Transductive kernel learning

Apprentissage transductif des noyaux

Dinh-Phong Vo
Hichem Sahbi

2012D003

mars 2012

Département Traitement du Signal et des Images
Groupe TII : Traitement et Interprétation des Images

Transductive Kernel Learning

Apprentissage Transductif des Noyaux

Dinh-Phong Vo DINH-PHONG.VO@TELECOM-PARISTECH.FR
Institut Mines-Télécom; Télécom ParisTech,
CNRS LTCI UMR 5141,
Paris, France

Hichem Sahbi HICHEM.SAHBI@TELECOM-PARISTECH.FR
CNRS LTCI UMR 5141,
Télécom ParisTech,
Paris, France

Abstract

Transductive inference techniques are nowadays becoming standard in machine learning due to their relative success in solving many real-world applications. Among them, kernel-based methods are particularly interesting but their success remains highly dependent on the choice of kernels. The latter are usually handcrafted or designed in order to capture better similarity in training data.

In this paper, we introduce a novel transductive learning algorithm for kernel design and classification. Our approach is based on the minimization of an energy function mixing i) a reconstruction term that factorizes a matrix of input data as a product of a learned dictionary and a learned kernel map ii) a fidelity term that ensures consistent label predictions with those provided in a ground-truth and iii) a smoothness term which guarantees similar labels for neighboring data and allows us to iteratively diffuse kernel maps and labels from labeled to unlabeled data. Solving this minimization problem makes it possible to learn both a decision criterion and a kernel map that guarantee linear separability in a high dimensional space and good generalization performance. Experiments conducted on object class segmentation, show improvements with respect to baseline as well as related work on the challenging VOC database.

Résumé

Les techniques d'inférence transductive sont devenus des standards incontournables en apprentissage automatique pour la résolution de problèmes multiples en reconnaissance des formes. Parmi ces techniques, les méthodes à noyaux sont particulièrement intéressantes mais leur succès dépend principalement du bon choix des noyaux. Ces derniers sont sélectionnés d'une façon add-hoc ou conçus afin de mieux caractériser la similarité entre les données.

Dans cet article, on introduit une nouvelle méthode d'apprentissage transductif pour la conception des noyaux. Cette approche est basée sur la minimisation d'une énergie qui mélange i) *un terme de reconstruction* qui factorise une matrice des données de départ comme un produit de deux matrices l'une correspond à un dictionnaire et l'autre au mapping du noyau appris, ii) *un terme d'attache aux données* qui garantit la consistance des labels prédits par rapport à ceux fournis par une vérité terrain et enfin iii) *un terme de lissage* garantissant une variation progressive des labels prédits pour des données similaires et permettant ainsi de diffuser itérativement le noyau appris et les labels vers les données non-étiquetées. La résolution de ce problème d'optimisation permet d'apprendre une fonction de décision et un noyau garantissant la séparabilité linéaire des données ainsi que de bonnes performances de généralisation. Les expériences effectuées, en segmentation en classes d'objets sur la base Pascal VOC, montrent des performances supérieures par rapport aux différentes "baselines" ainsi que des méthodes de l'état de l'art.

Keywords: Kernel Design and Learning, Transductive Inference, Matrix Factorization, Object Class Segmentation, Object Recognition.

1. Introduction

Existing machine inference techniques may be categorized into *inductive* and *transductive* (Vapnik, 1998). The former consists in finding a decision function from a labeled training set, and uses that function in order to generalize across unlabeled data. Among popular inductive techniques support vector machines (SVMs) (Vapnik, 1998; Schölkopf and Smola, 2001) are well studied and proved to be performant in many real-world applications including object recognition, text analysis and bioinformatics (Maji et al., 2008; Joachims, 2002; Asa et al., 2008). The success of SVMs is highly dependent on the choice of kernels; existing ones include the linear, the gaussian and the histogram intersection. However, usual kernels may not be appropriate in order to capture the actual and the "semantic" similarity between data for some specific tasks. Variants known as multiple kernels (MKL) (Rakotomamonjy et al., 2008; Lanckriet et al., 2004; Varma and Ray, 2007) consider convex (and possibly sparse) linear combinations of elementary kernels and proved to be more suitable.

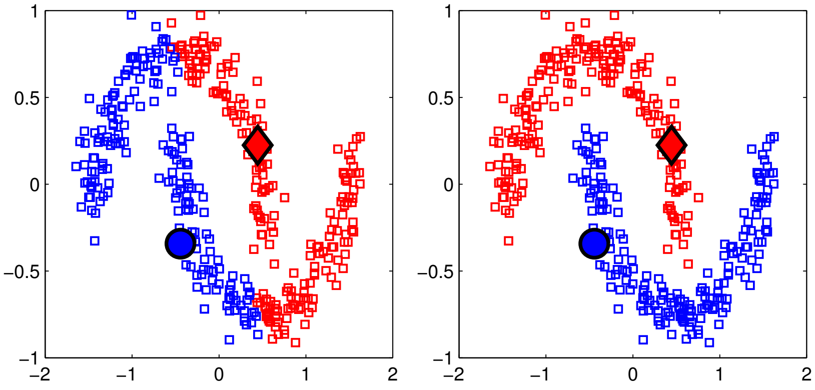
Even-though performant, the success of these methods, also depends on cardinality of the labeled data. In many applications such as object class segmentation (Duchenne et al., 2008), labeled data is rare and expensive; only a very small fraction of training data is labeled and the unlabeled data may not follow the same distribution as the labeled one, so learning kernels using inductive inference techniques is clearly not appropriate. Alternative approaches (Chapelle et al., 2006) may include the unlabeled data as a part of the learning process and this is known as transductive inference. The concept of transductive inference, or transduction, was pioneered by Vapnik (see for instance Vapnik (1998)). It relates to semi-supervised learning and relies on the i) smoothness assumption which states that close data in a high-density area of the input space, should have similar labels (Chapelle et al., 2006; Belkin et al., 2006) and ii) the cluster assumption which finds decision rules in low density areas of the input space (Seeger, 2001; Duchenne et al., 2008). In that context, transductive versions of SVMs were also introduced (Joachims, 1999); they build decision functions by optimizing the parameters of a learning model together with the labels of the unlabeled data. This turned out to be very useful in order to overcome the limited cardinality of the labeled data w.r.t the number of training parameters.

In this paper we introduce a novel transductive learning algorithm, for classification and kernel learning. Our method is based on a constrained matrix factorization which produces *a kernel map* that takes data from the input space into a high dimensional space in order to guarantee their linear separability while maximizing their margin. This margin property, however, and as known (Vapnik, 1998; Duchenne et al., 2008), does not necessarily guarantee good generalization performance on the unlabeled set, if the latter is drawn from a different probability distribution compared to the labeled data (see Fig. 1). Therefore and beside maximizing the margin, our transductive approach includes a regularization term that enforces smoothness in the resulting kernel map in order to correctly diffuse labels to the unlabeled data. Following our formulation, and in contrast to MKL, our learning model is not restricted to only convex linear combinations of existing kernels; indeed it is model-free. Furthermore, it also takes advantage from both labeled and unlabeled data and this results into better generalization performances as corroborated by our experiments.

The remainder of this paper is organized as follows. We introduce our transductive learning approach and kernel design in Section 2 and the implementation of our optimization procedure in Section 3. We illustrate the application of our method to object class segmentation in Section 4. We conclude the paper in Section 5 while providing a possible extension for a future work.

2. Inference and Kernel Design

Define $\mathcal{X} \subseteq \mathbb{R}^n$ as an input space corresponding to all the possible image features and let $\mathcal{S} = \{\mathbf{x}_1, \dots, \mathbf{x}_\ell, \dots, \mathbf{x}_m\}$ be a finite subset of $\mathcal{X}$ with an arbitrary order. This order



(a) Inductive (b) Transductive

Figure 1: This figure shows classification results on the "two moon" example in Belkin et al. (2006). (Left) An inductive method is used for classification (right) a transductive technique is used instead that also exploits the density of the unlabeled data. In this example labeled data are marked with "diamond" and "circle" and correspond to the positive and the negative classes respectively.

is defined so only the first ℓ labels of $\mathcal{S}$, denoted $\{y_1, \dots, y_\ell\}$ (with $y_i \in \{-1, +1\}$), are known. In many real-world applications only a few data is labeled (i.e., $\ell \ll m$) and its distribution may be different from the unlabeled data.
We can view $\mathcal{S}$ as a matrix $\mathbf{X}$ in which the i^{th} column corresponds to x_i. Our objective is to build both a decision criterion and an optimal *kernel map* in order to infer the unknown labels $\{y_{\ell+1}, \dots, y_m\}$.

2.1 Max-margin Inference and Kernel Design

Inductive learning aims to build a decision function f that predicts a label y for any given input data $\mathbf{x}$; this function is trained on $\mathcal{S}' = \{\mathbf{x}_1, \dots, \mathbf{x}_\ell\}$ and used in order to infer labels on $\mathcal{S} \backslash \mathcal{S}'$. In the max-margin classification (Vapnik, 1998), we consider ϕ as a mapping of the input data (in $\mathcal{X}$) into a high dimensional space $\mathcal{H}$. The dimension of $\mathcal{H}$ is usually sufficiently large (possibly infinite) in order to guarantee linear separability of data.

Assuming data linearly separable in $\mathcal{H}$, the max-margin inductive learning finds a hyperplane f (with a normal $\mathbf{w}$ and shift b) that separates ℓ training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{\ell}$ while maximizing their margin. The margin is defined as twice the distance between the closest training samples w.r.t f and the optimal $(\hat{\mathbf{w}}, \hat{b})$ correspond to

$$\begin{array}{ll} \underset{\mathbf{w},b}{\text{argmin}} & \frac{1}{2}\left\|\mathbf{w}\right\|_2^2 \\ \text{s.t} & y_i\left(\mathbf{w}'\phi(\mathbf{x_i})+b\right)\geq 1, \quad i=1,\ldots,\ell, \end{array} \tag{1}$$

which is the primal form of the hard margin support vector machine (Vapnik, 1998), $\|.\|_2^2$ is the ℓ_2-norm and $\mathbf{w}'$ is the transpose of $\mathbf{w}$. Given $\mathbf{x}_i \in \mathcal{S}\backslash\mathcal{S}'$, the class of $\mathbf{x}_i$ in $\{-1,+1\}$ is decided by the sign of $f(\mathbf{x}_i) = \mathbf{w}'\phi(\mathbf{x}_i)+b$. Following the kernel trick (Vapnik, 1998), one may show that $f(\mathbf{x}_i)$ can also be expressed as $\sum_{j=1}^{\ell}\alpha_j y_j \kappa(\mathbf{x}_i,\mathbf{x}_j)+b$, here $(\alpha_1\ldots\alpha_\ell)'$ is a vector of positive real-valued training parameters and $\kappa(\mathbf{x}_i,\mathbf{x}_j)=\langle\phi(\mathbf{x}_j),\phi(\mathbf{x}_i)\rangle$ is a symmetric, continuous, positive (semi-definite) kernel function (Schölkopf and Smola, 2001). The closed form of $\kappa(\mathbf{x}_i,\mathbf{x}_j)$ is defined among a collection of existing kernels including linear, gaussian and histogram intersection; but the underlying mapping $\phi(x)\in\mathcal{H}$ is usually *implicit*, i.e., it does exist but it is not necessarily known and may be infinite dimensional.

We propose in the remainder of this section a new approach that builds *explicit* and finite dimensional kernel map. In contrast to usual kernels, such as the gaussian, the VC-dimension (Vapnik, 1998), related to a finite dimensional kernel map, is finite[1]. According to Vapnik's VC-theory (Vapnik and Sterin, 1977), the finiteness of the VC-dimension avoids loose generalization bounds and may guarantee better performance.

Now, we turn the problem into finding the hyperplane f as well as a Gram (kernel) matrix $\mathbf{K}=\mathbf{\Phi}'\mathbf{\Phi}$ where each column $\mathbf{\Phi}_i$ corresponds to an explicit mapping of $\mathbf{x}_i$ into a high dimensional space (i.e., $\phi(\mathbf{x}_i)=\mathbf{\Phi}_i$). This mapping is designed in order to i) guarantee linear separability of data in $\mathcal{S}$, ii) to ensure good generalization performance by maximizing the margin, iii) to approximate the input data, and also iv) to ensure positive definiteness of $\mathbf{K}$ by construction, i.e., without adding further constraints. This results into the following constrained minimization problem

$$\begin{array}{ll} \underset{\mathbf{B},\mathbf{\Phi},\mathbf{w}}{\min} & \frac{1}{2}\left\|\mathbf{X}-\mathbf{B}\mathbf{\Phi}\right\|_F^2+\frac{1}{2}\left\|\mathbf{w}\right\|_2^2 \\ \text{s.t} & y_i\mathbf{w}'\mathbf{\Phi}_i\geq 1, \quad i=1,\ldots,\ell \\ & \|\mathbf{B}_j\|_2\leq 1, \quad j=1,\ldots,p, \end{array} \tag{2}$$

here $\|\mathbf{A}\|_F^2=\mathbf{tr}(\mathbf{A}\mathbf{A}')$ stands for the square of the Frobenius norm and $\mathbf{X}\approx\mathbf{B}\mathbf{\Phi}$ is factorized using an overcomplete basis $\mathbf{B}\in\mathbb{R}^{n\times p}$ (i.e., $p>n$) and a new kernel map $\mathbf{\Phi}\in\mathbb{R}^{p\times m}$. Without a loss of generality b is omitted in the above expression as it can be induced from $\mathbf{w}$ and the mapping $\mathbf{\Phi}$.

2.2 Enforcing Low Rank Kernels

As discussed earlier, and according to Vapnik (1998), the VC-dimension (related to a family of classifiers) depends also on the dimension of the learned kernel map and this

[1] The VC-dimension is the maximum number of data samples, that can be shattered, whatever their labels.

may affect generalization, especially if this dimension is very high. Since the actual (intrinsic) dimension of the learned kernel map $\mathbf{\Phi}$ is unknown, we choose the number of basis p to be sufficiently large such that the first inequality constraint in (2) can be satisfied and the left-hand side term tends to zero for an infinite number of solutions. First, p is overestimated to $\max(\ell, n) + 1$, and this guarantees that the above constrained minimization problem has a solution. Then, the actual (intrinsic) dimension is found by regularizing Eq. 2 by the Frobenius norm $\|\mathbf{\Phi}\|_F^2$ which has similar effect as the nuclear norm, (see Lemma 1 below).

Lemma 1 *For any matrix $\mathbf{\Phi} \in \mathbb{R}^{p \times m}$, the following inequalities hold*

$$\|\mathbf{\Phi}\|_F \leq \|\mathbf{\Phi}\|_* \leq \sqrt{r}\|\mathbf{\Phi}\|_F, \tag{3}$$

where the Frobenius norm $\|\mathbf{\Phi}\|_F = \sqrt{\sum_{i=1}^{\min\{p,m\}} \sigma_i^2}$;
the nuclear norm $\|\mathbf{\Phi}\|_* = \sum_{i=1}^{\min\{m,p\}} \sigma_i$; $r = \mathbf{rank}(\mathbf{\Phi}) = \mathbf{rank}(\mathbf{\Phi}'\mathbf{\Phi})$ and σ_i's are eigenvalues of the Gram matrix $\mathbf{K} = \mathbf{\Phi}'\mathbf{\Phi}$.

Proof See for instance Horn and Johnson (1990); Golub and Loan (1996). ■

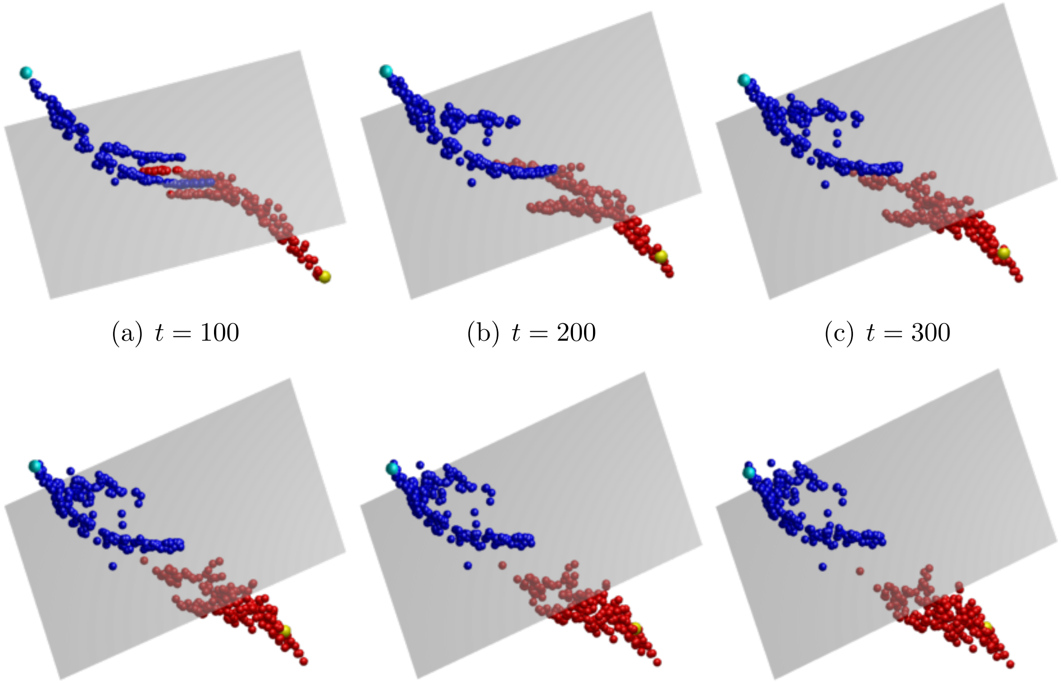
The problem is reformulated by adding an extra penalty $\frac{\mu}{2}\|\mathbf{\Phi}\|_F^2$ to the objective function (2) where $\mu \geq 0$ controls the rank of $\mathbf{K}$. Indeed, the squared Frobenius norm is exactly the ℓ_2-norm on the eigenvalues of $\mathbf{K}$ and it is less likely to shrink these eigenvalues into zeros compared to the ℓ_1-norm (which is the nuclear norm). Nevertheless, as will be shown later, it provides a closed form kernel solution and our experiments show that it indeed reduces the rank of the kernel map while allowing to learn effective classifiers.

2.3 Transductive Setting

For a better conditioning of (2), we implement in this section the smoothness assumption discussed in Section 1. This makes it possible to design smooth kernel maps and to assign similar predictions to neighboring data for a better generalization on the unlabeled ones (see toy example in Fig. 2).

We model the input data $\mathcal{S}$ using an adjacency graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where nodes $\mathcal{V} = \{v_1, \dots, v_m\}$ correspond to samples $\{x_i\}$ and edges $\mathcal{E} = \{e_{i,j}\}$ are the set of weighted links of $\mathcal{G}$. In the above definition, $\mathbf{x}_i \in \mathbb{R}^n$ is a feature vector (color, texture, etc.) while $e_{i,j} = (v_i, v_j, \mathbf{W}_{ij})$ defines a connection between v_i, v_j weighted by $\mathbf{W}_{i,j}$. The latter is defined as $\mathbf{W}_{ij} = 1_{\{v_j \in \mathcal{N}_k(v_i)\}} \cdot \exp\left(-\|\mathbf{x}_i - \mathbf{x}_j\|_2^2 / \sigma^2\right)$, here the neighborhood $\mathcal{N}_k(v_i)$ of a given node v_i, includes the set of the k-nearest neighbors of v_i. Notice that this neighborhood system is designed in order to guarantee that $\forall v_i, v_j \in \mathcal{V}$, $v_j \in \mathcal{N}_k(v_i)$ implies $v_i \in \mathcal{N}_k(v_j)$ and vice-versa.

Considering $f(x_i) = \mathbf{w}'\mathbf{\Phi}_i$ and $f(x_j) = \mathbf{w}'\mathbf{\Phi}_j$, we define our regularizer as $\frac{\gamma_s}{4}\sum_{i,j=1}^m (\mathbf{w}'\mathbf{\Phi}_i - \mathbf{w}'\mathbf{\Phi}_j)^2 \mathbf{W}_{ij}$, which may be rewritten as $\frac{\gamma_s}{2}\mathbf{w}'\mathbf{\Phi}\mathbf{L}\mathbf{\Phi}'\mathbf{w}$, here $\gamma_s \geq$


(a) $t = 100$ (b) $t = 200$ (c) $t = 300$

(d) $t = 400$ (e) $t = 500$ (f) $t = 600$

Figure 2: This figure shows the evolution of the learned kernel map through different iterations of our method (see Algorithm 1). This map is found for the popular "two moon" example in Belkin et al. (2006). The underlying 2D input data are not linearly separable, while the learned kernel map makes them linearly separable in a 3D space. In these experiments, only $\ell = 2$ samples were labeled (shown in blue and yellow resp. for the positive and the negative classes).

0 and $\mathbf{L}$ is the graph Laplacian defined by $\mathbf{L} = \mathbf{D} - \mathbf{W}$ and $\mathbf{D}_{ii} = \sum_{j=1}^{m} \mathbf{W}_{ij}$, $\mathbf{D}_{ij} = 0, \forall i \neq j$. When adding this regularizer in objective function (2) and replacing inequality constraints with the squared loss $\sum_{i=1}^{\ell}(y_i - \mathbf{w}'\mathbf{\Phi}_i)^2$, we obtain the complete form of our transductive learning problem

$$\begin{aligned} \min_{\mathbf{B},\mathbf{\Phi},\mathbf{w}} \quad & \tfrac{1}{2}\mathbf{w}'\big(\mathbf{I}_p + \mathbf{\Phi}\tilde{\mathbf{L}}\mathbf{\Phi}'\big)\mathbf{w} + \tfrac{1}{2}\left\|\mathbf{X} - \mathbf{B}\mathbf{\Phi}\right\|_F^2 \\ & -\gamma_c \mathbf{Y}'\mathbf{C}\mathbf{\Phi}'\mathbf{w} + \tfrac{\mu}{2}\|\mathbf{\Phi}\|_F^2, \\ \text{s.t} \quad & \|\mathbf{B}_j\|_2 \leq 1, \quad j = 1, \ldots, p, \end{aligned} \tag{4}$$

with $\mathbf{I}_p$ the $p \times p$ identity matrix, $\tilde{\mathbf{L}} = (\gamma_c\mathbf{C} + \gamma_s\mathbf{L})$ and $\mathbf{C}$ is the diagonal $m \times m$ matrix for which the i-th diagonal element is fixed to 1 for a labeled sample, and 0

```
Algorithm 1 TransRMF
Input: labeled ${(\mathbf{x}_i, y_i)}_{i=1}^{\ell}$ and unlabeled data ${\mathbf{x}_i}_{i=\ell+1}^{m}$
Initialization: set the adjacency matrix $\mathbf{W}$, $t \leftarrow 0$ and set $\mathbf{\Phi}^{(0)}$ to random full rank matrices.
Repeat steps (1+2) until convergence
    1. Update $\mathbf{w}^{(t+1)}$ and $\mathbf{B}^{(t+1)}$ using (5), (6) respectively.
    2. Update $\mathbf{\Phi}^{(t+1)}$ by taking the limit $\tilde{\mathbf{\Psi}}$ of (11), with $\mathbf{\Psi}^{(0)} = \mathbf{\Phi}^{(t)}$.
Output: kernel maps ${\tilde{\mathbf{\Phi}}_i}$ and labels ${y_i}$ with $y_i = \tilde{\mathbf{w}}'\tilde{\mathbf{\Phi}}_i$.
```

for an unlabeled one, and similarly, $\mathbf{Y}$ is the m-dimensional vector for which the i-th element is y_i for a labeled data, and 0 for an unlabeled one.

3. Optimization

It is clear that the minimization problem in (4) is not convex jointly w.r.t $\mathbf{B}, \mathbf{\Phi}, \mathbf{w}$. We consider an EM-like optimization procedure by solving three subproblems: we first maximize the margin $2/\left\|\mathbf{w}\right\|^2$ w.r.t $\mathbf{w}$ and we update the basis $\mathbf{B}$, then we minimize the regularization criterion, the rank and the reconstruction error w.r.t $\mathbf{\Phi}$. This process is repeated until convergence; i.e., all the unknowns remain unchanged from one iteration to another. Different steps of the algorithm are shown in Algorithm (1); the superscript (t) is added to $\mathbf{w}$, $\mathbf{B}$ and $\mathbf{\Phi}$ in order to show the evolution of their values through different iterations of the learning process.

3.1 Learning Basis and Classifier

Assuming fixed $\mathbf{\Phi}^{(t)}$ (denoted simply as $\mathbf{\Phi}$) and enforcing the gradient of (4) to vanish (w.r.t $\mathbf{w}$) leads to

$$\mathbf{w}^{(t+1)} = \gamma_c\big(\mathbf{I}_p + \mathbf{\Phi}\tilde{\mathbf{L}}\mathbf{\Phi}'\big)^{-1}\mathbf{\Phi}\mathbf{C}\mathbf{Y}. \tag{5}$$

Similarly, we find $\mathbf{B}^{(t+1)}$ as

$$\begin{array}{cl} \underset{\mathbf{B}}{\text{argmin}} & \frac{1}{2}\big\|\mathbf{X} - \mathbf{B}\mathbf{\Phi}\big\|_F^2 \\ \text{s.t} & \|\mathbf{B}_i\|_2^2 \leq 1, \quad i = 1, \ldots, p. \end{array} \tag{6}$$

We define the Lagrangian of (6) as

$$L(\mathbf{B}, \boldsymbol{\lambda}) = \frac{1}{2}\left\|\mathbf{X} - \mathbf{B}\mathbf{\Phi}\right\|_F^2 + \sum_{i=1}^{p} \lambda_i \left(\|\mathbf{B}_i\|_2^2 - 1\right), \tag{7}$$

where $\lambda_i \geq 0$ is the Lagrange multiplier associated with the i-th inequality constraint in (6). The dual function is given by $g(\boldsymbol{\lambda}) = \inf_{\mathbf{B}} L(\mathbf{B}, \boldsymbol{\lambda})$ and the minimizer $\mathbf{B}^*$ is obtained from (7) by taking derivative w.r.t $\mathbf{B}$

$$\mathbf{B}^* := \mathbf{X}\mathbf{\Phi}' \left(\mathbf{\Phi}\mathbf{\Phi}' + \mathrm{diag}(\boldsymbol{\lambda})\right)^{-1} \tag{8}$$

By replacing (8) into (7), the dual function is

$$\min_{\boldsymbol{\lambda} \geq \mathbf{0}} \quad \mathrm{tr}\left(\mathbf{X}\mathbf{\Phi}'(\mathbf{\Phi}\mathbf{\Phi}' + \mathrm{diag}(\boldsymbol{\lambda}))^{-1}\mathbf{\Phi}\mathbf{X}'\right) + \mathbf{1}'\boldsymbol{\lambda} \tag{9}$$

This problem is solved using Newtons method as in Lee et al. (2006). After minimizing $g(\boldsymbol{\lambda})$, we obtain the optimal basis $\mathbf{B}^{(t+1)}$ as $\mathbf{X}\mathbf{\Phi}' \left(\mathbf{\Phi}\mathbf{\Phi}' + \mathrm{diag}(\boldsymbol{\lambda}^*)\right)^{-1}$.

3.2 Learning Kernel Map

Considering fixed $\mathbf{B}^{(t+1)}$ and $\mathbf{w}^{(t+1)}$ (denoted simply as $\mathbf{B}$, $\mathbf{w}$ in the remainder of this section), and the previous kernel map solution $\mathbf{\Phi}^{(t)}$, our goal is to find $\mathbf{\Phi}^{(t+1)}$ by solving (4). Conditions for the existence of this new kernel map solution $\mathbf{\Phi}^{(t+1)}$ are given in the following proposition.

Proposition 2 *Let $\|.\|_1$ denote the entrywise ℓ_1-norm. Provided that the following inequality holds,*

$$\gamma_s < \|\mathbf{w}\mathbf{w}'\|_1^{-1}.\|\mathbf{W}\|_1^{-1}, \tag{10}$$

the optimization problem (4) admits a unique solution $\mathbf{\Phi}^{(t+1)} = \tilde{\mathbf{\Psi}}$ as the limit of

$$\mathbf{\Psi}^{(k+1)} = \psi\big(\mathbf{\Psi}^{(k)}\big), \tag{11}$$

here $\psi : \mathbb{R}^{p\times m} \to \mathbb{R}^{p\times m}$ is defined as $\psi(\mathbf{\Psi}) = \big(\psi_1(\mathbf{\Psi}) \ldots \psi_m(\mathbf{\Psi})\big)$, with each column vector $\psi_i(\mathbf{\Psi})$ as

$$\begin{aligned} \psi_i(\mathbf{\Psi}) \; &= \left(\mathbf{B}'\mathbf{B} + (\gamma_s \mathbf{D}_{ii} + \gamma_c \mathbf{C}_{ii})\mathbf{w}\mathbf{w}' + \mu \mathbf{I}_p\right)^{-1} \\ &\qquad\qquad . \left[\mathbf{B}'\mathbf{X} + \gamma_c \mathbf{w}\mathbf{Y}'\mathbf{C} + \gamma_s \mathbf{w}\mathbf{w}'\mathbf{\Psi}\mathbf{W}\right]_i, \end{aligned} \tag{12}$$

$[.]_i$ stands for the i-th column of a matrix. Furthermore, the kernel maps $\mathbf{\Psi}^{(k)}$ in (11) satisfy the convergence property:

$$\big\|\mathbf{\Psi}^{(k)} - \tilde{\mathbf{\Psi}}\big\|_1 \leq L^k \big\|\mathbf{\Psi}^{(0)} - \tilde{\mathbf{\Psi}}\big\|_1, \tag{13}$$

with $L = \gamma_s \|\mathbf{w}\mathbf{w}'\|_1.\|\mathbf{W}\|_1$ and $\mathbf{\Psi}^{(0)} = \mathbf{\Phi}^{(t)}$.

Proof Following (4), let us consider the function defined on the set of matrices in $\mathbb{R}^{p\times m}$

$$\begin{aligned} E : \mathbf{\Psi} \mapsto & \tfrac{1}{2}\mathbf{w}'\big(\mathbf{I}_p + \mathbf{\Psi}\tilde{\mathbf{L}}\mathbf{\Psi}'\big)\mathbf{w} + \tfrac{1}{2}\left\|\mathbf{X} - \mathbf{B}\mathbf{\Psi}\right\|_F^2 + \tfrac{\mu}{2}\|\mathbf{\Psi}\|_F^2 \\ & -\gamma_c \mathbf{Y}'\mathbf{C}\mathbf{\Psi}'\mathbf{w} \end{aligned} \tag{14}$$

The necessary condition of the fixed-point relation in (11) results from $\partial E/\partial \mathbf{\Psi} = 0$ (details about derivative are omitted in this proof). We will now prove that the function ψ is L-Lipschitzian, with $L = \gamma_s \|\mathbf{ww}'\|_1.\|\mathbf{W}\|_1$.
Let us denote the left-hand side (inverse) matrix in (12) simply as $\mathbf{Z}_i$ and introduce $g(\mathbf{\Psi}) = (g_1(\mathbf{\Psi}) \dots g_m(\mathbf{\Psi}))$ with $g_i(\mathbf{\Psi}) = \mathbf{Z}_i^{-1}\psi_i(\mathbf{\Psi})$.

Given two matrices $\mathbf{\Psi}^{(1)}$ and $\mathbf{\Psi}^{(2)}$ in $\mathbb{R}^{p\times m}$, we have

$$\begin{aligned}
& \sum_{i=1}^{m} \left\| \mathbf{Z}_i^{-1}\psi_i(\mathbf{\Psi}^{(1)}) - \mathbf{Z}_i^{-1}\psi_i(\mathbf{\Psi}^{(2)}) \right\|_1 \\
= & \sum_{i=1}^{m} \left\| g_i(\mathbf{\Psi}^{(1)}) - g_i(\mathbf{\Psi}^{(2)}) \right\|_1 \\
= & \quad \left\| g(\mathbf{\Psi}^{(1)}) - g(\mathbf{\Psi}^{(2)}) \right\|_1 \\
= & \gamma_s \left\| \mathbf{ww}'(\mathbf{\Psi}^{(1)} - \mathbf{\Psi}^{(2)})\mathbf{W} \right\|_1 \\
\leq & \gamma_s \left\| \mathbf{ww}' \right\|_1 . \left\| \mathbf{W} \right\|_1 . \left\| \mathbf{\Psi}^{(1)} - \mathbf{\Psi}^{(2)} \right\|_1 \\
\leq & L \left\| \mathbf{\Psi}^{(1)} - \mathbf{\Psi}^{(2)} \right\|_1, \text{ with } \; L = \gamma_s \left\| \mathbf{ww}' \right\|_1 . \left\| \mathbf{W} \right\|_1 .
\end{aligned} \tag{15}$$

By taking the free parameter μ (in $\mathbf{Z_i}$) sufficiently large

$$\begin{aligned}
& \sum_{i=1}^{m} \left\| \mathbf{Z}_i^{-1}\psi_i(\mathbf{\Psi}^{(1)}) - \mathbf{Z}_i^{-1}\psi_i(\mathbf{\Psi}^{(2)}) \right\|_1 \\
= & \sum_{i=1}^{m} \left\| \left[\psi_i(\mathbf{\Psi}^{(1)}) - \psi_i(\mathbf{\Psi}^{(2)})\right] . \mathbf{Z}_i^{-1} \right\|_1 \\
\geq & \sum_{i=1}^{m} \left\| \psi_i(\mathbf{\Psi}^{(1)}) - \psi_i(\mathbf{\Psi}^{(2)}) \right\|_1 \\
= & \left\| \psi(\mathbf{\Psi}^{(1)}) - \psi(\mathbf{\Psi}^{(2)}) \right\|_1
\end{aligned} \tag{16}$$

Combining (15), (16), we get

$$\left\| \psi(\mathbf{\Psi}^{(1)}) - \psi(\mathbf{\Psi}^{(2)}) \right\|_1 \leq L \left\| \mathbf{\Psi}^{(1)} - \mathbf{\Psi}^{(2)} \right\|_1$$

■

The process described in equation (11) allows us to recursively diffuse the kernel maps from the labeled to the unlabeled data, through the neighborhood system defined in the graph $\mathcal{G}$. This process is iterative and may require many steps before

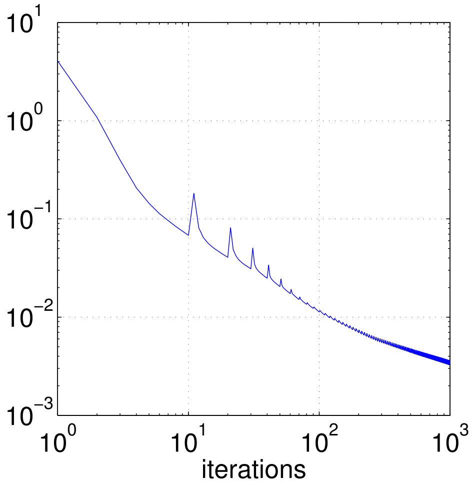


Figure 3: This figure illustrates the convergence process on the particular example of Fig. 1, i.e., the difference between current and previous estimate of kernel maps through different iterations.

convergence. The latter is reached when $\|\mathbf{\Psi}^{(k)} - \mathbf{\Psi}^{(k-1)}\| \leq \epsilon$; (in practice, $\epsilon = 10^{-2}$, and convergence usually happens in less than 100 iterations, see Fig. 3).

4. Experiments

We use the Pascal VOC 2011 dataset[2] in order to evaluate the performance of our transductive inference method on object class segmentation (OCS). For that purpose, we use 556 images from this dataset belonging to 21 categories; given an image, the goal is to assign each group of pixels (referred to as superpixel) to one of these 21 categories. In practice, a given image is subdivided into an irregular grid (neighborhood system) of 700 superpixels, each one is processed in order to extract various features (Tighe and Lazebnik, 2010) (see Table. 1 for more details).

For each image in VOC, we turn OCS into a transductive inference problem where only a small fraction of its underlying superpixels is labeled (see Fig. 4, third column). We train one transductive classifier (referred to as TransRMF) for each category and we combine these classifiers using the "winner-take-all" strategy in order to infer the category of a given unlabeled superpixel.
Following the evaluation protocol of Pascal VOC 2011, we use the standard segmentation accuracy for assessment. This measure is defined for each category $\mathcal{C}$ using intersection/union score, defined as the number of correctly labeled pixels of $\mathcal{C}$, di-

[2] http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2011/index.html

Table 1: This table shows the list of features used to describe superpixels. All feature vectors, excepting position, are normalized using ℓ_1-norm prior to their concatenation.

Type	**Description**	**Dimension**
Position	Absolute Mask	$8 \times 8 = 64$
	Top Height	1
	Bottom Height	1
Texture	Interior texton histogram	100
SIFT	Interior SIFT histogram	100
Color	RGB mean	3
	RGB std. dev.	3
	RGB Color Histograms	$11 \times 3 = 33$

vided by the number of pixels labeled with that category into different images and the ground truth. This accuracy is expressed as

$$\text{accuracy} = \frac{\text{true pos.}}{\text{true pos.} + \text{false pos.} + \text{false neg.}} \tag{17}$$

A mean accuracy is also considered as the expectation through different categories.

4.1 Settings and Performance

Different settings were experimented for our method including the size of the neighborhood (denoted k) when building the graph $\mathcal{G}$. The choice of these parameters will be discussed in the remainder of this section.

Graph topology. the degree of the graph k is very dependent on the topology of the data. An appropriate selection of k should avoid short-cuts (overestimated k) and missing-connections (under-estimated k).
In practice, we compared OCS accuracy for different values of k. The results show that the best performance is achieved when $k = 6$ (see Table 2).
Regularization & Rank reduction. Fig. (5) reports average accuracy for different values of the regularization parameter γ_s; note that $\gamma_s = 0$ corresponds to the *baseline* inductive setting (i.e., no regularization is applied). Fig. 4, shows the evolution of the underlying segmentation results w.r.t γ_s. According to these results, an underestimated γ_s results into noisy segmentation while an overestimated γ_s makes the segmentation results very smooth (with possibly lost details). In other word, as the

	10%	15%	20%	25%
k=12	61.95	66.26	69.06	70.88
k=6	**66.27**	**69.58**	**72.58**	**74.24**
k=3	65.97	69.16	72.54	74.12

Table 2: This table shows the average accuracy of OCS, with respect to k (degree of the graph $\mathcal{G}$) and for different percentage of labeled pixels.

regularization applies for both foreground and background classes, the smoothness favors the one with larger number of examples, i.e. likely to occupy more image area.

It is also clear that the transductive setting (i.e., $\gamma_s > 0$) outperforms the inductive one (i.e., $\gamma_s \to 0$). Fig. 5 also reports the average accuracy as an increasing function of γ_c (almost quasi-constant for larger values of γ_c). According to these experiments, we found that the best performances are achieved when $\gamma_s = 0.1$, $\gamma_c = 10$ and this satisfies our convergence criterion in Eq. (10) (see example in Fig. 3).

Finally, Fig. (5, right) and Fig. (6) show the evolution of accuracy and kernel rank w.r.t to the parameter μ. From these figures, it is clear that larger values of μ favor low rank kernels while maintaining high accuracy.

4.2 Comparison

We compare our TransRMF approach w.r.t to inductive as well as transductive approaches. Fig. (7) shows the average accuracies and comparison. For all these comparisons, $\gamma_s = 10^{-1}$, $\gamma_c = 10$, and $\mu = 10^{-10}$.

TransRMF vs inductive learning. In our experiments, inductive approaches include SVM classifiers (Vapnik, 1998) with four different kernels (linear, RBF, χ^2, and histogram intersection) and their linear combination using multiple kernel learning via SimpleMKL[3] (Rakotomamonjy et al., 2008) (see Fig. 7). Note that MKL has been extensively trained using several Gram matrices resulting from the combination of the four kernels mentioned earlier and the descriptors in Table 1.
As shown in Fig. 7-left, TransRMF outperforms the inductive classifiers, with various kernels as well as their combination using MKL, and the accuracy of the inductive techniques and TransRMF become more and more similar as the percentage of labeled data increases. *Our first conclusion is that TransRMF is very suitable to learn a classifier especially when the fraction of labeled data is very small and the second conclusion is that the learned kernel map is more appropriate for classification than linear combination of kernels.*

[3] http://asi.insa-rouen.fr/enseignants/~arakotom/code/mklindex.html

TransRMF vs related transductive methods. Transductive approaches, used for comparison, include Laplacian-SVM[4] (Melacci and Belkin, 2011) and transductive SVM[5] (Joachims, 1999). According to the result presented in Fig. 7-right, TransRMF consistently outperforms Transductive SVMs and Laplacian SVMs; note that the latter also relies on regularization with a setting similar to our, (i.e., the same graph Laplacian and graph $\mathcal{G}$) but *our method has an extra advantage of optimizing the kernel map resulting into a more suitable data representation for classification.*

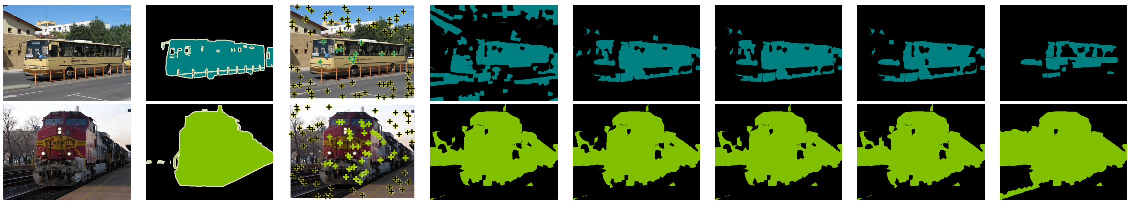


Figure 4: **Left to right:** original image; ground truth; annotation; segmentation results with $\gamma_s = 10^{-3}, 10^{-2}, 10^{-1}, 1, 10$.

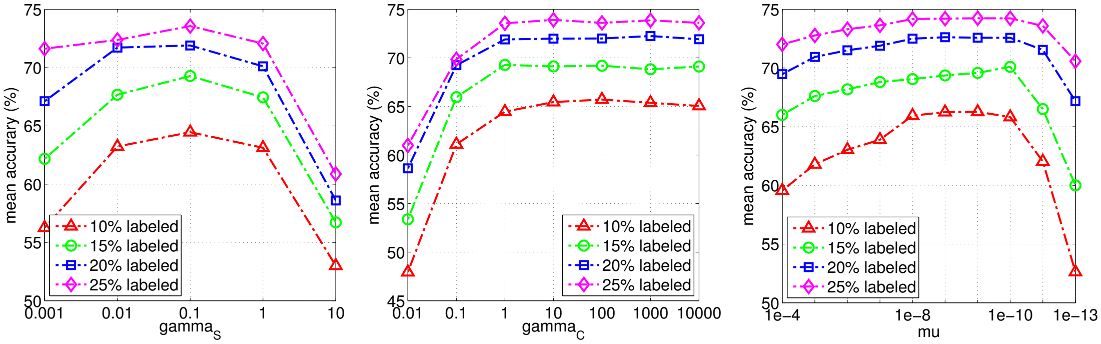
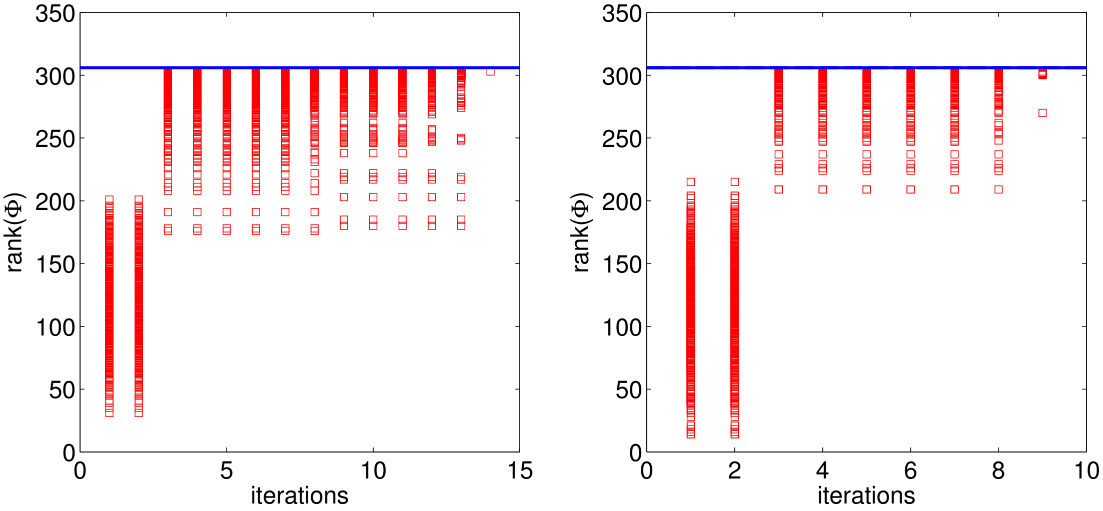


Figure 5: The evolutions of the average accuracy w.r.t the regularization term (left) and the fidelity term (middle) and the rank term (right) respectively.

5. Conclusion

We introduced in this paper, a new transductive learning approach for kernel design and classification. The strength of our contribution resides in the variational framework that allows us to explicitly design an optimal kernel map as a part of the learning process. When compared to baseline inductive methods, multiple kernel learning and

[4] http://www.dii.unisi.it/~melacci/lapsvmp/
[5] http://svmlight.joachims.org/



(a) $\mu = 10^{-4}$

(b) $\mu = 10^{-8}$

Figure 6: Figures above show the rank density of learned Φ's after every iteration of Alg. (1). Squares show the ranks of kernel maps Φ (related to 556 segmentation problems) over $t \leq 15$ iterations. In very first iterations, rank distribution is spanned over a wide range of low rank-numbers, i.e. 25 - 200, then it jumps to a more stable rank and finally approaches (not equal) to the upper-bound $\max(\ell, n) + 1$ (blue line). For large values of μ (Fig. 6(a)), TransRMF requires more iterations to achieve stable rank while smaller values of μ (Fig. 6(b)) help TransRMF cut off iterations as the rank converges faster, i.e. shorter strides of square dots compared to (Fig. 6(a)).

also related transductive methods, our approach shows superior accuracy on the challenging object class segmentation task.
As a future extension of this work, we will investigate the application of this method to other tasks including interactive image retrieval.

References

Asa, Ong Cheng Soon, Sonnenburg Sören, Schölkopf Bernhard, and Rätsch Gunnar Ben-Hur. Support vector machines and kernels for computational biology. *PLoS Comput Biol*, 4(10), 10 2008.

M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *J. Mach. Learn. Res.*, 7: 2399–2434, December 2006.

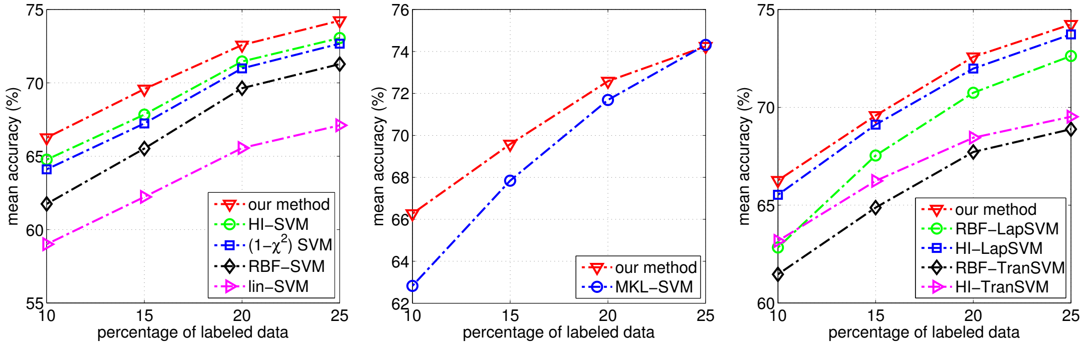


Figure 7: These diagrams show comparison of TransRMF against inductive methods in left (SVMs with various kernels) and multiple kernel learning (middle), and transductive learning (right) respectively. Note that all the performances are shown for different percentages of labeled data.

O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-Supervised Learning.* MIT Press, Cambridge, MA, 2006.

O. Duchenne, J.-Y. Audibert, R. Keriven, J. Ponce, and F. Segonne. Segmentation by transduction. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

Gene H. Golub and Charles F. Van Loan. *Matrix Computations.* The Johns Hopkins University Press, 3rd edition, 1996.

Roger A. Horn and Charles R. Johnson. *Matrix Analysis, chapter 5.* Cambridge University Press, 1990. ISBN 0521386322.

T. Joachims. Transductive inference for text classification using support vector machines. In *ICML*, pages 200–209, 1999.

T. Joachims. *Learning to Classify Text Using Support Vector Machines – Methods, Theory, and Algorithms.* Kluwer/Springer, 2002.

G-R Lanckriet, P. Bartlett, and M. Jordan. Learning the Kernel Matrix with Semidefinite Programming. *Journal of Machine Learning Research*, 5:27–72, 2004.

Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Y. Ng. Efficient sparse coding algorithms. In *NIPS*, pages 801–808, 2006.

S. Maji, A-C. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. In *CVPR*, 2008.

Stefano Melacci and Mikhail Belkin. Laplacian support vector machines trained in the primal. *J. Mach. Learn. Res.*, pages 1149–1184, July 2011.

A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet. SimpleMKL. *JMLR*, 9: 2491–2521, 2008.

B. Schölkopf and AJ. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond.* The MIT Press, December 2001.

M. Seeger. Learning with labeled and unlabeled data. *Technical Report, University of Edinburgh*, 2001.

J. Tighe and S. Lazebnik. Superparsing: Scalable nonparametric image parsing with superpixels. In *ECCV (5)*, pages 352–365, 2010.

V. Vapnik. *Statistical Learning Theory.* Wiley, New York, 1998.

V. Vapnik and A. Sterin. On structural risk minimization or overall risk in a problem of pattern recognition. *Automation and Remote Control*, 10(3):1495–1503, 1977.

M. Varma and D. Ray. Learning The Discriminative Power-Invariance Trade-Off. *2007 IEEE 11th ICCV*, pages 1–8, 2007.

Dépôt légal : 2012 – 1er trimestre
Imprimé à Télécom ParisTech – Paris
ISSN 0751-1345 ENST D (Paris) (France 1983-9999)

Télécom ParisTech

Institut TELECOM - membre de ParisTech

46, rue Barrault - 75634 Paris Cedex 13 - Tél. + 33 (0)1 45 81 77 77 - www.telecom-paristech.frfr

Département TSI