be approximated [13] by approximating the signature $k_h(\lambda)$ with $\hat{k}(\lambda)$ in a manner similar to (2). The resulting feature map is

$$\hat{\Psi}_\omega(x) = \begin{pmatrix} \sqrt{\alpha_\omega x} \cdot \cos(\omega \log x) \\ \sqrt{\alpha_\omega x} \cdot \sin(\omega \log x) \end{pmatrix}.$$

While for shift-invariant kernels optimizing the signature approximation was equivalent to optimizing the kernel approximation, we show that for homogeneous kernels the situation is different. We will demonstrate it on the L_∞ error measure, since it is independent of the data distribution. Derivation for other error measures is straightforward.

We first derive minimization $\min_{\hat{k}} \varepsilon_A$ of the absolute L_∞ error,

$$\begin{aligned} \varepsilon_A &= \max_{x,y \in (0,b]} |K_h(x, y) - \hat{K}_h(x, y)| = \\ &= \max_{x,y \in (0,b]} \sqrt{xy} \cdot \left| k_h\left(\log \frac{y}{x}\right) - \hat{k}\left(\log \frac{y}{x}\right) \right|. \end{aligned} \tag{18}$$

Let $\lambda = \log y - \log x \geq 0$ (which is equivalent to $y \geq x$) without a loss of generality, as k_h is symmetric. The error ε_A can be written as

$$\begin{aligned} \varepsilon_A &= \max_{y \in (0,b], \lambda \geq 0} y e^{-\lambda/2} |k_h(\lambda) - \hat{k}(\lambda)| \\ &= b \cdot \max_{\lambda \geq 0} e^{-\lambda/2} |k_h(\lambda) - \hat{k}(\lambda)|. \end{aligned}$$

This is achieved by optimizing the approximation of signature k_h with weighted C_∞ error function (6) with weight

$$w_A(\lambda) = e^{-\lambda/2}. \tag{19}$$

Similarly, we derive the minimization $\min_{\hat{k}} \varepsilon_R$ of the relative L_∞ error

$$\begin{aligned} \varepsilon_R &= \max_{x,y\in(0,b]} \frac{|K_h(x,y) - \hat{K}_h(x,y)|}{K_h(x,y)} = \\ &= \max_{\lambda} \frac{1}{k_h(\lambda)} |k_h(\lambda) - \hat{k}(\lambda)|. \end{aligned} \tag{20}$$

Optimizing of L_∞ for the relative kernel error (20) is equivalent to optimizing the weighted C_∞ error of the kernel signature approximation with weight

$$w_R(\lambda) = \frac{1}{k_h(\lambda)}. \tag{21}$$

While w_A is decreasing, that is, the fit should be tighter for small λ, w_R is increasing and a better fit should be at the tail of the kernel signature.

To apply the proposed approximation method, we need to select the size of the interval $[-M, M]$, where the kernel signature should be approximated. The optimal choice is $M = \log(b/m)$, where b is the largest expected input value and m is the smallest non-zero input value. For instance, for histograms with 8-bit entries, such as SIFT [17], $M = \log(255/1)$.

6. Kernels in more dimensions

Measurements in many applications take the form of high-dimensional vectors $\mathbf{x}, \mathbf{y} \in \mathcal{X}^n$, where $\mathcal{X}$ is $\mathbb{R}$ or $\mathbb{R}_0^+$ depending on the type of the input data. We will use x^i to denote i-th component of vector $\mathbf{x}$. So far only one dimensional kernels have been considered. These kernels can be extended to higher-dimensional input by either additive or multiplicative combination.

Additive kernels are defined as

$$K(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} K_i(x^i, y^i).$$

In computer vision, the following homogeneous additive kernels are commonly used: χ^2, intersection, Hellinger, and Jensen-Shannon kernels [13]. The feature

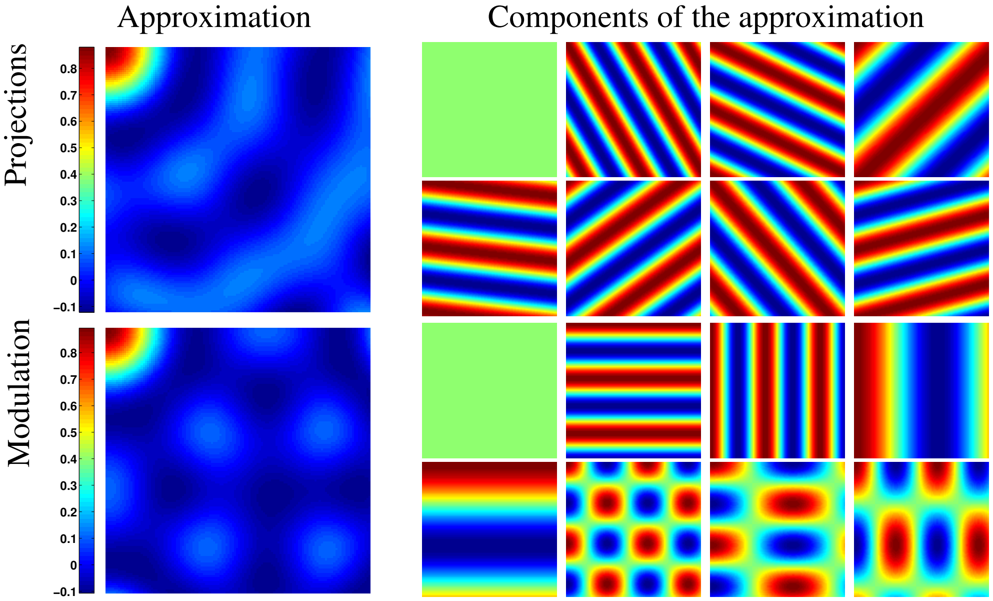


Figure 5: Example of the kernel signature approximation (first column) for a symmetric 2D RBF kernel by projections (top row) and by modulation (bottom row). Only one quadrant of the kernel signature is shown. The approximation results in 31D feature maps. Some of the components of the approximation, $\cos(\boldsymbol{\omega}^\top \mathbf{x})$ for projections and $\prod_i \cos(\omega^i x^i)$ for modulation, are shown on the right. The constant component adds one dimension to the feature maps. All other components of the projection approximation add two dimensions to the feature map. The components of the modulation approximation add 2 and 4 dimensions to the feature map for axis aligned 'waves', and 'blobs', respectively.

map for the additive kernel is a concatenation of feature maps for each dimension. The additive construction of the feature map increases the dimensionality D of the feature map linearly with the input dimension n, which is acceptable and we will not study the multi-dimensional additive feature maps further.

Multiplicative kernels, such as multi-dimensional RBF with diagonal Σ, can be written as

$$K(\mathbf{x}, \mathbf{y}) = \prod_{i=1}^{n} K_i(x^i, y^i).$$

The feature map is a tensor (Kronecker) product of the feature maps for each input dimension $\hat{\Psi}(\mathbf{x}) = \bigotimes \hat{\Psi}(x^i)$. Specifically, for two kernels K_1 and K_2 and input vectors $\mathbf{x} = (x^1, x^2)$, $\mathbf{y} = (y^1, y^2)$, we have

$$K(\mathbf{x}, \mathbf{y}) = K_1(x^1, y^1) \cdot K_2(x^2, y^2) \approx \left(\hat{\Psi}_{\Omega_1}(x^1) \otimes \hat{\Psi}_{\Omega_2}(x^2)\right)^\top \left(\hat{\Psi}_{\Omega_1}(y^1) \otimes \hat{\Psi}_{\Omega_2}(y^2)\right),$$

resulting in an explicit feature map

$$\hat{\Psi}_{\omega^1 \omega^2}(\mathbf{x}) = \hat{\Psi}_{\omega^1}(x^1) \otimes \hat{\Psi}_{\omega^2}(x^2) = \sqrt{\alpha_{\omega^1} \alpha_{\omega^2}} \begin{pmatrix} \cos(\omega^1 x^1) \cos(\omega^2 x^2) \\ \sin(\omega^1 x^1) \cos(\omega^2 x^2) \\ \cos(\omega^1 x^1) \sin(\omega^2 x^2) \\ \sin(\omega^1 x^1) \sin(\omega^2 x^2) \end{pmatrix}, \quad (22)$$

for every $\omega^1 \in \Omega_1$ and $\omega^2 \in \Omega_2$.

The construction of the multiplicative kernel, often called modulation, increases the dimensionality of the final feature map $\hat{\Psi}(\mathbf{x})$ exponentially with the number of input dimensions. Therefore, multiplicative kernels constructed by modulation are suitable only for low-dimensional input data. We will discuss multiplicative kernels in detail in section 6.2.

The proposed method for discrete optimization is not suitable for approximating kernels with high-dimensional input data. In practice, direct application is tractable for kernels up to 3 dimensions. Nevertheless, even with this limitation, there are practical applications that would benefit from our approach. Consider problems where low dimensional geometric data, such as the position of a point in a patch and the orientation of the gradient at that point, are to be encoded, *e.g.*, in interest point descriptors such as in [3] or [4]. Two different approaches using different forms of functions $\hat{k}$ approximating the kernel signature will be considered. The optimization formulation is essentially identical to the one-dimensional case, including the extension exploiting the entire continuous spectrum from section 4.2. The difference is in the construction of the frequency pool Ω and the discrete evaluation domain Z. The size of these sets is the bottleneck of the discrete optimization approach, as both sets grow fast with the increasing dimensionality n of the input data. Two different forms of functions $\hat{k}$ approximating the kernel signature will be considered. The approaches are compared in section 7.4.

6.1. Approximation by projections

A general method for direct approximation of the n-dimensional kernel signature uses form of the approximation $\hat{k}_P$ similar to [9]

$$\hat{k}_P(\boldsymbol{\lambda}) = \sum_{\boldsymbol{\omega}\in\Omega} \alpha_{\boldsymbol{\omega}} \cos(\boldsymbol{\omega}^\top \boldsymbol{\lambda}), \tag{23}$$

where $\boldsymbol{\lambda} = \mathbf{x} - \mathbf{y} \in \mathbb{R}^n$, $\Omega \subset \mathbb{R}^n$. Geometrically, $\hat{k}_P$ can be seen as projecting $\boldsymbol{\lambda}$ onto $\boldsymbol{\omega}$ and then encoding the projection by a cosine with frequency $\|\boldsymbol{\omega}\|$. Visualization for $n = 2$ is shown in Figure 5 (top row). Since (23) is only symmetric for each line passing through the origin, that is $\hat{k}_P(\boldsymbol{\lambda}) = \hat{k}_P(-\boldsymbol{\lambda})$, the evaluation set has to be constructed as $Z \subset \prod_{i=1}^{n-1}[-M_i, M_i] \times [0, M_n]$. The finite pool of frequencies is $\Omega \subset \mathbb{R}^n$. The projection method is capable of approximating multi-dimensional kernels, even those that are not multiplicative.

6.2. Approximation by modulation

For multiplicative kernels, the following form of $\hat{k}_M$ is useful

$$\hat{k}_M(\boldsymbol{\lambda}) = \sum_{\boldsymbol{\omega}\in\Omega} \alpha_{\boldsymbol{\omega}} \prod_{i}^{n} \cos(\omega^i \lambda^i). \tag{24}$$

The geometric interpretation of the form of $\hat{k}_M$ is shown in Figure 5 (bottom row). Since (24) is symmetric along all axes, that is

$$\hat{k}_M((\lambda^1, \ldots, \lambda^n)^\top) = \hat{k}_M((\pm\lambda^1, \ldots, \pm\lambda^n)^\top),$$

it is sufficient to optimize only in $Z \subset \prod_{i=1}^{n}[0, M_i]$. Let $\hat{\Psi}^i$ be a feature map optimized separately over the i-th dimension from frequencies Ω^i and corresponding weights α_{ω^i}, and let

$$\Omega_\otimes = \{\boldsymbol{\omega} = (\omega^1, , \ldots, \omega^n) \mid \omega^i \in \Omega^i\}, \tag{25}$$

$$\alpha_{\boldsymbol{\omega}} = \prod_{i=1}^{n} \alpha_{\omega^i}, \quad D_{\boldsymbol{\omega}} = \prod_{i=1}^{n} D_{\omega^i} \quad \text{for} \quad \boldsymbol{\omega} = (\omega^1, \ldots, \omega^n).$$

The feature map constructed from frequencies $\Omega_{\otimes}$ and weights $\alpha_{\boldsymbol{\omega}}$ is equivalent to $\hat{\Psi}(\mathbf{x}) = \bigotimes \hat{\Psi}(x^i)$.

The dimensionality of feature map $\hat{\Psi}$ can be reduced by dropping frequencies $\boldsymbol{\omega}$ with small coefficient $\alpha_{\boldsymbol{\omega}}$. Even though frequencies with small α_{ω^i} may still be important for approximation in dimension i, the product of such weights can exponentially reduce the impact. We refer to this greedy method as '$\bigotimes$ no LP' in the experiments. Better approximation results are achieved by executing the proposed LP optimization. Using the frequency pool $\Omega_{\otimes}$ (25) significantly increases the speed of the algorithm.

Note that frequencies from the modulation approach can be transformed into frequencies of the projection approach using identity

$$\cos(x)\cos(y) = \cos(x+y)/2 + \cos(x-y)/2. \tag{26}$$

However, while the left-hand side of the equation represents one entry in the frequency pool Ω for modulation, it generates two entries for the projection case. Overall, the projection approach is more general at the cost of larger LP problem (larger in both, the size of Ω and in the size of the evaluation set Z, as discussed in section 6.1). The visual difference of the approximation is shown in Figure 6 (top three plots).

Any multiplicative kernel can be expressed using either projections or modulations, see equations (25) and (26). However, the approximations delivered by both methods may not be multiplicative kernels, *i.e.* the kernels cannot be, in general, decomposed into a product of kernels acting on each dimension separately.

6.3. *Efficient approximation with L_∞ error*

In this section, an algorithm reducing the number of constraints in the linear program optimization is discussed. The approach is specific to the L_∞ error. The method is described for the multiplicative kernels using the modulation parametrization, but is applicable to any parametrization.

Minimizing the L_∞ error $C_\infty(k, \hat{k})$ (6) for n-dimensional kernel signature k is performed by a linear program that is minimizing the error ε subject to the constraints

$$|k(\boldsymbol{z}) - \hat{k}(\boldsymbol{z})| = \left| k(\boldsymbol{z}) - \sum_{\boldsymbol{\omega}\in\Omega} \alpha_{\boldsymbol{\omega}} \prod_{i}^{n} \cos(\omega^i z^i) \right| \leq \varepsilon \tag{27}$$

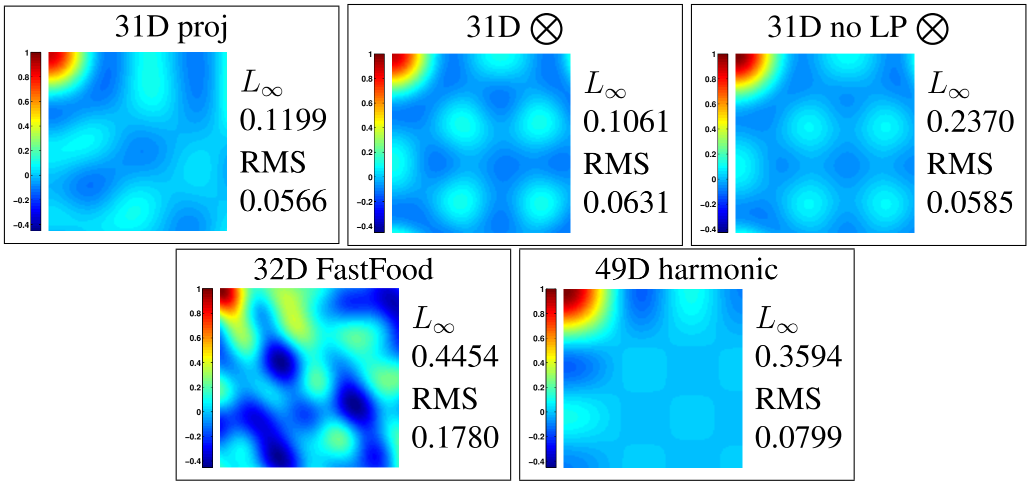


Figure 6: Comparison of kernel signature approximations of a symmetric 2D RBG kernel by projections, modulation, FastFood [18], and modulation of harmonic frequencies. Only one quadrant of the kernel signature is shown.

for every $\boldsymbol{z} \in Z \subset \mathbb{R}^n$, $\boldsymbol{z} = (z^1, \ldots, z^n)^\top$. The constraint (27) generates two rows in the constraint matrix in the following form

$$\begin{pmatrix} \prod_i^n \cos(\omega_1^i z^i), & \ldots, & \prod_i^n \cos(\omega_{|\Omega|}^i z^i), & -1 \\ -\prod_i^n \cos(\omega_1^i z^i), & \ldots, & -\prod_i^n \cos(\omega_{|\Omega|}^i z^i), & -1 \end{pmatrix} \begin{pmatrix} \alpha_{\boldsymbol{\omega}_1} \\ \vdots \\ \alpha_{\boldsymbol{\omega}_{|\Omega|}} \\ \varepsilon \end{pmatrix} \leq \begin{pmatrix} k(\boldsymbol{z}) \\ -k(\boldsymbol{z}) \end{pmatrix}. \quad (28)$$

The full constraint matrix has $|\Omega| + 1$ columns and $2|Z|$ rows. Both $|\Omega|$ and $|Z|$ grow exponentially with the dimension n of the kernel input. Increasing the size of the matrix makes the proposed optimization slower, or even intractable. Reducing the number of frequencies $|\Omega|$ (the number of columns in the constraint table) will compromise the quality of the approximation.

Provided that the frequencies used in Ω and the sampling rate of Z are chosen to avoid aliasing, most of the constraints (27) are trivially satisfied and are not active during the optimization. This observation gives rise to an efficient algorithm similar to the cutting plane algorithm [19, 20]. The optimization is executed iteratively, using only a subset $Z^i \subset Z$ to generate the constraints (27). The optimal solution for Z^i is a lower bound on the optimal solution for the full set Z. After each iteration, points in $z^* \in Z$ violating the constraints the most are added into Z^{i+1} to generate new constraints for the next iteration. If there is no point in Z,

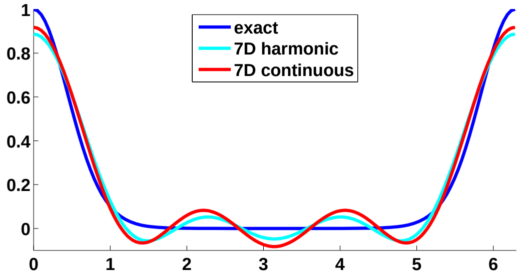


Figure 7: Approximating a kernel with a periodic signature.

for which the constraint is violated, the algorithm has converged to the optimal solution, as if all constraints were used. The convergence is guaranteed in a finite number of steps, in the worst case when all constraints are used, *i.e.* $Z^i = Z$, $i \leq |Z|$.

7. Experimental comparison

In this section, we evaluate the quality of the proposed feature map construction. First, a periodic kernel is considered. Then, 1D aperiodic function are approximated: a 1D RBF kernel and homogeneous kernels with detailed evaluation of the χ^2 kernel. Finally, the two dimensional RBF kernel approximation is evaluated, and the efficient algorithm optimizing L_∞ error on a 3D kernel is tested.

7.1. Periodic functions: a sanity check

In this section, we briefly compare the approximation achieved by the orthogonal projection onto the cosine basis with harmonic frequencies and the proposed method. While the orthogonal projection minimizes the sum of squared errors, the proposed method minimizes the L_∞ error.

As an example of a commonly used periodic kernel, the orientation kernel [4] is used. The orientation kernel is a periodic kernel with period of 2π derived from the Von Mises distribution, with kernel signature defined as

$$k_\theta(\lambda_\theta) = \frac{e^{\kappa \cos(\lambda_\theta)} - e^{-\kappa}}{2\sinh(\kappa)} \tag{29}$$

We compare the 7D explicit feature map approximation to the kernel with parameter κ set to $\kappa = 5$, see Figure 7. The L_∞ error of the methods is 0.1134 and 0.0824, while the RMS is 0.0542 and 0.0586 respectively.

	χ^2		intersect		J-S	
	L_∞	RMS	L_∞	RMS	L_∞	RMS
Ours 5D	**0.163**	**0.081**	**10.922**	**5.376**	**0.019**	**0.009**
VLFeat 5D	3.205	1.251	30.119	6.679	2.911	1.203
Ours 7D	**0.011**	**0.005**	**8.238**	**4.053**	**9e^{-4}**	**3e^{-4}**
VLFeat 7D	0.143	0.053	22.287	4.436	0.127	0.070

Table 1: Comparison of approximation precision of different homogeneous feature maps. Maximal error L_∞ and root mean square RMS error are compared on $x, y \in \{0, \ldots, 255\}$.

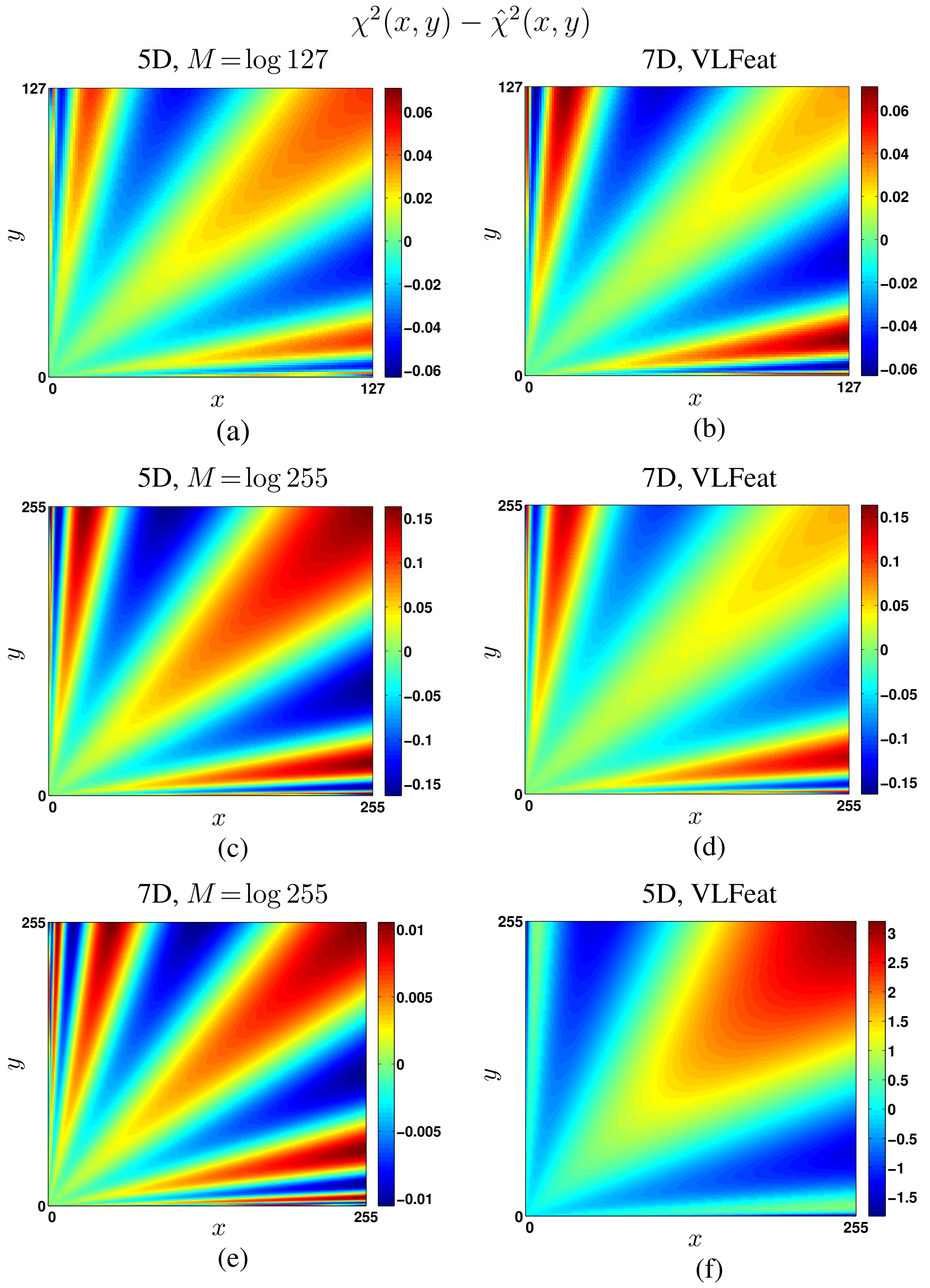
For periodic function, it is optimal to use the harmonic frequencies since they are periodic. The proposed continuous method results in harmonic frequencies, even if they were not included in the initial pool of frequencies.

7.2. RBF kernel

A number of different feature maps approximating a one-dimensional RBF kernel with $\sigma^2 = 0.2$ on interval $x, y \in [0, \pi]$ were compared. Figure 4(a) shows two examples of rather poor approximations of the underlying kernel signature: an orthogonal projection onto a cosine basis with angular frequencies $\Omega = \{0, \ldots, 5\}$ resulting in 11D feature map, and a random explicit feature map of 64 dimensions [21]. Since random explicit feature maps are only approximately shift-invariant, the plot shows $\hat{\Psi}_{\mathrm{RF}}(0)^\top \hat{\Psi}_{\mathrm{RF}}(\lambda)$. Figure 4(b) shows the values of the absolute error $k(\lambda) - \hat{k}(\lambda)$ for three comparable feature maps: an orthogonal projection onto a cosine basis with angular frequencies $\Omega = \{0, \ldots, 8\}$ resulting in 17D feature map (labelled 17D harmonic), the 11D feature map by the proposed discrete method (section 4.1), and the 11D feature map by the proposed continuous method (section 4.2). All three approximations would be indistinguishable from the exact k on Figure 4(a). It can be seen that the proposed continuous method is superior to the orthogonal projection, despite the fact that it uses only 11 dimensions compared to 17 of the orthogonal projection. The continuous method reduces the optimized C_∞ error function over the discrete method to approximately one half in this case (from $6.7 \cdot 10^{-3}$ to $3.3 \cdot 10^{-3}$).

7.3. Homogeneous kernels

We thoroughly evaluate the proposed method on a χ^2 kernel approximation $\chi^2(x, y) = 2xy/(x+y)$. We compared the approximation by the proposed method with the state-of-the-art method of [13] available in VLFeat [14]. Our approach allows the error function on the kernel to be optimized, while the competing method



(a)

(b)

(c)

(d)

(e)

(f)

Figure 8: Comparison of the absolute error of the χ^2 approximation. Left column is the proposed method optimizing (18), right column shows results for Vedaldi [13], VLFeat implementaion [14]. The first two rows compare the proposed 5D mapping to 7D mapping of VLFeat on $x, y \in \{0, \ldots, 127\}$ and $x, y \in \{0, \ldots, 255\}$, respectively. The third row shows the error of the 7D proposed mapping and 5D mapping of VLFeat for comparison.

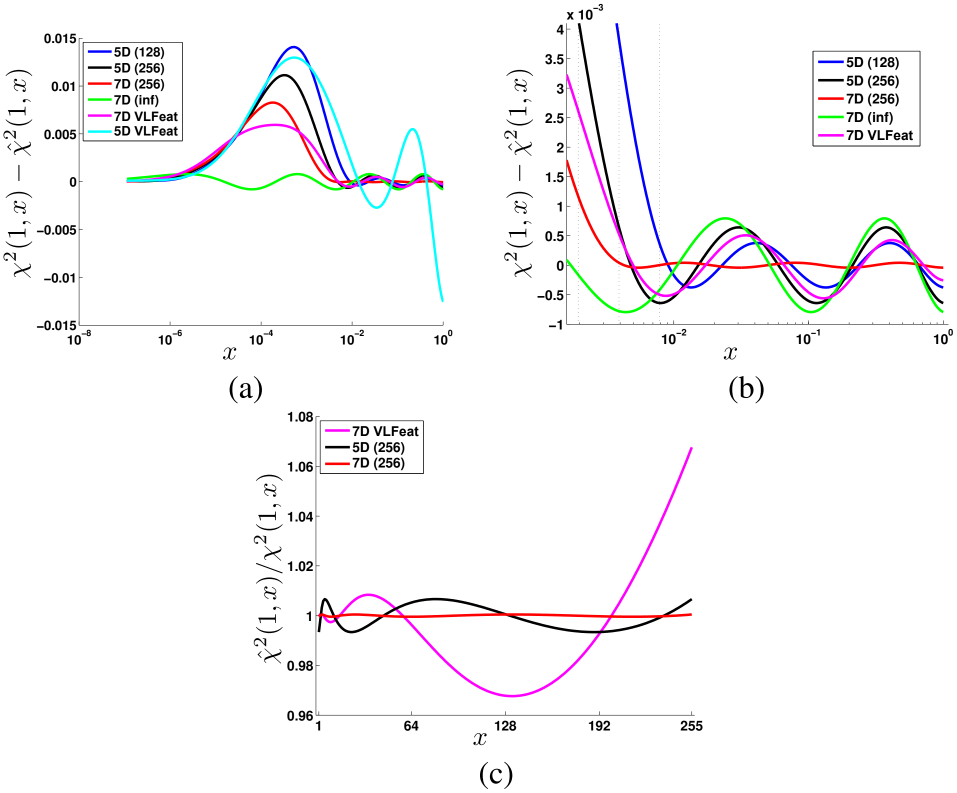


Figure 9: Comparison of different χ^2 approximations: (a) the absolute error for large ratios of the input values; (b) a close up with three dotted vertical lines at $1/512$, $1/256$ and $1/128$ respectively; (c) relative error of the approximation. The error range colour mapping is fixed for the first and second row. Logarithm of the number in brackets states the size M of the interval on which the kernel signatures were approximated.

approximates the kernel signature. The comparison on the commonly used homogeneous kernels is summarized in Table 1.

First, we compare the absolute error of the approximation. For this experiment, the ε_A error (equation (18)) was minimized for the proposed method. The approximation errors are plotted in Figure 8. The first row compares our 5D feature map to the 7D feature map of VLFeat [14] on input data $x, y \in \{0, \dots, 127\}$. Note that the input values can be arbitrarily scaled, and only the smallest non-zero ratio of the values is relevant. The kernel signature for the proposed method was optimized on the interval $[-M, M]$, where $M = \log 127$. Even though the proposed method provides a lower dimensional feature map and L_∞ error was optimized, it outperforms (on this interval) the method of VLFeat [14] in L_∞ error ($\max_{x,y} |\chi^2(x, y) - \hat{\chi}^2(x, y)|$: 0.048 vs. 0.071) as well as in L_2 error ($\sum_{x,y}(\chi^2(x, y) - \hat{\chi}^2(x, y))^2$: 9.121 vs. 11.272).

The middle row of Figure 8 compares the proposed 5D feature map ($M = \log 255$) and the 7D feature map of VLFeat [14] on the input data $x, y \in \{0, \dots, 255\}$. The 7D feature map provides a slightly better approximation than the 5D map; however, the error range of the two feature maps is approximately the same. Replacing a 7D feature map by 5D feature map reduces the memory requirements and kernel evaluation time by 28% .

For a full comparison, we have included (bottom row of Figure 8) the proposed 7D feature map ($M = \log 255$) with an order of magnitude lower error that the two previously compared feature maps, and also the 5D feature map of VLFeat [14] with an order of magnitude higher error than the two previous feature maps. From this experiment we see that (1) the proposed approximation outperforms the state-of-the-art feature maps, and (2) the feature map should be optimized for the input domain of particular application.

In the next experiment, we study how the approximation behaves outside the region for which it was optimized. The approximation error for different methods is plotted in Figure 9 for the values of the ratio x/y up to $1/10^7$. It can be observed that outside the optimal region, the error of the kernel signature approximation $|\hat{k}(\lambda) - k(\lambda)|$ increases and thus the error of the kernel $\sqrt{xy}|\hat{k}(\log y/x) - k(\log y/x)|$ also increases. Since $k(\lambda)$ decays and $\hat{k}(\lambda)$ is bounded, $|\hat{k}(\lambda) - k(\lambda)|$ is also bounded. As a result, for sufficiently large $|\lambda|$, the error of the kernel is dominated by $\sqrt{xy}$ and approaches zero. In Figure 9, the green curve corresponds to a kernel signature that has been optimized for a large enough interval so that the upper bound on the error is less than the optimized L_∞ inside the interval, thus having optimal error bound everywhere. This is only possible for error measures, such as ε_A (18), with decreasing weight $w(\lambda)$ (19).

The last experiment with the χ^2 kernel considers the relative error ε_R of the kernel fit (20). Three feature maps are compared: the proposed 5D and 7D maps constructed to minimize the relative error, and the 7D map by VLFeat [14]. The plot in Figure 9 (c) show that the proposed method significantly outperforms its competitor.

7.4. Symmetric RBF kernel in 2D

Four methods approximating the symmetric 2D RBF kernel with $\sigma^2 = 0.2$ with kernel input variables $\mathbf{x}, \mathbf{y} \in [0, \pi]^2$ were compared: two using the projection method described in section 6.1, and two using the modulation method (section 6.2). For the projection method, two different initializations of the frequency pool Ω were used: a general initialization by the discretization of angle and frequency $\|\boldsymbol{\omega}\|$ of $\boldsymbol{\omega}$, referred to as 'Proj'; and by frequencies equivalent to $\Omega_\otimes$ (25), obtained as a Cartesian product $\Omega_\otimes = \Omega_{1\mathrm{D}} \times \Omega_{1\mathrm{D}}$, where $\Omega_{1\mathrm{D}}$ corresponds to the 11D feature map form section 7.2 (referred to as 'Proj $\bigotimes$'). For both methods, the full LP optimization on 2D input, including the continuous extension, was performed.

Both modulation methods (section 6.2) were initialized by $\Omega_\otimes$. One method ('$\bigotimes$') exploits the full LP optimization on 2D input, including the continuous extension. For the last method ('$\bigotimes$ no LP') the feature map is selected greedily based on the estimate $\alpha_{\boldsymbol{\omega}} = \alpha_{\omega^1}\alpha_{\omega^2}$. The quantitative result are summarized in Table 2, and the qualitative results of approximations by 31 dimensional feature maps is shown in the leftmost column of Figure 6.

The fastest approach '$\bigotimes$ no LP' is the least precise. The most general approach 'Proj' is the slowest and performs slightly worse than the two approaches initialized by results of 1D optimization.

We made the following observations for the modulation methods: (1) the linear program selects different components than the greedy approach, (2) after the continuous extension, the frequencies in Ω are no longer on a grid, which is originally defined by the Cartesian product $\Omega^1 \times \Omega^2$.

Finally, the comparison with other methods shown in Figure 6 demonstrates that any of the proposed methods is superior to existing methods.

7.5. Efficient approximation with L_∞ error

In this section, an embedding encoding 2D position and an angle, $\mathbf{x} = (x, y, \theta) \in [0, \pi]^2 \times [0, 2\pi]$, is constructed. The dot product of the embedding approximates a multiplicative kernel, a product of Gaussian kernels with $\sigma^2 = 0.2$ on x and y

$D(\hat{k})$	Proj	Proj $\otimes$	$\otimes$	$\otimes$ no LP
31	0.1199	0.0984	0.1061	0.2370
73	0.0292	0.0148	0.0137	0.0439
101	0.0092	0.0038	0.0038	0.0122

Table 2: Comparison of the L_∞ error of the 2D RBF kernel approximation.

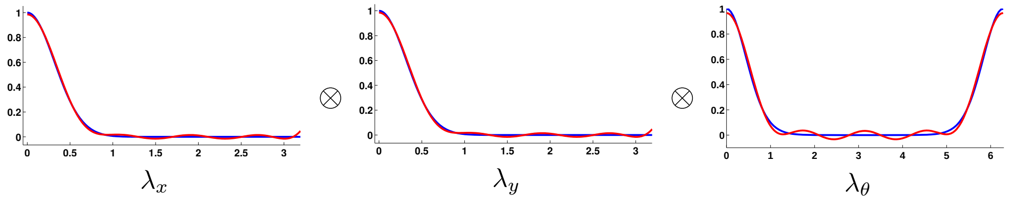


Figure 10: Visualization of modulated kernel composed of two position (x and y) and an orientation (θ) kernels (shown in blue). Each sub-kernel was initialized with a 9D feature map approximation (shown in red), the modulation results in $9^3 = 729$D feature map.

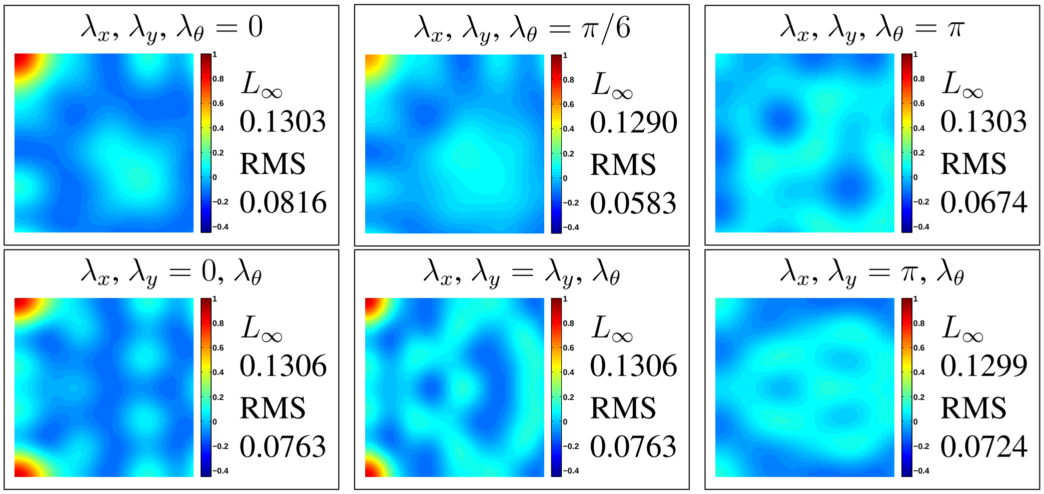


Figure 11: Approximation of a three dimensional kernel signature: 2D Gaussian in λ_x and λ_y modulated by a periodic orientation kernel in λ_θ (see Figure 10). Plots show slices through an approximation by a 101D explicit feature map. The top left slice (showing λ_x and λ_y for fixed $\lambda_\theta = 0$) is comparable to plots in Figure 6. The plots in the last column show areas where the kernel signature values should be flat zero.

(as in section 7.4) and an orientation kernel on θ with $\kappa = 5$ (as in section 7.1). Such a kernel can be used, for example, in a patch descriptor framework [4]. The components of the kernel signature are visualized in Figure 10.

The kernel was approximated using the efficient approximation of L_∞ introduced in section 4.2, including the extension to estimate the continuous frequencies. The frequency pool was initialized by the modulation of 9D feature maps for all three sub-kernels. The pure modulation leads to 729D explicit features maps. We show results of the proposed method resulting in a 101D feature map in Figure 11. After convergence, only 0.74% of the constraints (points of the evaluation set Z) were used. The approximation takes about 40 seconds in a Matlab implementation on a laptop (i5 CPU @ 2.60 GHz, 16GB RAM).

8. Conclusions

A novel method of data independent construction of low dimensional feature maps was proposed. The problem is cast as a linear program that jointly considers competing objectives: the quality of the approximation and the dimensionality of the feature map. The proposed discrete optimization exploits the entire continuous spectrum of frequencies and achieves considerably better approximations with feature maps of the same dimensionality or equally good approximations with lower dimensional feature maps compared to the state-of-the-art methods. It was also demonstrated that the proposed method allows for the optimization of meaningful errors measured on the homogeneous kernel output, rather than solely approximating the kernel signature.

Any application that uses explicit features maps would benefit from the results of this paper. The code is available [22].

Acknowledgement

I would like to thank Tomáš Werner for his valuable opinions and interesting discussions. This work was supported by MSMT LL1303 ERC-CZ grant.

References

[1] T. Joachims, Training linear SVMs in linear time, in: KDD, ACM, 2006, pp. 217–226.

[2] L. Bo, C. Sminchisescu, Efficient match kernel between sets of features for visual recognition, in: NIPS, 2009, pp. 135–143.

[3] L. Bo, X. Ren, D. Fox, Kernel descriptors for visual recognition, in: NIPS, 2010, pp. 244–252.

[4] A. Bursuc, G. Tolias, H. Jégou, Kernel local descriptors with implicit rotation matching, in: ICMR, 2015.

[5] G. Tolias, T. Furon, H. Jégou, Orientation covariant aggregation of local descriptors with embeddings, in: ECCV, 2014.

[6] H. Jégou, M. Douze, C. Schmid, P. Pérez, Aggregating local descriptors into a compact image representation, in: CVPR, 2010.

[7] F. Perronnin, Y. Liu, J. Sanchez, H. Poirier, Large-scale image retrieval with compressed fisher vectors, in: CVPR, 2010.

[8] B. Scholkopf, A. Smola, Learning with Kernels, MIT Press, 2002.

[9] A. Rahimi, B. Recht, Random features for large-scale kernel machines, in: NIPS, 2007, pp. 1177–1184.

[10] F. Li, C. Ionescu, C. Sminchisescu, Random fourier approximations for skewed multiplicative histogram kernels, in: DAGM, 2010.

[11] S. Maji, A. C. Berg, Max-margin additive classifiers for detection, in: ICCV, 2009, pp. 40–47.

[12] A. Vedaldi, A. Zisserman, Sparse kernel approximations for efficient classification and detection, in: CVPR, 2012.

[13] A. Vedaldi, A. Zisserman, Efficient additive kernels via explicit feature maps, TPAMI 34 (3).

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

[15] O. Chum, Low dimensional explicit feature maps, in: ICCV, 2015.

[16] R. J. Vanderbei, Linear Programming: Foundations and Extensions, Kluwer Academic Publishers, Boston, 1996.

[17] D. Lowe, Distinctive image features from scale-invariant keypoints, IJCV 60 (2) (2004) 91–110.

[18] Q. Le, T. Sarlós, A. Smola, Fastfood-approximating kernel expansions in loglinear time, in: ICML, 2013.

[19] J. Kelley, The cutting-plane method for solving convex programs, Journal of the Society for Industrial and Applied Mathematics 8 (4) (1960) 703–712.

[20] V. Franc, S. Sonnenburg, T. Werner, Cutting-Plane Methods in Machine Learning, The MIT Press, Cambridge,USA, 2012, Ch. 7, pp. 185–218.

[21] F. Li, C. Ionescu, C. Sminchisescu, Randfeat: Random fourier approximations for skewed multiplicative histogram kernels, release 1, http://sminchisescu.ins.uni-bonn.de/code/randfeat/randfeat-release1.tar.gz.

[22] O. Chum, Implementation of low dimensional explicit feature maps, http://cmp.felk.cvut.cz/˜chum/code/ld-efm.html.