Asymmetric Feature Maps with Application to Sketch Based Retrieval

Giorgos Tolias Ondřej Chum
Visual Recognition Group, Faculty of Electrical Engineering, Czech Technical University in Prague
{giorgos.tolias,chum}@cmp.felk.cvut.cz

Abstract

We propose a novel concept of asymmetric feature maps (AFM), which allows to evaluate multiple kernels between a query and database entries without increasing the memory requirements. To demonstrate the advantages of the AFM method, we derive a short vector image representation that, due to asymmetric feature maps, supports efficient scale and translation invariant sketch-based image retrieval. Unlike most of the short-code based retrieval systems, the proposed method provides the query localization in the retrieved image. The efficiency of the search is boosted by approximating a 2D translation search via trigonometric polynomial of scores by 1D projections. The projections are a special case of AFM. An order of magnitude speed-up is achieved compared to traditional trigonometric polynomials. The results are boosted by an image-based average query expansion, exceeding significantly the state of the art on standard benchmarks.

1. Introduction

Efficient match kernel [3] is a popular choice in applications evaluating complex similarity measures on large collections of objects, where an object is a set of elements. This includes local feature descriptors [3, 5] and image retrieval with short descriptors [38][1].

In efficient match kernel, all elements of the sets are mapped to a finite feature map [27, 39]. An inner product of the feature maps approximates evaluation of a specific kernel, defining similarity of the set elements. We propose an extension to this concept. In the asymmetric feature map, the query uses a different embedding than the database objects. The query embedding defines the kernel that is evaluated between the query and the database entries. Thus, multiple kernels can be evaluated while the memory requirements for the database remains the same (up to a scalar per kernel) as for a single kernel to be evaluated. The embeddings are obtained via joint kernel feature map optimization, which significantly improves the quality of kernel approximation for a fixed dimensionality of the feature map.

[1]The authors were supported by the MSMT LL1303 ERC-CZ grant.

The application domain of AFM is wide, in particular any method using efficient match kernel benefits from AFM. We evaluate the AFM on a sketch-based retrieval application. Sketch-based retrieval has received less attention than image retrieval and still remains challenging. Instead of a real image, the query consists of an abstract binary sketch. This allows the user to quickly outline an object, *e.g.* by a finger on a tablet or smart phone, and search for relevant images (see Figure 1). The progress in this area has more or less followed the footsteps of traditional image retrieval. The first systems employed global descriptors [8]. Then, the Bag-of-Words paradigm with local descriptors and feature quantization [17, 16, 30] was adopted.

Due to the absence of textural cues on the query side, the image representations are shape based. Bridging the representation gap between hand-drawn sketches and real images is one of the challenges making the task difficult. Matching based on shape information has been addressed previously. For instance, in object recognition and detection [2, 17, 23], a costly online matching is performed, which prevented the methods to scale to large image collections. Recent methods manage to index million [7] to billion [36] images for sketch-based retrieval, at the cost of sacrificed invariance to geometric transformations.

To demonstrate the impact of the AFM, we propose a short vector image representation allowing to index large image collections for sketch-based search. Scale and translation invariant real-time search allows to process an order of millions of images per one processor thread. The AFM based method achieves state-of-the-art results on standard benchmarks. The method runs at speed comparable to previously published approaches tailored to sketch-based

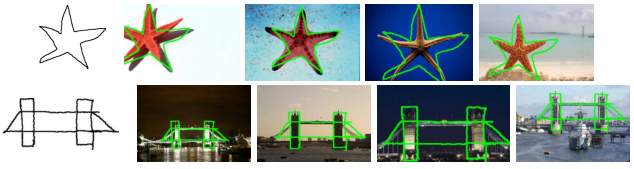

Figure 1. Scale and translation invariant query-by-sketch retrieval. An example of sketch queries and top-retrieved images with the sketch localization overlaid in green color.

search. Compared with methods based on efficient match kernel [38], the proposed method achieves order of magnitude speed-up. Unlike most of the methods using low-dimensional descriptors, the proposed method delivers localization of the object in both scale and space. The scale invariance is achieved by evaluating multiple kernels without the need to store multiple representations for database images. The translation invariance and object localization is provided by an efficient similarity evaluation on a 2D grid of translations. Namely, the four main contributions of this work are as follows. (1) Asymmetric explicit feature maps allowing the use of multiple kernel functions without constructing multiple representations for database items are proposed. (2) A joint kernel approximation approach for multiple kernels is derived, generalizing a recent approach of low dimensional explicit feature maps (LDFM) [9]. (3) The scoring through trigonometric polynomial introduced in [38] is further extended and a significant speed-up of its evaluation is proposed. (4) State-of-the-art sketch-based image retrieval based on the AFM, which is further boosted by query expansion which acts, not on the edge maps as standard sketch matching, but on the original images.

The rest of paper is organized as follows. Related work is discussed in Section 2 and the necessary background is presented in Section 3. Sections 4 and 5 describe our contributions on asymmetric explicit feature maps and on sketch retrieval, respectively, while the retrieval procedure and the experimental evaluation are analyzed in Section 6.

2. Related work

The most similar work to ours is the approach of Tolias *et al.* [38], where the trigonometric polynomial scores were introduced in the context of image retrieval (see Section 3.3 for technical details). Shape properties of local features, such as dominant orientation or position, are jointly encoded with the SIFT descriptor. Despite initially assuming aligned objects, their kernel descriptor comes with an efficient way to compute similarity over multiple image transformations. Compared to their method, asymmetric feature maps introduced in our paper: i) reduce the memory requirements of multi-scale search by roughly a factor of 3, and ii) achieve an order of magnitude speed-up through approximate translation search. The trigonometric polynomials have been also used by Bursuc *et al.* [5] in the context of rotation invariant feature descriptors. The descriptor has recently shown competitive results with CNN based approaches [1].

Since we demonstrate the advantages of AFM on sketch based retrieval, we provide a brief review of relevant literature on this topic. The line of research that focuses on sketches includes recognition [14, 40] or retrieval [24] of sketches. This paper addresses sketch-based image retrieval, which tries to match sketch queries to real images from a large collection. Following successful examples of traditional image retrieval, sketch-based methods employ global image representation [8, 29] or local descriptors and the Bag-of-Words model. In the latter case, representative methods employ local descriptors that are traditionally used on images [16, 30] or proposed particularly for this task [15, 28, 18, 6]. Some examples are HOG descriptors which are adapted for sketch retrieval [18] and were recently extended to capture color [4], symmetry-aware and flip invariant descriptors [6], and descriptors based on local contour fragments [28]. Generic approaches performing learning of discriminative features have been shown effective for sketch retrieval too [33].

Chamfer matching appears to be a good similarity measure for object shapes [37]. Recent attempts focus on Chamfer matching approximations in order to increase scalability. Cao *et al.* [7] binarize the distance transform map and manage to index two million images. However, their approach completely lacks invariance. The same holds for the work of Sun *et al.* [36] who increase the scale of the indexed collection up to one billion. Despite the achievement of scalability, rough approximations of Chamfer matching sacrifice accuracy. Recently, Parui and Mittal [25] proposed a similarity invariant approach able to index up to one million images. Their solution is based on dynamic programming to match chains of contour lines, while the main drawback is the costly off-line indexing.

3. Background

We briefly review the necessary background, which includes efficient match kernels [3], explicit feature maps [39] and efficient trigonometric polynomial scores [38].

3.1. Efficient Match Kernels

In many situations, an object is described by a set of measurements $\mathcal{P} = \{p \in \mathbb{R}^d\}$. Employing a mapping $\Psi : \mathbb{R}^d \to \mathbb{R}^D$ to the elements of $\mathcal{P}$, the set representation of efficient match kernels is defined as

$$\mathbf{V}(\mathcal{P}) = \sum_{p \in \mathcal{P}} \Psi(p). \tag{1}$$

Then, a dot product between the set representation yields the similarity between sets

$$\mathcal{S}(\mathcal{P}, \mathcal{Q}) = \mathbf{V}(\mathcal{P})^\top \mathbf{V}(\mathcal{Q}) = \sum_{p \in \mathcal{P}} \sum_{p \in \mathcal{Q}} \Psi(p)^\top \Psi(q). \tag{2}$$

Normalized similarity is computed by cosine similarity [38], *i.e.*, dot product of ℓ_2 normalized vectors,

$$\bar{\mathcal{S}}(\mathcal{P}, \mathcal{Q}) = \frac{\mathbf{V}(\mathcal{P})^\top \mathbf{V}(\mathcal{Q})}{\sqrt{\mathbf{V}(\mathcal{P})^\top \mathbf{V}(\mathcal{P})} \sqrt{\mathbf{V}(\mathcal{Q})^\top \mathbf{V}(\mathcal{Q})}}, \tag{3}$$

while another choice is to normalize by the set cardinality [3]. Herein, the cosine similarity is adopted ensuring self-similarity is normalized to one. A number of image representations, such as BOW [35, 11], Fisher vectors [26], or VLAD [20], can be interpreted as efficient match kernels.

3.2. Explicit feature maps

Let $K(p, q)$ be a one-dimensional (p is now scalar) positive definite stationary kernel [32] $K : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$. The value of a stationary kernel by definition depends only on the difference $\lambda = p - q$,

$$K(p, q) = K(p, p - \lambda) = k(\lambda), \tag{4}$$

where $k(\lambda)$ is a signature of kernel $K(p, q)$. Due to Bochner's theorem, kernel signature k can be written as

$$k(\lambda) = \int_0^\infty \alpha(\omega) \cos(\omega\lambda)\, \mathrm{d}\omega, \tag{5}$$

where $\alpha(\omega) : \mathbb{R}_0^+ \to \mathbb{R}_0^+$. The kernel signature is approximated by sum over a finite set Ω of frequencies

$$\hat{k}(\lambda) \approx \sum_{\omega \in \Omega} \alpha_\omega \cos(\omega\lambda), \tag{6}$$

where $\alpha_\omega \in \mathbb{R}_0^+$. Applying the trigonometric identity

$$\cos(p - q) = \cos(p)\cos(q) + \sin(p)\sin(q) \tag{7}$$

gives rise to feature map (or feature embedding) $\Psi_\omega : \mathbb{R} \to \mathbb{R}^2$ defined as

$$\Psi_\omega(p) = \left(\sqrt{\alpha_\omega}\cos(\omega p),\ \sqrt{\alpha_\omega}\sin(\omega p)\right)^\top. \tag{8}$$

The inner product of two such vectors reconstructs the terms of equation (6) since $\Psi_\omega(p)^\top \Psi_\omega(q) = \alpha_\omega \cos(\omega(p - q))$. Let the feature map $\Psi(p) : \mathbb{R} \to \mathbb{R}^D$ be constructed as a concatenation of $\Psi_\omega(p)$ for all $\omega \in \Omega$. Now, the inner product

$$\Psi(p)^\top \Psi(q) = \hat{k}(p - q) \approx K(p, q) \tag{9}$$

evaluates the approximation of the kernel signature (6) and hence approximates the original kernel K. The choice of the number of frequencies $|\Omega|$ determines the quality of the approximation and the dimensionality of the embedding. The dimensionality is $2|\Omega|$, or $2|\Omega| - 1$ if $0 \in \Omega$[2].

Feature map construction. We mention in detail (and compare) two approaches to construct the explicit feature maps. We do not consider random feature maps [27], which approximate the integral in (5) using Monte-Carlo methods. Such feature maps provide a poor approximation for low-dimensional feature maps.

Vedaldi and Zisserman [39] propose the following approximation to a kernel signature $k(\lambda)$ on an interval $\lambda \in [-\Lambda, \Lambda]$. First, a periodic function g with period 2Λ is constructed, so that $g(\lambda) = k(\lambda)$ for $\lambda \in [-\Lambda, \Lambda]$. The feature map is then efficiently obtained by approximating periodic g using harmonic frequencies only. This approach has been shown sub-optimal [9]. Further, the periodic function g is not even guaranteed to be positive definite.

A convex optimization approach is proposed by Chum [9]. The input domain of $\hat{k}(\lambda)$ is discretized to finite set $Z \subset [0, \Lambda]$. The quality of the approximation is measured at points in Z as, for example, an ℓ_∞ norm

$$C_\infty(k, \hat{k}) = \max_{\lambda \in Z} |k(\lambda) - \hat{k}(\lambda)|. \tag{10}$$

The set of frequencies $\Omega \subset \bar{\Omega}$ are selected from a pool of frequencies $\bar{\Omega}$, and corresponding weights $\alpha_\omega \geq 0, \omega \in \bar{\Omega}$ jointly through a solution of a linear program

$$\min_k C(k, \hat{k}) + \gamma \textstyle\sum_{\omega \in \bar{\Omega}} \alpha_\omega, \tag{11}$$

where $\gamma \in \mathbb{R}^+$ is a weight on the l_1 regularizer controlling the trade-off between the quality of the approximation and the sparsity of α_ω. This is the method we adopt and extend in this work.

3.3. Alignment using trigonometric polynomials

Tolias *et al.* [38] propose an image representation derived by efficient match kernels and explicit feature maps. We focus on the case that all measurements of set $\mathcal{P}$ are shifted by a constant value Δp; note that measurements p are now scalars. The similarity under such shift forms a trigonometric polynomial

$$\mathcal{S}(\mathcal{P}_{\Delta p}, \mathcal{Q}) = \sum_{\omega \in \Omega} \left(\beta_\omega \cos(\omega \Delta p) + \gamma_\omega \sin(\omega \Delta p)\right), \tag{12}$$

with $\mathcal{P}_{\Delta p} = \{p - \Delta p, p \in \mathcal{P}\}$. Parameters β_ω and γ_ω are given by dot products of relevant sub-vectors of $\mathbf{V}(\mathcal{P})$ and $\mathbf{V}(\mathcal{Q})$. Finally the similarity measure that is invariant under such shifting is given by $\mathcal{S}_1(\mathcal{P}_{\Delta p}, \mathcal{Q}) = \max_{\Delta p} \mathcal{S}(\mathcal{P}_{\Delta p}, \mathcal{Q})$.

We postpone further analysis of polynomials of scores until the image representation is introduced in Section 5.

4. Asymmetric feature maps

In this section, we introduce the concept of asymmetric feature maps. Unlike in classical explicit feature maps, a different feature map $\hat{\Psi}$ is used on the query side and a different one $\hat{\Psi}'$ is used on the database side. We show that with asymmetric feature maps, a number of different kernels can be efficiently evaluated between query and database vectors while keeping the database storage of fixed size. Compare the feature map in equation (8) to the following feature maps for the query and database side respectively

$$\hat{\Psi}_\omega(q) = \left(\alpha_\omega \cos(\omega q),\ \alpha_\omega \sin(\omega q)\right)^\top \tag{13}$$

$$\hat{\Psi}'_\omega(p) = \left(\cos(\omega p),\ \sin(\omega p)\right)^\top. \tag{14}$$

The inner products $\hat{\Psi}(q)^\top \hat{\Psi}'(p) = \Psi(q)^\top \Psi(p)$ are preserved. The kernel function is fully defined by the weights on the query side. No additional storage is required on the database side to evaluate the kernel. The same holds for efficient match kernels, as (1) is a normalized sum of feature maps. To evaluate the cosine similarity (3), only a single scalar per kernel $K^{(i)}$ needs to be stored for each database entry $\mathcal{P}$ – the ℓ_2 norm $\sqrt{\mathbf{V}^{(i)}(\mathcal{P})^\top \mathbf{V}^{(i)}(\mathcal{P})}$, which is computed offline.

[2] If $0 \in \Omega$, then $\alpha_0 \sin(0\lambda) = 0$ for all λ can be dropped from the explicit feature map.

Joint approximation of multiple kernels. In order to evaluate a number of different kernels $K^{(i)}(p,q)$ using the asymmetric feature maps, all respective explicit feature maps $\Psi^{(i)}$ have to be based on the same set of frequencies Ω. A naive approach would be to optimize the set of frequencies for one of the kernels and keep it fixed for other kernels. This approach, however, leads to poor approximation, as shown in Figure 2. We propose an extension to LDFM [9] to jointly approximate a set of kernels $K^{(i)}$ represented by their respective kernel signatures $k^{(i)}$, $i \in \{1 \dots n\}$. The quality of the approximation is measured by the sum of individual qualities (10)

$$C^*_\infty = \sum_{i=1}^{n} C_\infty(k^{(i)}, \hat{k}^{(i)}) = \sum_{i=1}^{n} \max_{\lambda \in Z} |k^{(i)}(\lambda) - \hat{k}^{(i)}(\lambda)|.$$

The optimization is performed by executing a linear program

$$\min_{\alpha^{(i)}_\omega \mid \omega \in \Omega} C^*_\infty + \gamma \sum_{\omega \in \Omega} \max_i \alpha^{(i)}_\omega, \tag{15}$$

where γ is a weight of the sparsity regularizer that controls the number of frequencies used, *i.e.* the dimensionality of the feature map. Following the approach of Chum [9], to ensure the required dimensionality of the feature map, a binary search for γ is performed.

Figure 2 presents the approximation of three different kernels using the same set of frequencies. We compare the approximation using only harmonic frequencies, the naive approximation mentioned above, and our joint approximation. The latter has a significantly better fit.

5. Sketch-Based Retrieval

In this section we present our sketch descriptor employing explicit feature maps and elaborate on the efficient trigonometric polynomial of scores to further approximate it. Our methodology is presented for the symmetric feature maps, while the asymmetric case is equivalent. We finally present efficient ways to perform the initial ranking and reranking for sketch-based image retrieval.

5.1. Sketch descriptor

Consider a binary sketch as a set of contour points, that is a set of pixels $\mathcal{P}$ that lie on the contour. A *contour pixel* $p \in \mathcal{P}$ is represented as $p = (p_x, p_y, p_\phi, p_w)$, where p_x and p_y are 2D image coordinates, p_ϕ is the gradient angle (or orientation) of the contour at (p_x, p_y), and p_w is a strength of the gradient. For real images, the contour parameters are obtained form an edge detector. For sketches, $p_w = 1$ is set for all contour pixels.

The similarity between contour pixels is computed using a multiplicative kernel composed of three one-dimensional kernels, spatial kernels over p_x, p_y, and an orientation kernel over p_ϕ. The 1D stationary kernels are denoted $K_x(p_x, q_x) = k_x(\lambda_x)$, $K_y(p_y, q_y) = k_y(\lambda_y)$, and $K_\phi(p_\phi, q_\phi) = k_\phi(\lambda_\phi)$ respectively. The *sketch descriptor* is a weighted sum of contour pixel feature maps[3]

$$\mathbf{V}(\mathcal{P}) = \sum_{p \in \mathcal{P}} p_w \Psi(p_x) \otimes \Psi(p_y) \otimes \Psi(p_\phi). \tag{16}$$

It is easy to show that *sketch similarity* (2) becomes

$$\mathcal{S}(\mathcal{P}, \mathcal{Q}) = \sum_{p \in \mathcal{P}} \sum_{q \in \mathcal{Q}} p_w q_w k_x(\lambda_x) k_y(\lambda_y) k_\phi(\lambda_\phi). \tag{17}$$

The orientation and spatial kernels are implemented by 1D RBF kernels with parameters σ_ϕ and $\sigma_x = \sigma_y$, respectively. The set of frequencies are denoted by Ω_ϕ and $\Omega_x = \Omega_y$, while the dimensionality of the corresponding embeddings is $D_x = 2|\Omega_x| - 1$ and $D_\phi = 2|\Omega_\phi| - 1$, respectively. Note that frequency $\omega = 0$ is always included. The sketch descriptor has dimensionality ${D_x}^2 D_\phi$.

The proposed representation constitutes a holistic representation encoding the global sketch shape. We now define a representation encoding only one of the spatial coordinates along with the orientation. It is equivalent to the projection of contour pixels on the horizontal/vertical image axis. The sketch descriptor derived by projection on the horizontal axis is given by

$$\mathbf{V}_x(\mathcal{P}) = \sum_{p \in \mathcal{P}} p_w \Psi(p_x) \otimes 1 \otimes \Psi(p_\phi), \tag{18}$$

where the $\otimes 1$ can be omitted and is only used to show, that the x-projection is a sub-vector of (16) and hence a special case of the proposed asymmetric feature map. This stems from the presence of the constant component of the feature map for y, corresponding to $0 \in \Omega_y$. An analogous derivation holds for $\mathbf{V}_y(\mathcal{P})$ and vertical projection.

5.2. Position alignment

The sketch descriptor encodes spatial coordinates and orientation of contour pixels. Therefore, alignment of objects is assumed, *i.e.* centered and up-right objects. Such an assumption does not hold in real image collections and introduces significant limitations. We now detail the polynomial of scores (mentioned in Section 3) proposed by Tolias *et al.* [38]. We show that translation invariance is achieved by polynomial of scores, and that its evaluation can be efficiently approximated to speed up the search process.

One dimensional. Consider the x-projected sketch descriptor $\mathbf{V}_x(\mathcal{P})$. Let $\mathcal{P}_{\Delta x}$ be the shifted version sketch $\mathcal{P}$ where all contour pixels are horizontally translated by Δx. Elementary trigonometric identities allow us to show that

$$\begin{aligned} \Psi^c_\omega(x - \Delta x) &= \Psi^c_\omega(x)\cos(\omega \Delta x) + \Psi^s_\omega(x)\sin(\omega \Delta x) \\ \Psi^s_\omega(x - \Delta x) &= \Psi^s_\omega(x)\cos(\omega \Delta x) - \Psi^c_\omega(x)\sin(\omega \Delta x), \end{aligned} \tag{19}$$

[3] We use Ψ to denote both the spatial and orientation feature map and simplify the notation. In fact, $\Psi(p_x)$ and $\Psi(p_y)$ approximate the spatial kernels k_x and k_y, respectively, which are identical, while $\Psi(p_\phi)$ the orientation kernel k_ϕ.

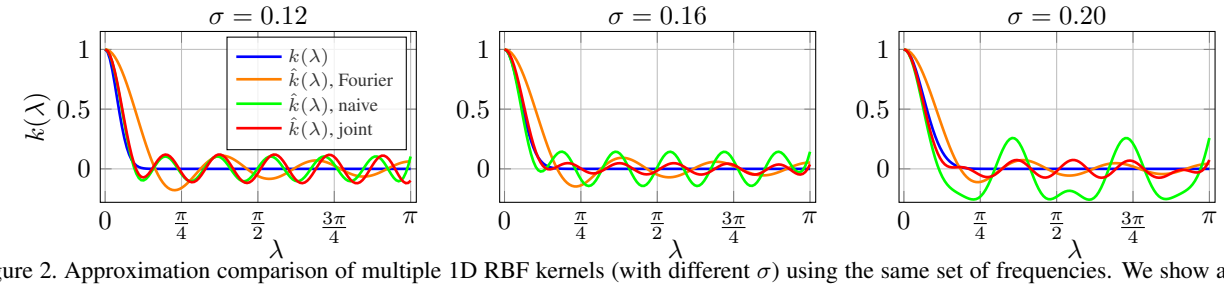

Figure 2. Approximation comparison of multiple 1D RBF kernels (with different σ) using the same set of frequencies. We show approximation using harmonic frequencies only, a naive approach of optimizing the leftmost kernel and using the same frequencies for all, and our joint approximation. Maximum value is normalized to one such that the errors are comparable. $|\Omega| = 7$ for all approximations.

where Ψ^c_ω and Ψ^s_ω denote the first and second dimension of Ψ_ω (8), respectively. Let $\mathbf{V}^c_\omega(\mathcal{P})$ be the sub-vector of $\mathbf{V}(\mathcal{P})$ comprised all elements that contain term $\Psi^c_\omega(x)$, and similarly for $\mathbf{V}^s_\omega(\mathcal{P})$. It turns out that the descriptor of the translated sketch is constructed from that of the original sketch

$$\begin{aligned}\mathbf{V}^c_\omega(\mathcal{P}_{\Delta x}) &= \mathbf{V}^c_\omega(\mathcal{P})\cos(\omega\Delta x) + \mathbf{V}^s_\omega(\mathcal{P})\sin(\omega\Delta x)\\ \mathbf{V}^s_\omega(\mathcal{P}_{\Delta x}) &= \mathbf{V}^s_\omega(\mathcal{P})\cos(\omega\Delta x) - \mathbf{V}^c_\omega(\mathcal{P})\sin(\omega\Delta x).\end{aligned} \quad (20)$$

The sketch similarity between sketches $\mathcal{P}$ and $\mathcal{Q}$ under horizontal translation Δx is a trigonometric polynomial

$$\mathcal{S}(\mathcal{P}_{\Delta x}, \mathcal{Q}) = \sum_{\omega\in\Omega_x} \left(\beta_\omega \cos(\omega\Delta x) + \gamma_\omega \sin(\omega\Delta x)\right), \quad (21)$$

with coefficients β_ω and γ_ω

$$\begin{aligned}\beta_\omega &= \mathbf{V}^c_\omega(\mathcal{P})^\top \mathbf{V}^c_\omega(\mathcal{Q}) + \mathbf{V}^s_\omega(\mathcal{P})^\top \mathbf{V}^s_\omega(\mathcal{Q})\\ \gamma_\omega &= \mathbf{V}^s_\omega(\mathcal{P})^\top \mathbf{V}^c_\omega(\mathcal{Q}) - \mathbf{V}^c_\omega(\mathcal{P})^\top \mathbf{V}^s_\omega(\mathcal{Q}).\end{aligned} \quad (22)$$

The coefficients β_ω and γ_ω of this polynomial are computed by two products of sub-vectors with D_ϕ dimensions. In total there are $N_1 = D_x$ coefficients to be computed. Finally, similarity for any translation with (21) has cost equal to N_1 scalar multiplications. If the candidate translations are fixed, then terms $\cos(\omega\Delta x)$ and $\sin(\omega\Delta x)$ can be precomputed. Normalized similarity comes at no extra cost since the ℓ_2 norm of sketch descriptor remains constant under translations (k_x is a stationary kernel):

$$\mathbf{V}(\mathcal{P}_{\Delta x})^\top \mathbf{V}(\mathcal{P}_{\Delta x}) = \mathbf{V}(\mathcal{P})^\top \mathbf{V}(\mathcal{P}). \quad (23)$$

Similarity that is invariant to horizontal translation is computed by maximizing (21) for all possible translations

$$\mathcal{S}_x(\mathcal{P}_{\Delta x}, \mathcal{Q}) = \max_{\Delta x} \mathcal{S}(\mathcal{P}_{\Delta x}, \mathcal{Q}). \quad (24)$$

Note that this similarity is also invariant to vertical translation as y coordinate is not encoded at all. However, this makes the representation less discriminative. The actual *sketch transformation* aligning the two shapes is given by $\hat{x}_1 = \arg\max_{\Delta x} \mathcal{S}(\mathcal{P}_{\Delta x}, \mathcal{Q})$. Similarity based on the vertical projection is defined in a similar way.

Two dimensional. Consider the full 2D translation $(\Delta x, \Delta y)$. Descriptor $\mathbf{V}(\mathcal{P})$ encoding both spatial coordinates is used. The corresponding second order trigonometric polynomial [38] of scores $\mathcal{S}(\mathcal{P}_{\Delta x,\Delta y}, Q)$ is constructed similarly to the first order one. The details are omitted for the sake of brevity. It allows for an efficient evaluation of similarity for multiple 2D translations in a sliding window manner. The cost to compute one of its coefficients is $4D_\phi$. There are $N_2 = 4(|\Omega_x| - 1)^2 + 4(|\Omega_x| - 1) + 1$ non-zero coefficients in total. The similarity computation for a single 2D translation has cost equal to N_2 scalar multiplications. Translation invariant similarity is given by $\mathcal{S}_{xy}(\mathcal{P}_{\Delta x,\Delta y}, \mathcal{Q}) = \max_{(\Delta x,\Delta y)} \mathcal{S}(\mathcal{P}_{\Delta x,\Delta y}, \mathcal{Q})$, and the transformation aligning the two shapes is given by $(\hat{x}_2, \hat{y}_2) = \arg\max_{(\Delta x,\Delta y)} \mathcal{S}(\mathcal{P}_{\Delta x,\Delta y}, \mathcal{Q})$.

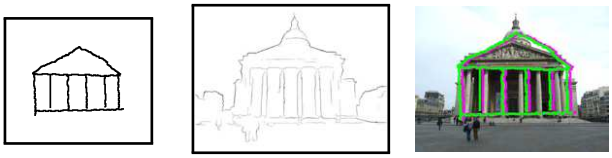

Figure 3. Sketch (left) and the edge map (middle) of a real image (right). We depict the translations maximizing similarity based on 1D projections (magenta) and the full 2D case (green).

In Figures 3 and 4 we present an alignment example between a sketch and a real image. Similarity is computed based on the horizontal and vertical projections, while also for the 2D case. Maximum similarity is met at translations that align the two silhouettes.

5.3. Efficient retrieval and query expansion

Herein, we propose three methods how to avoid exhaustive evaluation of $\mathcal{S}(\mathcal{P}_{\Delta x,\Delta y}, \mathcal{Q})$. First method efficiently selects a shortlist of images on which the score $\bar{\mathcal{S}}_{xy}$ is computed. The other two methods are designed to limit the number of possible translations over which $\mathcal{S}(\mathcal{P}_{\Delta x,\Delta y}, \mathcal{Q})$ is evaluated to obtain a good approximation of $\bar{\mathcal{S}}_{xy}$.

Shortlist by projections. The similarities $\bar{\mathcal{S}}_x$ and $\bar{\mathcal{S}}_y$ computed over the projections (22) provide an estimate of the $\bar{\mathcal{S}}_{xy}$. We propose to use this estimate for initial ranking and to compute the slow similarity $\bar{\mathcal{S}}_{xy}$ only on a shortlist of top S images. Experiments show that initial ranking by $\bar{\mathcal{S}}_x + \bar{\mathcal{S}}_y$ outperforms ranking that uses only one projection. To further speed-up the evaluation for large-scale collections, we propose *discriminative projection first* approach. In this method, one projection is computed over the whole dataset, creating a pre-shortlist of $3S$ images with

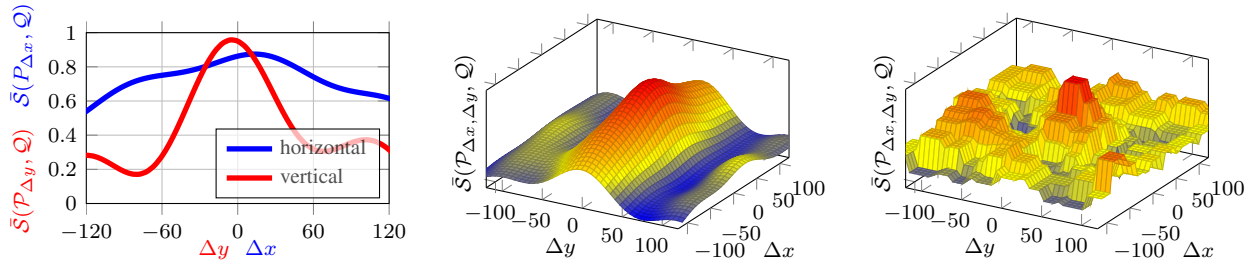

Figure 4. Alignment results for the example in Figure 3. Similarity as a function of translation (in pixels): independent 1D projections (left), the full 2D translation (middle), and the 2D binarized polynomial (right). At zero translation the centers of the sketch and the image are aligned. The detection in magenta color (Figure 3) is based on the similarity shown at the left, while the one in green is based on that shown in the middle.

the highest score. The second projection is only evaluated on this pre-shortlist. Now, the shortlist based on the value of $\bar{\mathcal{S}}_x + \bar{\mathcal{S}}_y$ is a sub-set of the pre-shortlist. The first projection used is query dependent, the one with higher variance in the relevant coordinate in the sketch query is used. Discriminative projection first is denoted as $\bar{\mathcal{S}}_> \xrightarrow{+} \bar{\mathcal{S}}_<$.

Re-ranking by local refinement. The 1D alignment provides, besides the scores, the scale and 1D translations $(\hat{x}_1, \hat{y}_1)$ maximizing the 1D projection scores, which often is a rough approximation of the full 2D alignment (see Figure 3 and 4). In this approach, the full similarity $\mathcal{S}(\mathcal{P}_{\Delta x,\Delta y}, \mathcal{Q})$ is only evaluated evaluated for a small neighborhood of $(\hat{x}_1, \hat{y}_1)$ on a fixed 2D grid. Sketch similarity computed by this method is denoted by $\mathcal{S}_{x/y}$.

Re-ranking by binary polynomial. We efficiently approximate the second order polynomial by a corresponding one that has binary coefficients and variables (*i.e.* $\cos(\omega_x \Delta x)\cos(\omega_y \Delta y)$ is binarized). We simply binarize both by a sign function. The similarity approximation for 2D translation is given by dot product between binary vectors which is faster to compute. Translation maximizing the binary approximation is found, and $\mathcal{S}(\mathcal{P}_{\Delta x,\Delta y}, \mathcal{Q})$ is only computed on a small neighborhood, as in the local refinement. Figure 4 shows an example where the position of the maximum on the 2D map of similarities for the binarized case remains close to that of the real valued one. Experiments show that the binary polynomials provide very good estimate of the translation. We denote this method by $\mathcal{S}_{xy\star}$.

Query expansion. Query Expansion (QE) is a standard approach to improve retrieval results by a new query that exploits the top-ranked results [10, 12, 21]. Unlike the original query, the QE is performed on image descriptors, the sketch descriptors are only used for localization. A global CNN image descriptor is used for QE, in particular off-the-shelf CroW [22] with VGG16 network [34]. The 512D image descriptor extracted per database image is compressed using product quantization [19] into 64 bytes. A basic version of an Average Query Expansion (AQE) [10] is used. CroW descriptors of the top results are averaged and a query is issued.

6. Experiments

We briefly summarize the design choices of the indexing and search procedure of our sketch-based retrieval. Then, we evaluate our method and compare to the state of the art.

Indexing (offline stage). All database images are down-sampled to have the longer side equal to 400 pixels. The edges are detected by off-the-shelf detector of Dollár and Zitnick [13]. The output edge strength is used as p_w, while all edges with strengths lower than 0.2 are completely discarded. A single sketch descriptor per database image is computed with AFM (14). Three kernels are used to search at three scales. Finally, the corresponding ℓ_2 norms for normalizing similarity (3) are computed and stored.

Query (online stage). The sketch query is cropped with a tight bounding box and resized similarly to database images. Two additional scales are given by down-sampling to 80% and 60%. Different query scales need to be matched with different kernels; smaller scale is matched with narrower kernel. The kernels shown in Figure 2 are used accordingly. The orientation kernel has $\sigma_\phi = 0.8$. One query descriptor per kernel is constructed (13). Additionally, each query is also horizontally mirrored.

The translations to be evaluated are fixed in a uniform way. Maximum translation is set to 80 pixels towards both directions and the step is 20. These are used for the maximum query size, while for different scales the maximum translation (step) is increased (decreased) linearly according to the relative query scale. That means, the localization is finer for smaller scales. Similarity is computed per scale independently and maximum similarity is kept.

The descriptor dimensionality is given by the number of frequencies $|\Omega_x|$ and $|\Omega_\phi|$. For instance, a compact setting of $|\Omega_x| = 5$ and $|\Omega_\phi| = 2$ lead to a 243D descriptor, while a high-performance settings of $|\Omega_x| = 6$ and $|\Omega_\phi| = 3$ lead to a 605D descriptor. In all cases, 9 additional scalars per image are stored (normalization of the 2D descriptor, normalizations of the 1D projections, all for 3 different scales).

Method identification. The following notation is used to identify the method, *ranking method* $\rightarrow$ *re-ranking method* (*number of re-ranked images*). Usage of average query expansion using n top images is denoted by QEn.

Method	P@20	Method	P@20
EI [7]	27.9	$\bar{\mathcal{S}}_{xy}$ $(5,2)$	57.9
Riemenschneider [28]	58.0	$\bar{\mathcal{S}}_{xy}$ $(6,3)$	61.4
SYM-FISH [6]	34.0	$\bar{\mathcal{S}}_{xy}$ $(5,2)$ + QE3	77.9
CS+GC [25]	49.3	$\bar{\mathcal{S}}_{xy}$ $(6,3)$ + QE3	**79.3**

Table 1. Performance comparison on the ETHZ extended shape dataset. Average precision at top 20 results is reported. We have not performed query mirroring for these results. The number of frequencies $(|\Omega_x|, |\Omega_\phi|)$ used is reported next to our methods.

Method	mAP	Method	mAP
GF-HOG [18]	12.2	$\bar{\mathcal{S}}_x + \bar{\mathcal{S}}_y \rightarrow \bar{\mathcal{S}}_{xy}$ (1k)	26.7
SHELO [29]	12.3	$\bar{\mathcal{S}}_x + \bar{\mathcal{S}}_y \rightarrow \bar{\mathcal{S}}_{xy}$ (5k)	29.2
LKS [30]	24.5	$\bar{\mathcal{S}}_{xy}$	30.4
GF-HOG [4]	18.2	$\bar{\mathcal{S}}_{xy}$ + QE3	**57.9**

Table 2. Performance comparison via mean Average Precision on the Flickr15k dataset.

6.1. Datasets and evaluation protocol

Constructing large scale ground-truth for sketch-based retrieval systems is not as easy as for traditional retrieval. One reason is the inherent abstraction of sketches. Moreover, positive images should not only comprised images of the same object/category, but also images depicting shapes similar to that of the query. Ground-truth at large scale should be on per query basis and this is not easy to achieve.

We initially evaluate our method on two image collections that are accompanied with ground-truth. These are the ETHZ extended shape dataset [31] and the Flickr15k dataset [18]. They consist of 285 images with 7 queries and 15K images with 330 queries (30 categories), respectively.

We further perform experiments on the large-scale dataset by Parui and Mittal [25] comprised 1.2M images and 175 queries, which has no available annotation. External annotators have manually evaluated the top results.

6.2. Evaluation and comparisons

Performance versus dimensionality. We construct the proposed sketch descriptor using our LDFM-based multiple kernel approximation and using the Fourier-based one. We compare performance for varying number of frequencies and present results in Figure 5 (left). The two methods have roughly the same performance for large number of frequencies where the kernel approximation is relatively good for both cases. The Fourier-based method significantly harms the performance for low number of frequencies due to its bad approximation. The orientation kernel is well approximated with few frequencies due to its wide shape (larger σ). We finally set $|\Omega_x| = 5$ and $|\Omega_\phi| = 2$ for the rest of our experiments, except if otherwise stated. Sketch descriptor $\mathbf{V}(\mathcal{P})$ has 243 dimensions, while $\mathbf{V}_x(\mathcal{P})$ only 27.

Ranking method. We compare ranking of the database with $\bar{\mathcal{S}}_{xy}$ and the projection-based approaches $\bar{\mathcal{S}}_x$ and $\bar{\mathcal{S}}_y$. In the latter case, only the top-ranked images are re-ranked by $\bar{\mathcal{S}}_{xy}$ to evaluate the performance loss. Results are shown in Figure 5 (middle). Ranking with sum of $\bar{\mathcal{S}}_y$ and $\bar{\mathcal{S}}_x$ appears significantly better than their individual use, while re-ranking one third of the database already recovers the performance loss. Speeding-up the ranking by $\bar{\mathcal{S}}_> \overset{+}{\rightarrow} \bar{\mathcal{S}}_<$, while in the end we re-rank 1k images, achieves mAP equal to 26.8. The drop is insignificant compared to the 26.9 in Figure 5 when re-ranking 1k images. Always ranking first with x or y projection, instead of our query dependent approach, gives 25.8 and 26.0 respectively.

Method	Dim	Time	DB	P@5	@10	@25	@50
$\bar{\mathcal{S}}_{xy}$ (1.2M)~~AFM~~ [38]	(8,3)	55.4	15.3	43.2	40.9	37.2	33.8
$\bar{\mathcal{S}}_{xy}$ (1.2M)~~AFM~~ [38]	(5,2)	20.2	3.3	25.8	24.7	22.5	20.2
$\bar{\mathcal{S}}_{xy}$ (1.2M)	(8,3)	55.4	5.1	50.1	46.7	42.0	37.2
$\bar{\mathcal{S}}_{xy}$ (1.2M)	(5,2)	20.2	1.1	45.8	44.1	38.5	35.4
$\bar{\mathcal{S}}_x + \bar{\mathcal{S}}_y \rightarrow \bar{\mathcal{S}}_{xy\star}$ (50k)	(6,3)	3.5	2.8	49.7	47.4	41.3	36.8
$\bar{\mathcal{S}}_> \overset{+}{\rightarrow} \bar{\mathcal{S}}_< \rightarrow \bar{\mathcal{S}}_{xy\star}$ (50k)	(6,3)	2.5	2.8	49.6	47.3	41.0	36.6
$\bar{\mathcal{S}}_> \overset{+}{\rightarrow} \bar{\mathcal{S}}_< \rightarrow \bar{\mathcal{S}}_{xy\star}$ (50k)†	(6,3)	2.5	0.7	50.3	47.3	41.5	36.7
$\bar{\mathcal{S}}_x + \bar{\mathcal{S}}_y \rightarrow \bar{\mathcal{S}}_{xy\star}$ (50k)	(5,2)	2.5	1.1	45.8	44.2	38.4	35.3
$\bar{\mathcal{S}}_> \overset{+}{\rightarrow} \bar{\mathcal{S}}_< \rightarrow \bar{\mathcal{S}}_{xy\star}$ (50k)	(5,2)	1.7	1.1	45.7	44.2	38.3	35.1
$\bar{\mathcal{S}}_> \overset{+}{\rightarrow} \bar{\mathcal{S}}_< \rightarrow \bar{\mathcal{S}}_{xy\star}$ (50k)†	(5,2)	1.7	0.3	45.6	43.5	38.0	35.0
$\bar{\mathcal{S}}_> \overset{+}{\rightarrow} \bar{\mathcal{S}}_< \rightarrow \bar{\mathcal{S}}_{xy\star}$ (50k)†+QE3	(6,3)	2.7	0.8	55.2	57.4	57.4	57.5
$\bar{\mathcal{S}}_> \overset{+}{\rightarrow} \bar{\mathcal{S}}_< \rightarrow \bar{\mathcal{S}}_{xy\star}$ (50k)†+QE10	(6,3)	2.7	0.8	63.0	63.4	64.8	65.2
$\bar{\mathcal{S}}_> \overset{+}{\rightarrow} \bar{\mathcal{S}}_< \rightarrow \bar{\mathcal{S}}_{xy\star}$ (50k)†+QE3	(5,2)	1.9	0.4	50.9	52.2	52.5	52.4
$\bar{\mathcal{S}}_> \overset{+}{\rightarrow} \bar{\mathcal{S}}_< \rightarrow \bar{\mathcal{S}}_{xy\star}$ (50k)†+QE10	(5,2)	1.9	0.4	56.4	56.8	57.3	57.8

Table 3. Performance, query time (in seconds) and database (DB) memory (in GB) requirements comparison on the 1.2M image dataset [25]. We report precision at n top ranked images (P@n). The number of frequencies $(|\Omega_x|, |\Omega_\phi|)$ is reported, which defines the final dimensionality (Dim = 1125, 605 or 243). ~~AFM~~: Asymmetric feature maps are not used. †: Vector components uniformly quantized into 1 byte.

Approximations. We perform re-ranking based on $\bar{\mathcal{S}}_{xy}$ and its two approximations. We use approximation $\bar{\mathcal{S}}_{xy\star}$ to efficiently search over all translations and scales, while we finally refine the translation of maximum similarity. On the other hand, $\bar{\mathcal{S}}_{x/y}$ is used to refine $(\hat{x}_1, \hat{y}_1)$ and acts only on the best scale found by the ranking method. In some cases the ranking method misses the correct scale and this is the main reason for the performance difference between the two. Results are shown in Figure 5 (right).

Comparisons to other methods. Comparison of our method to other methods is reported in Table 1 for the ETHZ extended shape dataset and in Table 2 for the Flickr15k dataset. The scores achieved without the QE are the highest reported on both benchmarks. The QE gives additional significant boost in the performance. On Flickr15k we remarkably outperform the previous state of the art by 24 points of mAP.

Large scale evaluation. We evaluate our method at large scale with the 1.2M dataset [25]. For each query, only top-ranked images are annotated as either negative, positive or similar. Images marked as similar are images of similar shape but different category than the query. Retrieval examples are shown in Figure 6 and performance comparison is presented in Table 3. We measure precision at top-ranked images per query and report average precision on top ranked

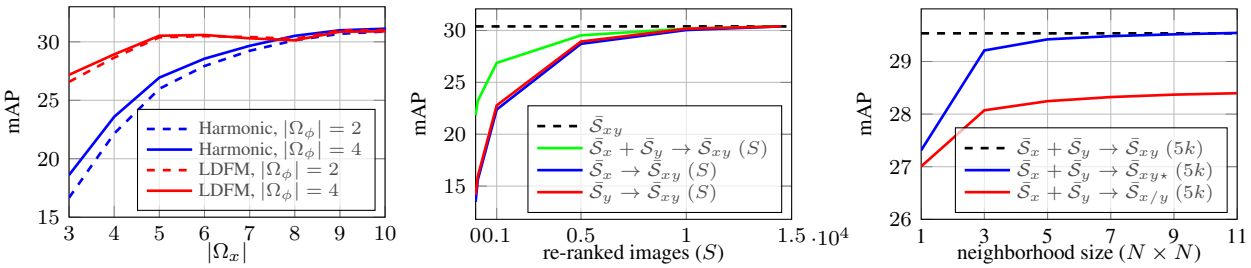

Figure 5. Performance comparison by measuring mean Average Precision (mAP) on the Flickr15k dataset. **Left**: Performance for increasing number of frequencies. Comparison between the Fourier-based approach [39] that uses harmonic frequencies and our joint optimization of the 3 kernel functions. Ranking is performed with $\bar{\mathcal{S}}_{xy}$. **Middle**: Comparison between the proposed methods for ranking the whole dataset. Re-ranking is additionally performed with $\bar{\mathcal{S}}_{xy}$ in all cases. We show mAP versus the number of re-ranked images. $S = 0$ signifies no re-ranking. **Right**: Performance of approximate re-ranking methods for increasing size of local refinement neighborhood. We show mAP versus the neighborhood size, while re-ranking 5k images.

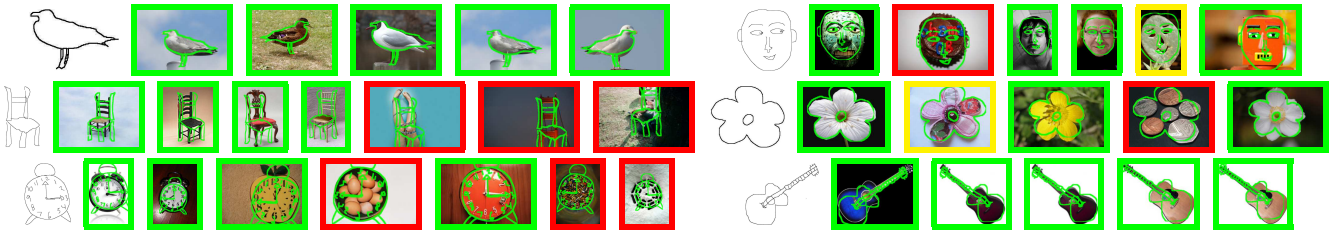

Figure 6. Examples of top-ranked retrieval images on the 1.2M dataset using our method. Localization of the sketch is shown in green color. Image borders denote positive (green), negative (red) and similar (yellow) image.

images over all queries.

We additionally evaluate performance when applying the trigonometric polynomial of Tolias *et al.* [38] to rank all database images. The proposed method by construction requires less memory and is significantly faster. It is also shown to perform better. The memory footprint is significantly decreased due to the asymmetry of our representation and due to good performance achieved with few frequencies. Encoding each vector component with 1 byte instead of single precision does not harm the performance.

Note that the discriminative-projection-first method only slightly decreases the performance, while it decreases the initial ranking time by 40%. Moreover, re-ranking only top 50k images performs with insignificant losses compared to ranking all images with the 2D polynomial. Finally, query expansion significantly improves the results. CNN descriptors are encoded with product quantization [19][4].

Query timings. The execution time was measured on the 1.2M image dataset using a single threaded MATLAB/Mex implementation on a 3.5GHz desktop machine. The results are summarized in Table 3. For $|\Omega_x| = 5$ and $|\Omega_\phi| = 2$, a query takes on average 1.81s for the initial ranking with $\bar{\mathcal{S}}_x + \bar{\mathcal{S}}_y$ and 0.72s for the top 50k re-ranking with $\bar{\mathcal{S}}_{xy\star}$ (with 3x3 neighborhood), giving a total time of 2.5s. Using $\bar{\mathcal{S}}_> \xrightarrow{+} \bar{\mathcal{S}}_<$, and computing the second projection only on $3 \cdot 50$k top-ranked images, ranking time drops to 1.05s. The values are independent of query complexity. The re-ranking using binary $\bar{\mathcal{S}}_{xy\star}$ is 17% faster compared to full $\bar{\mathcal{S}}_{xy}$. Applying the trigonometric polynomial scoring for ranking all images with the method of Tolias *et al.* [38] takes 20s, one order of magnitude slower than ours, for a low performance setup, while 55s with higher dimensionality and better performance which is still lower than ours.

The performance comparison to the work of Parui and Mittal [25] is not possible on the 1.2M dataset, as they use their own category-level ground truth, which is not publicly available. The comparison in terms of memory footprint (6.5GB is reported[25]) and execution time (1-5 sec per query is reported[25]) is favorable for the proposed method.

7. Conclusions

We have introduced a novel concept of asymmetric (explicit) feature maps. AFM allow to evaluate multiple kernels between a query and database entries with no additional memory requirements. The feature maps are optimally constructed by a joint kernel approximation, which turns out to be crucial for the accuracy. We have introduced a method of efficient approximation of scoring by trigonometric polynomials through 1D projections, which are a special case of asymmetric feature maps.

We have demonstrated the benefits of AFM on sketch-based image retrieval with short codes. We achieve state-of-the-art performance on a number of standard benchmarks. Compared with previous approaches using trigonometric polynomials [38], the proposed method achieves an order of magnitude speed-up, multiple-fold reduction in data storage, while improving the retrieval accuracy at the same time. The performance is further boosted by image-based average query expansion combined with AFM for object outline localization.

[4] After evaluation we discovered that the dataset contains a small amount of training ImageNet images, which can potentially affect with QE by CNN descriptors. Preliminary tests show that it affects insignificantly.

References

[1] Local features: State of the art, open problems and performance evaluation. http://www.iis.ee.ic.ac.uk/ComputerVision/DescrWorkshop/. 2

[2] A. C. Berg, T. L. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondences. In *CVPR*, 2005. 1

[3] L. Bo and C. Sminchisescu. Efficient match kernel between sets of features for visual recognition. In *NIPS*, Dec. 2009. 1, 2

[4] T. Bui and J. Collomosse. Scalable sketch-based image retrieval using color gradient features. In *ICCV*, 2015. 2, 7

[5] A. Bursuc, G. Tolias, and H. Jégou. Kernel local descriptors with implicit rotation matching. In *ICMR*, 2015. 1, 2

[6] X. Cao, H. Zhang, S. Liu, X. Guo, and L. Lin. Sym-fish: A symmetry-aware flip invariant sketch histogram shape descriptor. In *ICCV*. IEEE, 2013. 2, 7

[7] Y. Cao, C. Wang, L. Zhang, and L. Zhang. Edgel index for large-scale sketch-based image search. In *CVPR*. IEEE, 2011. 1, 2, 7

[8] A. Chalechale, G. Naghdy, and A. Mertins. Sketch-based image matching using angular partitioning. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 35(1):28–41, 2005. 1, 2

[9] O. Chum. Low dimensional explicit feature maps. In *ICCV*, 2015. 2, 3, 4

[10] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *ICCV*, 2007. 6

[11] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *ECCV Workshop Statistical Learning in Computer Vision*, May 2004. 2

[12] Q. Danfeng, S. Gammeter, L. Bossard, T. Quack, and L. V. Gool. Hello neighbor: Accurate object retrieval with k-reciprocal nearest neighbors. In *CVPR*, 2011. 6

[13] P. Dollár and C. L. Zitnick. Structured forests for fast edge detection. In *ICCV*, 2013. 6

[14] M. Eitz, J. Hays, and M. Alexa. How do humans sketch objects? *ACM Transactions on Graphics*, 2012. 2

[15] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa. An evaluation of descriptors for large-scale image retrieval from sketched feature lines. *Computers & Graphics*, 34(5):482–498, 2010. 2

[16] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa. Sketch-based image retrieval: Benchmark and bag-of-features descriptors. *Visualization and Computer Graphics, IEEE Transactions on*, 17(11):1624–1636, 2011. 1, 2

[17] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid. Groups of adjacent contour segments for object detection. *Trans. PAMI*, 30(1):36–51, 2008. 1

[18] R. Hu and J. Collomosse. A performance evaluation of gradient field hog descriptor for sketch based image retrieval. *CVIU*, 117(7):790–806, 2013. 2, 7

[19] H. Jégou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *Trans. PAMI*, 33(1):117–128, Jan. 2011. 6, 8

[20] H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *CVPR*, 2010. 2

[21] A. Joly and O. Buisson. Logo retrieval with a contrario visual query expansion. In *ACM Multimedia*, Oct. 2009. 6

[22] Y. Kalantidis, C. Mellina, and S. Osindero. Cross-dimensional weighting for aggregated deep convolutional features. In *ECCVW*, 2016. 6

[23] I. Kokkinos and A. Yuille. Inference and learning with hierarchical shape models. *IJCV*, 93(2):201–225, 2011. 1

[24] C. Ma, X. Yang, C. Zhang, X. Ruan, M.-H. Yang, and O. Coporation. Sketch retrieval via dense stroke features. In *BMVC*, 2013. 2

[25] S. Parui and A. Mittal. Similarity-invariant sketch-based image retrieval in large databases. In *ECCV*, pages 398–414. Springer, 2014. 2, 7, 8

[26] F. Perronnin and C. R. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, 2007. 2

[27] A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *NIPS*, 2007. 1, 3

[28] H. Riemenschneider, M. Donoser, and H. Bischof. Image retrieval by shape-focused sketching of objects. In *Computer Vision Winter Workshop*, 2011. 2, 7

[29] J. M. Saavedra. Sketch based image retrieval using a soft computation of the histogram of edge local orientations (s-helo). In *ICIP*, 2014. 2, 7

[30] J. M. Saavedra, J. M. Barrios, and S. Orand. Sketch based image retrieval using learned keyshapes (lks). In *BMVC*, 2015. 1, 2, 7

[31] K. Schindler and D. Suter. Object detection by global contour shape. *Pattern Recognition*, 41(12):3736–3748, 2008. 7

[32] B. Scholkopf and A. Smola. *Learning with Kernels*. MIT Press, 2002. 3

[33] A. Shrivastava, T. Malisiewicz, A. Gupta, and A. A. Efros. Data-driven visual similarity for cross-domain image matching. *ACM Transactions on Graphics*, 30(6):154, 2011. 2

[34] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2014. 6

[35] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *ICCV*, 2003. 2

[36] X. Sun, C. Wang, C. Xu, and L. Zhang. Indexing billions of images for sketch-based retrieval. In *ACM Multimedia*, 2013. 1, 2

[37] A. Thayananthan, B. Stenger, P. H. Torr, and R. Cipolla. Shape context and chamfer matching in cluttered scenes. In *CVPR*, 2003. 2

[38] G. Tolias, A. Bursuc, T. Furon, and H. Jégou. Rotation and translation covariant match kernels for image retrieval. *CVIU*, 140:9–20, 2015. 1, 2, 3, 4, 5, 7, 8

[39] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. *Trans. PAMI*, 34(3):480–492, Mar. 2012. 1, 2, 3, 8

[40] Q. Yu, Y. Yang, Y.-Z. Song, T. Xiang, and T. M. Hospedales. Sketch-a-net that beats humans. In *BMVC*, 2015. 2