Learning a Distance Metric from Relative Comparisons between Quadruplets of Images

Marc T. Law, Nicolas Thome, Matthieu Cord

Noname manuscript No.
(will be inserted by the editor)

Learning a Distance Metric from Relative Comparisons between Quadruplets of Images

Marc T. Law · Nicolas Thome · Matthieu Cord

Received: date / Accepted: date

Abstract This paper is concerned with the problem of learning a distance metric by considering meaningful and discriminative distance constraints in some contexts where rich information between data is provided. Classic metric learning approaches focus on constraints that involve pairs or triplets of images. We propose a general Mahalanobis-like distance metric learning framework that exploits distance constraints over up to four different images. We show how the integration of such constraints can lead to unsupervised or semi-supervised learning tasks in some applications. We also show the benefit on recognition performance of this type of constraints, in rich contexts such as relative attributes, class taxonomies and temporal webpage analysis.

Keywords Metric Learning · Relative Attributes · Web Mining · Change Detection

1 Introduction

Image representation for classification has been deeply investigated in recent years [13,47]. For instance, the traditional Bag-of-Visual-Words representation [54] has been extended for the coding step [21,64] as well as for the pooling [4], with bio-inspired models [50,58]. Nonetheless, the choice of a good similarity function is also crucial to compare, classify and retrieve images. Extensive work has been done (see the survey [34]) to learn a (dis)similarity function that is relevant to some specific tasks. One of the most standard forms of (dis)similarity functions used for learning is distance (pseudo-)metric.

Distance metric learning has been proven to be useful in many Computer Vision applications, such as image classification [12,20,45], image retrieval [12], face verification or person re-identification [22,46].

Each metric learning problem depends on both the application task and the way the input data is provided. In other words, it depends on the input data representation (e.g., unimodal or multimodal), the type of labels and/or relations between samples, the formulation of the metric, the resulting optimization problem and its computational complexity.

Binary (boolean) similarity labels on image pairs [63] are usually provided for the learning. In the context of face verification [22], binary similarity labels establish whether two images should be considered as equivalent (i.e., the two face images represent the same person) or not. Metrics are learned in order to minimize dissimilarities between similar pairs while separating dissimilar ones.

Recently, some attempts have been made to go beyond learning metrics using only pairwise similarity information. For instance, constraints that involve triplets of images have been considered to learn metrics [12,35,62]. These attempts follow the work of [32] that made the argument that humans are better at providing relative (hence triplet-wise) comparisons than absolute (i.e., pairwise) comparisons. Notably, the most natural way to generate triplet constraints is to exploit, if available, class membership information. The goal is then to have distances between images in the same class smaller than distances between images from different classes. More sophisticated triplet constraints can also be inferred from richer relationships. For example, Verma et al. [60] learn a similarity that depends on a class hierarchy: an image should be closer to another image from a sibling class than to any image from a more distant class in the hierarchy. In other contexts, such as learning attributes, one can exploit specific rankings between classes in order to learn a semantical metric and representation space [40,48].

In this paper, we focus on these rich contexts for learning similarity metrics. Instead of pairwise or triplet-wise tech-

Sorbonne Universités, UPMC Univ Paris 06, UMR 7606, LIP6, F-75005, Paris, France
E-mail: Marc.Law@lip6.fr, Nicolas.Thome@lip6.fr, Matthieu.Cord@lip6.fr

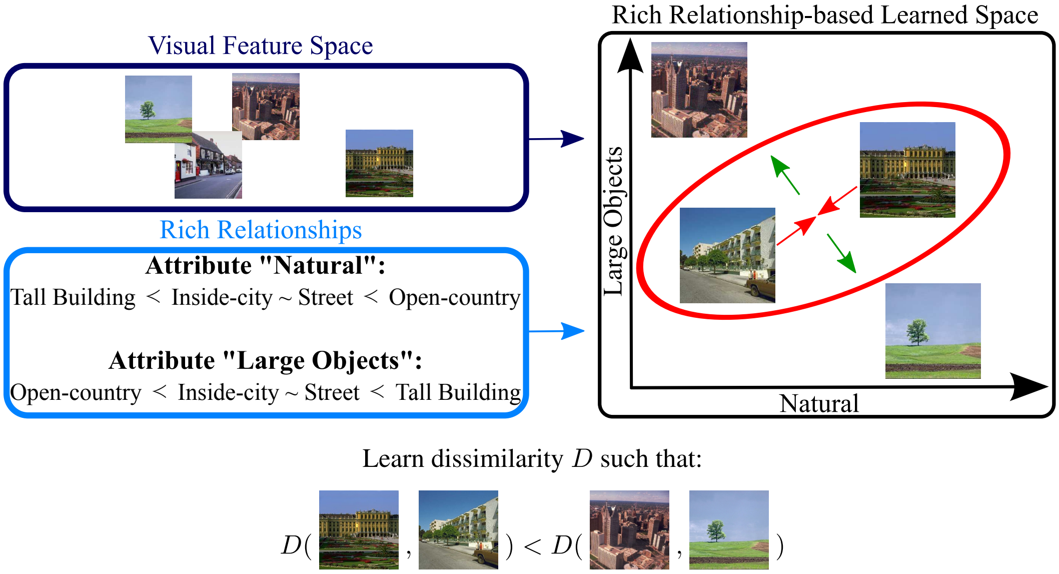

Visual Feature Space
Rich Relationships
Attribute "Natural":
Tall Building < Inside-city ~ Street < Open-country
Attribute "Large Objects":
Open-country < Inside-city ~ Street < Tall Building
Rich Relationship-based Learned Space
Large Objects
Natural
Learn dissimilarity D such that:
$D($, $) < D($, $)$

Fig. 1 Illustration of the quadruplet-wise (Qwise) strategy in a relative attribute context. The goal is to learn a projection of scene images by exploiting rich relationships (here relative attributes) over quadruplets of images such that samples satisfy the relationship constraints in the projected space.

niques, we propose to investigate meaningful relations between quadruplets of images. We first motivate why this type of constraints may be useful in different contexts. For this purpose, we illustrate in Fig. 1 our approach in the context of relative attributes [48] for which the goal is to learn a projection of visual image features into a high-level semantic space. Each dimension of this semantic space corresponds to the degree of presence of a given attribute (e.g., the presence of nature or large objects in the images). Four scene classes are considered in the figure: *tall building* (T), *inside city* (I), *street* (S) and *open country* (O). Class membership information and relative orderings on classes for the attributes “Natural” and ”Large objects” are also provided. In [48], they want the projected representations of images in the semantic space to satisfy the relative attribute constraints defined over their respective classes. They consider only inequality constraints (i.e., $(e) \prec (f)$: the presence of an attribute is stronger in class (f) than in class (e)) and pairwise equivalence constraints (i.e., $(f) \sim (g)$: the presence of an attribute is equivalent in class (f) and class (g)). In Fig. 1, the degrees of presence of nature and large objects in the *street* image and the *inside-city* image are clearly not equivalent. Learning a projection that enforces an equivalence (i.e., the same position) of these two images in the high-level semantic space, as proposed in [48], then seems limited. We argue in this paper that this type of absolute similarity information between the two images is restrictive, and thus noisy. Alternatively, a natural way to relax and exploit this equivalence information is to majorize the difference of attribute presence by considering pairs of classes for which the difference of attribute presence is greater. Such pairs of classes are easy to find when the following ordering is given: $(e) \prec (f) \sim (g) \prec (h)$. The difference between (f) and (g) is smaller than the difference between (h) and (e). Since the proposed relaxed constraints better describe relative orderings between the different images, they are more robust to noisy information.

This paper is an extension of our own previous work [40] where we proposed to exploit constraints that involve quadruplets of images to learn simple forms of distance metrics. We propose here to enrich the model in [40] by combining quadruplet-wise with pairwise constraints to learn a metric. In contexts where quadruplet-wise constraints can be automatically generated, this allows to learn a metric in an semi-supervised way. We also extend our model to learn a more general form of Mahalanobis distance metric. We present optimization techniques to deal with a large number of constraints and make the learning scheme more powerful. We extend the experiments in order to study the impact of the proposed constraints on recognition performance in different contexts.

The remainder of the paper is structured as follows. Section 2 presents related work on distance metric learning. We describe our learning problem in Section 3 and its optimization in Section 4. In Sections 5 to 7, we present experiments on temporal webpage analysis, class taxonomy and relative attribute applications. Finally, we offer our conclusions and plans for future research.

Notations: let $\mathbb{S}^d$ and $\mathbb{S}^d_+$ denote the sets of $d \times d$ real-valued symmetric and symmetric positive semidefinite (PSD) matrices, respectively. The set of considered images is $\mathcal{P} = \{\mathcal{I}_i\}_{i=1}^M$, each image $\mathcal{I}_i$ is represented by a feature vector

$\mathbf{x}_i \in \mathbb{R}^d$. For matrices $\mathbf{A} \in \mathbb{S}^d$ and $\mathbf{B} \in \mathbb{S}^d$, denote the Frobenius inner product by $\langle \mathbf{A}, \mathbf{B} \rangle = \text{tr}(\mathbf{A}^\top \mathbf{B})$ where tr denotes the trace of a matrix. $\Pi_{\mathcal{C}}(\mathbf{x})$ is the Euclidean projection of the vector or matrix $\mathbf{x}$ on the convex set $\mathcal{C}$ (see Chapter 8.1 in [8]). For a given vector $\mathbf{a} = (a_1, \dots, a_d)^\top \in \mathbb{R}^d$, $\text{Diag}(\mathbf{a}) = \mathbf{A} \in \mathbb{S}^d$ corresponds to a square diagonal matrix such that $\forall i, A_{ii} = a_i$ where $\mathbf{A} = [A_{ij}]$. For a given square matrix $\mathbf{A} \in \mathbb{R}^{d \times d}$, $\text{Diag}(\mathbf{A}) = \mathbf{a} \in \mathbb{R}^d$ corresponds to the diagonal elements of $\mathbf{A}$ set in a vector: i.e., $a_i = A_{ii}$. Finally, for $x \in \mathbb{R}$, let $[x]_+ = \max(0, x)$.

2 Related Work

The goal of distance metric learning is to produce a linear transformation of data which is optimized to fit semantical relationships between training samples. In this paper, the distance metric considered for learning is the widely used Mahalanobis-like distance metric $D_{\mathbf{M}}$ parameterized by a PSD matrix $\mathbf{M} \in \mathbb{S}^d_+$:

$$\begin{aligned} D^2_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_j) &= \Phi(\mathcal{I}_i, \mathcal{I}_j)^\top \mathbf{M} \, \Phi(\mathcal{I}_i, \mathcal{I}_j) \\ &= \langle \mathbf{M}, \Phi(\mathcal{I}_i, \mathcal{I}_j)\Phi(\mathcal{I}_i, \mathcal{I}_j)^\top \rangle \\ &= \langle \mathbf{M}, \mathbf{C}_{ij} \rangle \end{aligned} \tag{1}$$

where $\Phi(\mathcal{I}_i, \mathcal{I}_j) \in \mathbb{R}^d$ is the aggregation in a single vector of d elementary dissimilarity functions ϕ_k where $\forall k \in \{1, \dots, d\}$, $\phi_k : \mathcal{P} \times \mathcal{P} \to \mathbb{R}$. The commonly used function Φ is $\Phi(\mathcal{I}_i, \mathcal{I}_j) = \mathbf{x}_i - \mathbf{x}_j$. For convenience, we note the outer product $\mathbf{C}_{ij} = \Phi(\mathcal{I}_i, \mathcal{I}_j)\Phi(\mathcal{I}_i, \mathcal{I}_j)^\top$. Although most approaches learn the same form of distance metric, different types of information or optimization methods are used in the learning process.

2.1 Metric Learning via linear transformations and unsupervised approaches

Different optimization methods to learn a metric have been proposed in the literature. For instance, every symmetric PSD matrix $\mathbf{M} \in \mathbb{S}^d_+$ can be decomposed as the product $\mathbf{M} = \mathbf{L}^\top \mathbf{L}$ where $\mathbf{L} \in \mathbb{R}^{e \times d}$ and $e \geq \text{rank}(\mathbf{M}) = \text{rank}(\mathbf{L})$. As a consequence, learning a PSD matrix $\mathbf{M}$ and learning a linear transformation parameterized a matrix $\mathbf{L} \in \mathbb{R}^{e \times d}$ are two equivalent ways to learn a metric [62]. Indeed, every Mahalanobis-like distance metric can be rewritten[1] as a function of $\mathbf{L}$ (i.e., $D^2_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_j) = \|\mathbf{L}(\mathbf{x}_i - \mathbf{x}_j)\|^2$), and from any linear transformation parameterized by $\mathbf{L}$, any distance $D_{\mathbf{L}^\top \mathbf{L}} = D_{\mathbf{M}}$ can be induced. This is why eigenvector methods, such as *principal component analysis* (PCA) and *linear discriminant analysis* (LDA), that learn a linear transformation in order to satisfy some criterion (e.g., projecting the training inputs into a variance-maximizing subspace in the case of PCA) can be considered as metric learning approaches [62].

In addition to PCA and *manifold learning* approaches, for which the key idea is to learn an underlying low-dimensional manifold that preserves distances in the input space between observed data [6,57], several approaches learn a metric in an unsupervised manner (i.e., from an unlabeled dataset) by assuming the availability of several (partially) labeled datasets that share the same metric [17]. It is the case in the context of partitioning problems where a supervised learning framework aims at learning how to perform an unsupervised task [18,37]. This framework is also referred to as *supervised clustering* [17,18] and has been applied in different domains (e.g., video segmentation, image segmentation, change-point detection in bioinformatics [23]...).

We focus in this work on supervised learning methods where constraints over distances between training samples are given as input of the algorithm, and a metric is learned to satisfy most of them. The learning strategy is usually driven by the form of provided information and the application. When supervision is considered, the way the dataset is labeled, e.g., binary labels on pairwise or triplet-wise rankings, greatly affects the optimization problem formulation. In practice, the more informative constraints one gives, the better the performance of the learned metric is.

2.2 Pairwise optimization framework

In pairwise approaches [46,63], the problem is formulated as learning the PSD matrix $\mathbf{M} \in \mathbb{S}^d_+$ such that the distance metric $D^2_{\mathbf{M}}$ is optimized on a training set composed of a subset $\mathcal{S}$ of pairs of similar images and a subset $\mathcal{D}$ of pairs of dissimilar images. For instance, in the context of Mahalanobis Metric Learning for Clustering, Xing et al. [63] define the resulting convex objective function[2]:

$$\min_{\mathbf{M} \in \mathbb{S}^d_+} \sum_{(\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{S}} D^2_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_j) \quad s.t. \sum_{(\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{D}} \sqrt{D^2_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_j)} \geq 1 \tag{2}$$

The distances of similar pairs are minimized whereas dissimilar pairs are separated. A regularization term may be added: e.g., the term $\sum_{(\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{S}} D^2_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_j) = \text{tr}(\mathbf{M}\mathbf{A})$ with the PSD matrix $\mathbf{A} = \sum_{(\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{S}} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^\top$ can be seen as a regularizer [34] in Eq. (2). In [63] and in most metric learning algorithms, a (projected) gradient method is used to efficiently solve the optimization problem. A hinge loss or a generalized logistic loss function may be used to

[1] Note that the decomposition from $\mathbf{M}$ is not unique.

[2] The authors use the constraint $\sum_{(\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{D}} \sqrt{D^2_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_j)}$ instead of the usual squared Mahalanobis distance to avoid learning a matrix $\mathbf{M}$ that is rank 1.

express all the constraints (over $\mathcal{S}$ and $\mathcal{D}$) in a single objective function [46]. In the context of face verification, Mignon and Jurie [46] try to learn a metric such that the distances of similar images are smaller than a given threshold $b = 1$ whereas the distances of dissimilar images are greater than that threshold. They formulate their optimization problem as the sum of the loss related to each of these constraints:

$$\min_{\mathbf{M}\in\mathbb{S}^d_+} \sum_{(\mathcal{I}_i,\mathcal{I}_j)\in(\mathcal{S}\cup\mathcal{D})} \ell_\beta\left(y_{ij}(D^2_{\mathbf{M}}(\mathcal{I}_i,\mathcal{I}_j)-1)\right) \tag{3}$$

where $y_{ij} \in \{-1, 1\}$ indicates whether the images $(\mathcal{I}_i, \mathcal{I}_j)$ are dissimilar or not, and $\ell_\beta(x) = \frac{1}{\beta}\log(1+e^{\beta x})$ is the generalized logistic loss function and a smooth approximation of the hinge loss function $h(x) = \max(0, x)$. This learning process may be extended to kernel functions [26,46].

Many supervised approaches have been proposed recently to generate training sets $\mathcal{S}$ and $\mathcal{D}$. Most of those approaches use binary similarity labels: two images represent the same object or not [22,63], two images belong to the same class or not [46], an image is relevant to a query or not [12,20] .

2.3 Triplet-based methods

Another way to exploit labeled datasets is to consider a set $\mathcal{T}$ of triplets of images $\mathcal{T} = \{(\mathcal{I}_i, \mathcal{I}_i^+, \mathcal{I}_i^-)\}_{i=1}^N$ where the distance $D_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_i^+)$ between $(\mathcal{I}_i, \mathcal{I}_i^+)$ is smaller than the distance $D_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_i^-)$ between $(\mathcal{I}_i, \mathcal{I}_i^-)$. This type of constraints is easy to generate in classification contexts: the pair of images $(\mathcal{I}_i, \mathcal{I}_i^+) \in \mathcal{S}$ is sampled using images from the same class and $(\mathcal{I}_i, \mathcal{I}_i^-) \in \mathcal{D}$ from different classes [20, 35,59,62]. For instance, Large Margin Nearest Neighbor algorithm (LMNN) [62] learns a Mahalanobis distance for k-Nearest Neighbors (k-NN) approach using these triplet-wise training sets. More precisely, LMNN uses a scheme similar to Eq. (2) in order to enforce $D_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_i^-)$ to be larger than $D_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_i^+)$ where $\mathcal{I}_i^+$ is one of the k target nearest neighbors of $\mathcal{I}$. Their optimization problem can be formulated as:

$$\begin{aligned}\min_{\mathbf{M}\in\mathbb{S}^d_+,\xi} & \sum_{(\mathcal{I}_i,\mathcal{I}_i^+)\in\mathcal{S}} D^2_{\mathbf{M}}(\mathcal{I}_i,\mathcal{I}_i^+) + \sum_{(\mathcal{I}_i,\mathcal{I}_i^+,\mathcal{I}_i^-)\in\mathcal{T}} \xi_i \\ & \text{s.t.} D^2_{\mathbf{M}}(\mathcal{I}_i,\mathcal{I}_i^-) \geq 1 + D^2_{\mathbf{M}}(\mathcal{I}_i,\mathcal{I}_i^+) - \xi_i \\ & \quad \forall(\mathcal{I}_i,\mathcal{I}_i^+,\mathcal{I}_i^-)\in\mathcal{T}, \xi_i \geq 0\end{aligned} \tag{4}$$

where the same regularizer as in Eq. (2) is used.

In classification task, Frome et al. [19,20] also generate triplets of images using the same strategy as LMNN. However, their metric learning framework that is inspired by RankSVM [27] and based on a linear combination of patch-to-image distances, is different.

In image retrieval, the Online Algorithm for Scalable Image Similarity (OASIS) [12] learns a non-PSD square matrix $\mathbf{M}$ in the similarity function $S_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_j) = \mathbf{x}_i^\top \mathbf{M}\mathbf{x}_j$. For any triplet of images $(\mathcal{I}_i, \mathcal{I}_i^+, \mathcal{I}_i^-)$, a safety margin constraint is defined: $S_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_i^+) \geq S_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_i^-) + 1$, which is equivalent to $\mathbf{x}_i^\top \mathbf{M}(\mathbf{x}_i^+ - \mathbf{x}_i^-) \geq 1$. As explained by the authors [12], OASIS requires images represented as sparse vectors to be computationally efficient.

In the next subsection, we present different contexts where information richer than the sole membership of $(\mathcal{I}_i, \mathcal{I}_j)$ in $\mathcal{S}$ or $\mathcal{D}$ can be exploited to learn a distance metric. Such contexts involve, for instance, taxonomies which have a hierarchical structure and describe relationships between the different classes.

2.4 Exploiting rich relationships between samples

Some approaches investigate other types of information than class membership or richer semantic relationships in order to learn a metric that reflects more accurately global relations. For instance, in [61], a class taxonomy is used in order to get elements of related classes, close to each other. Verma et al. [60] extend this work by learning a local Mahalanobis distance metric for each category in a hierarchy. Shaw et al. [51] learn a distance metric from a network such that the learned distances are tied to the inherent connectivity structure of the network. Hwang et al. [24] learn discriminative visual representations while exploiting external semantic knowledge about object category relationships. Parikh and Grauman [48] use semantic comparisons between classes over different criteria. They consider totally ordered sets of classes that describe relations among classes. Based on these rich relations, they learn image representations by exploiting only pairwise class relations. We propose to explore this type of data knowledge in metric learning for image comparison.

2.5 Quadruplet-based methods

Noting that pairwise or triplet-wise approaches may, sometimes, be limited (see Section 1), our metric learning framework is based on constraints over quadruplets.

Relative distances that involve four samples have already been considered in the context of embedding problems. A classic approach of embedding problems is Multidimensional Scaling (MDS) that consists in assigning Euclidean coordinates to a set of objects such that a given set of dissimilarity, similarity or ordinal relations between the points are satisfied. Unlike metric learning approaches, classic embedding methods do not extend to new samples, a new embedding has to be learned each time a (new) test sample is added.

In the context of non-metric MDS, Shepard [52,53] considered in 1962 the following problem that involves quadruplets of samples:

Problem: Given a symmetric zero diagonal matrix of distances $\varDelta = [d_{ij}] \in \mathbb{S}^n$ between samples i and j, find the Euclidean coordinates $\mathbf{X} = [\mathbf{x}_i] \in \mathbb{R}^{d \times n}$ such that:

$$\forall i, j, k, l \quad \|\mathbf{x}_i - \mathbf{x}_j\|_2^2 < \|\mathbf{x}_k - \mathbf{x}_l\|_2^2 \Longleftrightarrow d_{ij} < d_{kl} \qquad (5)$$

In 1964, Kruskal posed the problem as an optimization problem and introduced an algorithm to solve it [33]. He formulated the input distance matrix $\varDelta$ as an exhaustive table of distances where all the values of d_{ij} are given as input. By noting the output distance matrix $\hat{\varDelta} \in \mathbb{S}^n$ which contains the distances $\hat{d}_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|^2$ of each pair of samples. The goal is to find an Euclidean embedding such that each distance $\hat{d}_{ij}$ is close enough to d_{ij}. This leads to the problem of minimizing the following criterion function called *stress*:

$$\sigma_1(\mathbf{X}) = \min_{\theta} \frac{\sum_{ij} \left(\|\mathbf{x}_i - \mathbf{x}_j\|^2 - \theta(d_{ij})\right)^2}{\sum_{ij} \|\mathbf{x}_i - \mathbf{x}_j\|^2} \qquad (6)$$

where θ is an arbitrary monotonic function. The problem in Eq. (6) consists in minimizing the distance between the scalar input value $\theta(d_{ij})$ and the distance between the samples i and j in the underlying low-dimensional space. The underlying idea is that if $\sigma_1(\mathbf{X})$ is minimized, then (most of) the constraints in Eq. (5) are satisfied. The smaller the value of the stress value in Eq. (6), the greater the correspondance between the matrices $\varDelta$ and $\hat{\varDelta}$. Noticing that the formulation of the problem formulated by Kruskal requires the magnitudes of all the distances d_{ij} as input, and not the relative orderings of distances as in Eq. (5), Agarwal et al. [3] propose to consider only ordinal information as input to learn a generalized non-metric multidimensional scaling. This work is extended to kernels in [43].

In the context of embedding problem, Hwang et al. [25] exploit analogy preserving constraints that involve four concepts (e.g., “a canine is to a dog as a feline is to cat” or “a fish is to water as a bird is to sky”). However, they are only interested in equivalence constraints.

In our previous work [40], we proposed to include constraints that involve up to four different images to learn a distance metric. Contrary to Eq. (5), we did not learn an embedding but a metric with different types of supervision. Constraints on quadruplets allow to better exploit rich relationships between samples in different contexts. In [40], we applied our framework to the contexts of relative attributes [48], hierarchical taxonomy classification [60] and temporal webpage analysis. For simplicity, we constrained our metric to be paramaterized by vectors instead of a full matrix. In this paper, we consider a metric that is parameterized by a full matrix as well. This metric formulation allows to better exploit correlations between feature images. We explain why our proposed constraints are a generalization of pairwise and tripletwise constraints. We also extend [40] with significant differences and contributions that we point out in the following. Especially, we:

• extend our proposed model so that absolute/non-relative distance constraints are considered in the learning framework. In particular, this allows to learn a distance threshold that separates similar pairs from dissimilar pairs when both quadruplet-wise and pairwise constraints are combined. This is particularly useful in the webpage analysis context framework proposed in [40] where unsupervised quadruplet-wise constraints, that are automatically generated using temporal information, are now combined with supervised pairwise constraints. By combining large unlabeled datasets with small labeled datasets, supervision cost (i.e., human annotation) is minimized while learning a meaningful metric.

• discuss optimization issues caused by a possibly very large number of constraints. We present optimization techniques, such as active set methods and the 1-slack cutting plane method, that can be useful to deal with a large number of constraints and make the learning scheme tractable.

• extend the experiments introduced in [40]:

1. temporal webpage analysis: we thoroughly study the benefits for recognition of (1) learning a metric parameterized by a full matrix and (2) combining unsupervised quadruplet-wise constraints with a relatively small number of supervised pairwise constraints.
2. hierarchical taxonomy classification: we demonstrate how (1) our method can deal with a large number of constraints and (2) full matrix distance metrics improve recognition over diagonal matrix distance metrics in the k-NN classification framework.
3. relative attributes: we analyse the robustness introduced by our proposed quadruplet-wise constraints. We present and compare different strategies for sampling constraints to compensate for labeling imprecisions. We investigate the impact of these strategies as a function of the number of exploited constraints.

3 Quadruplet-wise Similarity Learning Framework

3.1 Quadruplet Constraints

As explained in Section 2.5, our goal is to learn a metric that satisfies constraints that involve quadruplets of images.

In some cases (e.g., Fig. 1), pair or triplet constraints may be noisy or irrelevant, leading to less than optimal learning schemes when provided at a class level. On the other hand, working on appropriate dissimilarities between quadruplets of images limits the risk of incorporating misleading annotations. We are given a set $\mathcal{P}$ of images $\mathcal{I}_i$, and the target dissimilarity function $D : \mathcal{P} \times \mathcal{P} \to \mathbb{R}$ between pairs of images $(\mathcal{I}_i, \mathcal{I}_j)$, we note $D(\mathcal{I}_i, \mathcal{I}_j) = D_{ij}$. In this paper, interested in comparing pairs of dissimilarities (D_{ij}, D_{kl}).

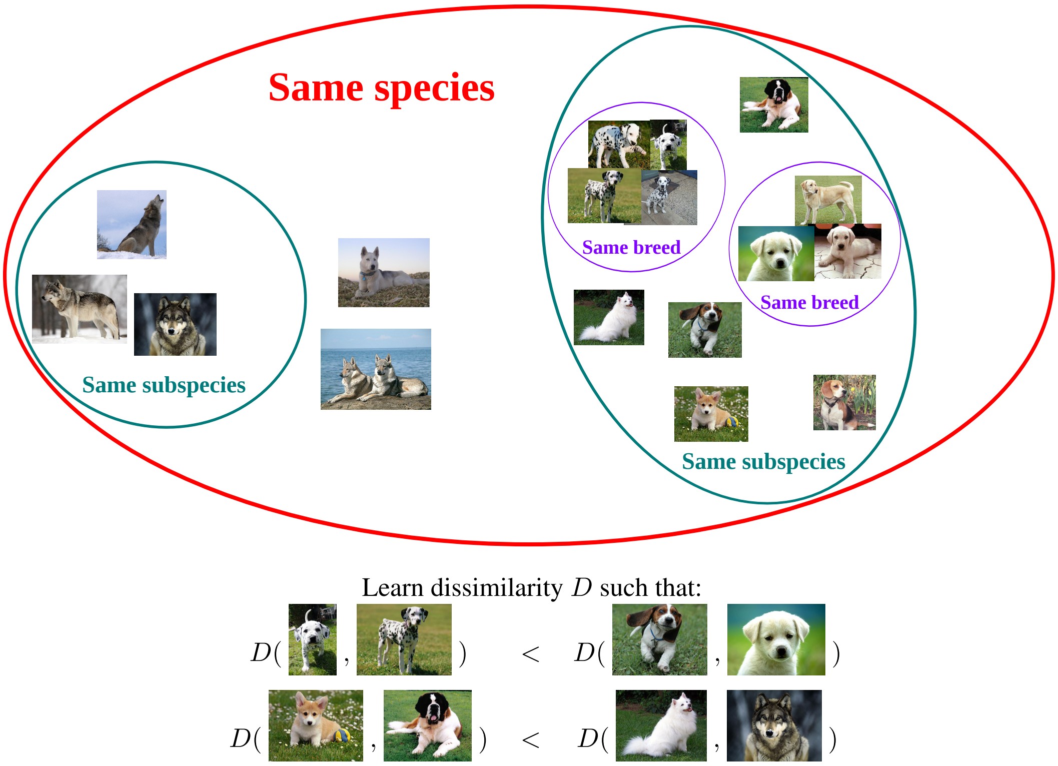

Same species
Same subspecies
Same subspecies
Same breed
Same breed
Learn dissimilarity D such that:
$D($, $)$ $<$ $D($, $)$
$D($, $)$ $<$ $D($, $)$

Fig. 2 Illustration of the quadruplet-wise (Qwise) strategy in a class taxonomy context. The goal is to learn a projection of animals of the same species such that members of the same breed are closer to each other than members from different breeds, and members from the same subspecies are closer to each other than member from different subspecies.

Each of them involves up to four different images $(\mathcal{I}_i, \mathcal{I}_j, \mathcal{I}_k, \mathcal{I}_l)$. Two types of relations $\mathcal{R}$ are considered between D_{ij} and D_{kl}: (1) strict inequality between dissimilarities: $D_{ij} < D_{kl}$, (2) non-strict inequality: $D_{ij} \leq D_{kl}$. Note that $D_{ij} = D_{kl}$ can be rewritten as two relations $D_{ij} \leq D_{kl}$ and $D_{ij} \geq D_{kl}$.

In order to deal with these constraints, we approximate them by creating the set of constraints $\mathcal{N}$ in this way:

$$\forall q = (\mathcal{I}_i, \mathcal{I}_j, \mathcal{I}_k, \mathcal{I}_l) \in \mathcal{N}, D_{kl} \geq D_{ij} + \delta_q \qquad (7)$$

where $\delta_q \in \mathbb{R}$ is a safety margin specific to the quadruplet q. The non-strict inequality constraint corresponds to $\delta_q = 0$. And the strict inequality constraint corresponds to $\delta_q > 0$, δ_q is usually set to 1 (i.e., $\delta_q = 1$).

Actually, Eq. (7) is a generalization of triplet-wise and pairwise constraints. Indeed:

- every triplet-wise constraint $D_{ik} \geq D_{ij} + \delta_q$ can be formulated by creating the quadruplet $q = (\mathcal{I}_i, \mathcal{I}_j, \mathcal{I}_i, \mathcal{I}_k) \in \mathcal{N}$.
- every pairwise constraint that involves a dissimilar pair of images $(\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{D}, D_{ij} \geq l$, where l is a given lower bound that represents the minimum value such that $(\mathcal{I}_i, \mathcal{I}_j)$ are considered as dissimilar, can be formulated by creating the quadruplet $q = (\mathcal{I}_i, \mathcal{I}_i, \mathcal{I}_i, \mathcal{I}_j) \in \mathcal{N}$ with $\delta_q = l$.
- every pairwise constraint that involves a similar pair of images $(\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{S}, u \geq D_{ij}$, where u is a given upper bound that represents the maximum value such that images $(\mathcal{I}_i, \mathcal{I}_j)$ are considered as similar, can be formulated by creating the quadruplet $q = (\mathcal{I}_i, \mathcal{I}_j, \mathcal{I}_i, \mathcal{I}_i) \in \mathcal{N}$ with $\delta_q = -u$.

Although quadruplet-wise constraints can be inferred from pairwise approaches [14,46], the converse is not true. Indeed, if the two pairs $(\mathcal{I}_i, \mathcal{I}_j)$ and $(\mathcal{I}_k, \mathcal{I}_l)$ are in $\mathcal{S}$ and $\mathcal{D}$, respectively, the following constraints $D_{ij} < D_{kl}$ can be inferred. However, the constraint $D_{ij} < D_{kl}$ does not imply that pairs $(\mathcal{I}_i, \mathcal{I}_j)$ and $(\mathcal{I}_k, \mathcal{I}_l)$ are in $\mathcal{S}$ and $\mathcal{D}$, respectively. In other words, from a quadruplet-wise constraint $D_{ij} < D_{kl}$, there is no need to determine arbitrary values of u and l such that $D_{ij} < u$ and $l < D_{kl}$ since u and l can take all the possible values (as long as $u \leq l$) and satisfy the quadruplet-wise constraint. Only the order of similarity between $(\mathcal{I}_i, \mathcal{I}_j)$ and $(\mathcal{I}_k, \mathcal{I}_l)$ is required. Since the provided annotations are less restrictive and thus less prone to noise, relative distances are particularly useful when human users that are not experts of the domain have to annotate similarity/relation information. A similar problem is pointed out in the context of relative attributes [48] in which boolean presence of an attribute is difficult to provide, whereas relative comparisons are easier and more natural for humans to annotate. Fig. 2 illustrates some examples of constraints for which a pairwise formulation is difficult, or at least for which constraints of relative distance comparisons seem more natural and intuitive. It shows different members of the *Ca-*

nis lupus species that are gathered together depending on their respective subspecies and breeds. By considering only pairwise similarity constraints, it is difficult to formulate the distance metric learning problem such that (1) members of the same breed are closer to each other than other members of the same subspecies are, and (2) members of the same subspecies are closer to each other than members from different subspecies. Depending on whether we consider members of the same subspecies as similar or dissimilar, the distance metric learned with pairwise constraints does not fully exploit the rich information given by the provided taxonomy. This limitation can be easily overcome by using relative distance comparison constraints as illustrated in Fig. 2.

We also note that quadruplet-wise constraints act as a complement to triplet-wise constraints to better describe rich relationships. For instance, the first quadruplet-wise constraint illustrated in Fig. 2 enforces the similarity between different Dalmatians. Indeed, we want animals of the same breed to be more similar to each other than animals of different breeds. Although this kind of information can be described with triplet-wise constraints by enforcing the distance between two Dalmatians to be smaller than the distance between one of these Dalmatians and an animal of another breed, quadruplet-wise constraints extend this kind of constraint by describing the fact that any pair of Dalmatians have to be closer to each other than any pair of animals of different breeds in general. Triplet-wise constraints then represent only a subset of the possible constraints that can describe such relationships.

We present in the following two different frameworks to learn a Mahalanobis distance metric that exploit this type of constraints. The first one considers the learning of a Mahalanobis distance metric parameterized by a full matrix $\mathbf{M} \in \mathbb{S}_+^d$. The second one considers the learning of a distance metric parameterized by one or many vectors that are learned independently.

3.2 Learning a Mahalanobis-like distance metric parameterized by a matrix

We present in this subsection the general Mahalanobis-like distance metric learning framework where a distance metric is parameterized by a full PSD matrix $\mathbf{M}$.

3.2.1 Optimization problem

The goal of our distance metric learning framework is to maximize the number of satisfied constraints in Eq. (7). However, the problem of maximizing the number of satisfied constraints in Eq. (7) is NP-hard [27], we then approximate it by using slack variables. By noting each quadruplet $q = (\mathcal{I}_i, \mathcal{I}_j, \mathcal{I}_k, \mathcal{I}_l) \in \mathcal{N}$, we optimize the following problem:

$$\begin{aligned} \min_{\mathbf{M} \in \mathbb{S}_+^d, \xi} \; & \boldsymbol{\Omega}(\mathbf{M}) + C_q \sum_{q \in \mathcal{N}} \xi_q \\ \text{s.t.} \forall q \in \mathcal{N}, \; & D_{\mathbf{M}}^2(\mathcal{I}_k, \mathcal{I}_l) \geq D_{\mathbf{M}}^2(\mathcal{I}_i, \mathcal{I}_j) + \delta_q - \xi_q \\ \forall q \in \mathcal{N}, \; & \xi_q \geq 0 \end{aligned} \tag{8}$$

where $\boldsymbol{\Omega}(\mathbf{M})$ is a regularization term and C_q a regularization parameter that controls the trade-off between fitting and regularization. Note that the problem in Eq. (8) is very similar to LMNN [62] (see Eq. (4)) with the exception that we exploit quadruplets of constraints instead of triplets.

Regularization: The choice of regularization has a significant impact on the learned distance model, both theorically and algorithmically. Different types of regularization have been proposed in the literature. Typically, the nuclear-norm regularizer $\boldsymbol{\Omega}(\mathbf{M}) = \|\mathbf{M}\|_*$ is known to prefer low-rank solutions. When $\mathbf{M} \in \mathbb{S}_+^d$ is PSD, the nuclear norm can be rewritten equivalently $\|\mathbf{M}\|_* = \text{tr}(\mathbf{M})$. The Frobenius norm regularizer $\boldsymbol{\Omega}(\mathbf{M}) = \frac{1}{2}\|\mathbf{M}\|_F^2 = \frac{1}{2}\langle \mathbf{M}, \mathbf{M} \rangle$ may be viewed as the matrix analog of the popular and standard squared-ℓ_2 regularization, particularly when $\mathbf{M}$ is a diagonal matrix since $\frac{1}{2}\|\mathbf{M}\|_F^2 = \frac{1}{2}\|\text{Diag}(\mathbf{M})\|_2^2$ in this case. In MMC [63] and LMNN [62], the term $\sum_{(\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{S}} D_{\mathbf{M}}^2(\mathcal{I}_i, \mathcal{I}_j) = \langle \mathbf{M}, \sum_{(\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{S}} \mathbf{C}_{ij} \rangle$ can also be seen as a regularizer (see Section 2.4.2.2 in [34]).

In the experiments, we use the same regularization as LMNN when we use LMNN as a baseline and we want to study the benefit of our proposed constraints in order to have a fair comparison. When we constrain the matrix $\mathbf{M} \in \mathbb{S}_+^d$ to be diagonal, we use the squared Frobenius norm in order to apply an efficient RankSVM [11] optimization scheme.

We will explain in Section 4.1 how to efficiently solve the problem in Eq. (8). We first propose to enrich the model with other types of constraints.

3.2.2 Combining pair and quadruplet constraints

As mentioned in Section 3.1, pairwise constraints can be rewritten as quadruplet-wise constraints. Nonetheless, in order to enhance the readability of the paper, we consider to explicitly distinguish the sets of similar image pairs $\mathcal{S}$ and of dissimilar image pairs $\mathcal{D}$ from the set of relative distance comparisons $\mathcal{N}$.

Especially, if we are provided with a set of similar pairs ($\mathcal{S}$) and a set of dissimilar pairs ($\mathcal{D}$), we expect the distances of similar pairs to be smaller than a given threshold u and the distances of dissimilar pairs to be greater than another threshold l (with $u \leq l$). To know whether a test pair is similar or dissimilar, one only needs to compute its distance and compare it to $b = \frac{u+l}{2}$. The resulting constraints can be written in this way:

$$\forall (\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{S} : D_{\mathbf{M}}^2(\mathcal{I}_i, \mathcal{I}_j) \leq u \tag{9}$$

$$\forall (\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{D} : D^2_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_j) \geq l \tag{10}$$

The integration of pairwise information in Eq. (8) then results in the following problem:

$$\begin{aligned} \min_{\mathbf{M} \in \mathbb{S}^d_+} & \boldsymbol{\Omega}(\mathbf{M}) + C_q \sum_{q \in \mathcal{N}} [\delta_q + \langle \mathbf{M}, \mathbf{C}_{ij} - \mathbf{C}_{kl} \rangle]_+ \\ & + C_p \sum_{(\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{S}} [\langle \mathbf{M}, \mathbf{C}_{ij} \rangle - u]_+ \\ & + C_p \sum_{(\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{D}} [l - \langle \mathbf{M}, \mathbf{C}_{ij} \rangle]_+ \end{aligned} \tag{11}$$

This problem is equivalent to Eq. (8) when $C_p = 0$ or $\mathcal{S} = \mathcal{D} = \emptyset$. It is convex w.r.t. $\mathbf{M}$. However, naive optimization methods can be comptutationally expensive to solve it. We discuss optimization schemes to efficiently solve this problem in Section 4.

We present an alternative distance metric formulation in order to obtain a convex optimization problem that can be solved efficiently.

3.3 Simplification of the model by optimizing over vectors

In order to obtain an efficient learning framework, we consider in this subsection cases where a distance metric is formulated as a function of one or many vectors. The distance metric is then learned by optimizing over those vectors.

We particularly focus on two contexts where the optimization process may be done efficiently [10, 11] by using this vector optimization approach and by learning a model with a relatively small number of parameters. The first one constrains the learned PSD matrix $\mathbf{M} \in \mathbb{S}^d_+$ to be diagonal (see Section 3.3.1). The second one considers that the training information is provided as multiple relative orderings; we then learn a linear transformation matrix whose rows each try to find a projection that satisfies a given relative ordering (see Section 3.3.2).

3.3.1 Learning a diagonal PSD matrix

In the first context, the PSD matrix $\mathbf{M} \in \mathbb{S}^d_+$ is constrained to be a diagonal matrix in Eq. (11). By noting $\mathbf{w} = \text{Diag}(\mathbf{M})$, it is easy to verify that, if $\mathbf{M}$ is a diagonal matrix, we have:

$$\begin{aligned} D^2_{\mathbf{M}}(\mathcal{I}_i, \mathcal{I}_j) &= \langle \mathbf{M}, \mathbf{C}_{ij} \rangle = \langle \text{Diag}(\mathbf{w}), \text{Diag}(\text{Diag}(\mathbf{C}_{ij})) \rangle \\ &= \mathbf{w}^\top [\Phi(\mathcal{I}_i, \mathcal{I}_j) \circ \Phi(\mathcal{I}_i, \mathcal{I}_j)] \end{aligned}$$

with $\text{Diag}(\mathbf{C}_{ij}) = \Phi(\mathcal{I}_i, \mathcal{I}_j) \circ \Phi(\mathcal{I}_i, \mathcal{I}_j)$ where $\circ$ is the Hadamard product (element-by-element product). For convenience, we also note $\Phi^{\circ 2}(\mathcal{I}_i, \mathcal{I}_j) = \Phi(\mathcal{I}_i, \mathcal{I}_j) \circ \Phi(\mathcal{I}_i, \mathcal{I}_j)$. The problem can then be rewritten as a function of $\mathbf{w}$.

In this context, the constraint $\mathbf{M} \in \mathbb{S}^d_+$ is equivalent to the constraint $\mathbf{w} \in \mathbb{R}^d_+$ (the elements of $\mathbf{w}$ are non-negative). Indeed, all the diagonal elements of a square diagonal matrix are its eigenvalues and a symmetric matrix is PSD iff all its eigenvalues are non-negative. We then consider the constraint $\mathbf{w} \in \mathbb{R}^d_+$ in this case.

3.3.2 Learning the rows of a linear transformation

If the provided annotations are M different dissimilarity functions (e.g., relative attributes), where each of them represents a relative ordering focused on a given criterion (e.g., $\mathcal{I}_i$ is more smiling than $\mathcal{I}_j$, $\mathcal{I}_i$ is younger than $\mathcal{I}_j$...), each row of the matrix $\mathbf{L} \in \mathbb{R}^{M \times d}$ can be learned independently. The m^{th} row of $\mathbf{L}$ (denoted $\mathbf{w}_m^\top$) satisfies the ordering of the m^{th} dissimilarity function $\mathscr{D}_{\mathbf{w}_m}(\mathcal{I}_i, \mathcal{I}_j) = \mathbf{w}_m^\top \Phi(\mathcal{I}_i, \mathcal{I}_j)$. The matrix $\mathbf{L}$ can then be written in this form:

$$\mathbf{L} = \begin{bmatrix} w_{1,1} & \dots & w_{1,d} \\ \vdots & \vdots & \vdots \\ w_{M,1} & \dots & w_{M,d} \end{bmatrix} = \begin{bmatrix} \mathbf{w}_1^\top \\ \vdots \\ \mathbf{w}_M^\top \end{bmatrix}, \ \mathbf{w}_m^\top : m^{\text{th}} \text{ row} \tag{12}$$

In the end, a linear transformation parameterized by the matrix $\mathbf{L}$ is learned, and, as explained in Section 2.1, learning a linear transformation is equivalent to learning a distance metric [62] parameterized by the matrix $\mathbf{M} = \mathbf{L}^\top \mathbf{L}$.

3.3.3 Unified problem formulation

In both cases mentioned above, the learning problem may be expressed as a linear combination of the parameter $\mathbf{w} \in \mathscr{C}^d$ where $\mathscr{C}^d$ is a d-dimensional convex set in $\mathbb{R}^d$. In this paper, the convex set $\mathscr{C}^d$ is either $\mathbb{R}^d$ or $\mathbb{R}^d_+$. Without loss of generality, we consider optimizing the following dissimilarity function:

$$\mathscr{D}_{\mathbf{w}}(\mathcal{I}_i, \mathcal{I}_j) = \mathbf{w}^\top \Psi(\mathcal{I}_i, \mathcal{I}_j) \text{ s.t. } \mathbf{w} \in \mathscr{C}^d \tag{13}$$

where

• $\Psi = \Phi^{\circ 2}$ and $\mathscr{C}^d = \mathbb{R}^d_+$ in the case $\mathbf{M} \in \mathbb{S}^d_+$ is a diagonal matrix (Section 3.3.1).

• $\Psi = \Phi$ and $\mathscr{C}^d = \mathbb{R}^d$ in the other case (Section 3.3.2).

We formulate our vector optimization problem as:

$$\begin{aligned} \min_{(\mathbf{w}, b, \xi)} & \frac{1}{2}(\|\mathbf{w}\|_2^2 + b^2) + C_q \sum_{q \in \mathcal{N}} \xi_q + C_p \sum_{(\mathcal{I}_i, \mathcal{I}_j) \in (\mathcal{S} \cup \mathcal{D})} \xi_{ij} \\ & \text{s.t.} \forall (\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{S}, \mathscr{D}_{\mathbf{w}}(\mathcal{I}_i, \mathcal{I}_j) \leq b - 1 + \xi_{ij} \\ & \quad \forall (\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{D}, \mathscr{D}_{\mathbf{w}}(\mathcal{I}_i, \mathcal{I}_j) \geq b + 1 - \xi_{ij} \\ & \quad \forall q \in \mathcal{N}, \mathscr{D}_{\mathbf{w}}(\mathcal{I}_k, \mathcal{I}_l) \geq \mathscr{D}_{\mathbf{w}}(\mathcal{I}_i, \mathcal{I}_j) + \delta_q - \xi_q \\ & \quad \xi_q \geq 0, \xi_{ij} \geq 0, \mathbf{w} \in \mathscr{C}^d, b \in \mathscr{C} \end{aligned} \tag{14}$$

It is very similar to Eq. (11) when the matrix $\text{Diag}(\mathbf{w}) = \mathbf{M}$ is constrained to be diagonal, $\boldsymbol{\Omega}(\mathbf{M}) = \frac{1}{2}\|\mathbf{M}\|_F^2 = \frac{1}{2}\|\mathbf{w}\|_2^2$,

$u = b - 1$ and $l = b + 1$. The only difference is the inclusion of the $b^2/2$ term in the regularizer. Note that both $\mathbf{w}$ and b are learned in Eq (14).

The problem is convex w.r.t. $\mathbf{w}$ and b, and the inclusion of the $b^2/2$ term in the regularizer does not affect generalization [31]. The optimization process is briefly discussed in Section 4.2 and a detailed discussion is provided in Section A.

4 Quadruplet-wise (Qwise) optimization scheme

We first focus on the case where $\mathbf{M} \in \mathbb{S}_+^d$ is a full (non-diagonal) matrix, then we discuss the case where the learned metric is parameterized by one vector of a set of vectors. Finally, we describe optimization issues that are common to both distance metric formulations.

4.1 Full matrix metric optimization

To solve the optimization problem of Eq. (11), we use the projected gradient method. A subgradient of Eq. (11) w.r.t. $\mathbf{M}$ is computed as follows:

$$\begin{aligned}\nabla =& C_p \left(\sum_{(\mathcal{I}_i,\mathcal{I}_j)\in\mathcal{S}^+} \mathbf{C}_{ij} - \sum_{(\mathcal{I}_i,\mathcal{I}_j)\in\mathcal{D}^+} \mathbf{C}_{ij} \right) \\ &+ C_q \sum_{q\in\mathcal{N}^+} (\mathbf{C}_{ij} - \mathbf{C}_{kl}) + \frac{\partial \boldsymbol{\Omega}(\mathbf{M})}{\partial \mathbf{M}}\end{aligned} \tag{15}$$

where $\mathcal{N}^+$, $\mathcal{S}^+$ and $\mathcal{D}^+$ are the subsets of violated constraints in $\mathcal{N}$, $\mathcal{S}$, $\mathcal{D}$ for a given value of $\mathbf{M}$, respectively, i.e., :

- $q \in \mathcal{N}^+ \iff q = (\mathcal{I}_i,\mathcal{I}_j,\mathcal{I}_k,\mathcal{I}_l) \in \mathcal{N}$ and $\delta_q + \langle \mathbf{M}, \mathbf{C}_{ij} - \mathbf{C}_{kl}\rangle > 0$
- $(\mathcal{I}_i,\mathcal{I}_j) \in \mathcal{S}^+ \iff (\mathcal{I}_i,\mathcal{I}_j) \in \mathcal{S}$ and $D^2_{\mathbf{M}}(\mathcal{I}_i,\mathcal{I}_j) > u$
- $(\mathcal{I}_i,\mathcal{I}_j) \in \mathcal{D}^+ \iff (\mathcal{I}_i,\mathcal{I}_j) \in \mathcal{D}$ and $D^2_{\mathbf{M}}(\mathcal{I}_i,\mathcal{I}_j) < l$

The value of $\frac{\partial \boldsymbol{\Omega}(\mathbf{M})}{\partial \mathbf{M}}$ depends on the choice of regularizer $\boldsymbol{\Omega}(\mathbf{M})$. For instance, $\frac{\partial \boldsymbol{\Omega}(\mathbf{M})}{\partial \mathbf{M}} = \mathbf{I}_d$ if $\boldsymbol{\Omega}(\mathbf{M}) = \text{tr}(\mathbf{M})$, $\frac{\partial \boldsymbol{\Omega}(\mathbf{M})}{\partial \mathbf{M}} = \mathbf{M}$ if $\boldsymbol{\Omega}(\mathbf{M}) = \frac{1}{2}\|\mathbf{M}\|_F^2$, and $\frac{\partial \boldsymbol{\Omega}(\mathbf{M})}{\partial \mathbf{M}} = \sum_{(\mathcal{I}_i,\mathcal{I}_j)\in\mathcal{S}} \mathbf{C}_{ij}$ if $\boldsymbol{\Omega}(\mathbf{M}) = \langle \mathbf{M}, \sum_{(\mathcal{I}_i,\mathcal{I}_j)\in\mathcal{S}} \mathbf{C}_{ij}\rangle$ in the case of MMC [63] and LMNN [62].

The whole algorithm of this subgradient method is presented in Algorithm 1 where η_t is the step size (see [7] for optimal stepsize strategies in subgradient methods). The complexity of Algorithm 1 is linear in the number of constraints and its complexity is dominated by the projection $\Pi_{\mathbb{S}_+^d}$ onto the PSD cone performed at each iteration (step 6). In the full matrix case, it requires an eigendecomposition of the matrix $(\mathbf{M}_t - \eta_t \nabla_t)$, whose complexity is cubic in the dimensionality d. This can be prohibitive if d is large. However, the dimensionality d of our input data is always smaller or equal to 1000 in our experiments. On a single 3,40 GHz computer, the eigendecomposition of a $10^3 \times 10^3$ matrix takes less than 0.1 second, which is tractable for our applications.
As one can see in Eq. (15), the subgradient related to the loss of each quadruplet of images $q = (\mathcal{I}_i,\mathcal{I}_j,\mathcal{I}_k,\mathcal{I}_l) \in \mathcal{N}$ is:

$$\frac{\partial \left[\delta_q + \langle \mathbf{M}, \mathbf{C}_{ij} - \mathbf{C}_{kl}\rangle\right]_+}{\partial \mathbf{M}} = \begin{cases} \mathbf{0} & \text{if } q \notin \mathcal{N}^+ \\ (\mathbf{C}_{ij} - \mathbf{C}_{kl}) & \text{if } q \in \mathcal{N}^+ \end{cases}$$

The value of the subgradient does not depend on the degree to which the constraint associated to the quadruplet $q \in \mathcal{N}$ is violated, but depends only on whether q is in $\mathcal{N}^+$ or not. Then let $h(\mathcal{N}^+)$ be a subgradient associated to the set $\mathcal{N}^+$, i.e., : $h(\mathcal{N}^+) = \sum_{q\in\mathcal{N}^+} (\mathbf{C}_{ij} - \mathbf{C}_{kl})$. Let $\mathcal{N}_t^+$ be the set of violated constraints in $\mathcal{N}$ at iteration t of the subgradient method. We note that:

$$h(\mathcal{N}_{t+1}^+) = h(\mathcal{N}_t^+) - h(\mathcal{N}_t^+ \setminus \mathcal{N}_{t+1}^+) + h(\mathcal{N}_{t+1}^+ \setminus \mathcal{N}_t^+)$$

Since the sets $(\mathcal{N}_t^+ \setminus \mathcal{N}_{t+1}^+)$ and $(\mathcal{N}_{t+1}^+ \setminus \mathcal{N}_t^+)$ are very small in practice, it is more efficient to store the matrix $h(\mathcal{N}_t^+)$ and compute $h(\mathcal{N}_t^+ \setminus \mathcal{N}_{t+1}^+)$ and $h(\mathcal{N}_{t+1}^+ \setminus \mathcal{N}_t^+)$ to obtain $h(\mathcal{N}_{t+1}^+)$ than naively computing $\sum_{q\in\mathcal{N}^+} (\mathbf{C}_{ij} - \mathbf{C}_{kl})$ for which the complexity is $O(|\mathcal{N}_{t+1}^+|d^2)$. Note that the same technique can be used for the sets $\mathcal{S}$ and $\mathcal{D}$ when they are not empty.

4.2 Simplified metric optimization

To solve the vector optimization problem in Eq. (14), we adapt the RankSVM model [27]. The complexity is linear in the number of constraints and large-scale efficient solvers have been proposed (e.g., Newton's method [11]). In order to exploit Newton's method, we use a Huber loss function instead of a hinge loss function like in Eq. (11). The optimization process is detailed in Section A and is a Newton adaptation of Algorithm 1 for vector optimization.

A small adaptation needs to be done to exploit the optimization techniques presented in Section 4.1 since we use Huber loss functions instead of a hinge loss. As the Huber loss function is composed of two linear parts (sets $\boldsymbol{\beta}_{i,y}^0$ and $\boldsymbol{\beta}_{i,y}^L$ in Section A.2) and a quadratic part, the technique presented in Section 4.1 for the hinge loss can be applied to the linear parts of the Huber loss function, which represent nearly all the domain of L_i^h.

4.3 Active sets

As the number of possible quadruplets can be very large, it is computationally prohibitive and sub-optimal to use all the quadruplets.

To overcome this limitation, we propose to add to our optimization schemes an *active set* strategy that exploits the

Algorithm 1 Projected Subgradient Method

```
Require: Sets $\mathcal{N}$, $\mathcal{D}$, $\mathcal{S}$ (some of them can be empty)
Iteration $t = 0$
Initialize $\mathbf{M}_t \in \mathbb{S}^d_+$ (e.g., $\mathbf{M}_t = \mathbf{0}$)
Initialize the step size $\eta_t > 0$ (e.g., $\eta_t = 1$)
repeat
    Compute $\nabla_t$ (subgradient w.r.t. $\mathbf{M}_t$, Eq. (15))
    $\mathbf{M}_{t+1} \leftarrow \Pi_{\mathbb{S}^d_+}(\mathbf{M}_t - \eta_t \nabla_t)$
    $t \leftarrow t + 1$
until $||\mathbf{M}_t - \mathbf{M}_{t-1}||_F^2 \leq \epsilon$
Return $\mathbf{M}_t$
```

fact that the great majority of training quadruplets do not incur margin violations. Only a small fraction of the quadruplets in $\mathcal{N}$ are in $\mathcal{N}^+$. In a similar manner as in LMNN [62], we check all the quadruplets and maintain an active list of those with margin violations: a full re-check is performed every 10-20 iterations, depending on fluctuations of the set $\mathcal{N}_t^+$. For intermediate iterations, we only check for margin violations from among those active quadruplets accumulated over previous iterations. When the optimization converges for a given active set $\mathcal{N}_t^+$, the most active constraints that are not in $\mathcal{N}_t^+$ are added in $\mathcal{N}_{t+1}^+$, note that $\mathcal{N}_t^+ \subset \mathcal{N}_{t+1}^+$. If all the possible active constraints are already in $\mathcal{N}_t^+$, then we have reached an optimal solution for the global optimization problem. Otherwise, some remaining active constraints are added to the current set $\mathcal{N}_t$ until convergence.

4.4 Structural Metric Learning

We present in this subsection an extension of our model based on structured output prediction for large margin methods [28, 30]. The proposed extension is inspired by [44] and learns a metric to predict a ranking over a set of samples. Its formulation allows to exploit efficient optimization techniques such as the 1-slack cutting-plane method [30].

The goal of Metric Learning to Rank (MLR) [44] is to learn a matrix $\mathbf{M} \in \mathbb{S}^d_+$ that minimizes a ranking loss function $\Delta : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ over permutations $\mathcal{Y}$ induced by distance. By considering a set $\mathcal{X}$ of queries x and a corpus $\mathcal{C}$ of points c_i that represent images or pairs of images, the structured output optimization problem can be expressed as:

$$\begin{aligned} &\min_{\mathbf{M}\in\mathbb{S}^d_+, \xi \geq 0} \boldsymbol{\Omega}(\mathbf{M}) + C\xi \\ &\qquad \text{s.t. } \forall x \in \mathcal{X}, y \in \mathcal{Y} \\ &\qquad\quad \langle \mathbf{M}, \psi(x, y_x)) - \psi(x, y) \rangle \geq \Delta(y_x, y) - \xi \end{aligned} \tag{16}$$

where $\boldsymbol{\Omega}(\mathbf{M}) > 0$ is a regularization term and $C > 0$ is a regularization parameter. The loss $\Delta(y_x, y)$ quantifies the penalty for making prediction y if the correct ranking output is y_x. Rankings are represented as a matrix of pairwise orderings $\mathcal{Y} \subset \{-1, 0, +1\}^{|\mathcal{C}| \times |\mathcal{C}|}$ between points c_i in $\mathcal{C}$. For any $y \in \mathcal{Y}$, $y_{ij} = +1$ if c_i is ranked ahead of c_j, $y_{ij} = -1$ if c_j is ranked ahead of c_i, and $y_{ij} = 0$ if c_i and c_j have equal rank.

The 1-slack approach in Eq. (16) shares a single slack variable ξ across all constraint batches, which are in turn aggregated by averaging over each point in the training set.

Let $\mathcal{C}_x^+$ and $\mathcal{C}_x^-$ denote the set of relevant and non-relevant images or pairs of images of $\mathcal{C}$ for the query x, respectively. In this paper, we consider the commonly used *partial order* feature map ψ:

$$\psi(x, y) = \sum_{c_i \in \mathcal{C}_x^+} \sum_{c_j \in \mathcal{C}_x^-} y_{ij} \left(\frac{\phi(x, c_i) - \phi(x, c_j)}{|\mathcal{C}_x^+| . |\mathcal{C}_x^-|} \right) \tag{17}$$

where $\phi(x, c_i)$ is a feature map which characterizes the relation between x and c_i. In the model proposed by [44], they consider that x and c_i are images represented by vectors $\mathbf{x}_x$ and $\mathbf{x}_i$ in the same space $\mathbb{R}^d$, and thus express ϕ as:

$$\phi_{\mathbb{R}^d}(x, c_i) = -(\mathbf{x}_x - \mathbf{x}_i)(\mathbf{x}_x - \mathbf{x}_i)^\top$$

In this case, they have $\langle \mathbf{M}, \phi_{\mathbb{R}^d}(x, c_i) \rangle = -D^2_{\mathbf{M}}(x, c_i)$.

In our case of quadruplets, we consider that c_i is a pair of images and is represented by a pair of vectors $(\mathbf{x}_{i_1}, \mathbf{x}_{i_2}) \in \mathbb{R}^d \times \mathbb{R}^d$. For convenience, we write $\mathbb{R}^d_2 = \mathbb{R}^d \times \mathbb{R}^d$. We then express ϕ as:

$$\phi_{\mathbb{R}^d_2}(x, c_i) = \phi_{\mathbb{R}^d_2}(x, i_1, i_2) = -(\mathbf{x}_{i_1} - \mathbf{x}_{i_2})(\mathbf{x}_{i_1} - \mathbf{x}_{i_2})^\top$$

In the *pairwise approach*, if the sets $\mathcal{S}$ and $\mathcal{D}$ of similar and dissimilar pairs are the only provided training labels, we consider that there exists only one query x. The sets are $\mathcal{C}_x^+ = \mathcal{S}$, $\mathcal{C}_x^- = \mathcal{D}$ and $\phi = \phi_{\mathbb{R}^d_2}$.

In the *tripletwise approach*, if training samples are provided as triplets of images (x, c_k, c_l), as in LMNN [62], where $(x, c_k) \in \mathcal{S}$ are similar and $(x, c_l) \in \mathcal{D}$ are dissimilar, we consider that each such image x is a query. The sets are then $\mathcal{C}_x^+ = \{c_k \mid (x, c_k) \in \mathcal{S}\}$, $\mathcal{C}_x^- = \{c_l \mid (x, c_l) \in \mathcal{D}\}$ and $\phi = \phi_{\mathbb{R}^d}$. This is the case considered in [44]. It generalizes LMNN when $\mathcal{C}_x^+$ is the set of k nearest neighbors of x in the original input space, and $\mathcal{C}_x^-$ is the set of images in categories different from x. Since LMNN extends linear ordinal regression SVM [29] by learning a PSD matrix instead of a vector (and using a different regularization term), MLR extends LMNN in the same way as structural SVM extends ordinal regression [29].

Quadruplet formulation: In our case where training labels are relative distances over quadruplets $(c_i, c_j, c_k, c_l) \in \mathcal{N}$, we consider that each query is a pair $x = (c_i, c_j)$, their corresponding positive and negative sets are $\mathcal{C}_x^+ = \{x\}$, $\mathcal{C}_x^- = \{(c_k, c_l) | (c_i, c_j, c_k, c_l)\} \in \mathcal{N}, x = (c_i, c_j)\}$, $\phi = \phi_{\mathbb{R}^d_2}$.

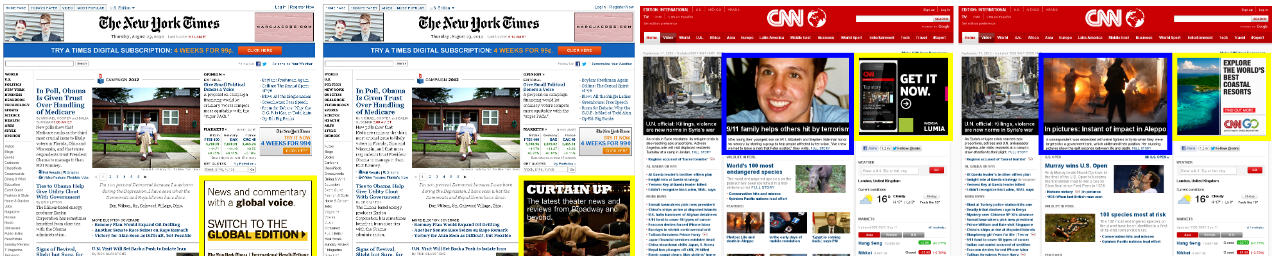

The New York Times
TRY A TIMES DIGITAL SUBSCRIPTION: 4 WEEKS FOR 99¢.
In Poll, Obama Is Given Trust Over Handling of Medicare
News and commentary with a global voice.
SWITCH TO THE GLOBAL EDITION ▸
CURTAIN UP
The latest theater news and reviews from Broadway and beyond.
CNN
U.N. official: Killings, violence are new norms in Syria's war
9/11 family helps others hit by terrorism
GET IT NOW.
LUMIA
In pictures: Instant of impact in Aleppo
EXPLORE THE WORLD'S BEST COASTAL RESORTS
CNN GO

Fig. 3 (left) A pair of successive versions of the New York Times homepage wherein only the advertisement (yellow region) is different. The change of advertisement does not affect the information shared by the page, the two versions are thus considered as similar. (right) A pair of successive versions of the CNN homepage. The change of news title (blue region), which is the main information shared by the page, makes the two versions dissimilar and is thus considered as an important change.

Optimization: In order to optimize Eq. (16), we use the same 1-slack cutting plane solver [30] as [44]. The difference with [44] is the formulation of the feature map induced by ϕ and the fact that we try to satisfy relative orderings of distances between image pairs. The structural formulation of our problem can greatly reduce the number of possible constraints compared to the non-structural formulation (in the same way as [29]). The 1-slack cutting plane method efficiently selects the most penalized constraints among a huge set of possible constraints, and optimizes the problem over a small set of active constraints. We can then solve problems that deal with huge numbers of quadruplets as training information.

5 Temporal Metric Learning for Webpages

We present in this section the first application of our metric learning method. We introduced in [40] a distance metric learning framework for webpage comparison and detection of important semantic regions. The goal is to determine whether semantical changes occurred between two successive versions of the same webpage or not. Fig. 3 illustrates two pairs of successive versions of webpages. On the left one, the change of advertisement (yellow region) is the only observable change. Since it does not change the content shared by the webpage, the two versions are considered as similar. A human (or indexing robot) then does not need to visit these two versions. On the contrary, on the right pair of Fig. 3, although an advertisement (yellow region) has also changed, the main news shared by the webpage (blue region) is different. The versions then both need to be visited and indexed. They are thus are considered as dissimilar. Several approaches that extract meaningful information in webpages admit the importance of visual information [42, 55, 56] since the layout is taken into account when pages are created. In order to exploit visual information, classic webpage analysis methods, such as the VIsion-based Page Segmentation algorithm (VIPS) [9], integrate visual descriptors based on the structure (e.g., position, width, border of regions or font colors) from the page source code rather than using computer vision-based features. We extend the webpage analysis experiments performed in [40] in several ways: 1) we propose a semi-supervised metric learning framework by combining unsupervised quadruplet constraints with pairwise constraints that are manually labeled; 2) we propose a novel heuristic to perform unsupervised change detection; 3) we combine both visual and structural [9] information.

5.1 Webpage change detection framework

5.1.1 Unsupervised constraints

Our approach relies on the assumption of monotony of changes, which is illustrated in Fig. 4 where four successive versions of the same webpage v_{t-1}, v_t, v_{t+1}, v_{t+2} are crawled with a sufficiently high frequency (each hour). Although the four versions are all different, one can see that v_t seems more similar to v_{t+1} than to v_{t+2}. Similarly, v_t and v_{t+1} are more similar than v_{t-1} and v_{t+2} are. By exploiting time information, one can automatically generate a set $\mathcal{B}$ of quadruplets of versions (v_t, v_{t+1}, v_r, v_s) where $r \leq t < s$. The goal is to learn a dissimilarity function D that satisfies most of the following constraints:

$$\forall (v_t, v_{t+1}, v_r, v_s) \in \mathcal{B} : D(v_t, v_{t+1}) \leq D(v_r, v_s) \qquad (18)$$

In order to satisfy these constraints, the metric D has to ignore random and periodic changes, which are often caused by advertisements. Fig. 4 illustrates a case where a car advertisement (at the right of the page) is identical in v_{t-1}, v_t and v_{t+2} and different in v_{t+1}. By ignoring this advertisement region, it is easier for D to satisfy the constraints in Eq. (18).

A trivial solution to satisfy all the constraints in Eq. (18) is a pseudometric such that: $\forall (v_i, v_j), D(v_i, v_j) = 0$. To avoid this degenerate solution, one can assume that there exists a change period $\gamma > 1$ such that for all $r \leq t < r + \gamma$ we have the strict inequality $D(v_t, v_{t+1}) < D(v_r, v_{r+\gamma})$. In other words, we assume that there exists a change period γ specific to the page such that the changes that occurred between the two versions v_r and $v_{r+\gamma}$ are more important

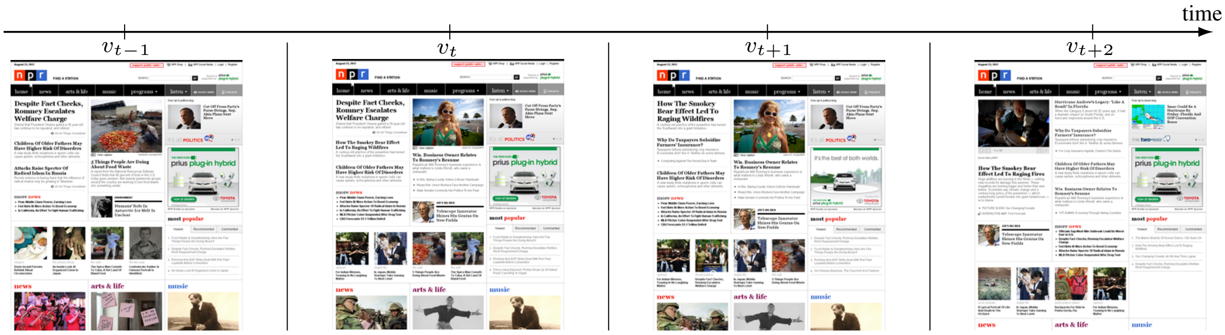


Fig. 4 Four successive versions of the NPR homepage. Although it is hard and expensive to ask users to label version pairs as similar or not, it is cheaper to infer that the dissimilarity between v_t and v_{t+1}, or even v_{t-1} and v_{t+1} is smaller than the dissimilarity between v_{t-1} and v_{t+2}.

than between directly successive versions v_t and v_{+1} where $r \leq t < r + \gamma$. Although v_t and v_{t+1} may be dissimilar, their dissimilarity is assumed smaller than the dissimilarity between v_r and $v_{r+\gamma}$[3]. In the same way as $\mathcal{B}$, we create a set $\mathcal{A}$ (with $\mathcal{A} \cap \mathcal{B} = \emptyset$) such that:

$$\forall (v_t, v_{t+1}, v_r, v_s) \in \mathcal{A} : D(v_t, v_{t+1}) + 1 \leq D(v_r, v_s) \quad (19)$$

where 1 is a safety margin, $r \leq t$ and $s \geq r + \gamma \geq t + 1$. The constraints defined in Eq. (19) penalize content that does not change much in some regions, although a change in the whole page is expected. This type of static content usually corresponds to menus: the algorithm learns to ignore these areas. Note that γ determines whether a quadruplet belongs to $\mathcal{B}$ or $\mathcal{A}$, and thus its related constraint (Eq. (18) or (19)). Nonetheless, since constraints satisfied in Eq. (19) are also satisfied in Eq. (18), choosing a value of γ greater than the actual change period of the page is not problematic. There is a straight connection between these equations and Eq. (7). Any quadruplet q in $\mathcal{B}$ can be formulated as $q \in \mathcal{N}$ with $\delta_q = 0$ and any quadruplet q in $\mathcal{A}$ can be formulated as $q \in \mathcal{N}$ with $\delta_q = 1$.

5.1.2 Pairwise supervised constraints

Additionally to the automatically generated constraints based on monotony of changes, richer information of whether a pair of versions is similar or dissimilar can be integrated. This information can be provided by human users (or heuristically determined). Let $\mathcal{S}$ be the set of pairs of versions annotated as (or assumed) similar and $\mathcal{D}$ the set of dissimilar version pairs, an interesting property of the function D would be to satisfy:

$$\forall (v_r, v_s) \in \mathcal{S} : D(v_r, v_s) + 1 \leq b \quad (20)$$

$$\forall (v_r, v_s) \in \mathcal{D} : b + 1 \leq D(v_r, v_s) \quad (21)$$

where 1 is a safety margin and $b \in \mathbb{R}$ a learned threshold. These two types of constraints (Eq. (20) and Eq. (21)) follow the classic approach in metric learning [26,63] that minimizes the distance of similar pairs while separating dissimilar pairs (in our case, keeping their distances beyond the threshold b). To know whether a test pair (v_r, v_s) is similar or not, one only has to study the sign of $D(v_r, v_s) - b$ (positive for dissimilar pairs, and negative for similar pairs).

5.1.3 Distance Metric formulation

We integrate the constraints mentioned from Eq. (18) to (21) in the learning framework described in Section 3 (see Eq.(11) and Eq. (14)) by considering $\mathcal{N} = \mathcal{A} \cup \mathcal{B}$. We consider the diagonal and full matrix Mahalanobis-like distance metric formulations where the:

- metric $\mathscr{D}_{\mathbf{w}}$ is parameterized by the d-dimensional vector $\mathbf{w} \in \mathbb{R}^d_+$. This metric tries to satisfy the ideal properties of the target function D (Eq. (18) to (21)). $\mathscr{D}_{\mathbf{w}}$ is a linear combination of d distances between versions v_i and v_j over d different spatial regions (one distance per region). These d distances are concatenated in the vector $d_{regions}(v_i, v_j) \in \mathbb{R}^d$. The computation of $d_{regions}$ is detailed is Section 5.2. $\mathscr{D}_{\mathbf{w}}$ is written:

$$\mathscr{D}_{\mathbf{w}}(v_i, v_j) = \mathbf{w}^\top d_{regions}(v_i, v_j) \quad (22)$$

where $\mathbf{w} \in \mathbb{R}^d_+$ is the weight vector: the value of the k-th element of $\mathbf{w}$ corresponds to the importance of change assigned to the k-th region of the page. An element of $\mathbf{w}$ close to 0 means that the corresponding region is ignored, whereas an element with a relatively high absolute value has more impact on the global dissimilarity function $\mathscr{D}_{\mathbf{w}}$. By avoiding $\mathbf{w}$ to have negative elements, the learned metric tends

[3] Different ways to set the parameter γ exist. It can for example be determined with prior knowledge about the page or it can be chosen heuristically following the observation in Adar et al. [1]: human users tend to visit more frequently webpages that often change. In other words, human users can be considered as intelligent web crawlers with a good crawling strategy. For instance, a page that is visited everyday by a lot of unique visitors can be assumed to be different everyday (in this case $\gamma = 24$ hours). This popularity information can be obtained from services that provide detailed statistics about the visits to a website (e.g., Google Analytics).

to ignore unimportant changes rather than penalizing them (which would mean negative scores in order to minimize the learned function).

• metric $D_{\mathbf{M}}$ is paramaterized by the symmetric PSD matrix $\mathbf{M} \in \mathbb{S}^d_+$. $D_{\mathbf{M}}$ is written:

$$D^2_{\mathbf{M}}(v_i, v_j) = \mathbf{c}_{ij}^\top \mathbf{M} \mathbf{c}_{ij} \tag{23}$$

where $\mathbf{c}_{ij} = (d_{regions}(v_i, v_j))^{\circ \frac{1}{2}}$ and $\circ\frac{1}{2}$ is the Hadamard square root (element-wise square root).

5.2 Visual and Structural Comparisons of Webpages

Visual distance representation: We present here how to compute a visual distance representation $d_{regions}$ (mentioned in Section 5.1.3) that relies on computer vision-based features. The method considers screen captures of page versions as images. Only the visible part of pages without scrolling is considered since it generally contains the main information shared by the page [41,55]. Our proposed method computes the GIST [47] descriptors of screen captures. GIST descriptor segments images by an m by m grid[4]. We formulate the vector $d_{regions}(v_i, v_j) \in \mathbb{R}^{m^2}$ as an m^2-dimensional vector for which each element corresponds to the squared ℓ_2-distance between bins that fall into the same cell of the grids of the screenshots of v_i and v_j. GIST descriptor was proven to provide very high accuracy for near-duplicate detection [16], which is close to our context of successive versions of the same document. The high efficiency, small memory usage and estimation of coarsely localized information of the global GIST descriptor, allowing to scale up to very large datasets [16], motivated this choice. Examples of our regular $m \times m$ segmentation are illustrated in Fig. 5.

Learning a multimodal visual/structural metric: We also propose to learn a multimodal distance metric $\mathscr{D}_{\mathcal{M}}$ by late fusion. It is expressed as a linear combination of visual and structural distance metrics:

$$\mathscr{D}_{\mathcal{M}}(v_i, v_j) = \alpha_1 \mathscr{D}_{\mathbf{w}}(v_i, v_j) + \alpha_2 \mathscr{D}_{\mathcal{H}}(v_i, v_j) + \alpha_3 \mathscr{D}_{\mathcal{U}}(v_i, v_j)$$

where the coefficients $\alpha_i \geq 0$ are learned with a binary SVM classifier that separates the pairs in $\mathcal{S}$ from pairs in $\mathcal{D}$. $\mathscr{D}_{\mathbf{w}}(v_i, v_j)$ is the learned visual distance metric mentioned in Eq. (22). $\mathscr{D}_{\mathcal{H}}$ (or $\mathscr{D}_{\mathcal{U}}$) is the Jaccard distance between hyperlinks (or image URLs) of v_i and v_j. $\mathscr{D}_{\mathcal{H}}$ and $\mathscr{D}_{\mathcal{U}}$ were shown to be discriminative for semantic change detection [41][5]

Computation time: the whole process of computation of distances between GIST descriptors, creation of constraints and learning of the diagonal matrix distance $\mathscr{D}_{\mathbf{w}}$ takes 0.7 seconds on a 3.4 GHz machine in MATLAB. It takes 4.5 seconds in the full matrix distance case. It can be done offline: only the learned parameter of the distance (**w** or **M**), the threshold b, the coefficients α_i in the late fusion setup, and the descriptors of test pairs are necessary for test.

5.3 Datasets and evaluation protocol

We hourly crawled different types of popular webpages (homepages or non-homepages of news or educational websites) as done in [1,5] for approximately 50 days: the version v_{t+1} is visited 1 hour after v_t. The crawled webpages[6] are the homepages of some news websites (e.g., CNN, BBC, National Public Radio (NPR), New York Times (NYT)), the finance section of Yahoo! News, the music section of NPR (that is not often updated) and educational webpages: the homepage of Boston's University and the open courseware page of the Massachusetts Institute of Technology (MIT).

To evaluate our approach with quantitative results, we labeled pairs of versions of some of these websites ($\sim 1,200$ per site). To simplify the labeling process, we select only news websites that are easier to annotate, and we choose as similarity criterion the presence of change of the main news in the page. Only the successive version pairs (v_t, v_{t+1}) of the CNN, BBC, NPR and New York Times homepages were labeled. We distinguish 4 labels for version pairs:

- *identical:* two given versions are identical.
- *similar:* a change not important enough to download both version occurs (e.g., a change of advertisement, see Fig. 3 (left)).
- *dissimilar:* the main news of the page changes. Particularly, we consider (v_t, v_{t+1}) as dissimilar only if textual news information is added in the page between v_t and v_{t+1}. We give more details about the annotation criterion in Section 5.4.
- *ambiguous:* two given versions are difficult to label as similar or dissimilar.

For each website, we create 10 train/test splits: for each split, we use 5 successive days for training, the 45 remaining days for test[7]. Ambiguous version pairs are ignored in the test evaluation process. However, they are used in automatically generated quadruplet-wise constraints to train important change maps. The identical versions are also ignored

[4] We use the publicly available code of Oliva and Torralba [47] in MATLAB to compute GIST descriptors. In particular, we choose the following setting: 8 oriented edge responses at 4 different scales. The computation time of the GIST descriptor of a page version (screen capture of about 1000 × 1000 pixels) using a 10 × 10 grid is 3.2 seconds.

[5] We also tried to include the Jaccard distance of words (similar to Dice's coefficient of words used in [2], with the exception that it satisfies the properties of a distance metric) but it does not improve performances.

[6] www.cnn.com, www.bbc.co.uk, www.npr.org, www.nytimes.com, finance.yahoo.com, www.npr.org/music, www.bu.edu, ocw.mit.edu

[7] We minimize the number of common versions used for training among the different splits: i.e., the first training split contains the first 5 days, the second one the 6th to 10th days, the third one the 11th to 15th days...

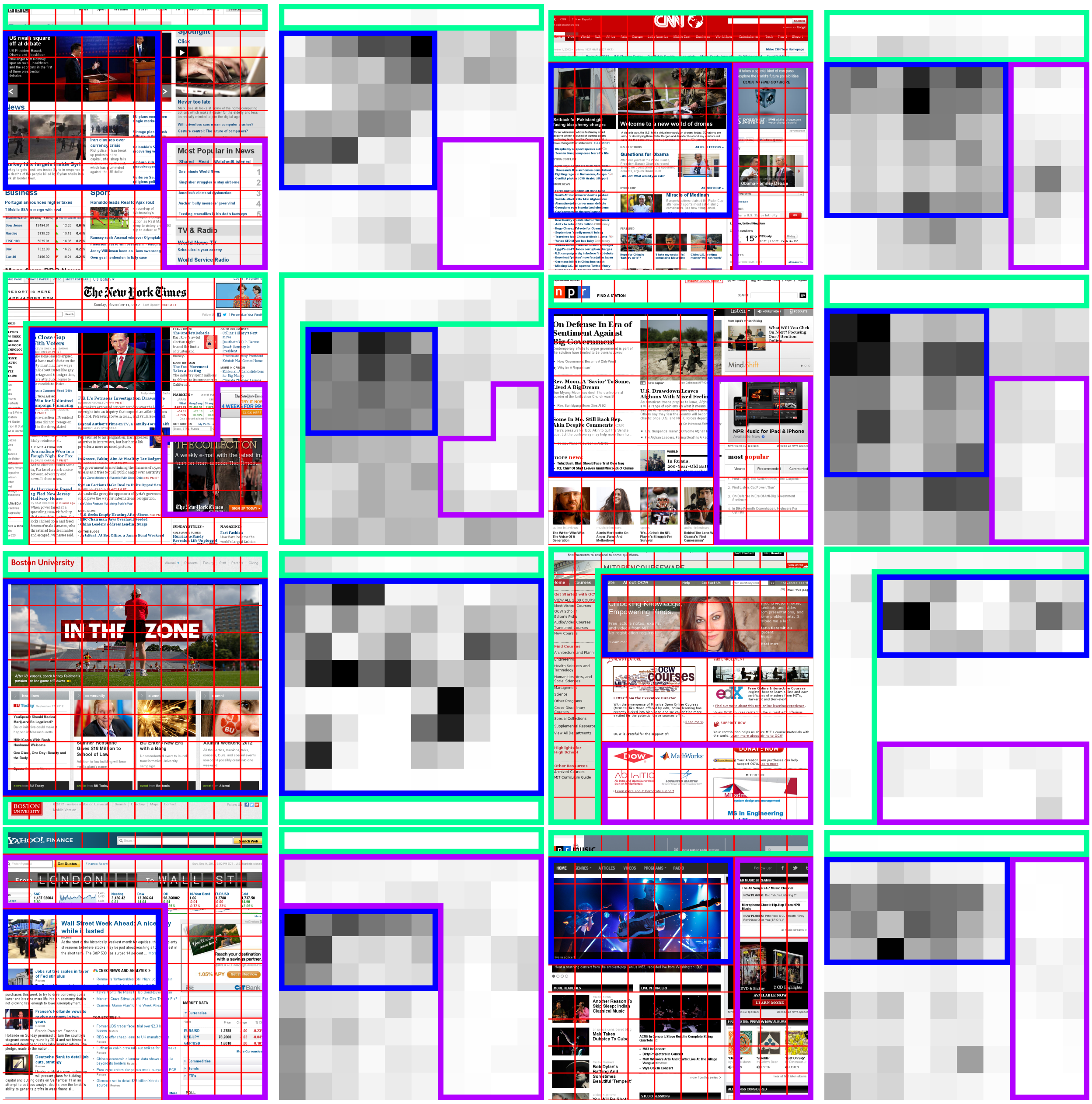


Fig. 5 Important change maps for the homepages of BBC, CNN, NYTimes, NPR, Boston's University, the open courseware page of the MIT, the finance section of Yahoo! News and the music section of NPR. (left) Webpage screenshot, with relevant area (news) in blue, unimportant parts (menu and advertisement) in green and purple, respectively. (right) Spatial weights of important change learned by our method with versions crawled during 5 days and without human annotations (higher values are darker).

for test because their distance would be 0 (the lowest possible value) with any distance metric; since they are easy examples (e.g., they would be the first retrieved similar pairs in the average precision evaluation), the performance measures would return very high scores by using them for test.

We compute the average precision for the similar class $\mathrm{AP}_{\mathcal{S}}$ by ranking distance values of test pairs of successive versions (v_t, v_{t+1}) in ascending order and the average precision for the dissimilar class $\mathrm{AP}_{\mathcal{D}}$ by ranking distance values of test pairs in descending order. The Mean Average Precision (MAP) is the mean of $\mathrm{AP}_{\mathcal{S}}$ and $\mathrm{AP}_{\mathcal{D}}$. Classification accuracy is also used: it is the mean of the accuracies of the class of similar pairs ($\mathcal{S}$) and of the class of dissimilar pairs ($\mathcal{D}$). Average precision is particularly useful to measure how much the relative orderings of distances are satisfied by the metric. Classification accuracy is useful to determine the most effective crawling strategies since it can measure how frequently a webpage changes in a given period.

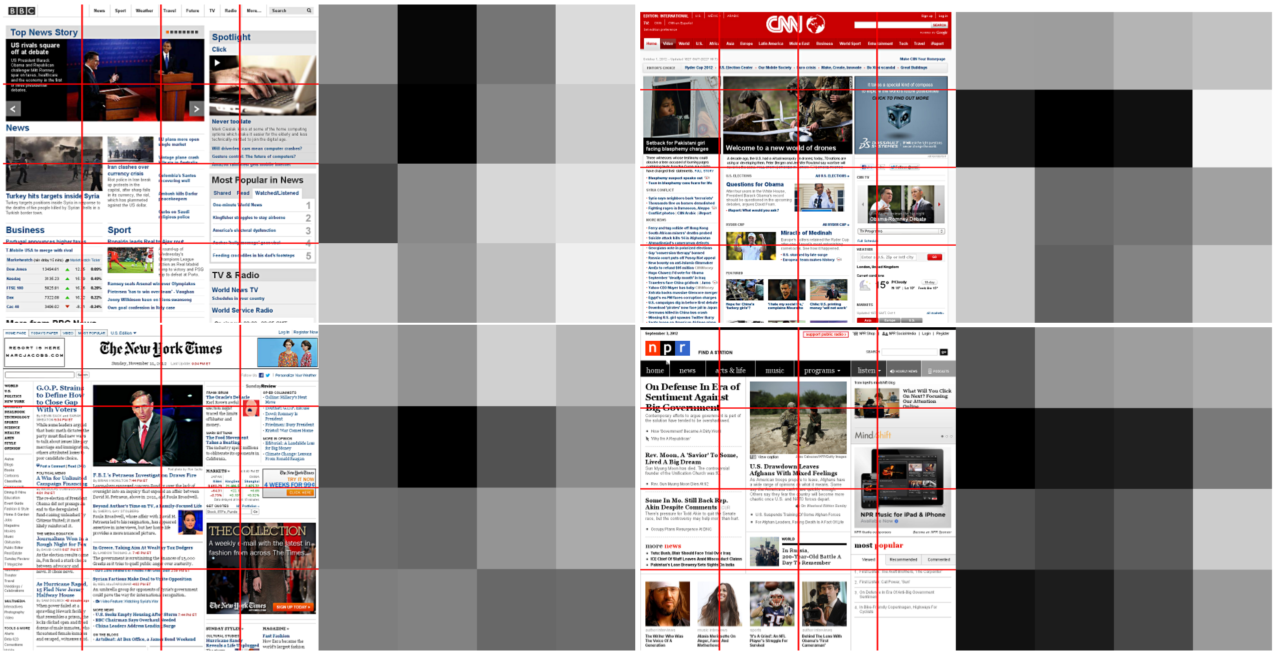


Fig. 6 Important change maps for the homepages of BBC, CNN, New York Times and NPR. (left) Webpage screenshot with webpage regular segmentation blocks (red lines). (right) Absolute values of the eigenvector of the dominant eigenvalue of the distance non-diagonal matrix learned by our method with versions crawled during 5 days and without human supervision (higher values are darker).

5.4 Qualitative results

We present in this subsection qualitative results when no human supervision is integrated in the learning process (i.e., we only consider the automatically generated constraints $\mathcal{A}$ and $\mathcal{B}$, the sets $\mathcal{D}$ and $\mathcal{S}$ are both empty).

A first qualitative evaluation is illustrated in Fig. 5. The figure shows the weights of regions learned for the 8 webpages mentioned in Section 5.3 without human supervision. In order to learn these weights/maps of importance, we sample version quadruplets (v_t, v_{t+1}, v_r, v_s) using Eq. (18) and Eq. (19) so that $r \geq t - 6$, $s \leq t + 7$, $\gamma = 4$. Images are segmented as a 10×10 or 8×8 grid. Training sets to learn these maps contain screenshots of pages visited every hour during 5 days. In terms of training constraints, we deal with less than $10,000$ constraints in our experiments, which makes the learning of the diagonal matrix metric $\mathcal{D}_{\mathbf{w}}$ very fast. The maps of importance in Fig. 5 plot the relative values of the parameter $\mathbf{w} \in \mathbb{R}_+^d$ of the learned metric. The highest positive values, represented by dark regions, correspond to important change regions of the page (e.g., news title). Menus and advertisements are ignored by the learned metric as expected.

We also tested our method on governmental websites but their change frequency is so low (the page often remains unchanged in 5 days) that a meaningful distance metric is not learnable in only 5 days. This is consistent with the observations of Adar et al. [2]: government domain addresses do not change as frequently or as much as pages in other domains do, and this may reflect the fact this type of site provides richer and less transient content that only requires small, infrequent updates.

Fig. 6 illustrates the eigenvector $\mathbf{v}_1$ of the largest eigenvalue λ_1 of $\mathbf{M}$ when we learn a full matrix metric $D_{\mathbf{M}}$. The matrix $\mathbf{M}' = \lambda_1 \mathbf{v}_1 \mathbf{v}_1^\top$ is the projection of $\mathbf{M}$ onto the set of rank-1 symmetric PSD matrices, and thus the nearest rank-1 matrix of $\mathbf{M}$ in the spectral norm. The vector $\mathbf{v}_1$ weighs the importance of spatial regions of the webpage since we have $D^2_{\mathbf{M}'}(v_i, v_j) = \lambda_1 (\mathbf{v}_1^\top \mathbf{c}_{ij})^2$. Fig. 6 shows that $\mathbf{v}_1$ correctly detects important change regions and ignores menus and advertisements.

5.5 Quantitative Results:

We present in this subsection quantitative results obtained by our method. We first present average precision scores obtained in the unsupervised set (i.e., $\mathcal{D} \cup \mathcal{S} = \emptyset$). Eventually, we present classification accuracy scores in the unsupervised and semi-supervised setups.

Average Precision: Table 1 compares the average precision scores obtained using different distance metrics:

- the Euclidean distance metric often used for the GIST descriptor [47].

- a triplet-based method for which the set $\mathcal{A}$ is used to generate triplet-wise constraints.

- our learned visual metric $\mathcal{D}_{\mathbf{w}}$ parameterized by a vector $\mathbf{w}$.

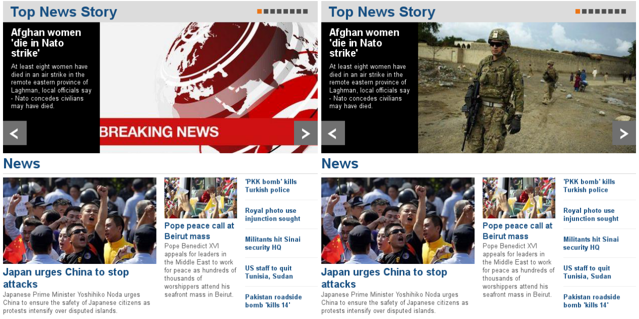
- our proposed visual metric $D_\mathbf{M}$ parameterized by the non-diagonal matrix $\mathbf{M}$ and learned using classic projected sub-gradient method (described in Algorithm 1).
- our proposed visual metric $D_\mathbf{M}^\textbf{struct}$ parameterized by the non-diagonal matrix $\mathbf{M}$ and learned using 1-slack cutting plane method (described in Section 4.4).

More precisely, Table 1 presents the recognition scores when screenshot images of webpages are segmented[8] as m^2 regions (i.e., $d_{regions}(v_i, v_j) \in \mathbb{R}^{m^2}$) where $m = 10$. The Euclidean distance metric is outperformed by all the learned metrics although its performance is good, which means that the Euclidean distance is fitted for change detection. The triplet-based method which exploits a small number of constraints is outperformed by quadruplet-wise methods that exploit a larger number of meaningful constraints. The full matrix distance metric $D_\mathbf{M}$ outperforms all the other methods. Particularly, it outperforms the diagonal matrix distance metric $\mathscr{D}_\mathbf{w}$ proposed in [40] due to the exploitation of correlations between the different spatial regions. The distance metric $D_\mathbf{M}^\textbf{struct}$ learned with structural metric learning returns slightly worse results than $D_\mathbf{M}$. This is due to the fact that the cutting plane method solves an approximation of the original problem (by exploiting a subset of active constraints). In general, $D_\mathbf{M}^\textbf{struct}$ also outperforms the other metrics.

The relatively low AP$_\mathcal{D}$ for the BBC homepage is due to false detections of semantical changes because of the similarity criterion used to label version pairs. As mentioned in Section 5.3, two versions are considered as dissimilar only if their textual news content is different. However, a specificity of the BBC website is that each *breaking news* story goes along with a *breaking news* logo that appears only within the hour after its publication. After this given period, the BBC *breaking news* logo is replaced by a related news picture. Fig. 7 illustrates one such example where the text content is identical (and the pair of versions is thus annotated as similar) but the *breaking news* logo is replaced by a news picture. Our method, which detects for Fig. 7 a visual change in an important region, returns a high dissimilarity value although the version pair is annotated as similar. The AP$_\mathcal{D}$ obtained by our method is then affected by this kind of noisy behavior of the BBC website. The other news websites studied in this paper do not have such a behavior and return better values of AP$_\mathcal{D}$. In a context where any new image about an important event has to be archived, the example illustrated in Fig. 7 would be considered as dissimilar, and the AP$_\mathcal{D}$ of BBC would be higher.

	National Public Radio (NPR)		
Method	AP$_\mathcal{S}$	AP$_\mathcal{D}$	MAP
Eucl. Distance	96.3 ± 0.2%	89.5 ± 0.5%	92.9 ± 0.3%
Triplet-based	98.0 ± 0.6%	92.5 ± 1.1%	95.2 ± 0.9%
Proposed $\mathscr{D}_\mathbf{w}$	98.6 ± 0.2%	94.3 ± 0.6%	96.5 ± 0.4%
Proposed $D_\mathbf{M}$	**98.7 ± 0.2%**	**94.5 ± 0.7%**	**96.6 ± 0.4%**
Proposed $D_\mathbf{M}^\textbf{struct}$	98.3 ± 0.3%	94.0 ± 0.6%	96.1 ± 0.5%
	New York Times		
Method	AP$_\mathcal{S}$	AP$_\mathcal{D}$	MAP
Eucl. Distance	69.8 ± 0.9%	79.5 ± 0.4%	74.6 ± 0.5%
Triplet-based	83.2 ± 1.4%	89.1 ± 2.7%	86.1 ± 2.0%
Proposed $\mathscr{D}_\mathbf{w}$	85.5 ± 5.4%	92.3 ± 4.1%	88.9 ± 4.6%
Proposed $D_\mathbf{M}$	**91.6 ± 4.4%**	**94.7 ± 2.4%**	**93.1 ± 3.4%**
Proposed $D_\mathbf{M}^\textbf{struct}$	90.5 ± 4.7%	94.0 ± 2.5%	92.2 ± 3.6%
	CNN		
Method	AP$_\mathcal{S}$	AP$_\mathcal{D}$	MAP
Eucl. Distance	68.1 ± 0.6%	85.9 ± 0.6%	77.0 ± 0.5%
Triplet-based	78.8 ± 1.9%	91.7 ± 1.7%	85.2 ± 1.8%
Proposed $\mathscr{D}_\mathbf{w}$	82.7 ± 4.1%	94.6 ± 1.8%	88.6 ± 2.9%
Proposed $D_\mathbf{M}$	**87.9 ± 3.1%**	**96.6 ± 0.6%**	**92.2 ± 1.9%**
Proposed $D_\mathbf{M}^\textbf{struct}$	87.4 ± 3.2%	96.3 ± 0.6%	91.9 ± 1.9%
	BBC		
Method	AP$_\mathcal{S}$	AP$_\mathcal{D}$	MAP
Eucl. Distance	91.1 ± 0.3%	76.7 ± 0.6%	83.9 ± 0.4%
Triplet-based	92.5 ± 0.4%	80.1 ± 1.0%	86.3 ± 0.6%
Proposed $\mathscr{D}_\mathbf{w}$	92.8 ± 0.4%	79.3 ± 1.3%	86.1 ± 0.8%
Proposed $D_\mathbf{M}$	**93.0 ± 0.6%**	**82.5 ± 1.3%**	**87.7 ± 1.0%**
Proposed $D_\mathbf{M}^\textbf{struct}$	92.8 ± 0.6%	81.8 ± 1.4%	87.3 ± 1.0%

Table 1 Test average precisions obtained by the classic Euclidean distance and by learned metrics in the fully unsupervised setup.



Top News Story

Afghan women 'die in Nato strike'

At least eight women have died in an air strike in the remote eastern province of Laghman, local officials say - Nato concedes civilians may have died.

BREAKING NEWS

News

Japan urges China to stop attacks

Japanese Prime Minister Yoshihiko Noda urges China to ensure the safety of Japanese citizens as protests intensify over disputed islands.

Pope peace call at Beirut mass

Pope Benedict XVI appeals for leaders in the Middle East to work for peace as hundreds of thousands of worshippers attend his seafront mass in Beirut.

'PKK bomb' kills Turkish police

Royal photo use injunction sought

Militants hit Sinai security HQ

US staff to quit Tunisia, Sudan

Pakistan roadside bomb 'kills 14'

Top News Story

Afghan women 'die in Nato strike'

At least eight women have died in an air strike in the remote eastern province of Laghman, local officials say - Nato concedes civilians may have died.

News

Japan urges China to stop attacks

Japanese Prime Minister Yoshihiko Noda urges China to ensure the safety of Japanese citizens as protests intensify over disputed islands.

Pope peace call at Beirut mass

Pope Benedict XVI appeals for leaders in the Middle East to work for peace as hundreds of thousands of worshippers attend his seafront mass in Beirut.

'PKK bomb' kills Turkish police

Royal photo use injunction sought

Militants hit Sinai security HQ

US staff to quit Tunisia, Sudan

Pakistan roadside bomb 'kills 14'

Fig. 7 The important region of two successive versions of the BBC homepage. A specificity of the BBC website is that it always uses its ”breaking news” logo to introduce recent breaking news and removes it after a short period. In this case, since the textual content of the main news is unchanged, we consider the two versions are similar. However, in a Web archiving context, these two versions are considered as dissimilar since a relevant visual information is updated. Our algorithm tends to detect a visual change in the important change region although the news is the same.

Classification Accuracy: We now present classification accuracy results in the unsupervised and semi-supervised setups. For the sake of clarity of the paper and scalability of the method, we present in the following only the results obtained with the diagonal Qwise visual distance metric $\mathscr{D}_\mathbf{w}$ and with GIST descriptor. The relative quantitative performances of other models follow the same tendencies as in Table 1.

When human annotations to distinguish similar pairs from dissimilar pairs are not provided (i.e., $\mathcal{S} \cup \mathcal{D} = \emptyset$), a dis-

[8] We experimented with different values of m (i.e., $m = 4, 8$ and 10), and this setting returned the best recognition performance for all the distance metrics. All the distance metrics benefit from greater values of m, which means that they need to focus on highly detailed small regions of pages.

tance $\mathscr{D}_\mathbf{w}$ can be learned from the training set $\mathcal{N} = \mathcal{A} \cup \mathcal{B}$ composed of quadruplets of successive versions of the same webpage (crawled for 5 days in our experiments). However, no threshold (b in Eq. (20) and Eq. (21)) is learned to distinguish similar pairs from dissimilar pairs. In other words, pair distances can be compared to one another but our learned metric cannot determine whether or not important changes occurred in a given pair of versions. We present here how to learn a change detection algorithm (that can distinguish similar pairs of versions from dissimilar pairs) without exploiting information provided by human users. In particular, we propose to learn a change detection algorithm that exploits the metric $\mathscr{D}_\mathbf{w}$ learned from the set $\mathcal{N}$ to automatically generate the training sets $\mathcal{S}$ (class -1) and $\mathcal{D}$ (class $+1$). Once these sets are created, we learn a binary classifier that discriminates pairs in $\mathcal{S}$ from pairs in $\mathcal{D}$. Assuming that the metric $\mathscr{D}_\mathbf{w}$ learned in Eq. (14) provides lowest distance values for similar pairs and highest values for dissimilar pairs, the training pairs in $\mathcal{S}$ and $\mathcal{D}$ can be automatically inferred from the training set of page versions in $\mathcal{N}$. Let k be the cardinality of the created sets $\mathcal{S}$ and $\mathcal{D}$ ($k = |\mathcal{S}| = |\mathcal{D}|$). The k version pairs (v_t, v_{t+1}) (among the $24 \times 5 = 120$ possible pairs) with highest values of $\mathscr{D}_\mathbf{w}(v_t, v_{t+1})$ form $\mathcal{D}$, whereas the k version pairs with values $\mathscr{D}_\mathbf{w}(v_t, v_{t+1})$ closest to 0 (and that are not completely identical) form $\mathcal{S}$. Any binary classifier that exploits the generated training samples in $\mathcal{S}$ and $\mathcal{D}$ can be learned. We learn a linear SVM classifier that discriminates pairs in $\mathcal{D}$ from pairs in $\mathcal{S}$.

Fig. 8 and Table 2 report classification accuracies in the unsupervised setup described above. We learn a linear SVM with the automatically created sets $\mathcal{S}$ and $\mathcal{D}$ using the $|\mathcal{S}| = |\mathcal{D}| = k = 25$ version pairs with lowest and highest distances, respectively.

Fig. 8 illustrates the change detection accuracy as a function of the grid resolution used to segment webpage screenshots (i.e., the number of regions in webpages). Change detection improves as the grid resolution increases. At a grid resolution of 4×4, the change detection is already better for all websites than a naive classifier that randomly determines whether a test pair is similar and would reach 50% accuracy. We reach accuracies up to 87% on NPR with a 10×10 grid resolution. Table 2 compares accuracies (using a 10×10 grid resolution) depending on whether visual features are used independently (as in Fig. 8) or combined with structural distances. The combination of structural and visual distances improves the accuracy up to 2% on CNN.

All these results illustrate the ability of our model to learn a change detection algorithm without human supervision.

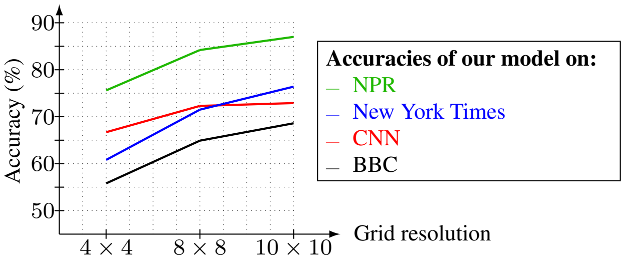


Fig. 8 Test accuracies in the similarity detection task without human supervision as the grid resolution of the GIST descriptor increases ($k = 25$).

Web Site	Visual Method	Multimodal Vis./Struct. Method
NPR	**87.0**	86.7
NYTimes	76.4	**77.0**
CNN	72.9	**75.0**
BBC	**68.6**	**68.6**

Table 2 Test accuracies (in %) in the fully unsupervised setup using only visual descriptors or combining them with structural metrics. A 10×10 grid resolution is considered ($k = 25$).

We now show how the results may be improved by exploiting little human supervision. Fig 9 reports classification accuracies on the different websites as a function of the number of annotated pairs per class ($k = |\mathcal{S}| = |\mathcal{D}|$) [9]. Using $k = 5$ annotated pairs per class improves accuracy by 5% when compared to the unsupervised method ($k = 0$), and using 20 annotated pairs further improves recognition by 5.5%. However, we reach a ceiling for $k > 20$, around which the accuracy does not improve significantly. Using a small number of annotated pairs is then sufficient. Moreover, note that the selected pairs in $\mathcal{S}$ and $\mathcal{D}$ are randomly chosen among the $24 \times 5 = 120$ possible pairs. Active strategies can be performed to minimize integrated human supervision.

Table 3 compares the accuracies obtained with the change detection method proposed in [41] and with our method that combines a learned visual metric with structural distances. To the best of our knowledge, the approach in [41] is the only machine learning method proposed for change detection in the context of Web archiving. This approach combines unlearned visual and structural distances to learn a linear SVM. The approach in [41] exploits SIFT and color-based bags-of-words representations which are slow to compute (see [39, Section 5] for details on computation time). For the sake of scalability, we consider instead GIST descriptor which is more appropriate to our large scale problem [16] and can be related to SIFT-based BoWs [54] since it provides gradient information for the different spatial grids in the image. As shown in [39], the recognition performance of similarity between pairs of webpage versions is dominated by gradient-based descriptors. Color descriptors can also be included at the expense of additional computation time.

[9] The accuracies reported with zero annotated pair sample per class correspond to those of Section 5.5, Fig. 8 and Table 2.

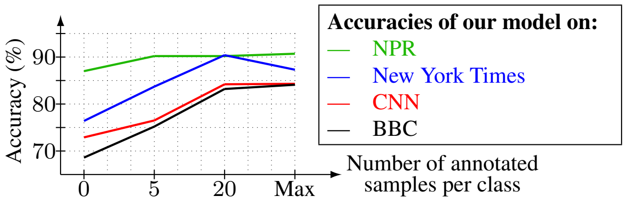


Fig. 9 Test accuracies in the semi-supervised setup for similarity detection as the number of annotated samples increases. A 10 × 10 grid GIST descriptor is used.

	Number of annotated samples per class			
	Law et al. [41]		Proposed method	
Web Site	5	20	5	20
NPR	81.4	86.1	**90.6**	**90.6**
New York Times	65.3	68.3	**83.4**	**90.2**
CNN	70.2	71.6	**77.4**	**85.1**
BBC	69.8	72.3	**80.0**	**83.9**

Table 3 Test accuracies (in %) in the (semi-)supervised setup of the baseline method described in [41] and our method using the same visual and structural descriptors.

Our proposed approach outperforms the approach in [41] by a margin of 12%. Moreover, combining structural and visual distances (see Table 3) improves recognition over visual distances alone (see Fig 9) with a global margin of 1% for all websites. This result is consistent with the observations in [41] that structural and visual distances are complementary. In conclusion, our semi-supervised method outperforms the unsupervised approach and the approach in [41] that does not focus on important regions.

In conclusion of these Webpage experiments, we have shown that:

- the metric learned with our proposed strategy allows to detect important regions in webpages. The learned metric also implicitly returns small distances for semantically similar pairs of versions and larger values for semantically distant versions.
- the metrics learned in an unsupervised way perform very well and their recognition performance is improved with very little human supervision.
- our sampling strategy allows to create a lot of significant constraints. This is particularly useful when triplet-wise sampling strategies generate a relatively small number of constraints.
- the learned distance metric can be extended by combining both visual and structural information metrics.

6 Hierarchical Metric Learning

6.1 Creation of constraints

In this section, the goal is to learn a distance metric that is relevant to a given hierarchical object class taxonomy. More precisely, our objective is to learn a metric such that images from close (e.g., sibling) classes with respect to the class semantic hierarchy are more similar than images from more distant classes. Our strategy is illustrated in Fig. 2 where different subclasses of the general class *Canis lupus* are gathered together depending on their subspecies and their breed, which corresponds to subclasses and subsubclasses in the taxonomy, respectively.

Given a semantic taxonomy expressed by a tree of classes, let us consider two sibling classes c_a and c_b and a class c_d that is not their sibling (we call it a cousin class). We generate two types of quadruplet-wise constraints in order to:

(1) Enforce the dissimilarity between two images from the same class to be smaller than between two others from sibling classes. If $(\mathcal{I}_i, \mathcal{I}_j)$ are both sampled from c_a, and $(\mathcal{I}_k, \mathcal{I}_l)$ are sampled from $c_a \times c_b$, we want $D_{ij} < D_{kl}$. These constraints are similar to the ones exploited by LMNN with the exception that we use quadruplets of images and that LMNN does not exploit taxonomy information: i.e., we sample $\mathcal{I}_l$ from a sibling class of c_a whereas LMNN samples $\mathcal{I}_l$ from any class different from c_a.

(2) Enforce the dissimilarity between two images from sibling classes to be smaller than between two images from cousin classes. If $(\mathcal{I}_i, \mathcal{I}_j)$ are sampled from $c_a \times c_b$ and $(\mathcal{I}_k, \mathcal{I}_l)$ from $c_a \times c_d$, we want $D_{ij} < D_{kl}$. These constraints are strongly related to the taxonomy information and allow to discriminate images from sibling classes better than from any other class. They follow the idea that semantically close objects should be closer with the learned distance metric.

In order to limit the number of training constraints, we sample the image $\mathcal{I}_j$ such that $\mathcal{I}_j$ is one of the k nearest neighbors of $\mathcal{I}_i$: $\mathcal{I}_j$ is sampled in the same class in the case (1) and in a sibling class in the case (2).

We consider both the diagonal PSD matrix and the full matrix distance metric formulations described in Section 3. The experiments are performed on datasets where billions of constraints can be generated. To have a tractable framework, we use the optimization strategies mentioned in Section 4.

6.2 Classification results

To evaluate our metric learning for class hierarchy, we follow the subtree classification task described in [60]. There are 9 datasets which are all subsets of ImageNet [15]): *Amphibian*, *Fish*, *Fruit*, *Furniture*, *Geological Formation*, *Musical Instrument*, *Reptile*, *Tool*, *Vehicle*. Each of these 9 datasets contains 8 to 40 different classes and from 8000 to 54000 images each. We use the train, validation and test sets de-

Subtree Dataset	Non-linear SVM	TaxEmb	Verma et al. [60]	Qwise (Diagonal Matrix)	Qwise (Full Matrix)
Amphibian	38%	38%	41%	**43.5**%	**43.5**%
Fish	34%	37%	39%	41%	**41.6**%
Fruit	22.5%	20%	**23.5**%	21.1%	21.1%
Furniture	44%	41%	46%	48.8%	**48.9**%
Geological Formation	50.5%	50.5%	52.5%	**56.1**%	**56.1**%
Musical Instrument	30.5%	23%	32.5%	**32.9**%	**32.9**%
Reptile	21.5%	18.5%	22%	23.0%	**23.1**%
Tool	27.5%	24.5%	**29.5**%	26.4%	26.7%
Vehicle	30.5%	22.5%	27%	**34.7**%	**34.7**%
Average Accuracy	33.2%	30.6%	34.8%	36.4%	**36.5**%

Table 4 Standard classification accuracy for the various datasets using the SVM classification framework for the 9 datasets from ImageNet.

fined in [60], and also the same publicly available features[10]: 1000 dimensional SIFT-based Bag-of-Words (BoW) [54].

We learn a PSD matrix $\mathbf{M} \in \mathbb{S}_+^d$ that exploits the constraints described in introduction and that we decompose[11] as $\mathbf{M} = \mathbf{L}^\top \mathbf{L}$. The matrix $\mathbf{L}$ is used to project input data in another representation space which is the input space of another classifier. We choose a standard classifier (linear SVM) to perform classification.

When we constrain $\mathbf{M} \in \mathbb{S}_+^d$ to be diagonal, we formulate our metric $D_{\mathbf{M}}^2(\mathcal{I}_i, \mathcal{I}_j) = \mathcal{D}_{\mathbf{w}}(\mathcal{I}_i, \mathcal{I}_j) = \mathbf{w}^\top \Psi(\mathcal{I}_i, \mathcal{I}_j)$ where $\Psi(\mathcal{I}_i, \mathcal{I}_j) = (\mathbf{x}_i - \mathbf{x}_j) \circ (\mathbf{x}_i - \mathbf{x}_j)$ and $\mathbf{w} = \text{Diag}(\mathbf{M})$. Once the diagonal PSD matrix $\mathbf{M} \geq \mathbf{0}$ is learned, we project the input space using the linear transformation parameterized by the diagonal matrix $\mathbf{M}^{1/2} = \mathbf{L} \in \mathbb{R}^{d \times d}$ such that $\forall i \in \{1, \dots, d\}, \mathbf{L}_{ii} = \sqrt{\mathbf{M}_{ii}}$ (note that $\mathbf{L}^\top \mathbf{L} = \mathbf{M}$).

Table 4 presents the results reported in [60] (a nonlinear SVM, TaxEmb [61] and the method proposed in [60]) and our Qwise methods (diagonal matrix [40], and full matrix).

The model of Verma et al. [60] and TaxEmb [61] also exploit class taxonomy information to learn hierarchical similarity metrics or embedding. It is worth mentioning that Verma et al. [60] have a complex learning framework: they learn a local metric parameterized by a full PSD matrix for each class (leaf of the subtree), which can lead to overfitting. Our Qwise-learning model is simpler since we learn only one global metric for each subtree. Moreover, when we use a diagonal matrix model, the number of parameters only grows linearly with the input space dimension. Both proposed methods obtain surprisingly very similar results with a global accuracy of $36.4 \sim 36.5\%$, which is 1.6% better than the method of Verma et al. [60]. Both proposed methods outperform all the reported methods, globally and on each dataset except Fruit and Tool. All these results validate the fact that the proposed constraints are useful when richer information compared to class membership information is provided.

6.3 Further analysis with k-NN classification

We now further analyze our metrics. We use the same k-NN classification[12] for all the compared approaches to focus on the discussion of the metrics.

Table 5 reports the results obtained with the Euclidean distance ($\mathbf{M} = \mathbf{I}_d$), LMNN [62], and our Qwise (diagonal and full matrix models). All the learned models outperform the Euclidean distance in this setup for the mentioned datasets. Full matrix models that exploit correlations between features outperform metric learning models that learn a diagonal distance matrix. We note that our proposed methods, that exploit hierarchical taxonomy information, slightly outperform LMNN that uses only class membership information. It is worth mentioning that this gain is not straightforward as our proposed constraints focus on preserving semantic distances w.r.t. the provided taxonomy rather than performing k-NN classification task. Moreover, although the method in Verma et al. [60] exploits a k-NN classification framework, it cannot be directly compared to the results in Table 5 since it exploits an ad hoc k-NN classifier which is optimized for the learned metric and is not the same as the one used by the methods reported in Table 5. All the methods in Table 5 exploit the same k-NN classifier as LMNN and can thus be compared to one another.

To better observe the preservation of relationship (in the hierarchy) between the predicted class and the ground truth class instead of only focusing on the correct assignment of an image to its class, we use the modified accuracy $\text{Acc}_c = 1 - \frac{1}{m} \sum_{t=1}^{m} \Delta(c, \hat{y}_t^c)$ where c and $\hat{y}_t^c$ denote the ground truth and predicted class labels of the t^{th} test example, respectively, and m is the total number of test examples in the class c. We consider:

$$\Delta(c, \hat{y}_t^c) = \begin{cases} 0 & \text{if } \hat{y}_t^c = c \\ 0.5 & \text{if } \hat{y}_t^c \text{ is a sibling class of } c \\ 1 & \text{otherwise} \end{cases} \tag{24}$$

The proposed evaluation metric Δ takes class hierarchy information into account. In particular, Eq. (24) can be seen

[10] `http://www.image-net.org/challenges/LSVRC/2010/`

[11] We use the eigendecomposition $\mathbf{M} = \mathbf{UDU}^\top$ where $\mathbf{D}$ is a diagonal matrix, and we formulate $\mathbf{L} = \mathbf{D}^{1/2}\mathbf{U}^\top$.

[12] We report the results for 10 nearest neighbor classification (which performs better than 1-NN, 5-NN and 50-NN).

Model	Amphibian	Fish	Fruit	Furn.	Geological F.	Musical I.	Reptile	Tool	Vehicle	AVG
Euclidean distance 10-NN	35.5	33.9	16.9	36.6	43.3	27.0	17.2	24.5	20.2	28.3
LMNN Diagonal Matrix 10-NN	39.0	37.4	19.5	39.4	47.2	27.5	19.8	23.8	22.9	30.8
LMNN Full Matrix 10-NN	**41.8**	38.3	21.1	41.1	49.5	28.5	21.2	24.0	28.0	32.6
Qwise Diagonal Matrix 10-NN	39.3	37.6	20.6	40.0	47.6	28.0	20.7	23.8	24.8	31.4
Qwise Full Matrix 10-NN	**41.8**	**38.5**	**21.7**	**41.6**	**51**	**29.3**	**21.8**	**24.2**	**29.3**	**33.2**

Table 5 Standard classification accuracy for the various datasets using the k-NN classification framework.

Model	Amphibian	Fish	Fruit	Furn.	Geological F.	Musical I.	Reptile	Tool	Vehicle	AVG
Euclidean distance 10-NN	50.1	35.3	32.1	42.2	45.1	28.5	21.3	26.2	29.1	34.4
LMNN Diagonal Matrix 10-NN	53.0	42.0	34.2	42.7	48.5	30.2	22.4	25.5	32.2	36.7
LMNN Full Matrix 10-NN	56.0	42.3	34.5	44.1	51.1	31.7	22.4	25.7	32.8	37.8
Qwise Diagonal Matrix 10-NN	54.8	42.5	39.1	44.8	50.0	33.1	24.4	25.6	33.2	38.6
Qwise Full Matrix 10-NN	**56.7**	**43.6**	**39.7**	**46.9**	**53.2**	**34.1**	**25.5**	**26.1**	**34.7**	**40.1**

Table 6 Classification accuracy that takes class hierarchy information into account for the various datasets using the k-NN classification framework (see text).

as a special case of the *Context Sensitive Loss* (CSL) function[13] used in [60]. In [60], they consider that $\Delta(c, \hat{y}_t^c)$ is the height of the lowest common ancestor of the pair $(c, \hat{y}_t^c)$ normalized by the maximum tree height of the taxonomy tree when $c \neq \hat{y}_t^c$. Eq. (24) corresponds to the same CSL loss as in [60] when the maximum tree height of the category taxonomy is 2. Since the constraints that we consider in Section 6.1 focus on pairs of images in the same category or in sibling categories, the loss defined in Eq. (24) is more appropriate to evaluate that our learned metric better takes into account images in the two lowest levels of the taxonomy.

Table 6 reports the corresponding scores. Qwise outperforms other methods, and the gap with LMNN is more significant. This demonstrates that the proposed constraints allow to better fit semantic relationships between classes. This result corroborates the claim of [60] that exploiting class taxonomy to learn a metric is beneficial for recognition.

7 Metric learning and Relative Attributes

In this section, we present and compare different strategies for sampling quadruplet-wise constraints in the context of relative attributes. We show that relaxing strong equivalence constraints by quadruplet-wise constraints introduces robustness and compensates for labeling imprecisions.

Relative attributes have been introduced in [48]. Attributes are human-nameable concepts used to describe images. For instance, in Fig. 10, the attribute a_m = "presence of smile" allows to rank 4 celebrity classes from the least to the most smiling. Instead of considering attributes as boolean values as done in [38] (i.e., the concept is present in the image or not), Parikh and Grauman [48] learn for each attribute a_m a vector $\mathbf{w}_m \in \mathbb{R}^d$ such that the score $\mathbf{w}_m^\top \mathbf{x}_i \in \mathbb{R}$ represents the degree of presence of a_m in the image $\mathcal{I}_i$.

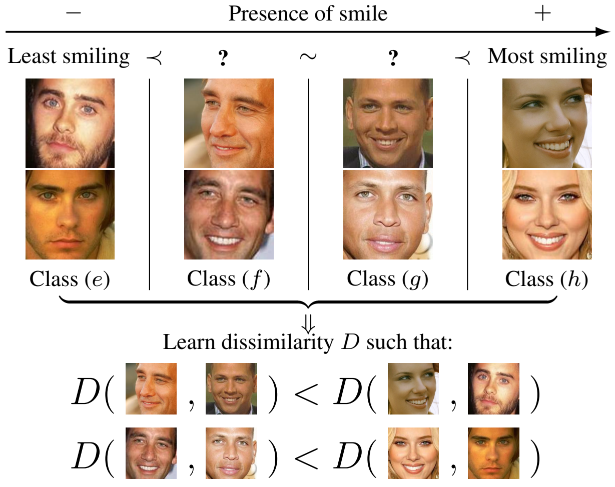


Fig. 10 Quadruplet-wise (Qwise) strategy on 4 face classes ranked according to the degree of presence of smile. Qwise strategy defines quadruplet-wise constraints to express that dissimilarities between examples from (f) and (g) should be smaller than dissimilarities between examples from (e) and (h).

To learn $\mathbf{w}_m$, they use original training sets about relative ordering between classes such as the one presented in Fig. 10: $(e) \prec (f) \sim (g) \prec (h)$. In [48], only pairwise relations are considered for learning:

- $(e) \prec (f)$ meaning that images in class (f) have stronger presence of the attribute a_m than images in class (e).
- $(f) \sim (g)$ meaning that images in (f) and (g) have equivalent strength of presence of the attribute a_m.

Let M be the total number of attributes that are considered for a given dataset. Once the optimal weight vectors $\mathbf{w}_m$ are learned for all the attributes a_m with $m \in \{1, \dots, M\}$, each image $\mathcal{I}_i$ is described by a high level feature representation:

$$\mathbf{h}_i = [\mathbf{w}_1^\top \mathbf{x}_i, \dots, \mathbf{w}_m^\top \mathbf{x}_i, \dots, \mathbf{w}_M^\top \mathbf{x}_i]^\top \in \mathbb{R}^M$$

This corresponds to learning a linear transformation parameterized by $\mathbf{L} \in \mathbb{R}^{M \times d}$ such that $\mathbf{h}_i = \mathbf{L}\mathbf{x}_i$ where the m^{th}

[13] As mentioned in [60, footnote 1], there exist several ways to define a CSL and one of these ways is chosen in [60].

row of $\mathbf{L}$ is $\mathbf{w}_m^\top$ (see Eq. (12)). As explained in Section 2.1, their problem can then be cast as a metric learning problem.

7.1 Integrating quadruplet-wise constraints

Following our vector formalism defined in Section 3.3.2, we consider to learn for each attribute a_m the signed dissimilarity function $D_{\mathbf{w}_m}$ such that $D_{\mathbf{w}_m}(\mathcal{I}_i, \mathcal{I}_j) = \mathbf{w}_m^\top \Psi(\mathcal{I}_i, \mathcal{I}_j)$, with $\Psi(\mathcal{I}_i, \mathcal{I}_j) = \mathbf{x}_i - \mathbf{x}_j$. The sign of $D_{\mathbf{w}_m}(\mathcal{I}_i, \mathcal{I}_j)$ determines the relative ordering of presence of the attribute a_m between the images $\mathcal{I}_i$ and $\mathcal{I}_j$. For instance, $D_{\mathbf{w}_m}(\mathcal{I}_i, \mathcal{I}_j) > 0$ means that the presence of a_m is stronger in $\mathcal{I}_i$ than in $\mathcal{I}_j$.

The provided information concerning the degree of presence of an attribute in an image is given at a class level: pairwise constraints may be noisy or irrelevant, leading to less than optimal learning scheme. Considering triplet-wise constraints (e.g., class (x) is more similar to (y) than to (z)) could be helpful but still generates inconsistent constraints in some cases: in Fig. 10 (second row), Owen (f) seems to be smiling more like Johansson (h) than like Rodriguez (g). To further exploit the available ordered set of classes and overcome these limitations, we consider relations between quadruplets. Two types of Qwise constraints may be derived from the training set.

7.1.1 Replacing ordered pairs by quadruplets

The first type of relation that we consider in this section is: $(e) \prec (f) \prec (g) \prec (h)$. We do the following assumption: any image pair from the extreme border classes (e) and (h) is more dissimilar than any image pair from the intermediate classes (f) and (g). This information can be written:

$$\forall (\mathcal{I}_i, \mathcal{I}_j, \mathcal{I}_k, \mathcal{I}_l) \in (g) \times (f) \times (h) \times (e) \;\; D_{kl} > D_{ij} \quad (25)$$

By sampling such quadruplets from the whole set of relative orderings over classes (e.g., Table 7, see experiments for details), we build our Qwise set $\mathcal{N}$ such that for all quadruplet q in $\mathcal{N}$, we have $\delta_q = 1$ in Eq. (14).

7.1.2 Flexible constraints instead of equivalence constraints

OSR Attributes	Relative Ordering of Classes
Natural	$T \prec I \sim S \prec H \prec C \sim O \sim M \sim F$
Open	$T \prec F \prec I \sim S \prec M \prec H \sim C \sim O$
Perspective	$O \prec C \prec M \sim F \prec H \prec I \prec S \prec T$
Large-Objects	$F \prec O \prec M \prec I \sim S \prec H \sim C \prec T$
Diagonal-Plane	$F \prec O \prec M \prec C \prec I \sim S \prec H \prec T$
Close-Depth	$C \prec M \prec O \prec T \sim I \sim S \sim H \sim F$

PubFig Attributes	Relative Ordering of Classes
Masculine-Looking	$S \prec M \prec Z \prec V \prec J \prec A \prec H \prec C$
White	$A \prec C \prec H \prec Z \prec J \prec S \prec M \prec V$
Young	$V \prec H \prec C \prec J \prec A \prec S \prec Z \prec M$
Smiling	$J \prec V \prec H \prec A \sim C \prec S \sim Z \prec M$
Chubby	$V \prec J \prec H \prec C \prec Z \prec M \prec S \prec A$
Visible-Forehead	$J \prec Z \prec M \prec S \prec A \sim C \sim H \sim V$
Bushy-Eyebrows	$M \prec S \prec Z \prec V \prec H \prec A \prec C \prec J$
Narrow-Eyes	$M \prec J \prec S \prec A \prec H \prec C \prec V \prec Z$
Pointy-Nose	$A \prec C \prec J \sim M \sim V \prec S \prec Z \prec H$
Big-Lips	$H \prec J \prec V \prec Z \prec C \prec M \prec A \prec S$
Round-Face	$H \prec V \prec J \prec C \prec Z \prec A \prec S \prec M$

Table 7 Relative orderings used in [48] for the OSR dataset (categories: coast (C), forest (F), highway (H), inside-city (I), mountain (M), open-country (O), street (S) and tall-building (T)) and the PubFig dataset (categories: Alex Rodriguez (A), Clive Owen (C), Hugh Laurie (H), Jared Leto (J), Miley Cyrus (M), Scarlett Johansson (S), Viggo Mortensen (V) and Zac Efron (Z)).

The second type of relations is: $(e) \prec (f) \sim (g) \prec (h)$, which means that the presence of the attribute a_m is equivalent for any pair of images $(\mathcal{I}_i, \mathcal{I}_j) \in (f) \times (g)$. To take into account the fact that the dissimilarity D_{ij} between $\mathcal{I}_i$ and $\mathcal{I}_j$ is signed whereas the provided information is not, we consider the following constraint[14]: $D_{kl} > |D_{ij}|$ where $(\mathcal{I}_k, \mathcal{I}_l) \in (h) \times (e)$. In order to have a convex problem, we rewrite it as two constraints:

$$\begin{cases} D_{kl} \geq D_{ij} + 1 \\ D_{kl} \geq D_{ji} + 1 \end{cases} \quad (26)$$

We thus generate two quadruplets in $\mathcal{N}$ from Eq. (26).

7.2 Classification Experiments

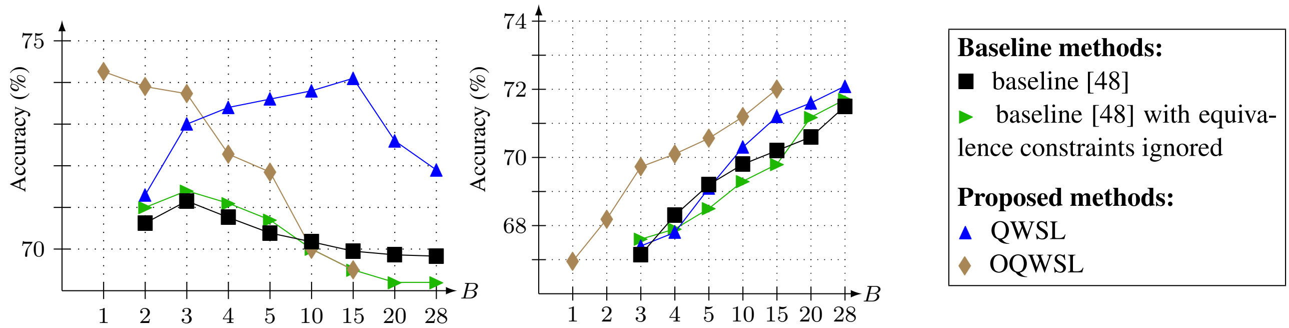
To evaluate and compare our Qwise scheme, we follow a classification framework inspired from [48] for scene and face recognition on the OSR [47] and Pubfig [36] datasets.

Datasets: We experiment with the two datasets used in [48]: Outdoor Scene Recognition (OSR) [47] containing 2688 images from 8 scene categories and a subset of Public Figure Face (PubFig) [36] containing 771 images from 8 face categories. We use the image features made publicly available by [48]: a 512-dimensional GIST [47] descriptor for OSR and a concatenation of the GIST descriptor and a 45-dimensional Lab color histogram for PubFig. Relative orderings of classes according to some semantic attributes are also available (see Table 7).

7.2.1 Recognition with Gaussian Models

We study here the impact of our proposed constraints on the original relative attribute problem [48].

[14] It is not necessary to discuss the sign of D_{kl} since $\mathcal{I}_k$ was annotated to have stronger presence of a_m than $\mathcal{I}_l$. We infer $D_{kl} > 0$.



(a) **OSR Dataset**

(b) **PubFig Dataset**

Fig. 11 Recognition performance of the baseline [48] and the proposed methods on OSR dataset (a) and PubFig dataset (b) as a function of B (the number of pairs of classes used to generate relative constraints per attribute). Accuracies smaller than 69% are not reported for $B = 1$ on OSR. Accuracies smaller than 66% are not reported for $B = 1$ or $B = 2$ on PubFig.

Baseline: As a baseline, we use the relative attribute learning problem of Parikh and Grauman [48] that exploits relative attribute orderings between classes (see Table 7) to generate pairwise constraints. A Gaussian model is learned to perform recognition, as explained below.

Qwise Method: We use for **OSR** and **Pubfig** the quadruplet-wise constraints defined in Section 7.1. The Qwise scheme uses only relative attribute information to learn a linear transformation. Particularly, we distinguish two QWise adaptations of the problem of [48] named **QWSL** and **OQWSL**:

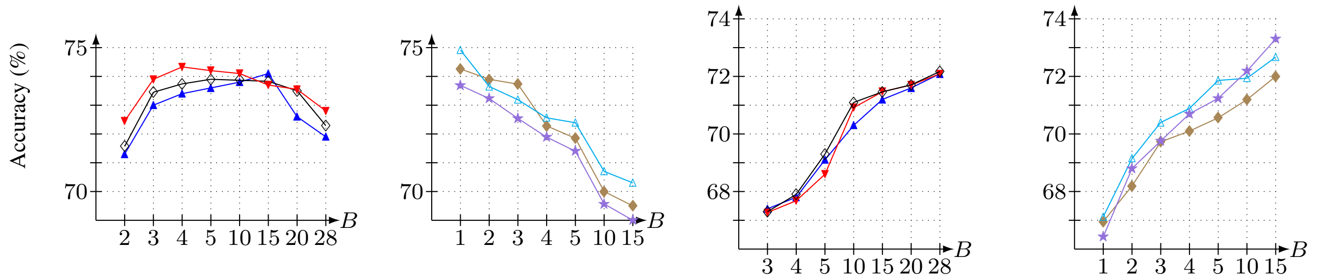
- *QWSL*: this method replaces pairwise equivalence constraints as explained in Section 7.1.2 (Eq. (26)) and exploits the same pairwise ordered constraints as [48]. By relaxing only restrictive pairwise equivalence constraints, this method is more robust to the annotation problems described in Fig 10.
- *OQWSL*: this method exploits only quadruplet-wise constraints for training. The pairwise equivalence constraints are relaxed as explained in Section 7.1.2, and pairwise ordered constraints are replaced by quadruplet-wise constraints as explained in Section 7.1.1. On some datasets, the pairwise ordered annotations performed by humans may be noisy in the same way as equivalence constraints. The purpose of this method is to relax the pairwise constraints generated by these possibly noisy annotations.

Learning setup: We use the same experimental setup as [48] to learn our Qwise metric. $N = 30$ training images are used per class, the rest is for testing. Let B be the number of pairs of classes that we select to learn the projection direction $\mathbf{w}_m$ of attribute a_m, From each of the B selected pairs of classes, we extract $N \times N$ image pairs or quadruplets to create training constraints. To carry out fair comparisons, we generate one Qwise constraint for each pairwise constraint generated by [48] using the strategies described in Section 7.1.1. In this way, we have the same number of constraints. Once all the M projection directions $\mathbf{w}_m$ are learned, a multivariate Gaussian distribution is learned for each class c_s of images: the mean $\boldsymbol{\mu}_s \in \mathbb{R}^M$ and covariance matrix $\boldsymbol{\Sigma}_s \in \mathbb{R}^{M \times M}$ are estimated using the $\mathbf{h}_i$ of all the training images $\mathcal{I}_i$ in c_s. A test image $\mathcal{I}_t$ is assigned to the class corresponding to the highest likelihood. The performance is measured as the average classification accuracy across all classes over 10 random train/test splits.

Values of B: when at least one of the two images $\mathcal{I}_i$ and $\mathcal{I}_j$ belongs to extreme border classes (e.g., the most or least smiling classes), a pair of images $(\mathcal{I}_k, \mathcal{I}_l)$ such that $D_{kl} > D_{ij}$ cannot be sampled. We ignore the constraint in this case: since we cannot generate Qwise constraints from a pairwise constraint that involves extreme border classes, the maximum possible value for B is $\binom{C-2}{2} = 15$ for OQWSL where $C = 8$ is the number of classes. Otherwise, the maximum possible value for B is $\binom{C}{2} = 28$.

Results: The comparison of our proposed methods and the baseline [48] is illustrated in Fig. 11 for the OSR dataset and PubFig dataset.

- *Pairwise baseline study:* we first study for the baseline [48] the impact of the pairwise equivalence constraints (i.e., $(f) \sim (g)$) on recognition performance to better analyze the benefit of our Qwise constraints. On both OSR and PubFig, recognition performance is comparable when pairwise equivalence constraints are exploited and when they are not. This proves that equivalence constraints are not informative and do not appropriately exploit the provided equivalence information. In the following, we study the impact on performance recognition induced by the integration of our proposed Qwise constraints:
- *OSR:* On OSR, our methods reach an accuracy of 74.3% and 74.1%, which is 3% better than the optimal baseline accuracies. QWSL is more robust as B increases, it seems to benefit both from the precision of strict order pairwise constraints and from the flexibility applied on problematic equivalent pairs of classes. OQWSL performs surprisingly well with a set of 4 classes ($B = 1$) per attribute, attesting that our Qwise scheme performs well with a small number of constraints.
- *PubFig:* On PubFig, since there are not many equivalence constraints (see Table 7), QWSL mostly uses the same pair-



(a) **QWSL** on OSR

(b) **OQWSL** on OSR

(c) **QWSL** on PubFig

(d) **OQWSL** on PubFig

Fig. 12 Recognition performance of our proposed methods for different neighbor sampling strategies (see text) on OSR dataset ((a) & (b)) and PubFig dataset ((c) & (d)) as a function of B (the number of pairs of classes used to generate relative constraints per attribute). Accuracies smaller than 69% are not reported for $B = 1$ on OSR. Accuracies smaller than 66% are not reported for $B = 1$ or $B = 2$ on PubFig.

wise constraints as the baselines and then performs similarly. OQWSL reaches 72% accuracy, which is 2% better than baselines with comparable B (number of constraints). Moreover, when combining OQWSL and pairwise ordered constraints for extreme border classes, our method reaches 74.5% accuracy.

The recognition performance of all the baselines and proposed methods decreases with large values of B on OSR but increases on PubFig, which suggests that the provided annotations of OSR are noisy, or at least not reliable. QWSL is more robust and performs at least as well as baselines on both datasets. However, OQWSL is clearly better that all the other methods on PubFig with comparable B.

In conclusion, our approach outperforms the baselines on both OSR and PubFig with a margin of 3% accuracy, reaching state-of-the-art results in this original setup[15] [48].

This proves that relaxing noisy pairwise constraints by intuitive quadruplet-wise constraints introduces robustness and compensates for labeling imprecisions described in Section 7.1.

Impact of the distance of surrounding classes to create quadruplets: We have a totally ordered set of classes per attribute to describe relations. We only studied the case where we upper bound the dissimilarity between two classes with their nearest neighbor classes in the ordered set. What happens if we choose more distant classes in the set to create quadruplets? Fig. 12 shows that our methods are very robust to the distance of surrounding classes. In the figures, the methods **(O)QWSL-1**, **(O)QWSL-2**, **(O)QWSL-3** correspond to different sampling strategies to generate a given quadruplet $q = (\mathcal{I}_i, \mathcal{I}_j, \mathcal{I}_k, \mathcal{I}_l)$ from a given pair $(\mathcal{I}_i, \mathcal{I}_j)$. For a given $p \in \{1, 2, 3\}$, (O)QWSL-p corresponds to sampling the images $\mathcal{I}_k$ and $\mathcal{I}_l$ from the p^{th} closest classes of the classes of $\mathcal{I}_i$ and $\mathcal{I}_j$.[16]

Except in Fig 12 (b) where OQWSL-3 performs little worse than OQWSL-1, choosing further neighbors gives better results than choosing nearest neighbors. Our best accuracies are obtained by doing so: QWSL-2 in Fig. 12 (a), OQWSL-2 in Fig. 12 (b) and OQWSL-3 in Fig. 12 (d). Our performances are about 4% and 1.5% better than the optimal baselines accuracies on OSR and PubFig respectively (3.5% better on PubFig with comparable B). The reason of this phenomenon seems to be the high intra-class variance. In general, using two close classes seems to be the right choice to learn a good margin between classes. However, if the generated training constraints are noisy, the quality of the learned projection direction $\mathbf{w}$ is affected.

In conclusion, Qwise constraints allow to refine relations between samples and can improve recognition.

7.2.2 Comparison of different classification models

Learning for each class a multivariate Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_s, \boldsymbol{\Sigma}_s)$ can be seen as learning a Mahalanobis distance metric $D_{\boldsymbol{\Sigma}_s^{-1}}(\mathbf{x}, \boldsymbol{\mu}_s)$. We propose to compare the performances of models learned with our constraints when combined with metric LMNN [62]. LMNN exploits only class membership information in order to learn a Mahalanobis-like distance metric. For each image, LMNN tries to satisfy the condition that members of a predefined set of target neighbors (of the same class) are closer than samples from other classes. In [62], those neighbors are chosen using the ℓ_2-distance in the input space.

Setup: The high level features $\mathbf{h}_i \in \mathbb{R}^M$ learned with our method are used as input of LMNN. We call this strategy **Qwise + Pairwise + LMNN** (Q+Pwise + LMNN) since we combine both Qwise and pairwise constraints. Depending

[15] A different setup is used in [49] where additional feedback improves recognition.

[16] For instance, if we have $(k) \prec (i) \prec (e) \prec (f) \sim (g) \prec (h) \prec (j) \prec (l)$, the classes (i) and (j) and the second closest classes of (f) and (g). The classes (k) and (l) are their third closest classes.

	OSR	Pubfig
LMNN [62]	$71.2 \pm 2.0\%$	$71.5 \pm 1.6\%$
LMNN-G	$70.7 \pm 1.9\%$	$69.9 \pm 2.0\%$
RA (Parikh's code [48])	$71.3 \pm 1.9\%$	$71.3 \pm 2.0\%$
RA + LMNN	$71.8 \pm 1.7\%$	$74.2 \pm 1.9\%$
OQSWL + LMNN-G	$73.5 \pm 1.7\%$	$74.1 \pm 1.8\%$
OQWSL + LMNN	$73.9 \pm 1.9\%$	$75.7 \pm 1.8\%$
QWSL + LMNN-G	$\mathbf{74.6 \pm 1.7}\%$	$74.8 \pm 1.7\%$
QWSL + LMNN	$74.3 \pm 1.9\%$	$77.0 \pm 1.9\%$
Qwise + Pairwise (Q+Pwise)	$74.1 \pm 2.1\%$	$74.5 \pm 1.3\%$
Q+Pwise + LMNN-G	$\mathbf{74.6 \pm 1.7}\%$	$76.5 \pm 1.2\%$
Q+Pwise + LMNN	$74.3 \pm 1.9\%$	$\mathbf{77.6 \pm 2.0}\%$

Table 8 Test classification accuracies on the OSR and Pubfig datasets for different methods.

on the dataset, we use a different definition of Q+Pwise based on the results obtained in Section 7.2.1.

• For OSR, Q+Pwise is QWSL which obtained the best results with a Gaussian model and proved to be robust.

• For Pubfig, Q+Pwise is OQWSL to which we add pairwise inequality constraints that are applied to extreme border categories for each attribute.

In both cases, our method combines quadruplet-wise and pairwise constraints. We denote:

- *LMNN*: the methods for which a k-NN classifier is used (since LMNN is designed for k-NN classification).

- *LMNN-G*: the methods for which a linear transformation is learned but used with a multivariate Gaussian model instead of a k-NN classifier. We propose these methods in order to have the same classifier as [48] and be fair in comparison.

- *RA + LMNN* is a combination of the baselines [48] and [62] that first exploits pairwise constraints based on relative attribute annotations to learn a representation of images in attribute space, and second, learns a metric in attribute space with LMNN.

We use the publicly available codes of [48] and [62]. For comparison, we also report as baselines the combination of OQWSL (which exploits only quadruplet-wise constraints) and QWSL with LMNN.

Results: Table 8 reports the classification scores for the different baselines, Q+Pwise, and Q+Pwise+LMNN.

On OSR and Pubfig, Q+PWise reaches an accuracy of 74.1% and 74.5%, respectively. It outperforms the baselines [48] and [62] on both datasets by a margin of 3% accuracy. Moreover, performance is further improved when relative attributes and LMNN are combined. Particularly, an improvement of about 3% is obtained on Pubfig, reaching 77.6%. Relative attribute annotations (used for Qwise learning) and class membership information (used for LMNN) then seem complementary. It can also be noted that the combination of pairwise and Qwise constraints obtain the best results (compared to OQWSL).

In conclusion, we have proposed and compared different strategies for sampling constraints to compensate for labeling imprecisions. Relaxing strong equivalence constraints by quadruplet-wise constraints improves recognition.

8 Conclusion and Perspectives

In this paper, we have proposed a general and efficient Mahalanobis distance metric learning framework that exploits constraints over quadruplets of images. Our approach can easily combine relative and absolute distance constraints. We experimentally show in different scenarios (i.e., relative attributes, metric learning on class hierarchy and temporal webpage analysis) that it is specifically adapted to incorporate knowledge from rich or complex semantic label relations.

In the context of relative attributes, we have shown that some pairwise comparisons of images are limited and can be improved by relaxing the quadruplet-wise relaxed constraints. In the context of hierarchical classification, class taxonomies can be used to better describe semantical relationships between images.

We have proposed a novel webpage change detection method that exploits temporal relationships between versions and detects important change regions. This method can be easily learned in a unsupervised or semi-supervised way and exploit structural information of webpages. Particularly, the change detection algorithm learned without human supervision obtains good recognition results on different websites. In order to improve recognition, it can also exploit a small number of human annotations that are performed globally on page version pairs instead of requiring annotation of each semantical block of each page as usually done. Since our method mostly relies on visual comparisons on rendered pages, it is generic and robust to the way the analyzed pages are coded. Structural distances, that use the source code of webpages, are also easy to integrate in our framework. The possible applications of our approach are diverse: Web crawling and search engine improvements, navigation in Web archives, improvement of mobile phone applications that load the important content of webpages...

Future work includes the learning of non-linear distance metrics and more general types of constraints that exploit sets of images. Also, the implementation of a webpage segmentation method dedicated to change detection by using our algorithm as a preprocessing step will be investigated.

Acknowledgements This work was partially supported by the SCAPE Project cofunded by the European Union under FP7 ICT2009.4.1 (Grant Agreement nb 270137).

A Solver for the vector optimization problem

We describe here the optimization process when the goal is to learn a dissimilarity function $\mathscr{D}_{\mathbf{w}}$ parameterized by a vector $\mathbf{w}$.

A.1 Primal form of the optimization problem

We first rewrite Eq. (14) in the primal form in order to use the efficient and scalable primal Newton method [11].

The first two constraints of Eq. (14) over $\mathcal{S}$ and $\mathcal{D}$ try to satisfy Eq. (9) and Eq. (10). They are equivalent to $y_{ij}(\mathscr{D}_{\mathbf{w}}(\mathcal{I}_i, \mathcal{I}_j) - b) \geq 1 - \xi_{ij}$ where $y_{ij} = 1 \iff (\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{D}$ and $y_{ij} = -1 \iff (\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{S}$. Eq. (14) can then be rewritten equivalently :

$$\begin{aligned} \min_{(\mathbf{w},b)} \ & \frac{1}{2}(\|\mathbf{w}\|_2^2 + b^2) + C_p \sum_{(\mathcal{I}_i, \mathcal{I}_j) \in \mathcal{S} \cup \mathcal{D}} L_1(y_{ij}, \mathscr{D}_{\mathbf{w}}(\mathcal{I}_i, \mathcal{I}_j) - b) \\ & + C_q \sum_{q \in \mathcal{N}} L_{\delta_q}(1, \mathscr{D}_{\mathbf{w}}(\mathcal{I}_k, \mathcal{I}_l) - \mathscr{D}_{\mathbf{w}}(\mathcal{I}_i, \mathcal{I}_j)) \\ & s.t. \mathbf{w} \in \mathscr{C}^d, b \in \mathscr{C} \end{aligned} \tag{27}$$

where L_1 and L_{δ_q} are loss functions and $y_{ij} \in \{-1; 1\}$. In particular, for Eq. (14) and Eq. (27) to be strictly equivalent, they have to correspond to the classic hinge loss function $L_\delta(y, t) = \max(0, \delta - yt)$. We actually use a differentiable approximation of this function to have good convergence properties [10,11].

For convenience, we rewrite some variables:

- $\boldsymbol{\omega} = [\mathbf{w}^\top, b]^\top$ is the concatenation of $\mathbf{w}$ and b in a single $(d+1)$-dimensional vector. We note $e = d + 1$ and then have $\boldsymbol{\omega} \in \mathbb{R}^e$.
- $\mathbf{c}_{ij} = [(\Psi(\mathcal{I}_i, \mathcal{I}_j))^\top, -1]^\top$ is the concatenation vector of $\Psi(\mathcal{I}_i, \mathcal{I}_j)$ and -1. We also have $\mathbf{c}_{ij} \in \mathbb{R}^e$.
- $p = (\mathcal{I}_i, \mathcal{I}_j) \iff \mathbf{c}_p = \mathbf{c}_{ij}$ and $y_p = y_{ij}$.
- $q = (\mathcal{I}_i, \mathcal{I}_j, \mathcal{I}_k, \mathcal{I}_l) \iff \mathbf{z}_q = \mathbf{x}_{kl} - \mathbf{x}_{ij}$.

Eq. (27) can be rewritten equivalently with these variables:

$$\begin{aligned} \min_{\boldsymbol{\omega} \in \mathscr{C}^e} \ & \frac{1}{2}\|\boldsymbol{\omega}\|_2^2 + C_p \sum_{p \in \mathcal{S} \cup \mathcal{D}} L_1(y_p, \boldsymbol{\omega}^\top \mathbf{c}_p) \\ & + C_q \sum_{q \in \mathcal{N}} L_{\delta_q}(1, \boldsymbol{\omega}^\top \mathbf{z}_q) \end{aligned} \tag{28}$$

By choosing such a regularization, our scheme may be compared to a RankSVM [11], with the exception that the loss function L_{δ_q} works on quadruplets. The complexity of this convex problem w.r.t. $\boldsymbol{\omega}$ is linear in the number of constraints (i.e., the cardinality of $\mathcal{N} \cup \mathcal{D} \cup \mathcal{S}$). It can be solved with a classic or stochastic (sub)gradient descent w.r.t. $\boldsymbol{\omega}$ depending on the number of constraints. The number of parameters to learn is small and grows linearly with the input space dimension, limiting overfitting [46]. It can also be extended to kernels [11].

We describe in the following how to apply Newton method [31, 10,11] to solve Eq. (28) with good convergence properties. The primal Newton method [11] is known to be fast for SVM classifier and RankSVM training. As our vector model is an extension of the RankSVM model, the learning is then also fast.

A.2 Loss functions

Let us first describe loss functions that are appropriate for Newton method. Since the hinge loss function is not differentiable, we use differentiable approximations of L_1 and L_{δ_q} inspired by the Huber loss function.

Algorithm 2 Projected Newton Step

```
Require: Sets $\mathcal{S}$, $\mathcal{D}$, $\mathcal{A}$, $\mathcal{B}$ (some of them can be empty)
Iteration $t = 0$
Initialize $\boldsymbol{\omega}_t \in \mathscr{C}^e$ (e.g., $\boldsymbol{\omega}_t = \mathbf{1}$)
Initialize the step size $\eta_t > 0$ (e.g., $\eta_t = 1$)
repeat
    Compute $\nabla_t$ and $\mathbf{H}_t$ (gradient and hessian w.r.t. $\boldsymbol{\omega}_t$)
    $\boldsymbol{\omega}_{t+1} \leftarrow \Pi_{\mathscr{C}^e}(\boldsymbol{\omega}_t - \eta_t \mathbf{H}_t^{-1} \nabla_t)$
    $t \leftarrow t + 1$
until $||\boldsymbol{\omega}_t - \boldsymbol{\omega}_{t-1}||_2^2 \leq \epsilon$
Return $\boldsymbol{\omega}_t$
```

For simplicity, we also constrain the domain of δ_q to be 0 or 1 (i.e., $\delta_q \in \{0, 1\}$). The set $\mathcal{N}$ can then be partitioned as two sets $\mathcal{A}$ and $\mathcal{B}$ such that for all:

- $q \in \mathcal{N}, \delta_q = 1 \iff q \in \mathcal{A}$
- $q \in \mathcal{N}, \delta_q = 0 \iff q \in \mathcal{B}$

In Eq. (28), we consider $t_p = \boldsymbol{\omega}^\top \mathbf{c}_p$ or $t_q = \boldsymbol{\omega}^\top \mathbf{z}_q$. Without loss of generality, let us consider t_r with $r \in \boldsymbol{\beta}$ (with $\boldsymbol{\beta} = \mathcal{S}, \mathcal{D}, \mathcal{A}$ or $\mathcal{B}$) and $y \in \{-1, +1\}$. Our loss functions are written:

$$L_1^h(y, t_r) = \begin{cases} 0 & \text{if } \ yt_r > 1 + h \ \ \text{set: } \boldsymbol{\beta}_{1,y}^0 \\ \frac{(1+h-yt_r)^2}{4h} & \text{if } \ |1 - yt_r| \leq h \ \text{set: } \boldsymbol{\beta}_{1,y}^Q \\ 1 - yt_r & \text{if } \ yt_r < 1 - h \ \ \text{set: } \boldsymbol{\beta}_{1,y}^L \end{cases} \tag{29}$$

$$L_0^h(y, t_r) = \begin{cases} 0 & \text{if } \ yt_r > 0 \qquad \text{set :} \boldsymbol{\beta}_{0,y}^0 \\ \frac{t_r^2}{4h} & \text{if } \ |-h - yt_r| \leq h \ \text{set :} \boldsymbol{\beta}_{0,y}^Q \\ -h - yt_r & \text{if } \ yt_r < -2h \qquad \text{set :} \boldsymbol{\beta}_{0,y}^L \end{cases} \tag{30}$$

where $h \in [0.01, 0.5]$. In all our experiments, we set $h = 0.05$.

As described in [10], L_1^h is inspired from the Huber loss function, it is a differentiable approximation of the hinge loss ($L_1(y, t) = \max(0, 1 - yt)$) when $h \to 0$. Similarly, L_0^h is a differentiable approximation when $h \to 0$ of $L_0(y, t) = \max(0, -yt)$, the adaptation of the hinge loss that considers the absence of security margin. Given set $\boldsymbol{\beta}$ and $y \in \{-1, +1\}$, we can infer three disjoint sets:

- $\boldsymbol{\beta}_{i,y}^0$ is the subset of elements in $\boldsymbol{\beta}$ that have zero loss in $L_i^h(y, \cdot)$.
- $\boldsymbol{\beta}_{i,y}^Q$ is the subset of elements in $\boldsymbol{\beta}$ that are in the quadratic part of $L_i^h(y, \cdot)$.
- $\boldsymbol{\beta}_{i,y}^L$ is the subset of elements in $\boldsymbol{\beta}$ in the non-zero loss linear part of $L_i^h(y, \cdot)$.

A.3 Gradient and Hessian Matrices

By considering $L_1 = L_1^h$ and $L_0 = L_0^h$ in Eq. (28), the gradient $\nabla \in \mathbb{R}^e$ of Eq. (28) w.r.t. $\boldsymbol{\omega}$ is:

$$\begin{aligned} \nabla = \boldsymbol{\omega} + \frac{C_p}{2h} & \sum_{p \in (\mathcal{S} \cup \mathcal{D})_{1,y_p}^Q} (\boldsymbol{\omega}^\top \mathbf{c}_p - (1+h)y_p)\mathbf{c}_p \\ & - C_p \sum_{p \in (\mathcal{S} \cup \mathcal{D})_{1,y_p}^L} y_p \mathbf{c}_p + \frac{C_q}{2h} \sum_{q \in \mathcal{A}_{1,1}^Q} (\boldsymbol{\omega}^\top \mathbf{z}_q - (1+h))\mathbf{z}_q \\ & + \frac{C_q}{2h} \sum_{q \in \mathcal{B}_{0,1}^Q} (\boldsymbol{\omega}^\top \mathbf{z}_q)\mathbf{z}_q - C_q \sum_{q \in (\mathcal{A}_{1,1}^L \cup \mathcal{B}_{0,1}^L)} \mathbf{z}_q \end{aligned} \tag{31}$$

and the Hessian matrix $\mathbf{H} \in \mathbb{R}^{e\times e}$ of Eq. 28 w.r.t. $\boldsymbol{\omega}$ is:

$$\mathbf{H} = \mathbf{I}_e + \frac{C_p}{2h} \sum_{p\in(\mathcal{S}\cup\mathcal{D})^{Q}_{1,y_p}} \mathbf{c}_p\mathbf{c}_p^\top + \frac{C_q}{2h} \sum_{q\in(\mathcal{A}^{Q}_{1,1}\cup\mathcal{B}^{Q}_{0,1})} \mathbf{z}_q\mathbf{z}_q^\top \quad (32)$$

where $\mathbf{I}_e \in \mathbb{R}^{e\times e}$ is the identity matrix. $\mathbf{H}$ is the sum of a positive definite matrix ($\mathbf{I}_e$) and of positive semi-definite matrices. $\mathbf{H}$ is then positive definite, and thus invertible (because every positive definite matrix is invertible).

Proof: $\mathbf{H}$ can be written $\mathbf{H} = \mathbf{I}_e + \mathbf{B}$ with $\mathbf{B} \in \mathbb{R}^{e\times e}$ a positive semi-definite matrix. For all vector $\mathbf{z} \in \mathbb{R}^e$, we have $\mathbf{z}^\top\mathbf{H}\mathbf{z} = \mathbf{z}^\top\mathbf{I}_e\mathbf{z} + \mathbf{z}^\top\mathbf{B}\mathbf{z}$. By definition of positive (semi-)definiteness, we have the following property: for all nonzero $\mathbf{z} \in \mathbb{R}^e$, $\mathbf{z}^\top\mathbf{I}_e\mathbf{z} > 0$ and $\mathbf{z}^\top\mathbf{B}\mathbf{z} \geq 0$. Then for all nonzero $\mathbf{z} \in \mathbb{R}^e$, $\mathbf{z}^\top\mathbf{H}\mathbf{z} > 0$. $\mathbf{H}$ is then a positive definite matrix. □

The global learning scheme is described in Algorithm 2. The step size $\eta_t > 0$ can be set to 1 and unchanged as in [10], or optimized at each iteration through line search (see Section 9.5.2 in [8]). The parameter $\epsilon \geq 0$ determines the stopping criterion by controlling the ℓ_2-norm of the difference of $\boldsymbol{\omega}$ between iteration t and $t-1$.

Complexity: Computing the Hessian takes $O(\sigma e^2)$ time (where $\sigma = |(\mathcal{S}\cup\mathcal{D})^{Q}_{1,y_p}| + |(\mathcal{A}^{Q}_{1,1}\cup\mathcal{B}^{Q}_{0,1})|$) and solving the linear system is $O(e^3)$ because of the inversion of $\mathbf{H}_t \in \mathbb{R}^{e\times e}$. This can be prohibitive if e is large but we restrict $e \leq 1001$ in our experiments; the inversion of $\mathbf{H}_t$ is then very fast. Other optimization methods are proposed in [11] (e.g., a truncated Newton method) if e is large.

It can be noticed that Newton method is appropriate for unconstrained problems, where the inclusion of $\mathbf{H}^{-1}$ at each iteration allows to converge faster to the global minimum. When $\mathscr{C}^e$ is $\mathbb{R}^e_+$, Eq. (28) is a constrained problem and the minimum of the unconstrained problem is not necessarily the minimum of the constrained problem. In Eq. (28), since our loss functions are linear almost everywhere on their domain, the Hessian of the problem is close to the identity matrix and it is affected almost exclusively by the regularization term. This is why applying a projected Newton method is not a major issue in our case. If computing the inverse of the Hessian is too much expensive, the Hessian can be omitted and a classic projected gradient method can be used.

References

1. Adar, E., Teevan, J., Dumais, S.: Resonance on the web: web dynamics and revisitation patterns. In: ACM CHI Conference on Human Factors in Computing Systems (CHI) (2009)
2. Adar, E., Teevan, J., Dumais, S., Elsas, J.: The web changes everything: understanding the dynamics of web content. In: ACM WSDM Conference Series Web Search and Data Mining (WSDM). ACM (2009)
3. Agarwal, S., Wills, J., Cayton, L., Lanckriet, G., Kriegman, D.J., Belongie, S.: Generalized non-metric multidimensional scaling. In: International Conference on Artificial Intelligence and Statistics (AISTATS, pp. 11–18 (2007)
4. Avila, S., Thome, N., Cord, M., Valle, E., Araújo, A.d.A.: Pooling in image representation: The visual codeword point of view. Computer Vision and Image Understanding (CVIU) **117**(5), 453–465 (2013)
5. Ben Saad, M., Gançarski, S.: Archiving the Web using Page Changes Pattern: A Case Study. In: Joint Conference on Digital Library (JCDL) (2011)
6. Borg, I., Groenen, P.: Modern multidimensional scaling: Theory and applications. Springer Series in Statistics (2005)
7. Boyd, S., Vandenberghe, L.: Subgradient. Notes for EE364b, Stanford University, Winter 2006-07 (2008). URL `http://see.stanford.edu/materials/lsocoee364b/01-subgradients_notes.pdf`
8. Boyd, S.P., Vandenberghe, L.: Convex optimization. Cambridge university press (2004)
9. Cai, D., Yu, S., Wen, J., Ma, W.: Vips: a vision-based page segmentation algorithm. Microsoft Technical Report, MSR-TR-2003-79-2003 (2003)
10. Chapelle, O.: Training a support vector machine in the primal. Neural Computation **19**(5), 1155–1178 (2007)
11. Chapelle, O., Keerthi, S.S.: Efficient algorithms for ranking with svms. Inf. Retrieval **13**(3), 201–215 (2010)
12. Chechik, G., Sharma, V., Shalit, U., Bengio, S.: Large scale online learning of image similarity through ranking. The Journal of Machine Learning Research (JMLR) **11**, 1109–1135 (2010)
13. Cord, M., Cunningham, P.: Machine learning techniques for multimedia. Springer (2008)
14. Davis, J.V., Kulis, B., Jain, P., Sra, S., Dhillon, I.S.: Information-theoretic metric learning. In: International Conference on Machine Learning (ICML) (2007)
15. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2009)
16. Douze, M., Jégou, H., Sandhawalia, H., Amsaleg, L., Schmid, C.: Evaluation of gist descriptors for web-scale image search. In: ACM International Conference on Image and Video Retrieval (CIVR) (2009)
17. Finley, T., Joachims, T.: Supervised clustering with support vector machines. In: International Conference on Machine Learning (ICML), pp. 217–224. ACM (2005)
18. Finley, T., Joachims, T.: Supervised k-means clustering. Cornell Computing and Information Science Technical Report (2008)
19. Frome, A., Singer, Y., Malik, J.: Image retrieval and classification using local distance functions. In: Advances in Neural Information Processing Systems (NIPS) (2006)
20. Frome, A., Singer, Y., Sha, F., Malik, J.: Learning globally-consistent local distance functions for shape-based image retrieval and classification. In: IEEE International Conference on Computer Vision (ICCV) (2007)
21. Goh, H., Thome, N., Cord, M., Lim, J.: Unsupervised and supervised visual codes with restricted boltzmann machines. In: European Conference on Computer Vision (ECCV) (2012)
22. Guillaumin, M., Verbeek, J., Schmid, C.: Is that you? metric learning approaches for face identification. In: IEEE International Conference on Computer Vision (ICCV) (2009)
23. Hocking, T.D., Schleiermacher, G., Janoueix-Lerosey, I., Boeva, V., Cappo, J., Delattre, O., Bach, F., Vert, J.P.: Learning smoothing models of copy number profiles using breakpoint annotations. BMC bioinformatics **14**(1), 164 (2013)
24. Hwang, S.J., Grauman, K., Sha, F.: Learning a tree of metrics with disjoint visual features. In: Advances in Neural Information Processing Systems (NIPS) (2011)
25. Hwang, S.J., Grauman, K., Sha, F.: Analogy-preserving semantic embedding for visual object categorization. In: International Conference on Machine Learning (ICML) (2013)
26. Jain, P., Kulis, B., Grauman, K.: Fast image search for learned metrics. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2008)
27. Joachims, T.: Optimizing search engines using clickthrough data. In: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 133–142. ACM (2002)
28. Joachims, T.: A support vector method for multivariate performance measures. In: Proceedings of the 22nd international conference on Machine learning, pp. 377–384. ACM (2005)
29. Joachims, T.: Training linear svms in linear time. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 217–226. ACM (2006)
30. Joachims, T., Finley, T., Yu, C.N.J.: Cutting-plane training of structural svms. Machine Learning **77**(1), 27–59 (2009)

31. Keerthi, S.S., DeCoste, D.: A modified finite newton method for fast solution of large scale linear svms. Journal of Machine Learning Research **6**(1), 341 (2005)
32. Kendall, M.G., Gibbons, J.D.: Rank correlation methods. Oxford University Press (1990)
33. Kruskal, J.B.: Nonmetric multidimensional scaling: a numerical method. Psychometrika **29**(2), 115–129 (1964)
34. Kulis, B.: Metric learning: a survey. Found. and Trends in Machine Learning **5**(4), 287–364 (2012)
35. Kumar, M., Torr, P., Zisserman, A.: An invariant large margin nearest neighbour classifier. In: IEEE International Conference on Computer Vision (ICCV) (2007)
36. Kumar, N., Berg, A., Belhumeur, P., Nayar, S.: Attribute and simile classifiers for face verification. In: IEEE International Conference on Computer Vision (ICCV) (2009)
37. Lajugie, R., Bach, F., Arlot, S.: Large-margin metric learning for constrained partitioning problems. In: International Conference on Machine Learning (ICML), pp. 297–305 (2014)
38. Lampert, C.H., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2009)
39. Law, M.T., Sureda Gutierrez, C., Thome, N., Gançarski, S., Cord, M.: Structural and visual similarity learning for web page archiving. In: 10th workshop on Content-Based Multimedia Indexing (CBMI) (2012)
40. Law, M.T., Thome, N., Cord, M.: Quadruplet-wise image similarity learning. In: IEEE International Conference on Computer Vision (ICCV), pp. 249–256 (2013)
41. Law, M.T., Thome, N., Gançarski, S., Cord, M.: Structural and visual comparisons for web page archiving. In: ACM Symposium on Document Engineering (DocEng) (2012)
42. Luo, P., Fan, J., Liu, S., Lin, F., Xiong, Y., Liu, J.: Web article extraction for web printing: a dom+ visual based approach. In: ACM Symposium on Document Engineering (DocEng). ACM (2009)
43. McFee, B., Lanckriet, G.: Partial order embedding with multiple kernels. In: International Conference on Machine Learning (ICML), pp. 721–728. ACM (2009)
44. McFee, B., Lanckriet, G.: Metric learning to rank. In: International Conference on Machine Learning (ICML) (2010)
45. Mensink, T., Verbeek, J., Perronnin, F., Csurka, G.: Metric learning for large-scale image classification: generalizing to new classes at near-zero cost. In: European Conference on Computer Vision (ECCV) (2012)
46. Mignon, A., Jurie, F.: Pcca: A new approach for distance learning from sparse pairwise constraints. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2012)
47. Oliva, A., Torralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. International Journal of Computer Vision (IJCV) **42**(3), 145–175 (2001)
48. Parikh, D., Grauman, K.: Relative attributes. In: IEEE International Conference on Computer Vision (ICCV) (2011)
49. Parkash, A., Parikh, D.: Attributes for classifier feedback. In: European Conference on Computer Vision (ECCV) (2012)
50. Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., Poggio, T.: Robust object recognition with cortex-like mechanisms. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **29**(3), 411–426 (2007)
51. Shaw, B., Huang, B.C., Jebara, T.: Learning a distance metric from a network. In: Advances in Neural Information Processing Systems (NIPS), pp. 1899–1907 (2011)
52. Shepard, R.N.: The analysis of proximities: Multidimensional scaling with an unknown distance function. i. Psychometrika **27**(2), 125–140 (1962)
53. Shepard, R.N.: The analysis of proximities: Multidimensional scaling with an unknown distance function. ii. Psychometrika **27**(3), 219–246 (1962)
54. Sivic, J., Zisserman, A.: Video Google: A text retrieval approach to object matching in videos. In: IEEE International Conference on Computer Vision (ICCV) (2003)
55. Song, R., Liu, H., Wen, J., Ma, W.: Learning block importance models for web pages. In: World Wide Web Conference (WWW) (2004)
56. Spengler, A., Gallinari, P.: Document structure meets page layout: Loopy random fields for web news content extraction. In: ACM Symposium on Document Engineering (DocEng) (2010)
57. Tenenbaum, J., De Silva, V., Langford, J.: A global geometric framework for nonlinear dimensionality reduction. Science **290**(5500), 2319–2323 (2000)
58. Theriault, C., Thome, N., Cord, M.: Extended coding and pooling in the hmax model. IEEE Transactions on Image Processing **22**(2), 764–777 (2013)
59. Torresani, L., Lee, K.: Large margin component analysis. In: Advances in Neural Information Processing Systems (NIPS) (2007)
60. Verma, N., Mahajan, D., Sellamanickam, S., Nair, V.: Learning hierarchical similarity metrics. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2012)
61. Weinberger, K., Chapelle, O.: Large margin taxonomy embedding with an application to document categorization. Advances in Neural Information Processing Systems (NIPS) **21**, 1737–1744 (2008)
62. Weinberger, K., Saul, L.: Distance metric learning for large margin nearest neighbor classification. The Journal of Machine Learning Research (JMLR) **10**, 207–244 (2009)
63. Xing, E., Ng, A., Jordan, M., Russell, S.: Distance metric learning, with application to clustering with side-information. In: Advances in Neural Information Processing Systems (NIPS) (2002)
64. Yang, J., Yu, K., Gong, Y., Huang, T.: Linear spatial pyramid matching using sparse coding for image classification. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2009)