Learning Multi-View Neighborhood Preserving Projections

Novi Quadrianto NOVI.QUADRIANTO@GMAIL.COM
SML NICTA & RSISE ANU, Canberra ACT, Australia

Christoph H. Lampert CHL@IST.AC.AT
IST Austria (Institute of Science and Technology Austria), Klosterneuburg, Austria

Abstract

We address the problem of metric learning for multi-view data, namely the construction of embedding projections from data in different representations into a shared feature space, such that the Euclidean distance in this space provides a meaningful within-view as well as between-view similarity. Our motivation stems from the problem of cross-media retrieval tasks, where the availability of a joint Euclidean distance function is a prerequisite to allow fast, in particular hashing-based, nearest neighbor queries.

We formulate an objective function that expresses the intuitive concept that matching samples are mapped closely together in the output space, whereas non-matching samples are pushed apart, no matter in which view they are available. The resulting optimization problem is not convex, but it can be decomposed explicitly into a convex and a concave part, thereby allowing efficient optimization using the convex-concave procedure. Experiments on an image retrieval task show that nearest-neighbor based cross-view retrieval is indeed possible, and the proposed technique improves the retrieval accuracy over baseline techniques.

1. Introduction

In this work we study the problem of data retrieval in a multi-view setting, i.e. for data collections that contain entries in different representations. Typical examples are multi-media databases, which contain images, audio files and videos, or simply the world wide web, seen as a huge collection of data, created by content providers as well as users in various, typically mutually incompatible data formats. Retrieving relevant data from such heterogeneous sources has become a task of major interest to large internet-based companies as well as to home users, and the application of machine learning techniques is one of the most promising approaches in this area.

We concentrate on the aspect of jointly learning a distance function for multi-view data in which Euclidean distance comparisons are meaningful not only within a single view, but also between different views. Such a representation allows one to subsequently rely on conventional retrieval techniques. Since compatibility to these is an underlying motivation of our work, we start by introducing these before motivating and introducing our new contribution.

1.1. Efficient Nearest Neighbor Retrieval

Despite certain success in the development of supervised learning techniques for retrieval tasks, for example *learning to rank* (see (Liu, 2009) for an overview), the majority of retrieval techniques today rely on some form of *(k-)nearest neighbor* search. Supervision is still useful to improve the quality of retrieved results, but typically this is done in a query-independent way through *metric learning*, which transforms the original representation of the data samples into a new, preferably low-dimensional, representations in which similar samples have a small Euclidean distance to each other, and dis-similar samples are separated by a large distance. The Euclidean metric is not chosen arbitrarily here, but motivated by considerations of efficiency: exhaustive nearest neighbors search does not scale well to the target regime of millions of database samples, and even fast approaches based on specially designed data structures, such as *kd-trees* (Friedman et al., 1977), are not efficient anymore when they have to deal with data in hundreds or even thousands of dimensions.

Appearing in *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, WA, USA, 2011.

Approximate nearest neighbor techniques are applied instead, and these are available mainly for queries with respect to the Euclidean distance, e.g. *randomized neighborhood graphs* (Arya et al., 1998), *navigating nets* (Krauthgamer & Lee, 2004) and *cover trees* (Beygelzimer et al., 2006). Methods for fast (approximate) nearest neighbor retrieval for other distance functions are rare in comparison, and typically still limited, e.g., to metrics (Omohundro, 1987), and Bregman divergences (Cayton, 2008).

For datasets of millions or even billions of entries, even approximate search-based techniques are typically infeasible, and one has to resort to *hashing* approaches. Based on the original introduction of *locality sensitive hashing* (Indyk & Motawani, 1998), several methods have been developed that transform an original feature representation into a *hash code*, i.e. a short binary string that can act as an index to directly access elements in a database, e.g. (Salakhutdinov & Hinton, 2007; Weiss et al., 2009; Kulis & Darrell, 2009).

1.2. Metric Learning

Approximate nearest neighbor search in general, and hashing-based approaches in particular, provide a powerful and well developed tool for efficient information retrieval from large databases. However, they typically rely on the availability of a meaningful Euclidean metric between the data samples. If such a metric is not readily available, *metric learning* can be applied to construct one. Basic techniques in this area are unsupervised, such as *PCA* for dimensionality reduction and the suppression of noise in already vectorial data, or *kernel-PCA* (Schölkopf et al., 1998) to construct a vector representation based from a kernel function.

Supervised techniques typically work by identifying linear projection directions that make related samples similar, and unrelated sample dis-similar in the output space, for example based on a maximum margin criteria (Schultz & Joachims, 2002; Shalev-Shwartz et al., 2004). Neighborhood component analysis (Goldberger et al., 2004) achieves a similar goal by (approximately) minimizing the leave-one-out classification error, and Weinberger & Saul (2009) combines both aspects.

Relatively little prior work, however, exists regarding the integration of multi-view data, and the methods that do exists are often tailored with specific application domains in mind: Vinokourov et al. (2002) and Hardoon et al. (2004) use canonical correlation analysis to construct a joint feature space from image and text data. Text-to-Image retrieval can then be performed by maximizing the inner product between text queries and database images. Lampert & Kroemer (2010) use maximum covariance analysis in a similar setup to jointly reduce the dimension of only weakly paired multi-view data. Ham et al. (2005) use pairwise correspondences between two views to perform *manifold alignment* based on a correlation-like measure. The resulting alignment, however, does not extend to out-of-sample examples. Jeon et al. (2003) and Monay & Gatica-Perez (2007) learn joint probabilistic models between image and text annotation, from which the best matching image for a subsequent text query can be found by marginalization. None of these approaches learn a Euclidean output metric, so they can not directly be combined with existing hashing-based retrieval techniques.

In a recent abstract, Hadsell et al. (2010) express a goal similar to ours: to learn a similarity function between multiple views that can be used as replacement for the Euclidean metric. It will be interesting to compare their approach to ours, however, so far no details or results of their work are available.

2. The Model

Here we describe our model of learning a shared latent space from the multiple representations of the objects. For the purpose of explaining our basic idea, we focus on the case when we want to learn a shared latent space from two data sources or views. We discuss the general setting of more than two data sources in Section 5.

We are given two sets of m observed data points, $\{x_1, \ldots, x_m\} \subset \mathcal{X}$ and $\{y_1, \ldots, y_m\} \subset \mathcal{Y}$ describing the same objects. For example, for image objects, $\mathcal{X}$ can be features extracted based on the content of the image itself and $\mathcal{Y}$ can be texts surrounding the image on a webpage. Note that the dimensionality of $\mathcal{X}$ and $\mathcal{Y}$ in general are not the same. We assume that for each $x_i \in \mathcal{X}$ there exists a set $\mathcal{S}_{x_i}$ of data points from $\mathcal{Y}$ that are deemed similar to x_i. In the simplest form, the set $\mathcal{S}_{x_i}$ is a singleton and contains only a data point y_j describing the same object. We explore other types of neighborhood sets in Section 4.

For $\mathcal{X} = \mathbb{R}^{d_1}$ and $\mathcal{Y} = \mathbb{R}^{d_2}$, we seek projection functions,

$$g_1 : \mathbb{R}^{d_1} \to \mathbb{R}^D \text{ and } g_2 : \mathbb{R}^{d_2} \to \mathbb{R}^D \qquad (1)$$

with potentially $D \ll \min(d_1, d_2)$ that respect the neighborhood relationship $\{\mathcal{S}_{x_i}\}_{i=1}^m$. We further assume a linear parameterization of the functions $g_1^w(x_i) := \langle w_1, \phi(x_i) \rangle$ for H_1 basis functions $\{\phi_h(x_i)\}_{h=1}^{H_1}$ and $w_1 \in \mathbb{R}^{D \times H_1}$ and similarly $g_2^w(y_i) := \langle w_2, \psi(y_i) \rangle$ for H_2 basis functions $\{\psi_h(y_i)\}_{h=1}^{H_2}$ and $w_2 \in \mathbb{R}^{D \times H_2}$.

2.1. Regularized Risk Functional

Our regularized objective function for learning a shared representation has the form

$$L(w_1, w_2, \mathcal{X}, \mathcal{Y}, \mathcal{S}) := \sum_{i,j=1}^{m} L^{i,j}(w_1, w_2, x_i, y_j, \mathcal{S}_{x_i}) + \eta\Omega(w_1) + \gamma\Omega(w_2), \tag{2}$$

where $L^{i,j}(.)$ is the loss function, $\Omega(.)$ is a regularizer on the parameters and the trade-off variables η and γ control the relative influence of loss and regularization terms. For $\Omega(.)$ one typically chooses the ℓ_2 norm, or the ℓ_1 norm if one wants to induce sparsity in the parameters. The loss function expresses the properties we expect the projected data to have, in our case that enforce similar objects across different views are mapped to nearby points, whereas dis-similar objects across different views to be pushed apart.

2.2. Loss Function

We choose the loss function as

$$L^{i,j}(w_1, w_2, x_i, y_j, \mathcal{S}_{x_i}) = \tag{3}$$

$$\frac{\mathbf{I}_{[\![y_j \in \mathcal{S}_{x_i}]\!]}}{2} \times L_1^{i,j} + \frac{\left(1 - \mathbf{I}_{[\![y_j \in \mathcal{S}_{x_i}]\!]}\right)}{2} \times L_2^{i,j} \tag{4}$$

with (5)

$$L_1^{i,j} = \|g_1^{w_1}(x_i) - g_2^{w_2}(y_j)\|_{\text{Fro}}^2 \tag{6}$$

$$L_2^{i,j}(\beta_d) = \begin{cases} -\frac{1}{2}\beta_d^2 + \frac{a\lambda^2}{2}, & \text{if } 0 \le |\beta_d| < \lambda \\ \frac{|\beta_d|^2 - 2a\lambda|\beta_d| + a^2\lambda^2}{2(a-1)}, & \text{if } \lambda \le |\beta_d| \le a\lambda \\ 0, & \text{if } |\beta_d| \ge a\lambda, \end{cases} \tag{7}$$

where $\beta_d = \|g_1^{w_1}(x_i) - g_2^{w_2}(y_j)\|_{\text{Fro}}$ for appropriately chosen constants a and λ. The above loss function consists of the similarity term $L_1^{i,j}$ that enforces similar objects to be at proximal locations in the latent space and the dis-similarity term $L_2^{i,j}$ that pushes dis-similar objects away from each other. For similar objects, the optimization problem in (2) is convex in w_1 for a fixed w_2 and vice versa. This form of the similarity loss function has been used in many metric learning literatures (for example Hadsell et al. (2010)). In many cases, however, the form of dis-similar loss function is taken to be a truncated version of the negative of the similarity loss function, i.e. $\max(0, 1 - \|g_1^{w_1}(x_i) - g_2^{w_2}(y_j)\|_{\text{Fro}}^2)$. While this is intuitive, it leads to an undesirable non-convex optimization problem without appealing decomposition properties. The loss function we introduce is also non-convex, but it has a decomposition form that is amenable to efficient iterative optimization. We call this function as a smoothly clipped inverted squared deviation (SCISD) function (see Figure 1(a)). This SCISD function decompose into a difference of two concave functions, thus (2) can be solved efficiently by using the concave convex procedure (CCCP) (Yuille & Rangarajan, 2003). This is discussed in greater detail in Section 3.

3. The Optimization

As discussed in Section 2, the optimization problem in (2) for dis-similar objects is non-convex in w_1 for a fixed w_2 and vice versa. One potential approach to solve the optimization problem is to use successive linear lower bounds on $L_2^{i,j}(.)$ and to solve the resulting convex problem and this is known as the concave convex procedure (CCCP). CCCP works as follow: for a given function $f(x) = g(x) - h(x)$, where g is convex and $-h$ is concave, an upper bound can be found by

$$f(x) \le g(x) - h(x_0) - \langle \partial h(x_0), x - x_0 \rangle . \tag{8}$$

This upper bound is convex and can be minimized effectively over a convex domain. Subsequently one finds a new location x_0 and the entire procedure is repeated. This procedure is guaranteed to converge to a local optimum or a saddle point (Sriperumbudur & Lanckriet, 2009).

To apply CCCP in our situation, we first confirm that the function $L_2^{i,j}(\beta_d)$ can be written as the difference of two concave functions (see Figure 1):

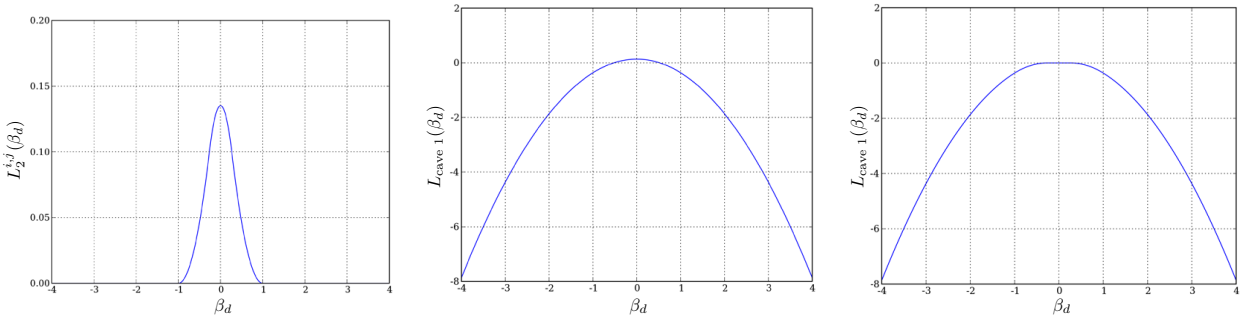
$$L_2^{i,j}(\beta_d) = L_{\text{cv}}^1(\beta_d) - L_{\text{cv}}^2(\beta_d), \text{with} \tag{9}$$

$$L_{\text{cv}}^1(\beta_d) = -\frac{1}{2}\beta_d^2 + \frac{a\lambda^2}{2} \tag{10}$$

$$L_{\text{cv}}^2(\beta_d) = \begin{cases} 0, & \text{if } 0 \le |\beta_d| < \lambda \\ \frac{2a\lambda|\beta_d| - |\beta_d|^2 - \beta_d^2(a-1) - a\lambda^2}{2(a-1)}, & \text{if } \lambda \le |\beta_d| \le a\lambda \\ -\frac{1}{2}\beta_d^2 + \frac{a\lambda^2}{2}, & \text{if } |\beta_d| \ge a\lambda, \end{cases} \tag{11}$$

Together with the convex regularization functions, the term $-L_{\text{cv}}^2(\beta_d)$ forms the convex part of the objective function while the term $L_{\text{cv}}^1(\beta_d)$ contributes the concave part. Each iteration of CCCP approximates the concave part (the $L_{\text{cv}}^1(\beta_d)$ term) by its tangent (linear upper bound), that is $\langle \partial h(x_0), x \rangle$ in (8). With this linearization, the convex upper bound of (2) with respect to w_1 is then in the form of

$$\sum_{i,j=1}^{m} \frac{\mathbf{I}_{[\![y_j \notin \mathcal{S}_{x_i}]\!]}}{2} \left[-L_{\text{cv}}^2(\beta_d) + (\langle w_1, x_i \rangle - \langle w_2, y_j \rangle) x_i^T \right] + \sum_{i,j=1}^{m} \frac{\mathbf{I}_{[\![y_j \in \mathcal{S}_{x_i}]\!]}}{2} \times L_1^{i,j} + \eta\Omega(w_1) \tag{12}$$



(a) Difference of Concave, $L_2^{i,j}(\beta_d) = L_{\text{cv}}^1(\beta_d) - L_{\text{cv}}^2(\beta_d)$ (b) $L_{\text{cv}}^1(\beta_d)$ (c) $L_{\text{cv}}^2(\beta_d)$

Figure 1. Smoothly Clipped Inverted Squared Deviation, SCISD ($a = 3.7$ and $\lambda = 1/a$). This proposed function is suitable for pushing dis-similar objects away from each other. Furthermore, it admits a desirable concave-convex decomposition property.

Algorithm 1 Multi-View Neighborhood Preserving Projection

Input: Data sources $X = \{x_1, \ldots, x_m\}$ and $Y = \{y_1, \ldots, y_m\}$, an inter-view neighborhood relationship $\{\mathcal{S}_{x_i}\}_{i=1}^m$, number of alternations N
Output: w_1^* and w_2^*
Initialize w_1 and w_2
for $t = 1$ to N **do**
  Solve the convex optimization problem in (12) w.r.t. w_1 and obtain w_1^t
  Solve the convex optimization problem in (13) w.r.t. w_2 and obtain w_2^t
end for

and similarly with respect to w_2 it has the following form

$$\sum_{i,j=1}^{m} \frac{\mathbf{I}_{[\![y_j \notin \mathcal{S}_{x_i}]\!]}}{2} \left[-L_{\text{cv}}^2(\beta_d) + (\langle w_2, y_j\rangle - \langle w_1, x_i\rangle) y_j^T\right] + \sum_{i,j=1}^{m} \frac{\mathbf{I}_{[\![y_j \in \mathcal{S}_{x_i}]\!]}}{2} \times L_1^{i,j} + \gamma\Omega(w_2). \quad (13)$$

Our neighborhood preserving projection learning essentially consists of alternating convex optimizations over w_1 and over w_2 (see Algorithm 1). Several existing convex solvers can be used to solve each of the convex optimization steps. Since we have a decomposable loss function, we can also apply a stochastic gradient style of optimization (Bottou & Bousquet, 2007), thereby facilitating large scale learning.

Algorithm 2 Hybrid-{PCA and Multi-NPP}

Input: Data sources $X = \{x_1, \ldots, x_m\}$ and $Y = \{y_1, \ldots, y_m\}$ and an inter-vew neighborhood relationship $\{\mathcal{S}_{x_i}\}_{i=1}^m$
Output: w_1^{PCA} and w_2^*
Initialize w_2
Solve the optimization problem in (13) w.r.t. w_2 while fixing $w_1 = w_1^{\text{PCA}}$ and obtain w_2^*

4. Experiments

Datasets We use 1000 images from the Israeli-Images dataset described in Bekkerman & Jeon (2007)[1]. We use 80% of the dataset as training sets and the remainder as test sets. The images consist of 11 classes, i.e. {`birds, desert, flowers, trees, food, housing, christianity, islam, judaism, personalities and symbols`} with close to uniform class proportions. To simulate a multi-view setting, we choose to represent each image by one feature vector based on global color information and one feature vector based on local SIFT descriptors that mainly encode surface texture. Note that images that are deemed similar in the color space could be categorized as dis-similar in the SIFT space, for example a black *animal* jaguar and a black *automobile* jaguar. Any multi-view learning projection should exploit this seemingly contradictory information beyond simply maximizing correlation between features in different views. As global image representation, we extract HSV color histograms using 8 uniformly spaced bins for Hue, 4 for Saturation and 2 for Value and form the normalized histogram. This means the dimensionality

[1] `http://www.cs.umass.edu/~ronb/image_clustering.html`

Table 1. Accuracy ± standard deviation. Cross-view retrieval via hybrid-{PCA and Multi-NPP}. This hybrid between our proposed learning setup and PCA achieves between-view retrieval results as good as ordinary within-view retrieval (vide Table 2 and 3).

Crossing Type	#dim	5-NN	10-NN	30-NN	50-NN	70-NN	100-NN
Color Query	10	24.2±2.59	24.9±2.72	26.3±2.82	26.4±2.56	25.8±1.90	25.8±1.73
- SIFT Database	30	28.9±2.64	29.8±3.04	30.6±2.51	30.0±2.63	29.9±2.75	29.4±2.38
	50	30.0±3.20	29.2±3.12	30.2±3.42	29.6±3.74	29.6±4.04	29.0±3.51
SIFT Query	10	18.8±3.59	19.1±3.14	19.4±3.71	19.8±3.91	19.7±4.19	19.9±3.92
- Color Database	30	24.0±3.30	24.3±3.44	24.8±3.57	24.6±3.87	24.8±3.81	24.8±3.42
	50	27.8±4.27	26.8±4.28	27.0±3.09	27.4±3.78	27.8±3.90	27.9±3.82

of our color features is 64. As local image representation, we extract colorSIFT descriptors (Van De Sande et al., 2010) and construct a codebook of 300 visual words using k-means clustering algorithm. Images are represented by normalized histograms of visual words occurrence, resulting in a features dimension of 300.

Algorithms We compare our proposed projection method with Principal Component Analysis (PCA) and Canonical Correlation Analysis (CCA) baselines. PCA finds un-correlated lower-dimensional projections by performing a variance-maximizing rotation. CCA seeks projections of paired datasets such that the correlation between the projected representations is maximized. In the context of a multi-view setting, PCA finds lower-dimensional representations from each of the view independently whereas CCA finds the representation by considering the multi-view datasets as paired datasets. Our Multi-NPP method learns lower-dimensional projections which preserves neighborhood structure across multiple views as defined by the set $\mathcal{S}_{x_i}$. In this experiment, this inter-view neighborhood set contains y_j data points from the other view having the same class label as i-th object. As well, we sample the non-neighboring points such that the cardinality of the non-neighbor set is in the same order as the neighbor set.

Model Selection We set the parameters of the SCISD function to be at $a = 3.7$ and $\lambda = 1/a$ (see Figure 1(a)). This means we try to push non-neighboring objects to be a unit apart from the object of interest. We perform a cross validation model selection approach in choosing the regularization parameters, η and γ. However, we find that our proposed method is mildly sensitive to the parameters and instead we fix both values at 10^{-6}.

Evaluation Metric Given a training set (or database), we use each sample in the test set as a query to retrieve k most similar samples or nearest neighbors in the database. We use k nearest neighbor classification metric to assess the quality of the retrieval results.

Results The experimental results of 10 repeated trials are summarized in Table 2–5. We project the original data representations to $\{10, 30, 50\}$-dimensional space. Table 2 and Table 3 simulate the standard retrieval setting when the query and the database have the same feature representations, color and SIFT, respectively. Our Multi-NPP method performs comparably to PCA and CCA approaches in color representation (Table 2) and outperforms the two baselines by a significant margin in SIFT representation (Table 3). For color space (Table 2) projecting data to lower-dimensional representation only slightly degrades the feature representations in the original space. For SIFT space, Table 3, simultaneous learning of lower-dimensional representations across 2 views improves performance (Multi-NPP). Table 4 and 5 simulate our motivating example of a cross-view retrieval task when the query and the database are described by different feature representations. It is clear that for this particular task, Multi-NPP exceeds random performance of PCA baseline and near to random performance of CCA baseline.

We also assess the performance of a variant of our method where instead of performing alternating optimizations between two projection matrices for each view, only a single convex optimization is performed. This is achieved by finding un-correlated lower-dimensional projections via PCA of one of the view and subsequently learning neighborhood preserving projections of the other view defined on the uncorrelated PCA space (refer to Algorithm 2). In the cross-view retrieval task, this turns out both computationally attractive and highly effective. Table 1 summarizes the experimental results for $\{10, 30, 50\}$ lower-dimensional projections. For 50-dimensional space, the retrieval performance of cross-view setting is on

Table 2. Accuracy ± standard deviation. View 1 - a 64-dimensional histogram vector of local color features. View 2 - a 300 dimensional histogram vector of global SIFT features. `Original`: original data space, `PCA`: principal component analysis lower-dimensional projection, `CCA`: canonical correlation analysis lower-dimensional projection and `Multi-NPP` : our neighborhood preserving lower-dimensional projection. Retrieval setting of color feature queries (test points) from a pool of color feature database (training set). Color Query - Color Database. The best result over all methods for a particular problem is highlighted in **boldface**.

Method	#dim	5-NN	10-NN	30-NN	50-NN	70-NN	100-NN
Original	64	31.4±2.52	31.3±3.87	30.4±3.55	28.6±3.29	26.2±3.93	25.2±3.51
PCA	10	**28.9±2.25**	**30.1±2.35**	**29.4±3.08**	**28.2±2.57**	**27.3±2.90**	24.8±1.79
	30	**31.0±3.27**	**32.6±3.63**	**30.4±3.82**	28.2±3.51	26.8±4.26	26.2±3.29
	50	**31.3±3.12**	**31.4±3.46**	**30.3±2.99**	28.6±3.32	26.2±3.61	25.2±3.44
CCA	10	24.8±3.86	24.7±3.42	24.0±3.78	24.4±3.75	22.8±4.18	21.6±4.18
	30	28.8±3.71	27.9±3.86	26.1±4.81	24.4±5.04	23.2±4.33	20.6±5.37
	50	29.9±3.43	28.2±2.78	26.6±4.06	24.7±4.61	24.2±4.88	20.9±5.14
Multi-NPP	10	26.4±4.33	27.6±3.39	27.4±3.54	27.3±2.97	25.9±3.07	**25.1±2.43**
	30	29.0±3.15	29.9±3.28	29.9±3.54	**29.6±3.53**	**28.2±3.44**	**27.8±4.44**
	50	30.0±3.90	29.5±2.81	30.2±3.98	**28.7±3.40**	**27.7±2.48**	**26.2±2.71**

Table 3. Retrieval setting of SIFT feature queries (test points) from a pool of SIFT feature database (training set). SIFT Query - SIFT Database.

Method	#dim	5-NN	10-NN	30-NN	50-NN	70-NN	100-NN
Original	300	32.2±2.37	33.2±3.18	30.2±4.00	29.7±4.58	28.2±3.74	25.7±4.02
PCA	10	29.6±1.99	30.2±3.18	29.9±2.84	29.6±3.49	28.8±2.07	28.2±2.15
	30	31.4±2.60	32.6±2.97	30.8±4.05	29.6±2.86	29.0±3.78	26.4±3.34
	50	31.8±3.30	32.8±3.33	30.2±4.05	29.4±3.36	28.9±4.02	26.4±3.23
CCA	10	16.7±1.88	17.7±2.48	19.1±2.00	20.0±2.07	18.8±2.53	19.0±2.71
	30	18.8±3.03	20.4±2.95	20.9±3.82	20.2±3.36	19.2±2.36	18.9±1.78
	50	19.4±3.14	21.7±3.91	20.6±3.08	18.8±1.32	19.1±2.65	18.9±2.97
Multi-NPP	10	**31.4±3.92**	**32.9±3.16**	**33.4±3.62**	**32.5±2.70**	**32.9±3.49**	**32.0±3.47**
	30	**34.5±2.77**	**35.4±3.70**	**34.4±2.53**	**34.0±2.86**	**33.5±2.28**	**33.0±2.45**
	50	**34.0±2.76**	**35.4±2.83**	**34.2±1.67**	**34.4±2.86**	**33.6±2.71**	**32.8±2.64**

Table 4. Retrieval setting of color feature queries (test points) from a pool of SIFT feature database (training set). Color Query - SIFT Database.

Method	#dim	5-NN	10-NN	30-NN	50-NN	70-NN	100-NN
PCA	10	9.3±1.66	9.3±2.03	10.0±2.31	9.5±1.83	9.2±1.86	9.2±1.87
	30	8.9±1.62	10.2±1.90	11.0±2.05	10.0±1.74	10.2±1.90	10.7±2.05
	50	9.4±1.17	10.7±1.38	10.5±2.04	11.0±1.50	10.7±1.55	10.4±2.29
CCA	10	15.4±4.27	15.8±4.53	15.9±4.59	15.6±3.80	14.8±4.27	14.8±3.90
	30	15.3±4.31	17.6±5.58	16.8±5.04	17.9±5.15	16.8±5.90	16.8±5.44
	50	16.2±4.83	16.8±5.27	18.2±6.30	17.8±6.02	18.0±6.19	17.6±6.15
Multi-NPP	10	**18.6±2.07**	**18.9±2.28**	**18.7±2.21**	**19.2±2.28**	**19.0±2.32**	**19.0±1.99**
	30	**20.8±3.16**	**20.8±3.25**	**21.0±2.60**	**20.9±2.58**	**21.4±3.13**	**20.0±3.01**
	50	**20.4±3.43**	**20.4±2.88**	**21.8±3.21**	**21.8±3.25**	**21.8±2.90**	**22.0±3.42**

Table 5. Retrieval setting of SIFT feature queries (test points) from a pool of color feature database (training set). SIFT Query - Color Database.

Method	#dim	5-NN	10-NN	30-NN	50-NN	70-NN	100-NN
PCA	10	8.2±2.54	9.2±3.35	9.4±3.36	9.4±3.30	8.8±3.35	9.4±2.72
	30	9.1±3.19	9.4±2.58	9.6±2.80	9.6±2.56	9.9±2.18	10.0±3.20
	50	8.6±2.65	9.8±2.47	9.8±3.33	9.6±2.91	9.8±2.66	9.8±2.66
CCA	10	12.5±2.98	13.8±2.36	13.8±2.82	14.3±2.90	13.4±2.69	13.8±2.82
	30	13.7±3.11	13.6±2.85	15.7±3.31	14.0±2.81	14.2±2.65	14.2±2.50
	50	13.2±1.77	13.2±2.32	13.4±2.62	12.9±2.71	12.7±3.04	12.7±1.77
Multi-NPP	10	**19.0±3.63**	**20.8±3.52**	**22.0±3.98**	**22.8±3.84**	**23.3±3.92**	**22.8±3.96**
	30	**21.8±3.65**	**23.8±3.10**	**25.1±3.28**	**24.6±3.53**	**23.9±3.15**	**23.9±2.50**
	50	**22.6±2.07**	**22.9±1.93**	**22.4±4.30**	**26.1±3.61**	**23.6±3.88**	**22.8±3.69**

the par with the standard retrieval setting. Essentially, this shows that nearest-neighbor based cross-view retrieval is indeed possible.

5. Extensions

Kernelization Multi-NPP can easily be extended by using kernel methods to work in a nonlinear feature space, as opposed to the original input space. Since our objective function in (2) consists of a loss component and a strictly monotonic increasing regularization component, the generalized Representer Theorem is thus applicable for our setting (Schölkopf et al., 2000). Consequently, the solution of (2) lies in the span of m particular kernels centered on the given training data points and thus the projection solution admits a representation of the form

$$w_1 = \sum_{i=1}^{m} \alpha_i k(x_i, \cdot), \quad \text{and } w_2 = \sum_{j=1}^{m} \beta_i l(y_j, \cdot), \quad (14)$$

for a positive-definite kernel k on $\mathcal{X}$ and a kernel l on $\mathcal{Y}$. With this, the regularized objective function in (2) can be expressed purely in term of inner products thus kerneled.

Beyond 2-View Section 2 describes the Multi-NPP objective function for the case of 2-view problem. For the case with more than two data sources we build an objective function analogous to (2) to by summing up the terms of all pairwise objectives.

6. Discussion and Conclusion

We have proposed Multi-NPP, a new metric learning technique for multi-view data. The method jointly learns projection directions for all views into a shared feature space. In combination with an existing fast nearest neighbor technique, e.g. hashing-based, this allows fast query-by-example operations not only for data within the same view, but also between views.

In order to achieve this goal, we introduced a new objective function for metric learning. It is based on the classical principle of pulling samples together, if they are related, and pushing them apart if they are not. Our contribution lies in a new loss function for the "pushing" part, which can be expressed explicitly as the difference of two concave functions. This allows application of the convex-concave procedure for optimization. As a technique based on explicit linear projections, Multi-NPP can easily be kernelized, in order to perform non-linear projection and handle non-vectorial data.

Our experiments show improved performance over PCA or CCA as baselines. We obtain a result of particular interest for the case in which one knows a priori which view will be used for queries later. In this case, a hybrid between the proposed learning setup and PCA achieves between-view retrieval results as good as ordinary within-view retrieval.

Despite the success in our experiments, there are still open questions that we plan to study in future work. In particular, we plan to explore further the question of how to avoid local extrema in the non-convex optimization, and how to select a kernel and regularization parameters in an unsupervised scenario. Furthermore, we plan to extend our model to merging the phases of metric learning and the hash code generation into a single learning problem, instead of treating them as separate modules.

Acknowledgments

The authors would like to thank Francesco Dinuzzo, Tiberio Caetano, Dale Schuurmans, and Kristian Kersting for discussions. NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program. NQ is partly supported by Microsoft Research Asia Fellowship.

References

Arya, S., Mount, D. M., Netanyahu, N. S., Silverman, R., and Wu, A. Y. An optimal algorithm for approximate nearest neighbor search in fixed dimensions. *Journal of the ACM*, 45(6):891–923, 1998.

Bekkerman, Ron and Jeon, Jiwoon. Multi-modal clustering for multimedia collections. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, pp. 1–8, 2007.

Beygelzimer, A., Kakade, S., and Langford, J. Cover trees for nearest neighbor. In *International Conference on Machine Learning*, pp. 97–104, 2006.

Bottou, L. and Bousquet, O. The tradeoffs of large scale learning. In Platt, J. C., Koller, D., Singer, Y., and Roweis, S.T. (eds.), *Advances in Neural Information Processing Systems 20*, pp. 161–168, 2007.

Cayton, L. Fast nearest neighbor retrieval for Bregman divergences. In *International Conference on Machine Learning*, pp. 112–119, 2008.

Friedman, J. H., Bentley, J. L., and Finkel, R. A. An algorithm for finding best matches in logarithmic

expected time. *ACM transaction on mathematical software*, 3(3):209–226, 1977.

Goldberger, J., Roweis, S., Hinton, G., and Salakhutdinov, R. Neighbourhood component analysis. In *Advances in Neural Information Processing Systems 17*, 2004.

Hadsell, Raia, Matei, Bogdan, and Sawhney, Harpreet. Learning a multi-modal similarity metric with application to 2d-3d matching. In *Snowbird Learning Workshop*, 2010.

Ham, Jihun, Lee, Daniel D., and Saul, Lawrence K. Semisupervised alignment of manifolds. In *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics*, 2005.

Hardoon, D., Szedmak, S., and Shawe-Taylor, J. Canonical correlation analysis: An overview with application to learning methods. *Neural Computation*, 16:2639–2664, 2004.

Indyk, P. and Motawani, R. Approximate nearest neighbors: Towards removing the curse of dimensionality. In *Proceedings of the* 30^{th} *Symposium on Theory of Computing*, pp. 604–613, 1998.

Jeon, J., Lavrenko, V., and Manmatha, R. Automatic image annotation and retrieval using cross-media relevance models. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pp. 119–126, 2003.

Krauthgamer, R. and Lee, J.R. Navigating nets: Simple algorithms for proximity search. In *Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 798–807, 2004.

Kulis, Brian and Darrell, Trevor. Learning to hash with binary reconstructive embeddings. In *Advances in Neural Information Processing Systems 22*, pp. 1042–1050, 2009.

Lampert, C. and Kroemer, O. Weakly-Paired Maximum Covariance Analysis for Multimodal Dimensionality Reduction and Transfer Learning. In *European Conference on Computer Vision*, pp. 566–579, 2010.

Liu, T.Y. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3 (3):225–331, 2009.

Monay, F. and Gatica-Perez, D. Modeling semantic aspects for cross-media image indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(10):1802–1817, 2007.

Omohundro, S. Efficient algorithms with neural network behaviour. *Complex Systems*, 1:273, 1987.

Salakhutdinov, R. and Hinton, G. Semantic Hashing. In *SIGIR workshop on Information Retrieval and applications of Graphical Models*, 2007.

Schölkopf, B., Smola, A., and Müller, K.R. Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, 10(5):1299–1319, 1998.

Schölkopf, B., Herbrich, R., Smola, A. J., and Williamson, R. C. A generalized representer theorem. Technical Report 2000-81, NeuroCOLT, 2000. Also appears in *Proceedings of the Annual Conference on Learning Theory 2001.*

Schultz, M. and Joachims, T. Learning a distance metric from relative comparisons. In *Advances in Neural Information Processing Systems 16*, 2002.

Shalev-Shwartz, S., Singer, Y., and Ng, A.Y. Online and batch learning of pseudo-metrics. In *International Conference on Machine Learning*, pp. 743–750, 2004.

Sriperumbudur, Bharath and Lanckriet, Gert. On the convergence of the concave-convex procedure. In Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C. K. I., and Culotta, A. (eds.), *Advances in Neural Information Processing Systems 22*, pp. 1759–1767. MIT Press, 2009.

Van De Sande, K.E.A., Gevers, T., and Snoek, C.G.M. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, 2010.

Vinokourov, A., Shawe-Taylor, J., and Cristianini, N. Inferring a semantic representation of text via cross-language correlation analysis. In Becker, S., Thrun, S., and Obermayer, K. (eds.), *Advances in Neural Information Processing Systems 15*, 2002.

Weinberger, K.Q. and Saul, L.K. Distance metric learning for large margin nearest neighbor classification. *The Journal of Machine Learning Research*, 10:207–244, 2009.

Weiss, Yair, Torralba, Antonio, and Fergus, Rob. Spectral hashing. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L. (eds.), *Advances in Neural Information Processing Systems 21*, pp. 1753–1760, 2009.

Yuille, A.L. and Rangarajan, A. The concave-convex procedure. *Neural Computation*, 15:915–936, 2003.