

---

# Robust Matching and Recognition using Context-Dependent Kernels

---

Hichem Sahbi<sup>1</sup>

Jean-Yves Audibert<sup>2,3</sup>

Jaonary Rabarisoa<sup>2</sup> and Renaud Keriven<sup>2</sup>

SAHBI@TELECOM-PARISTECH.FR

AUDIBERT@CERTIS.ENPC.FR

{RABARISO,KERIVEN}@CERTIS.ENPC.FR

<sup>1</sup> CNRS - LTCI, UMR 5141, Telecom ParisTech.

<sup>2</sup> Certis - Ecole des Ponts, 6 avenue Blaise Pascal, Cité Descartes, 77455 Marne-la-Vallée France

<sup>3</sup> Willow - ENS / INRIA, 45 rue d'Ulm, 75005 Paris, France

## Abstract

*The success of kernel methods including support vector machines (SVMs) strongly depends on the design of appropriate kernels. While initially kernels were designed in order to handle fixed-length data, their extension to unordered, variable-length data became more than necessary for real pattern recognition problems such as object recognition and bioinformatics.*

*We focus in this paper on object recognition using a new type of kernel referred to as “context-dependent”. Objects, seen as constellations of local features (interest points, regions, etc.), are matched by minimizing an energy function mixing (1) a fidelity term which measures the quality of feature matching, (2) a neighborhood criteria which captures the object geometry and (3) a regularization term. We will show that the fixed-point of this energy is a “context-dependent” kernel (“CDK”) which also satisfies the Mercer condition. Experiments conducted on object recognition show that when plugging our kernel in SVMs, we clearly outperform SVMs with “context-free” kernels.*

## 1. Introduction

Object recognition is one of the biggest challenges in vision and its interest is still growing (Everingham et al., 2007). Among existing methods, those based on machine learning (ML), show a particular interest as they are performant and theoretically well grounded (Bishop, 2007). ML approaches, such as the popular support vector networks (Boser et al., 1992), basically require the design of similarity measures, also referred to as *kernels*, which should provide high values when two objects share similar structures/appearances and should be invariant, as much as possible, to the linear

and non-linear transformations. Kernel-based object recognition methods were initially *holistic*, i.e., each object is mapped into one or multiple fixed-length vectors and a similarity, based on color, texture or shape (Swain & Ballard, 1991; Chapelle et al., 1999), is then defined. *Local* kernels, i.e., those based on bags or local sets were introduced in order to represent data which cannot be represented by ordered and fixed-length feature vectors, such as graphs, trees, interest points, etc (Gartner, 2003). It is well known that both holistic and local kernels should satisfy certain properties among them the positive definiteness, low complexity for evaluation, flexibility in order to handle variable-length data and also invariance. Holistic kernels have the advantage of being simple to evaluate, discriminating but less flexible than local kernels in order to handle invariance<sup>1</sup>. While the design of kernels gathering flexibility, invariance and low complexity is a challenging task; the proof of their positive definiteness is sometimes harder (Cuturi, 2005). This property also known as the Mercer condition ensures, according to Vapnik’s SVM theory (Vapnik, 1998), optimal generalization performance and also the uniqueness of the SVM solution.

Consider a database of objects (images), each one seen as a constellation of local features, for instance interest points (Schmid & Mohr, 1997; Lowe, 2004), extracted using any suitable filter (Harris & Stephens, 1988). Again, original holistic kernels explicitly (or implicitly) map objects into fixed-length feature vectors and take the similarity as a decreasing function of any well-defined distance (Barla et al., 2002). In contrast to holistic kernels, local ones are designed in order to handle variable-length and unordered

---

Appearing in *Proceedings of the 25<sup>th</sup> International Conference on Machine Learning*, Helsinki, Finland, 2008. Copyright 2008 by the author(s)/owner(s).

---

<sup>1</sup>In case of object recognition, invariance means robustness to occlusion, geometric transformations and illumination.

data. Two families of local kernels can be found in the literature; those based on statistical “length-insensitive” measures such as the Kullback Leibler divergence, and those which require a preliminary step of alignment. In the first family, the authors in (Kondor & Jebara, 2003; Moreno et al., 2003) estimate for each object (constellation of local features) a probability distribution and compute the similarity between two objects (two distributions) using the “Kullback Leibler divergence” in (Moreno et al., 2003) and the “Bhattacharyya affinity” in (Kondor & Jebara, 2003). Only the function in (Kondor & Jebara, 2003) satisfies the Mercer condition and both kernels were applied for image recognition tasks. In (Wolf & Shashua, 2003), the authors discuss a new type of kernel referred to as “principal angles” which is positive definite. Its definition is based on the computation of the principal angles between two linear subspaces under an orthogonality constraint. The authors demonstrate the validity of their method on visual recognition tasks including classification of motion trajectory and face recognition. In this family of kernels, the main drawback resides is the strong assumption about the used probabilistic models in order to approximate the set of local features which may not hold true in practice.

In the second family, the “max” kernel (Wallraven et al., 2003) considers the similarity function, between two feature sets, as the sum of their matching scores and unlike discussed in (Wallraven et al., 2003) this kernel is actually not Mercer (Bahlmann et al., 2002). In (Lyu, 2005), the authors introduced the “circular-shift” kernel defined as a weighted combination of Mercer kernels using an exponent. The latter is chosen in order to give more prominence to the largest terms so the resulting similarity function approximates the “max” and also satisfies the Mercer condition. The authors combined local features and their relative angles in order to make their kernel rotation invariant and they show its performance for the particular task of object recognition. In (Boughorbel et al., 2005), the authors introduced the “intermediate” matching kernel, for object recognition, which uses virtual local features in order to approximate the “max” while satisfying the Mercer condition. Recently, (Grauman & Darrell, 2007) introduced the “pyramid-match” kernel, for object recognition and document analysis, which maps feature sets using a multi-resolution histogram representation and computes the similarity using a weighted histogram intersection. The authors showed that their function is positive definite and can be computed linearly with respect to the number of local fea-

tures. Other matching kernels include the “dynamic programming” function which provides, in (Bahlmann et al., 2002), an effective matching strategy for handwritten character recognition, nevertheless the Mercer condition is not guaranteed.

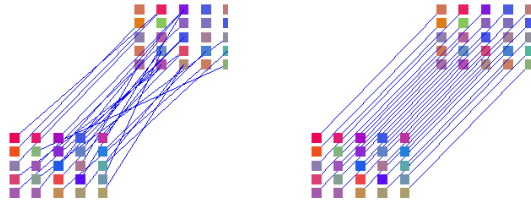


Figure 1. This figure shows a comparison of the matching results when using a naive matching strategy without geometry, (which consists in finding the set of possible matches by minimizing a distance between the color descriptors) and our “context-dependent” matching.

### 1.1. Motivation and Contribution

The success of the second family of local kernels strongly depends on the quality of alignments which are difficult to obtain mainly when images contain redundant and repeatable structures. Regardless the Mercer condition, a naive matching kernel (such as the “max”), which looks for all the possible alignments and sums the best ones, will certainly fail and results into many false matches (see Figures 1 and 2, left). The same argument is supported in (Schmid & Mohr, 1997), for the general problem of visual features matching, about the strong spatial correlation between interest points and the corresponding close local features in the image space. This limitation also appears in closely related areas such as text analysis, and particularly string alignment. A simple example, of aligning two strings (“Sir” and “Hi Sir”) using a simple similarity measure  $\mathbb{1}_{\{c_1=c_2\}}$  between any two characters  $c_1$  and  $c_2$ , shows that without any extra information about the *context* (i.e., the sub-string) surrounding each character in (“Sir” and “Hi Sir”), the alignment process results into false matches (See Table 1). *Hence, it is necessary to consider the context as a part of the alignment process when designing kernels.*

In this paper, we introduce a new kernel, called “context-dependent” (or “CDK”) and defined as the fixed-point of an energy function which balances an “alignment quality” term and a “neighborhood” criteria. The alignment quality is inversely proportional to the expectation of the Euclidean distance between the

Naive matching	'H'	'i'	'S'	'i'	'r'
'S'	0	0	-	1	0
'i'	0	<b>1</b>	-	0	<b>1</b>
'r'	0	0	-	0	1
Context-dependent	-	-	-	-	-
'S'	0	0	-	.38	0
'i'	0	<b>.36</b>	-	0	<b>.39</b>
'r'	0	0	-	0	.38

Table 1. This table shows a simple comparison between similarity measures when using naive matching (upper table) and context-dependent matching (lower table).

most likely aligned features (see Section 2) while the neighborhood criteria measures the spatial coherence of the alignments; given a pair of features  $(f_p, f_q)$  with a high alignment quality, the neighborhood criteria is proportional to the alignment quality of all the pairs close<sup>2</sup> to  $(f_p, f_q)$ . *The general form of “CDK” captures the similarity between any two features by incorporating also their context, i.e., the similarity of the surrounding features.* Our proposed kernel can be viewed as a variant of “dynamic programming” kernel (Bahlmann et al., 2002) where instead of using the ordering assumption we consider a neighborhood assumption which states that two points match if they have similar features and if they satisfies a neighborhood criteria i.e., their neighbors match too. This also appears in other well studied kernels such as Fisher (Jaakkola et al., 1999), which implements the conditional dependency between data using the Markov assumption. “CDK” also implements such dependency with an extra advantage of being the fixed-point and the (sub)optimal solution of an energy function closely related to the goal of our application. This goal is to gather the properties of flexibility, invariance and mainly discrimination by allowing each local feature to consider its context in the matching process. Notice that the goal of this paper is not to extend local features to be global and doing so (as in (Mortensen et al., 2005; Amores et al., 2005)) makes local features less invariant, but rather to design a similarity kernel (“CDK”) which captures the context while being invariant. Even though we investigate “CDK” in the particular task of object recognition, we can easily extend it to handle closely related areas in machine learning such as text alignment for document retrieval (Nie et al., 1999), machine translation (Sim et al., 2007) and bioinformatics (Scholkopf et al., 2004).

In the remainder of this paper we consider the

<sup>2</sup>The closeness is defined in Section 2.

following terminology and notation. A feature refers to a local interest point  $x_i^p = (\psi_g(x_i^p), \psi_f(x_i^p), y_p)$ , here  $i$  stands for the  $i^{th}$  sample of the subset  $\mathcal{S}_p = \{x_1^p, \dots, x_n^p\}$  and  $y_p \in \mathbb{N}^+$  is a unique indicator which provides the class or the subset including  $x_i^p$ .  $\psi_g(x_i^p) \in \mathbb{R}^2$  stands for the 2D coordinates of the interest-point  $x_i^p$  while  $\psi_f(x_i^p) \in \mathbb{R}^s$  corresponds to the descriptor of  $x_i^p$  (for instance the 128 coefficients of the SIFT(Lowe, 2004)). We define  $\mathcal{X}$  as the set of all possible features taken from all the possible images in the world and  $X$  is a random variable standing for a sample in  $\mathcal{X}$ . We also consider  $k_t : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  as a symmetric function which, given two samples  $(x_i^p, x_j^q)$ , provides a similarity measure. Other notations will be introduced as we go along through different sections of this paper which is organized as follows. We first introduce in Section 2, our energy function which makes it possible to design our context-dependent kernel and we show that this kernel satisfies the Mercer condition so we can use it for support vector machine training and other kernel methods. In Section 3 we show the application of this kernel in object recognition. We discuss in Section 4 the advantages and weaknesses of this kernel and the possible extensions in order to handle other tasks such as string matching and machine translation. We conclude in Section 5 and we provide some future research directions.

## 2. Kernel Design

Define  $\mathcal{X} = \cup_{p \in \mathbb{N}^+} \mathcal{S}_p$  as the set of all possible interest points taken from all the possible objects in the world. We assume that all the objects are sampled with a given cardinality i.e.,  $|\mathcal{S}_p| = n$ ,  $|\mathcal{S}_q| = m$ ,  $\forall p, q \in \mathbb{N}^+$  ( $n$  and  $m$  might be different). Our goal is to design a kernel  $K$  which provides the similarity between any two objects (subsets)  $\mathcal{S}_p, \mathcal{S}_q$  in  $\mathcal{X}$ .

**Definition 1 (Subset Kernels)** *let  $\mathcal{X}$  be an input space, and consider  $\mathcal{S}_p, \mathcal{S}_q \subseteq \mathcal{X}$  as two finite subsets of  $\mathcal{X}$ . We define the similarity function or kernel  $K$  between  $\mathcal{S}_p = \{x_i^p\}$  and  $\mathcal{S}_q = \{x_j^q\}$  as  $K(\mathcal{S}_p, \mathcal{S}_q) = \sum_i^n \sum_j^m k(x_i^p, x_j^q)$ .*

here  $k$  is symmetric and continuous on  $\mathcal{X} \times \mathcal{X}$ , so  $K$  will also be continuous and symmetric. Since  $K$  is defined as the cross-similarity  $k$  between all the possible sample pairs taken from  $\mathcal{S}_p \times \mathcal{S}_q$ , it is obvious that  $K$  has the big advantage of not requiring any (hard) alignment between the samples of  $\mathcal{S}_p$  and  $\mathcal{S}_q$ . Nevertheless, for a given  $\mathcal{S}_p, \mathcal{S}_q$ , the value of  $K(\mathcal{S}_p, \mathcal{S}_q)$  should be dominated by  $\sum_i \max_j k(x_i^p, x_j^q)$ , so  $k$  should be appropriately designed (see Section 2.1).

Let  $X$  be a random variable standing for samples taken from  $\mathcal{S}_p$  and  $X'$  is defined in a similar way for the subset  $\mathcal{S}_q$ . We design our kernel  $k(x_i^p, x_j^q) = \mathbb{P}(X' = x_j^q, X = x_i^p)$  as the *joint probability* that  $x_j^q$  matches  $x_i^p$ . Again, it is clear enough (see Figures 1,2 and Table 1) that when this joint probability is estimated using only the sample coordinates (without their contexts), this may result into many false matches and wrong estimate of  $\{\mathbb{P}(X' = x_j^q, X = x_i^p)\}_{i,j}$ .

Before describing the whole design of  $k$ , we start with our definition of context-dependent kernels.

**Definition 2 (Context-Dependent Kernels)** *we define a context-dependent kernel  $k$  as any symmetric, continuous and recursive function  $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  such that  $k(x_i^p, x_j^q)$  is equal to*

$$c(x_i^p, x_j^q) \times h \left( \sum_{k,\ell} k(x_k^p, x_\ell^q) \mathbb{V}(x_i^p, x_k^p, x_j^q, x_\ell^q) \right),$$

here  $c$  is a positive (semi) definite and context-free (non-recursive) kernel,  $\mathbb{V}(x, x', y, y')$  is a monotonic decreasing function of any (pseudo) distance involving  $(x, x', y, y')$  and  $h(x)$  is monotonically increasing.

## 2.1. Approach

We consider the issue of designing  $k$  using a variational framework. Let  $\mathcal{I}_p = \{1, \dots, n\}$ ,  $\mathcal{I}_q = \{1, \dots, m\}$ ,  $\mu = \{k(x_i^p, x_j^q)\}$ ,  $d(x_i^p, x_j^q) = \|\psi_f(x_i^p) - \psi_f(x_j^q)\|_2$  and  $\mathcal{N}_p(x_i^p) = \{x_k^p \in \mathcal{S}_p : k \neq i, \|\psi_g(x_i^p) - \psi_g(x_k^p)\|_2 \leq \epsilon_p\}$  ( $\epsilon_p$  defines a neighborhood and  $\mathcal{N}_q$  is defined in the same way for  $\mathcal{S}_q$ ). Consider  $\alpha, \beta \geq 0$ ,  $(i, j) \in \mathcal{I}_p \times \mathcal{I}_q$ ,  $\mu = \{k(x_i^p, x_j^q)\}$  is found by solving

$$\begin{aligned} \min_{\mu} \quad & \sum_{i \in \mathcal{I}_p, j \in \mathcal{I}_q} k(x_i^p, x_j^q) d(x_i^p, x_j^q) + \\ & \beta \sum_{i \in \mathcal{I}_p, j \in \mathcal{I}_q} k(x_i^p, x_j^q) \log(k(x_i^p, x_j^q)) + \\ \alpha \quad & \sum_{i \in \mathcal{I}_p, j \in \mathcal{I}_q} k(x_i^p, x_j^q) \left( - \sum_{\substack{x_k^p \in \mathcal{N}_p(x_i^p), \\ x_\ell^q \in \mathcal{N}_q(x_j^q)}} k(x_k^p, x_\ell^q) \right) \\ \text{s.t.} \quad & k(x_i^p, x_j^q) \in [0, 1], \quad \sum_{i,j} k(x_i^p, x_j^q) = 1 \end{aligned} \quad (1)$$

The first term measures the quality of matching two

descriptors  $\psi_f(x_i^p)$ ,  $\psi_f(x_j^q)$ . In the case of SIFT, this is considered as the distance,  $d(x_i^p, x_j^q)$ , between the 128 SIFT coefficients of  $x_i^p$  and  $x_j^q$ . A high value of  $d(x_i^p, x_j^q)$  should result into a small value of  $k(x_i^p, x_j^q)$  and vice-versa.

The second term is a regularization criteria which considers that without any a priori about the aligned samples, the probability distribution  $\{k(x_i^p, x_j^q)\}$  should be flat so the negative of the entropy is minimized. This term also helps defining a simple solution and solving the constrained minimization problem easily. The third term is a neighborhood criteria which considers that a high value of  $k(x_i^p, x_j^q)$  should imply high kernel values in the neighborhoods  $\mathcal{N}_p(x_i^p)$  and  $\mathcal{N}_q(x_j^q)$ . This criteria makes it possible to consider the context (spatial configuration) of each sample in the matching process.

We formulate the minimization problem by adding an equality constraint and bounds which ensure that  $\{k(x_i^p, x_j^q)\}$  is a probability distribution.

**Proposition 1** (1) admits a solution in the form of a context-dependent kernel  $k_t(x_i^p, x_j^q) = v_t(x_i^p, x_j^q) / Z_t$ , with  $t \in \mathbb{N}^+$ ,  $Z_t = \sum_{i,j} v_t(x_i^p, x_j^q)$  and  $v_t(x_i^p, x_j^q)$  defined as

$$\begin{aligned} & \exp \left( - \frac{d(x_i^p, x_j^q)}{\beta} - 1 \right) \times \\ & \exp \left( \frac{2\alpha}{\beta} \sum_{k,\ell} \mathbb{V}(x_i^p, x_k^p, x_j^q, x_\ell^q) k_{t-1}(x_k^p, x_\ell^q) \right) \end{aligned} \quad (2)$$

which is also a Gibbs distribution.

**Proof.** the proof, lengthy, is omitted and it is available in a research report.

In (2), we set  $v_0$  to any positive definite kernel (see proposition 3) and we define  $\mathbb{V}(x_i^p, x_k^p, x_j^q, x_\ell^q)$  as  $g(x_i^p, x_k^p) \times g(x_j^q, x_\ell^q)$  where  $g$  is a decreasing function of any (pseudo) distance involving  $(x_i^p, x_k^p)$ , *not necessarily symmetric*. In practice, we consider  $g(x_i^p, x_k^p) = \mathbb{1}_{\{i=k\}} \times \mathbb{1}_{\{x_k^p \in \mathcal{N}_p(x_i^p)\}}$ . It is easy to see that  $k_t$  is a P-kernel on any  $\mathcal{S}_p \times \mathcal{S}_q$  (Haussler, 1999) (as the joint probability over sample pairs taken from any  $\mathcal{S}_p$  and  $\mathcal{S}_q$  sums to one), so the value of the subset kernel  $K(\mathcal{S}_p, \mathcal{S}_q)$  defined in (1) is constant and *useless*. To make  $k_t$  (up to a factor) a P-kernel on  $\mathcal{X} \times \mathcal{X}$  (and not on  $\mathcal{S}_p \times \mathcal{S}_q$ ), we cancel the equality constraint in (1) and we can prove in a similar way that  $k_t(x_i^p, x_j^q)$  is equal to  $v_t(x_i^p, x_j^q)$  which is still a context-dependent kernel.

## 2.2. Mercer Condition

Let  $\mathcal{X}$  be an input space and let  $k_t : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  be symmetric and continuous.  $k_t$  is Mercer, i.e., positive (semi) definite, if and only if any Gram (kernel scalar product) matrix built by restricting  $k_t$  to any finite subset of  $\mathcal{X}$  is positive (semi) definite. A Mercer kernel  $k_t$  guarantees the existence of a reproducing kernel Hilbert space  $\mathcal{H}$  where  $k_t$  can be written as a dot product i.e.,  $\exists \Phi_t : \mathcal{X} \rightarrow \mathcal{H}$  such that  $\forall x, x' \in \mathcal{X}$ ,  $k_t(x, x') = \langle \Phi_t(x), \Phi_t(x') \rangle$ .

**Proposition 2** *ex. (S-Taylor & Cristianini, 2000)*  
 the sum and the product of any two Mercer kernels is a Mercer kernel. The exponential of any Mercer kernel is also a Mercer kernel.

**Proof.** see, for instance, (S-Taylor & Cristianini, 2000).

Now, let us state *our result* about the positive definiteness of the ‘‘CDK’’ kernel.

### Proposition 3

consider  $g : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ , let  $\mathbb{V}(x_i^p, x_k^p, x_j^q, x_\ell^q) = g(x_i^p, x_k^p)g(x_j^q, x_\ell^q)$ , and  $k_0$  positive definite. The kernel  $k_t$  is then positive definite.

**Proof.** Initially ( $t = 0$ ),  $k_0$  is per definition a positive definite kernel. By induction, let us assume  $k_{t-1}$  a Mercer kernel i.e.,  $\exists \Phi_{t-1} : \mathcal{X} \rightarrow \mathcal{H}$  such that  $k_{t-1}(x, x') = \langle \Phi_{t-1}(x), \Phi_{t-1}(x') \rangle$ ,  $\forall x, x' \in \mathcal{X}$ . Now, the sufficient condition will be to show that  $(\sum_{y, y'} \mathbb{V}(x, y, x', y') k_{t-1}(y, y'))$  is also a Mercer kernel. Then, by the closure of the exponential and the product (see proposition 2),  $k_t$  will then be Mercer. We need to show

$$\forall x_1, \dots, x_d \in \mathcal{X}, \forall c_1, \dots, c_d \in \mathbb{R},$$

$$(*) = \sum_{i,j} c_i c_j \left( \sum_{y, y'} \mathbb{V}(x_i, y, x_j, y') k_{t-1}(y, y') \right) \geq 0$$

We have

$$\begin{aligned} (*) &= \sum_{i,j} c_i c_j \sum_{y, y'} g(x_i, y) g(x_j, y') k_{t-1}(y, y') \\ &= \sum_{y, y'} \left( \sum_i c_i g(x_i, y) \right) \times \\ &\quad \left( \sum_j c_j g(x_j, y') \right) k_{t-1}(y, y') \\ &= \sum_{y, y'} \gamma_y \gamma_{y'} k_{t-1}(y, y') \\ &= \left\| \sum_y \gamma_y \Phi_{t-1}(y) \right\|_{\mathcal{H}} \geq 0. \end{aligned}$$

**Corollary 1**  $K$  defined in (1) is also a Mercer kernel.

**Proof.** the proof is straightforward for the particular case  $n = m$ . As  $k_t(x_i^p, x_j^q) = \langle \Phi_t(x_i^p), \Phi_t(x_j^q) \rangle$ , we can write  $K(\mathcal{S}_p, \mathcal{S}_q) = \sum_{i,j} \langle \Phi_t(x_i^p), \Phi_t(x_j^q) \rangle = \langle \sum_i \Phi_t(x_i^p), \sum_j \Phi_t(x_j^q) \rangle$  and this corresponds to a dot product in some Hilbert space. The proof can be found in (S-Taylor & Cristianini, 2000) for the general case of finite subsets of any length.

## 2.3. Algorithm and Setting

The factor  $\beta$ , in  $k_t$ , acts as a scale parameter and it is selected using

$$\beta \leftarrow \mathbb{E}_r \left[ \mathbb{E}_{\{X_1^r, X_2^r : d(X_1^r, X_2^r) \leq \epsilon\}} [d(X_1^r, X_2^r)] \right] \quad (3)$$

here  $\mathbb{E}$  denotes the expectation and  $X_1^r$  (also  $X_2^r$ ) denotes a random variable standing for samples in  $\mathcal{S}_r$ . The coefficient  $\alpha$  controls the tradeoff between the alignment quality and the neighborhood criteria. It is selected by cross-validation and it should guarantee  $k_t(x_i^p, x_j^q) \in [0, 1]$ . If  $A = \sup_{i,j} \sum_{k,\ell} g(x_i^p, x_k^p) \times g(x_j^q, x_\ell^q)$ ,  $\alpha$  should then be selected in  $[0, \frac{\beta}{2A}]$ .

Consider  $P, Q$  as the intrinsic adjacency matrices of  $\mathcal{S}_p$  and  $\mathcal{S}_q$  respectively defined as  $P_{i,k} = g(x_i^p, x_k^p)$ ,  $Q_{j,\ell} = g(x_j^q, x_\ell^q)$ . Let  $U$  denotes the unit matrix and consider  $D_{i,j} = d(x_i^p, x_j^q)$ ,  $\mu_{i,j}^{(t)} = k_t(x_i^p, x_j^q)$ . Now,  $\mu_{i,j}^{(t)}$  is iteratively found using Algorithm (‘‘CDK’’) (see table 2) and converges to a fixed point (see. Section 2.4).

---

### Algorithm (CDK)

---

#### Initialization:

Set  $\beta$  using (3) and  $\alpha \in [0, \frac{\beta}{2A}]$   
 Set  $\mu^{(0)} \leftarrow k_0, t \leftarrow 0$

**Repeat** until  $t \rightarrow T_{max}$  or  $\|\mu^{(t)} - \mu^{(t-1)}\|_2 \rightarrow 0$

$$\mu^{(t)} \leftarrow \exp \left( -D/\beta + \frac{2\alpha}{\beta} P \mu^{(t-1)} Q - U \right)$$


---

Table 2. The ‘‘CDK’’ kernel evaluation.

## 2.4. Convergence

Let us assume  $0 \leq g \leq 1$ , and remind  $\mu^{(t)} \in \mathbb{R}^{n \times m}$  be the vector of components  $\mu_{i,j}^{(t)} = k_t(x_i^p, x_j^q)$ . Introduce the mapping  $f : \mathbb{R}^{n \times m} \rightarrow \mathbb{R}^{n \times m}$  defined by its



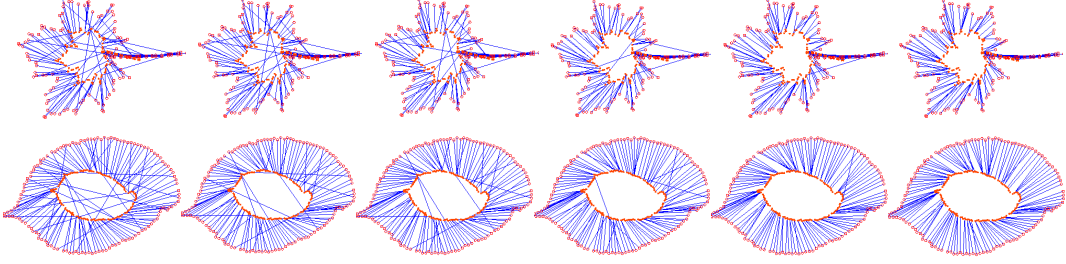


Figure 3. This figure shows the evolution of context-dependent silhouette matching on the Swedish set, for different and increasing values of  $\alpha$ . We clearly see that when  $\alpha$  increases the matching results are better. We set  $\beta = 0.1$  and  $t = 1$ .

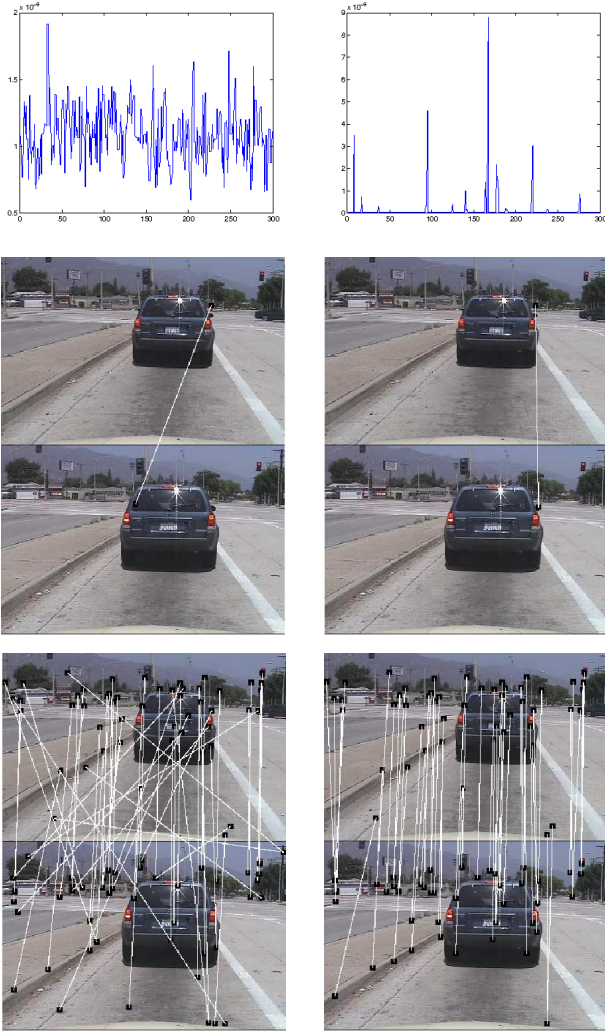


Figure 2. This figure shows a comparison of the matching results when using a naive matching strategy without geometry and our “context-dependent” kernel matching. (Top figures) show the distribution of the kernel values  $k(x_i, x_j)$ ,  $j \in \mathcal{I}_q$  using a context-free kernel (left) and our “CDK” kernel (right). We can clearly see that the highest value changes its location so the matching results are now corrected (as shown in middle figures). (Bottom) other matching results.

component  $f_{i,j}(v)$  as

$$\exp\left(-1 - \frac{d(x_i^p, x_j^q)}{\beta} + \frac{2\alpha}{\beta} \sum_{k,\ell} g(x_i^p, x_k^p) g(x_j^q, x_\ell^q) v_{k,\ell}\right)$$

By construction of the kernel  $k_t$ , we have  $\mu^{(t)} = f(\mu^{(t-1)})$ . Let  $A$  and  $B$  satisfy

$$\sup_{1 \leq i \leq n, 1 \leq j \leq m} \sum_{k,\ell} g(x_i^p, x_k^p) g(x_j^q, x_\ell^q) \leq A \quad (4)$$

$$\sum_{i,j} \exp\left(-1 - \frac{d(x_i^p, x_j^q)}{\beta}\right) \leq B \quad (5)$$

Consider  $L = \frac{2B\alpha}{\beta} \exp\left(\frac{2\alpha A}{\beta}\right)$ , and let

$\mathcal{B} = \{v \in \mathbb{R}^{n \times m} : \forall 1 \leq i \leq n, 1 \leq j \leq m, |v_{i,j}| \leq 1\}$  be the  $\|\cdot\|_\infty$ -ball of radius 1. Finally, let  $\|\cdot\|_1$  denote the 1-norm on  $\mathbb{R}^{n \times m}$ :  $\|u\|_1 = \sum_{1 \leq i \leq n, 1 \leq j \leq m} |u_{i,j}|$ .

**Proposition 4** *If  $\|\mu^{(0)}\|_\infty \leq 1$  and  $2\alpha A \leq \beta$ , then we have  $f(\mathcal{B}) \subset \mathcal{B}$ , and on  $\mathcal{B}$ ,  $f$  is  $L$ -Lipschitz for the norm  $\|\cdot\|_1$ .*

*In particular, if  $L < 1$ , then there exists a unique  $\tilde{v} \in \mathcal{B}$  such that  $f(\tilde{v}) = \tilde{v}$ , and the sequence  $(\mu^{(t)})$  satisfies*

$$\|\mu^{(t)} - \tilde{v}\|_1 \leq L^t \|\mu^{(0)} - \tilde{v}\|_1 \xrightarrow{t \rightarrow +\infty} 0. \quad (6)$$

**Proof.** The first assertion is proved by induction by checking that for  $\|v\|_\infty \leq 1$ , we have

$$\begin{aligned} f_{i,j}(v) &\leq \exp\left(-1 + \frac{2\alpha}{\beta} \sum_{k,\ell} g(x_i^p, x_k^p) g(x_j^q, x_\ell^q) v_{k,\ell}\right) \\ &\leq \exp\left(-1 + \frac{2\alpha}{\beta} A\right) \leq 1. \end{aligned}$$

For the second assertion, note that for any  $v$  in  $\mathcal{B}$ , we have  $|\frac{\partial f_{i,j}}{\partial v_{k,\ell}}(v)| \leq \exp\left(-1 - \frac{d(x_i^p, x_j^q)}{\beta}\right)$ . For any  $v, v'$  in  $\mathcal{B}$ , we have

$$\|f(v) - f(v')\|_1 = \sum_{i,j} |f_{i,j}(v) - f_{i,j}(v')| = (**)$$

$$\begin{aligned}
(**) &\leq \sum_{i,j} \exp\left(-1 - \frac{d(x_i^p, x_j^q)}{\beta}\right) \frac{2\alpha}{\beta} \exp\left(\frac{2\alpha}{\beta} A\right) \\
&\quad \times \sum_{k,\ell} \left|g(x_i^p, x_k^p)g(x_j^q, x_\ell^q)v_{k,\ell} \right. \\
&\quad \quad \left. - g(x_i^p, x_k^p)g(x_j^q, x_\ell^q)v'_{k,\ell}\right| \\
&\leq \sum_{i,j} \exp\left(-1 - \frac{d(x_i^p, x_j^q)}{\beta}\right) \frac{2\alpha}{\beta} \exp\left(\frac{2\alpha}{\beta} A\right) \|v - v'\|_1 \\
&\leq L \|v - v'\|_1
\end{aligned}$$

which proves the second assertion. The last assertion directly comes from the fixed-point theorem.

### 3. Performance

Experiments were conducted on the Swedish set (15 classes, 75 images per category) and a random subset of MNIST digit database (10 classes, 200 images per category). Each class in Swedish (resp. MNIST) is split into 50+25 (resp. 100+100) contours for training and testing. Interest points were sampled from each contour in MNIST (resp. Swedish) and encoded using the 60 (resp. 16) coefficients of the shape-context descriptor (Belongie et al., 2000).

#### 3.1. Generalization and Comparison

We evaluate  $k_t$ ,  $t \in \mathbb{N}^+$  using two initializations: (i) linear  $k_0(x, x') = k_l(x, x') = \langle x, x' \rangle$  (ii) and polynomial  $k_0(x, x') = k_p(x, x') = (\langle x, x' \rangle + 1)^2$ . Our goal is to show the improvement brought when using  $k_t$ ,  $t \in \mathbb{N}^+$ , so we compared it against the standard context-free kernels  $k_l$  and  $k_p$  (i.e.,  $k_t$ ,  $t = 0$ ). For this purpose, we trained a “one-versus-all” SVM classifier for each class in both MNIST and Swedish using the subset kernel  $K(\mathcal{S}_p, \mathcal{S}_q) = \sum_{x \in \mathcal{S}_p, x' \in \mathcal{S}_q} k_t(x, x')$ . The performance are measured, on different test sets, using  $n$ -fold cross-validation error ( $n = 5$ ).

We remind that  $\beta$  is set using (3) as the left-hand side of  $k_t$  corresponds to the Gaussian kernel with scale  $\beta$ . In practice,  $\beta = 0.1$ . The influence (and the performance) of the right-hand side of  $k_t$  increases as  $\alpha$  increases (see. Figure 3), nevertheless the convergence of  $k_t$  to a fixed point is guaranteed only if  $\alpha \in [0, \frac{\beta}{2A}]$ . Therefore, it becomes obvious that  $\alpha$  should be set to  $\frac{\beta}{2A}$  where  $A = \sup_{i,j} \sum_{k,\ell} g(x_i^p, x_k^p) \times g(x_j^q, x_\ell^q)$  (in practice,  $0 \leq g \leq 1$  and  $A = 1$ ).

Table (3) shows the 5-fold cross validation errors on MNIST and Swedish for different iterations; we clearly see the out-performance and the improvement of the

	INITIALIZATION	LINEAR	POLYNOMIAL
ITERATIONS (MNIST)			
$k_0$		11.4 ± 4.42	9.15 ± 4.63
$k_1$		8.80 ± 4.77	5.6 ± 2.72
$k_2$		6.90 ± 3.55	5.8 ± 2.36
$k_3$		6.90 ± 3.41	5.2 ± 2.07
$k_4$		<b>6.90 ± 3.41</b>	<b>5.2 ± 2.07</b>
ITERATIONS (SWEDISH)			
$k_0$		11.7 ± 2.88	6.53 ± 6.34
$k_1$		6.00 ± 2.30	3.33 ± 2.73
$k_2$		<b>3.06 ± 1.88</b>	<b>3.33 ± 2.73</b>

Table 3. This table shows the mean and the standard deviation of the 5-fold error on the MNIST (top) and Swedish (bottom) databases. Poly and Lin stand respectively for the polynomial and the linear kernels which are used as initialization of the “CDK” kernel. We can see a clear and a consistent gain through different iterations and also the convergence of the errors.

“CDK” kernel ( $k_t$ ,  $t \in \mathbb{N}^+$ ) with respect to the context-free kernels used for initialization ( $k_0 = k_l$  and  $k_0 = k_p$ ).

### 4. Remarks and Discussion

The adjacency matrix  $P$ , in  $k_t$ , provides the intrinsic properties and also characterizes the geometry of an object  $\mathcal{S}_p$ . Let us remind  $\mathcal{N}_p(x_i^p) = \{x_k^p \in \mathcal{S}_p : k \neq i, \|\psi_g(x_i^p) - \psi_g(x_k^p)\|_2 \leq \epsilon_p\}$  and  $P_{i,j} = \mathbb{1}_{\{x_j^q \in \mathcal{N}_p(x_i^p)\}}$ . It is easy to see that  $P$  is translation and rotation invariant and can also be made scale invariant when  $\epsilon_p$  is adapted to the scale of  $\psi_g(x_i^p)$ . It follows that the right-hand side of our kernel is invariant to any 2D similarity transformation. Notice, also, that the left-hand side of  $k_t$  involves similarity invariant descriptors  $\psi_f(x_i^p)$ ,  $\psi_f(x_j^q)$  so  $k_t$  (and  $K$ ) is similarity invariant.

One current limitation of our kernel  $k_t$  resides in its evaluation complexity. Assuming  $k_{t-1}$  known, for a given pair  $x_i^p$ ,  $x_j^q$ , this complexity is  $O(\max(N^2, s))$ , where  $s$  is the dimension of  $\psi_f(x_i^p)$  and  $N = \max_{i,p} \#\{\mathcal{N}_p(x_i^p)\}$ . It is clear enough that when  $N < \sqrt{s}$ , the complexity of evaluating our kernel is strictly equivalent to that of usual kernels such as the linear. Nevertheless, the worst case ( $N \gg \sqrt{s}$ ) makes our kernel evaluation prohibitive and this is mainly due to the right-hand side of  $k_t(x_i^p, x_j^q)$  which requires the evaluation of kernel sums in a hypercube of dimension 4. A simple and straightforward generalization of the integral image (see for instance (Viola & Jones, 2001)) will reduce this complexity to  $O(s)$ .

Finally, the out-performance of our kernel comes essentially from the inclusion of the context. This strongly improves the precision and helps including the intrinsic properties (geometry) of objects. Even though tested only on visual object recognition, our kernel can be extended to many other pattern analysis problems such as bioinformatics, speech and text. For instance, in text analysis and particularity machine translation (Sim et al., 2007), the design of a similarity kernel between words in two different languages, can be achieved using any standard dictionary. Of course, the latter defines similarity between any two words  $(w_e, w_f)$  independently from their bilingual training text (or bitext), i.e., the phrases where  $(w_e, w_f)$  might appear and this results into bad translation performances. A better estimate of similarity between two words  $(w_e, w_f)$ , can be achieved using their context i.e., the set of words which coocure frequently with  $(w_e, w_f)$  (Koehn et al., 2003).

## 5. Conclusion

We introduced in this paper a new type of kernels referred to as context-dependent. Its strength resides in the improvement of the alignments between interest points and this is considered as a preliminary step in order to increase the robustness and the precision of object recognition.

We have also shown that our kernel is Mercer and applicable to SVM learning. The latter, achieved for shape recognition problems, has better performance than SVMs with context-free kernels. Future work includes the comparison of our kernel with other context-free kernels and its application in scene and object understanding using other standard databases.

## Acknowledgements

This work was supported in part by the Agence Nationale de la Recherche project “Modèles Graphiques et Applications”.

## References

- Amores, J., Sebe, N., & Radeva, P. (2005). Fast spatial pattern discovery integrating boosting with constellations of contextual descriptors. *In CVPR*.
- Bahlmann, C., Haasdonk, B., & Burkhardt, H. (2002). On-line handwriting recognition with support vector machines, a kernel approach. *IWFHR*, 49–54.
- Barla, A., Odone, F., & Verri, A. (2002). Hausdorff kernel for 3d object acquisition and detection. *In ECCV, LNCS 2353*, 20–33.
- Belongie, S., Malik, J., & Puzicha, J. (2000). Shape context: A new descriptor for shape matching and object recognition. *In NIPS*.
- Bishop, C. (2007). Pattern recognition and machine learning. *Springer*.
- Boser, B., Guyon, I., & Vapnik, V. (1992). An training algorithm for optimal margin classifiers. *In Fifth Annual ACM Workshop on Computational Learning Theory, Pittsburgh*, 144–152.
- Boughorbel, S., Tarel, J., & Boujemaa, N. (2005). The intermediate matching kernel for image local features. *In IJCNN*.
- Chapelle, O., Haffner, P., & Vapnik, V. (1999). Svms for histogram-based image classification. *Transaction on Neural Networks*, 10.
- Cuturi, M. (2005). Etude de noyaux de semi-groupe pour objets structures dans le cadre de l'apprentissage statistique. *PhD thesis Gostatistique, ENSMP*.
- Everingham, M., Gool, L. V., Williams, C., Winn, J., & Zisserman, A. (2007). The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. <http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html>.
- Gartner, T. (2003). A survey of kernels for structured data. *Multi Relational Data Mining*, 5, 49–58.
- Grauman, K., & Darrell, T. (2007). The pyramid match kernel: Efficient learning with sets of features. *In JMLR*, 8, 725–760.
- Harris, C., & Stephens, M. (1988). A combined corner and edge detector. *Alvey Vision Conference*, 147–151.
- Haussler, D. (1999). Convolution kernels on discrete structures. *Technical Report UCS-CRL-99-10, UC Santa Cruz*.
- Jaakkola, T., Diekhans, M., & Haussler, D. (1999). Using the fisher kernel method to detect remote protein homologies. *ISMB*, 149–158.
- Koehn, P., Och, F. J., & Marcu, D. (2003). Statistical phrase-based translation. *In Proceedings of HLT/NAACL*.
- Kondor, R., & Jebara, T. (2003). A kernel between sets of vectors. *In proceedings of ICML*.



- Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *In IJCV*, 60, 91–110.
- Lyu, S. (2005). Mercer kernels for object recognition with local features. *In the proceedings of CVPR*.
- Moreno, P., Ho, P., & Vasconcelos, N. (2003). A kullback-leibler divergence based kernel for svm classification in multimedia applications. *In NIPS*.
- Mortensen, E., Deng, H., & Shapiro, L. (2005). A sift descriptor with global context. *In CVPR*, 184–190.
- Nie, J.-Y., Simard, M., Isabelle, P., & Durand, R. (1999). Cross-language information retrieval based on parallel texts and automatic mining of parallel texts from the web. *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*.
- S-Taylor, J., & Cristianini, N. (2000). Support vector machines and other kernel-based learning methods. *Cambridge University Press*.
- Schmid, C., & Mohr, R. (1997). Local greyvalue invariants for image retrieval. *In PAMI*, 19, 530–535.
- Scholkopf, B., Tsuda, K., & Vert, J.-P. (2004). Kernel methods in computational biology. *MIT Press*.
- Sim, K., Byrne, W., Gales, M., Sahbi, H., & Woodland, P. (2007). Consensus network decoding for statistical machine translation system combination. *In ICASSP*.
- Swain, M., & Ballard, D. (1991). Color indexing. *International Journal of Computer Vision*, 7, 11–32.
- Vapnik, V. N. (1998). Statistical learning theory. *A Wiley-Interscience Publication*.
- Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *In CVPR*.
- Wallraven, C., Caputo, B., & Graf, A. (2003). Recognition with local features: the kernel recipe. *ICCV*, 257–264.
- Wolf, L., & Shashua, A. (2003). Learning over sets using kernel principal angles. *In JMLR*, 4, 913–931.