
Multimodal similarity-preserving hashing

Jonathan Masci^{1,2,3}

jonathan@idsia.ch

Michael M. Bronstein²

michael.bronstein@usi.ch

Alexander A. Bronstein⁴

bron@eng.tau.ac.il

Jürgen Schmidhuber^{1,2,3}

juergen@idsia.ch

¹Istituto Dalle Molle di Studi sull’Intelligenza Artificiale (IDSIA), Manno, Switzerland

²Faculty of Informatics, Università della Svizzera Italiana (USI), Lugano, Switzerland

³Scuola Universitaria Professionale della Svizzera Italiana (SUPSI), Lugano, Switzerland

⁴School of Electrical Engineering, Tel Aviv University, Tel Aviv, Israel.

Abstract

We introduce an efficient computational framework for hashing data belonging to multiple modalities into a single representation space where they become mutually comparable. The proposed approach is based on a novel coupled siamese neural network architecture and allows unified treatment of intra- and inter-modality similarity learning. Unlike existing cross-modality similarity learning approaches, our hashing functions are not limited to binarized linear projections and can assume arbitrarily complex forms. We show experimentally that our method significantly outperforms state-of-the-art hashing approaches on multimedia retrieval tasks.

1 Introduction

Similarity is a fundamental notion underlying a variety of computer vision, pattern recognition, and machine learning tasks ranging from retrieval, ranking, classification, and clustering to object detection, tracking, and registration. In all these problems, one has to quantify the degree of similarity between objects usually represented as feature vectors. While in some cases domain-specific knowledge dictates a natural similarity function, most generally a “natural” measure of similarity is rather elusive and cannot be constructed without side information provided e.g. through human annotation. An even more challenging setting frequently arises in tasks involving multiple media or data coming from different modalities. For example, a medical image of the same organ can be obtained using different physical processes such as CT and MRI; a multimedia search engine may perform queries in a corpus consisting of audio, video, and textual information. While domain knowledge can be used to construct reasonable similarity functions for each data modality, it is much more challenging to create a consistent and meaningful similarity measure *across* them.

Previous work The idea of constructing similarity measures suitable to specific data has been thoroughly explored by the statistics and machine learning communities. One can roughly divide similarity learning methods into unsupervised and supervised. The former class uses only the data with no additional side information. Unsupervised methods include PCA and its kernelized version (Schoelkopf et al. (1997)) that approximate the data globally by their second-order statistics either in the original Euclidean space or in a feature space represented by a kernel; and various local embedding methods such as the locally linear embedding (Roweis & Saul (2000)), Laplacian eigenmaps (Belkin & Niyogi (2003)), and diffusion maps (Coifman & Lafon (2006)), which are all based on the assumptions that the data residing in a high-dimensional Euclidean space actually belong to a low-dimensional manifold, a parametrization of which is looked for. Unsupervised methods are inherently limited due to their inability to incorporate side information into the learning process.

Supervised methods can be further subdivided according to the type of side information they rely on. Class labels is the most straightforward way of specifying side information, and is used in methods dating back to LDA (Johnson & Wichern (2002)) and its kernelized version (Mika et al. (1999)) as well as more modern approaches of Xing et al. (2002); Weinberger & Saul (2009). Other methods accept side information in the form of knowingly similar and dissimilar pairs (Davis et al. (2007)) or triplets of the form “ x is more similar to y than z ” (Shen et al. (2009); McFee & Lanckriet (2009)). A family of methods referred to as multidimensional scaling (MDS) rely on metric dissimilarity values supplied on a training set of pairs of data vectors, and seek for a Euclidean representation reproducing them as faithfully as possible (Borg & Groenen (2005)). Once the embedding into the representation space has been learned, similarity to new, unseen data is computed either directly (if the metric admits a parametric representation), or using an out-of-sample extension.

Similarity learning methods can also be classified by the type of the produced similarity functions. A significant class of practical methods learns a linear projection making the Euclidean metric optimal – this is essentially equivalent to learning a Mahalanobis distance (Weinberger & Saul (2009); Shen et al. (2009)). Kernelized versions of these approaches are often available in the cases where the data have an intricate structure that cannot be captured by a linear transformation.

Hashing approaches represent the data as binary codes to which the Hamming metric is subsequently applied as the measure of similarity. These methods include the family of locality sensitive hashing (LSH) (Gionis et al. (1999)), and the recently introduced spectral hashing (Weiss et al. (2008)). These approaches are mainly used to construct an efficient approximation of some trusted standard similarity such as the Jaccard index or the cosine distance, and are inapplicable if side information has to be relied upon. Shakhnarovich et al. (2003) proposed to construct optimal LSH-like hashes (referred to as *similarity-sensitive hashing* or SSH) using supervised learning. More efficient approaches have been subsequently proposed by Torralba et al. (2008) and Strecha et al. (2012).

The extension of similarity learning to multi-modal data has been addressed in the literature only very recently. Bronstein et al. (2010) used a supervised learning algorithm based on boosting to construct hash functions of data belonging to different modalities in a way that makes them comparable using the Hamming metric. This method can be viewed as an extension of SSH to the multimodal setting, dubbed by the authors *cross-modal SSH* (CM-SSH), and it enjoys the compactness of the representation and the low complexity involved in distance computation. McFee & Lanckriet (2011) proposed to learn multi-modal similarity using ideas from multiple kernel learning (Bach et al. (2004); McFee & Lanckriet (2009)). Multi-modal kernel learning approaches have been proposed by Lee et al. (2009) for medical image registration, and by Weston et al. (2010). The main disadvantage of the latter is the fact that it is limited to linear projections only. The framework proposed by McFee & Lanckriet (2011) can be kernelized, but it involves the computationally expensive semidefinite programming, which limits scalability. Also, both algorithms produce continuous Mahalanobis metrics, which is disadvantageous both in computational and storage complexity especially when dealing with large-scale data. The appealing property of similarity-preserving hashing methods like the CM-SSH Bronstein et al. (2010) is the compactness of the representation and the low complexity involved in distance computation.

Contributions This paper is motivated by the work of Bronstein et al. (2010) on multimodal similarity-preserving hashing. We propose a novel multi-modal similarity learning framework based on neural networks. Our approach has several advantages over the state-of-the-art. First, we combine intra- and inter-modal similarity into a single framework. This allows exploiting richer information about the data and can tolerate missing modalities; the latter is especially important in sensor networks where one or more sensors may fail or in application like multimedia retrieval where it is hard to obtain reliable samples of cross-modal similarity. We show that previous works can be considered as particular cases of our model. Second, we solve the full optimization problem without resorting to relaxations as in SSH-like methods; it has been recently shown that such a relaxation degrades the hashing performance (see e.g., Strecha et al. (2012); Masci et al. (2011)). Third, we introduce a novel coupled siamese neural network architecture to solve the optimization problem underlying our multi-modal hashing framework. Fourth, the use of neural networks can be very naturally generalized to more complex non-linear projections using multi-layered networks, thus allowing to produce embeddings of arbitrarily high complexity. We show experimental result on several standard multi-modal datasets demonstrating that our approach compares favorably to state-of-the-art algorithms.

2 Background

Let $X \subseteq \mathbb{R}^n$ and $Y \subseteq \mathbb{R}^{n'}$ be two spaces representing data belonging to different modalities (e.g., X are images and Y are text descriptions). Note that even though we assume that the data can be represented in the Euclidean space, the similarity of the data is not necessarily Euclidean and in general can be described by some metrics $d_X : X \times X \rightarrow \mathbb{R}_+$ and $d_Y : Y \times Y \rightarrow \mathbb{R}_+$, to which we refer as *intra-modal dissimilarities*. Furthermore, we assume that there exists some *inter-modal dissimilarity* $d_{XY} : X \times Y \rightarrow \mathbb{R}_+$ quantifying the “distance” between points in different modality. The ensemble of intra- and inter-modal structures d_X, d_Y, d_{XY} is not necessarily a metric in the strict sense. In order to deal with these structures in a more convenient way, we try to represent them in a common metric space. In particular, the choice of the Hamming space offers significant advantages in the compact representation of the data as binary vectors and the efficient computation of their similarity.

Multimodal similarity-preserving hashing is the problem of represent the data from different modalities X, Y in a common space $\mathbb{H}^m = \{\pm 1\}^m$ of m -dimensional binary vectors with the Hamming metric $d_{\mathbb{H}^m}(a, b) = \frac{m}{2} - \frac{1}{2} \sum_{i=1}^m a_i b_i$ by means of two embeddings, $\xi : X \rightarrow \mathbb{H}^m$ and $\eta : Y \rightarrow \mathbb{H}^m$ mapping similar points as close as possible to each other and dissimilar points as distant as possible from each other, such that $d_{\mathbb{H}^m}(\xi(x), \xi(x')) \approx d_X(x, x')$, $d_{\mathbb{H}^m}(\eta(y), \eta(y')) \approx d_Y(y, y')$, and $d_{\mathbb{H}^m}(\xi(x), \eta(y)) \approx d_{XY}(x, y)$. In a sense, the embeddings act as a *metric coupling*, trying to construct a single metric that preserves the intra- and inter-modal similarities. A simplified setting of the multimodal hashing problem used in Bronstein et al. (2010) is *cross-modality similarity-preserving hashing*, in which only the inter-modal dissimilarity d_{XY} is taken into consideration and d_X, d_Y are ignored. To the best of our knowledge, the full multimodal case has never been addressed before.

For simplicity, in the following discussion we assume the side information given as the intra- and inter-modal dissimilarities to be binary, $d_X, d_Y, d_{XY} \in \{0, 1\}$, i.e., a pair of points can be either similar or dissimilar. This dissimilarity is usually unknown and hard to model, however, it should be possible to sample d_X, d_Y, d_{XY} on some subset of the data $X' \subset X, Y' \subset Y$. This sample can be represented as sets of similar pairs of points (*positives*) $\mathcal{P}_X = \{(x \in X', x' \in X') : d_X(x, x') = 0\}$, $\mathcal{P}_Y = \{(y \in Y', y' \in Y') : d_Y(y, y') = 0\}$, and $\mathcal{P}_{XY} = \{(x \in X', y \in Y') : d_{XY}(x, y) = 0\}$, and likely defined sets $\mathcal{N}_X, \mathcal{N}_Y$, and \mathcal{N}_{XY} of dissimilar pairs of points (*negatives*). In many practical applications such as image annotation or text-based image search, it might be hard to get the inter-modal positive and negative pairs, but easy to get the intra-modal ones.

The problem of multimodal similarity-preserving hashing boils down to find two embeddings $\xi : X \rightarrow \mathbb{H}^m$ and $\eta : Y \rightarrow \mathbb{H}^m$ such that $m d_{\mathbb{H}^m}(\xi(x), \eta(y)) \approx d_{XY}(x, y)$ minimizing the aggregate of false positive and false negative rates,

$$\begin{aligned} \min_{\xi, \eta} \quad & \mathbb{E}\{d_{\mathbb{H}^m}(\xi(x), \xi(x')) | \mathcal{P}_X\} + \mathbb{E}\{d_{\mathbb{H}^m}(\eta(y), \eta(y')) | \mathcal{P}_Y\} + \mathbb{E}\{d_{\mathbb{H}^m}(\xi(x), \eta(y)) | \mathcal{P}_{XY}\} - \\ & \mathbb{E}\{d_{\mathbb{H}^m}(\xi(x), \xi(x')) | \mathcal{N}_X\} - \mathbb{E}\{d_{\mathbb{H}^m}(\eta(y), \eta(y')) | \mathcal{N}_Y\} - \mathbb{E}\{d_{\mathbb{H}^m}(\xi(x), \eta(y)) | \mathcal{N}_{XY}\}. \end{aligned} \quad (1)$$

Cross-modality similarity sensitive hashing Bronstein et al. (2010) studied the particular case of cross-modal similarity hashing (without incorporating intra-modality similarity), with linear embeddings of the form $\xi(\mathbf{x}) = \text{sign}(\mathbf{Px} + \mathbf{a})$ and $\eta(\mathbf{y}) = \text{sign}(\mathbf{Qy} + \mathbf{b})$. Their CM-SSH algorithm constructs the dimensions of ξ and η one-by-one using boosting. At each iteration, one-dimensional embeddings $\xi_i(\mathbf{x}) = \text{sign}(\mathbf{p}_i \mathbf{x} + a_i)$ and $\eta_i(\mathbf{y}) = \text{sign}(\mathbf{q}_i \mathbf{y} + b_i)$ are found using a two-stage scheme: first, the embeddings are linearized as $\xi_i(\mathbf{x}) \approx \mathbf{p}_i^\top \mathbf{x}$ and $\eta_i(\mathbf{y}) \approx \mathbf{q}_i^\top \mathbf{y}$ and the resulting objective is minimized to find the projection

$$\min_{\mathbf{p}_i, \mathbf{q}_i} \mathbb{E}\{\mathbf{x}^\top \mathbf{p}_i^\top \mathbf{q}_i \mathbf{y} | \mathcal{P}_{XY}\} - \mathbb{E}\{\mathbf{x}^\top \mathbf{p}_i^\top \mathbf{q}_i \mathbf{y} | \mathcal{N}_{XY}\}, \quad (2)$$

(here \mathbf{p}_i and \mathbf{q}_i are unit vectors representing the i th row of the matrices \mathbf{P} and \mathbf{Q} , respectively, and the expectations are weighted by per-sample weights adjusted by the boosting). With such an approximation, the optimal projection directions \mathbf{p} and \mathbf{q} have a closed-form expressions using SVD of the positive and negative covariance matrices. At the second stage, the thresholds a_i and b_i are found by one-dimensional search.

This approach has several drawbacks. First, CM-SSH solves a particular setting of problem (1) with $\mathcal{P}_{XY}, \mathcal{N}_{XY}$ only, thus ignoring the intra-modality similarities. Second, the assumption of

separability (treating each dimension separately) and the linearization of the objective replace the original problem with a relaxed version, whose optimization produces suboptimal solutions that tend to increase the hash sizes (or alternatively, for a fixed hash length m , the method manifests inferior performance; see Masci et al. (2011)). Finally, this approximation is limited to a relatively narrow class of linear embeddings that often do not capture well the structure of the data.

3 Multimodal NN hashing

Our approach for multimodal hashing is related to supervised methods for dimensionality reduction and in particular extends the framework of (Schmidhuber & Prelinger (1993); Hadsell et al. (2006); Taylor et al. (2011)), also known as the *siamese architecture*. These methods learn a mapping onto a usually low-dimensional feature space such that similar observations are mapped to nearby points in the new manifold and dissimilar observations are pulled apart. In our simplest setting, the linear embedding $\xi = \text{sign}(\mathbf{P}\mathbf{x} + \mathbf{a})$ is realized as a neural network with a single layer (where \mathbf{P} represent the linear weights and \mathbf{a} is the bias) and a sign activation function (in practice, we use a smooth approximation $\text{sign}(x) \approx \tanh(\beta x)$). The parameters of the embedding can be learned using the back-propagation algorithm (Werbos (1974)) minimizing the loss

$$\mathcal{L}_X = \frac{1}{2} \sum_{(\mathbf{x}, \mathbf{x}') \in \mathcal{P}_X} \|\xi(\mathbf{x}) - \xi(\mathbf{x}')\|_2^2 + \frac{1}{2} \sum_{(\mathbf{x}, \mathbf{x}') \in \mathcal{N}_X} \max\{0, m_X - \|\xi(\mathbf{x}) - \xi(\mathbf{x}')\|_2\}^2 \quad (3)$$

w.r.t. the network parameters (\mathbf{P}, \mathbf{a}) . Note that for binary vectors (when $\beta = \infty$), the squared Euclidean distance in (3) is equivalent up to constants to the Hamming distance. The second term in (3) is a *hinge-loss* providing robustness to outliers and produces a mapping for which negatives are pulled m_X apart. The system is fed with pairs of samples which share the same parametrization and for which a corresponding dissimilarity is known, 0 for positives and 1 for negatives (thus the name *siamese network*, e.g. two inputs and a common output vector).

Coupled siamese architecture In the multimodal setting, we have two embeddings ξ and η , each cast as a siamese network with parameters (\mathbf{P}, \mathbf{a}) and (\mathbf{Q}, \mathbf{b}) , respectively. Such an architecture allows to learn similarity-sensitive hashing for each modality independently by minimizing the loss functions $\mathcal{L}_X, \mathcal{L}_Y$. In order to incorporate inter-modal similarity, we couple the two siamese networks by the cross-modal loss

$$\mathcal{L}_{XY} = \frac{1}{2} \sum_{(\mathbf{x}, \mathbf{y}) \in \mathcal{P}_{XY}} \|\xi(\mathbf{x}) - \eta(\mathbf{y})\|_2^2 + \frac{1}{2} \sum_{(\mathbf{x}, \mathbf{y}) \in \mathcal{N}_{XY}} \max\{0, m_{XY} - \|\xi(\mathbf{x}) - \eta(\mathbf{y})\|_2\}^2, \quad (4)$$

thus jointly learning two sets of parameters for each modality. We refer to this model, which generalizes the siamese framework, as *coupled siamese networks* for which a schematic representation is shown Figure 1.

Our implementation differs from the original architecture of Hadsell et al. (2006) in the choice of the output activation function (we use tanh activation that encourages binary representations rather than a linear output layer). This way the maximum distance is bounded by $\sqrt{4m}$ and by simply enlarging the margin between dissimilar pairs we enforce the learning of codes which differ by the sign of their components. Once the model is learned, hashes are produced by thresholding the output.

The reader should also note that the hamming distance is equal to the squared euclidean distance. Hence the loss function in eq 5, when $\beta \rightarrow +\infty$, margins = 0 and $\alpha = 1$ coincides with eq 1. However for optimization reasons a margin needs to be added.

Training The training of our coupled siamese network is performed by minimizing

$$\min_{\mathbf{P}, \mathbf{a}, \mathbf{Q}, \mathbf{b}} \mathcal{L}_{XY} + \alpha_X \mathcal{L}_X + \alpha_Y \mathcal{L}_Y, \quad (5)$$

where α_X, α_Y are weights determining the relative importance of each modality. The loss (5) can be considered as a generalization of the loss in (1), which is obtained by setting $\alpha_X = \alpha_Y = 1$, margins = 0, and $\beta = \infty$. Furthermore, setting $\alpha_X = \alpha_Y = 0$, we obtain the particular setting of cross-modal loss, whose relaxed version is minimized by the CM-SSH algorithm of Bronstein

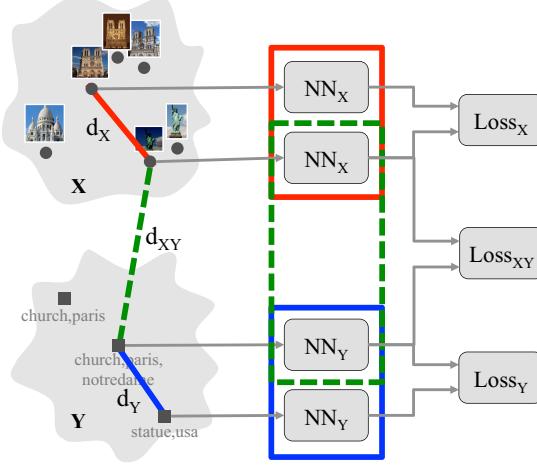


Figure 1: Schematic representation of our coupled siamese network framework. Two pairs of nets, one for modality X and one for modality Y are coupled together by a cross-modal loss. The system learns two set of parameters.

et al. (2010). It is also worth repeating that in many practical cases, it is very hard to obtain reliable cross-modal training samples ($\mathcal{P}_{XY}, \mathcal{N}_{XY}$) but much easier to obtain intra-modal samples ($\mathcal{P}_X, \mathcal{N}_X, \mathcal{P}_Y, \mathcal{N}_Y$). In the full multimodal setting ($\alpha_X, \alpha_Y > 0$), the terms $\mathcal{L}_X, \mathcal{L}_Y$ can be considered as a *regularization*, preventing the algorithm from over fitting.

We apply the back-propagation algorithm (Werbos (1974); LeCun (1985); Rumelhart et al. (1986)) to get the gradient of our model w.r.t. the embedding parameters. The gradients of the intra- and inter-modal loss functions w.r.t. to the parameters of ξ are given by

$$\begin{aligned} \nabla \mathcal{L}_X &= \begin{cases} (\xi(\mathbf{x}) - \xi(\mathbf{x}'))(\nabla \xi(\mathbf{x}) - \nabla \xi(\mathbf{x}')) & (\mathbf{x}, \mathbf{x}') \in \mathcal{P}_X \\ (\xi(\mathbf{x}) - \xi(\mathbf{x}') - m_X)(\nabla \xi(\mathbf{x}) - \nabla \xi(\mathbf{x}')) & (\mathbf{x}, \mathbf{x}') \in \mathcal{N}_X \text{ and } \\ & m_X > \|\xi(\mathbf{x}) - \xi(\mathbf{x}')\|_2 \\ 0 & \text{else} \end{cases} \\ \nabla \mathcal{L}_{XY} &= \begin{cases} (\xi(\mathbf{x}) - \eta(\mathbf{y}))\nabla \xi(\mathbf{x}) & (\mathbf{x}, \mathbf{y}) \in \mathcal{P}_{XY} \\ (\xi(\mathbf{x}) - \eta(\mathbf{y}) - m_{XY})\nabla \xi(\mathbf{x}) & (\mathbf{x}, \mathbf{y}) \in \mathcal{N}_{XY} \text{ and } m_{XY} > \|\xi(\mathbf{x}) - \eta(\mathbf{y})\|_2 \\ 0 & \text{else} \end{cases} \end{aligned}$$

where the term $\nabla \xi = \partial \xi / \partial (\mathbf{P}, \mathbf{a})$ is the usual back-propagation step of a neural network. Equivalent derivation is done for the parameters of η . The model can be easily learnt jointly using any gradient-based technique such as conjugate gradient or stochastic gradient descent.

Non-linear embeddings Our model straightforwardly generalizes to non-linear embeddings using multi-layered network architecture. The proposed framework is in fact general and any class of neural networks can be applied to arbitrarily increase the complexity of the embedding. Deep and hierarchical models are able to model highly non-linear embeddings and scale well to large-scale data by means of fully online learning, where the parameters are updated after every input tuple presentation. This allows to sample a possibly huge space with constant memory requirements.

4 Results

We tested our algorithm on cross-modal data retrieval tasks using standard datasets from the shape retrieval and multimedia retrieval communities. We compared three algorithms: our coupled siamese framework in the full multimodal setting (MM-NN) and its reduced version (CM-NN), as well as CM-SSH. The single-layer version (denoted L1) of CM-NN and MM-NN realizes a linear embedding function and compares directly with CM-SSH. Two-layered version (L2) allows to obtain more

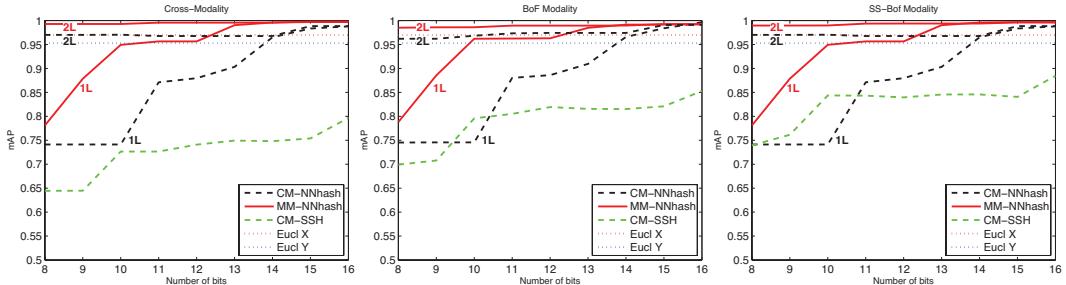


Figure 2: Mean average precision (mAP) vs. hash length m for the ShapeGoogle retrieval experiment. *Left:* cross-modal (BoF–SS-BoF); *center:* BoF; *right:* SS-BoF. L1 and L2 refer to single and two-layer neural networks, respectively. Performance of raw descriptors in each modality with L_2 distance is shown in dotted.

complex non-linear embeddings. For training the neural networks, we used conjugate gradients. The hash functions learned by each of the methods were applied to the data in the datasets, and the Hamming distance was used to rank the matches. Retrieval performance was evaluated using *mean average precision* $mAP = \sum_{r=1}^R P(r) \cdot rel(r)$, where $rel(r)$ is the relevance of a given rank (one if relevant and zero otherwise), R is the number of retrieved results, and $P(r)$ is *precision at r*, defined as the percentage of relevant results in the first r top-ranked retrieved matches.

ShapeGoogle In the first experiment, we reproduced the multimodal shape retrieval experiment of Bronstein et al. (2010) using the ShapeGoogle dataset Bronstein et al. (2011), containing 583 geometric shapes of 12 different classes subjected to synthetic transformations as well as 456 unrelated shapes (“distractors”). The goal was to correctly match a transformed shape from the query set to the rest of the dataset. The shapes were represented using 32-dimensional bag of geometric features (BoF) and 64-dimensional spatially-sensitive bags of geometric features (SS-BoF). To learn the hashing functions, we used positive and negative sets of size $|\mathcal{P}| = 10^4$ and $|\mathcal{N}| = 5 \times 10^4$, respectively. For CM-SSH, we used the code with settings provided by Bronstein et al. (2010). For MM-NN, we used single-layer architecture with margins $m_X = m_Y = 1$, $m_{XY} = 3$ and $\alpha_X = 0.1$, $\alpha_Y = 0.3$ which we empirically found to be the best combination (additional results with different parameters are shown in Table 3 and in supplementary materials). In addition, we also show a two-layer architecture with 128 hidden nodes. For CM-NN we used $m_{XY} = 3$ as for the single layer case. MM-NN used $m_X = m_Y = 1$, $m_{XY} = 3$ and $\alpha_X = 0.3$, $\alpha_Y = 0.3$.

Figure 2 and Table 1 shows the performance of different methods as function of hash length m . First, we can see that NN-based methods (CM-NN and MM-NN) dramatically outperform the boosting-based CM-SSH for a fixed hash length. MM-NN achieves almost perfect performance using only 12 bits (for comparison, CM-SSH requires almost 100 bits to achieve similar performance). The reason is likely to be the fact that CM-SSH resorts to relaxation of the problem thus producing a suboptimal solution, while NN-hash solved the “true” optimization problem. Secondly, adding another layer to the neural network we obtain a non-linear hashing function, which performs dramatically better than a single-layered architecture, achieving near-perfect performance with 8 bits. Thirdly, fully multimodal method (MM-NN) consistently outperforms the cross-modal version (CM-NN). We attribute this fact to the use of the intra-modal losses, acting as regularization.

	BoF-SS-BoF	BoF	SS-BoF	BoF-SS-BoF	BoF	SS-BoF
Raw	—	96.97%	95.29%	—	96.97%	95.29%
MM-NN	L1	95.63%	96.29%	95.63%	99.69%	99.18%
	L2	99.56%	98.94%	99.37%	99.76%	99.26%
CM-NN	L1	87.97%	88.60%	87.97%	98.77%	99.64%
	L2	96.77%	97.40%	96.77%	98.83%	99.08%
CM-SSH	74.09%	81.93%	83.95%	79.68%	85.29%	88.45%

12-bit hash

16-bit hash

Table 1: Performance of different methods (mAP) on the ShapeGoogle retrieval experiment. L1 and L2 refer to neural networks with 1 or 2 layers.

Figure 3 visually exemplifies a retrieval experiment for MM-NNhash where the query shape on the left-most side is compared with a similar shape, middle, and a dissimilar one, right-most. The produced hash vectors on the bottom row are shown along with the BoF and the SS-BoF descriptors.

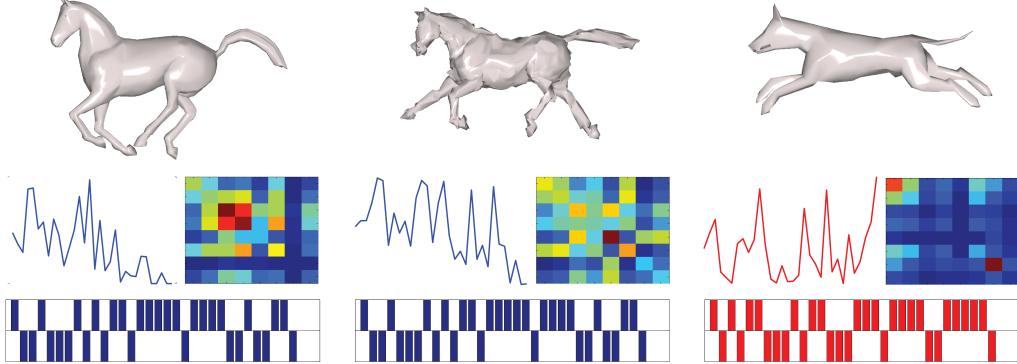


Figure 3: ShapeGoogle retrieval example. *Top*: original shapes, *middle*: BoF descriptor (left) and SS-BoF (right), *bottom*: MM-NNhash binary descriptors.

Importance of intra-modal regularization is exemplified in Table 2. In this experiment, we performed training of a 1 layer net using a subset of the cross-modal data ($\mathcal{P}_{XY}, \mathcal{N}_{XY}$), while keeping the intra-modal data in the MM-NN. The CM-MN method manifested significant performance drop (attributed most likely to overfitting), while the performance of MM-NN remains practically unchanged.

	BoF-SS-BoF		BoF		SS-BoF	
	1/2	1/10	1/2	1/10	1/2	1/10
MM-NN	99.53%	99.28%	99.67%	99.63%	99.53%	99.26%
CM-NN	98.44%	95.03%	98.96%	94.35%	98.51%	95.83%

Table 2: Performance of different methods (mAP) on the ShapeGoogle retrieval experiment where only a subset of the cross-modal correspondences is kept. Hash of length $m = 16$ is used and the best configuration is found through grid search.

Choice of parameters The theoretically smallest hash length must be $m = \lceil \log_2(\#classes) \rceil$. However, since we are using a simple embedding, in practice an m about 5–10 times larger may be required to achieve satisfactory results. Table 3 and Figure 4 show the performance of the NNhash methods under different choices of the parameters. We can see that the addition of intra-modal regularization makes the cross-modal performance less sensitive to the choice of the parameters, and that MM-NN produces higher cross-modal performance than CM-MM for similar margin settings.

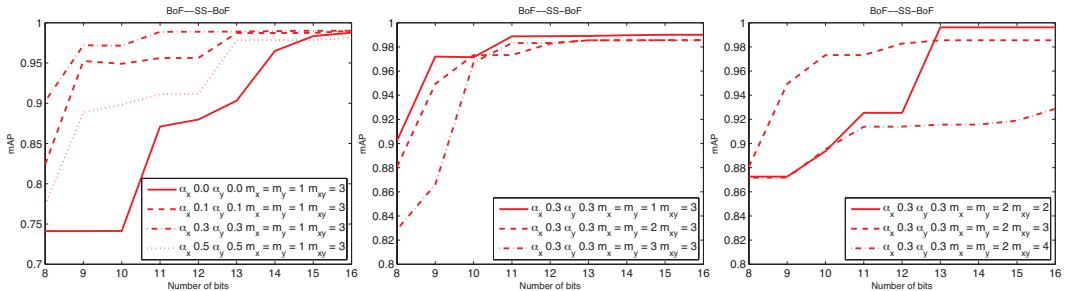


Figure 4: mAP for the ShapeGoogle experiment with various configuration of the hyper-parameters of the system.

α_X	α_Y	m_X	m_Y	m_{XY}	BoF-SS-BoF		BoF		SS-BoF		
					12 bit	16 bit	12 bit	16 bit	12 bit	16 bit	
0.0	0.5	1	1	2	90.67%	99.38%	91.26%	98.87%	90.67%	99.38%	
0.5	0.0	2	2	3	93.12%	99.39%	94.83%	99.35%	91.79%	99.25%	
0.1	0.1	1	1	3	95.62%	99.00%	96.05%	98.93%	95.62%	99.00%	
0.3	0.3	2	2	2	92.54%	99.62%	92.51%	99.45%	92.54%	99.62%	
0.3	0.3	1	1	3	98.88%	99.01%	98.97%	99.24%	98.88%	99.01%	
0.3	0.3	2	2	3	98.28%	98.55%	98.74%	98.51%	98.28%	99.48%	
0.5	0.5	3	3	3	95.59%	99.07%	95.46%	99.13%	95.59%	98.93%	
					2	96.03%	96.96%	96.52%	96.97%	95.51%	96.72%
0	0	0	0		3	87.97%	98.77%	88.60%	99.64%	87.97%	98.99%
					5	91.22%	95.60%	93.75%	97.06%	91.24%	95.60%

Table 3: Performance of different methods in the ShapeGoogle cross-modal retrieval experiment using several hash lengths and various selection of the parameters for single layer nets. The settings with $\alpha_X = \alpha_Y = m_X = m_Y = 0$ correspond to CM-NN.

NUS In the second experiment, we used the NUS dataset of Chua et al. (2009), containing about 250K annotated images from Flickr. The images are manually categorized into 81 classes (one image can belong to more than a single class) and represented as 500-dimensional bags of SIFT features (BoF, used as the first modality) and 1000-dimensional bags of text tags (Tags, used as the second modality). The dataset was split into approximately equal parts for testing and training. We used positive and negative sets of size $|\mathcal{P}| = |\mathcal{N}| = 5 \times 10^5$. Positive pairs were images belonging at least to one common class; negative pairs were images belonging to disjoint sets of classes. For MM-NN, we used the margins $m_{XY} = 7, m_X = m_Y = 3$ and $\alpha_X = \alpha_Y = 0.3$. CM-NN used $m_{XY} = 7$. Testing was performed using a query and database sets of size approximately 10^4 and 1.8×10^5 , respectively. First ten matches were found using approximate nearest neighbors Arya et al. (1998). Matches that had at least one class in common with the query were considered correct.

Table 4 compares the performance of different methods. MM-NN outperforms other approaches in all quality criteria. Figure 5 shows examples of top matches using MM-NN.

	BoF-Tags	Tags-BoF	BoF	Tags
Raw	—	—	57.0%	78.9%
MM-NN	64.3%	55.00%	79.39%	87.41%
CM-NN	61.1%	51.3%	75.62%	86.23%
CM-SSH	53.7%	50.2%	54.0%	76.0%

Table 4: mAP of different methods for the NUS experiment. Hashes of length 64 were used.

Figure 6 shows retrieval results using as queries artificially created Tag vectors containing specific words such as “cloud”. These tags are hashed using η and matched to BoFs hashed using ξ . The retrieved results are meaningful and most of them belong to the same class. It is especially interesting to note that MM-NN, apart from the Labels in the ground-truth which are in general noisy, produced relevant results in both cases. Figure 7 shows image annotation results. We retrieve the top five Tags matches from a BoF query and assign the ten most frequent annotations to the image. We clearly see that MM-NN produces better annotations than CM-SSH also in this case.

Wiki In the third experiment, we reproduced the results of Rasiwasia et al. (2010) using the dataset of 2866 annotated images from Wikipedia. The images are categorized in 10 classes and represented as 128-dimensional bags of SIFT features (Image modality) and 10-dimensional LDA topic model (Text modality). The dataset was split into disjoint subsets of 2173 and 693 for training and testing, respectively. We used positive and negative sets of size $|\mathcal{P}| = 1 \times 10^4, |\mathcal{N}| = 1 \times 10^5$. Table 5 shows the mAP for the Image–Text and Text–Image cross-modal retrieval experiment. For reference, we also reproduce the results reported in Rasiwasia et al. (2010) using correlation matching (CM), semantic matching (SM), and semantic correlation matching (SCM). MM-NN slightly outperforms SCM on average. We should stress however that these results are not directly comparable with ours: while Rasiwasia et al. (2010) find a Euclidean embedding, we use Hamming embedding (in general, a more difficult problem). While having similar performance to SCM, the significant advantage of

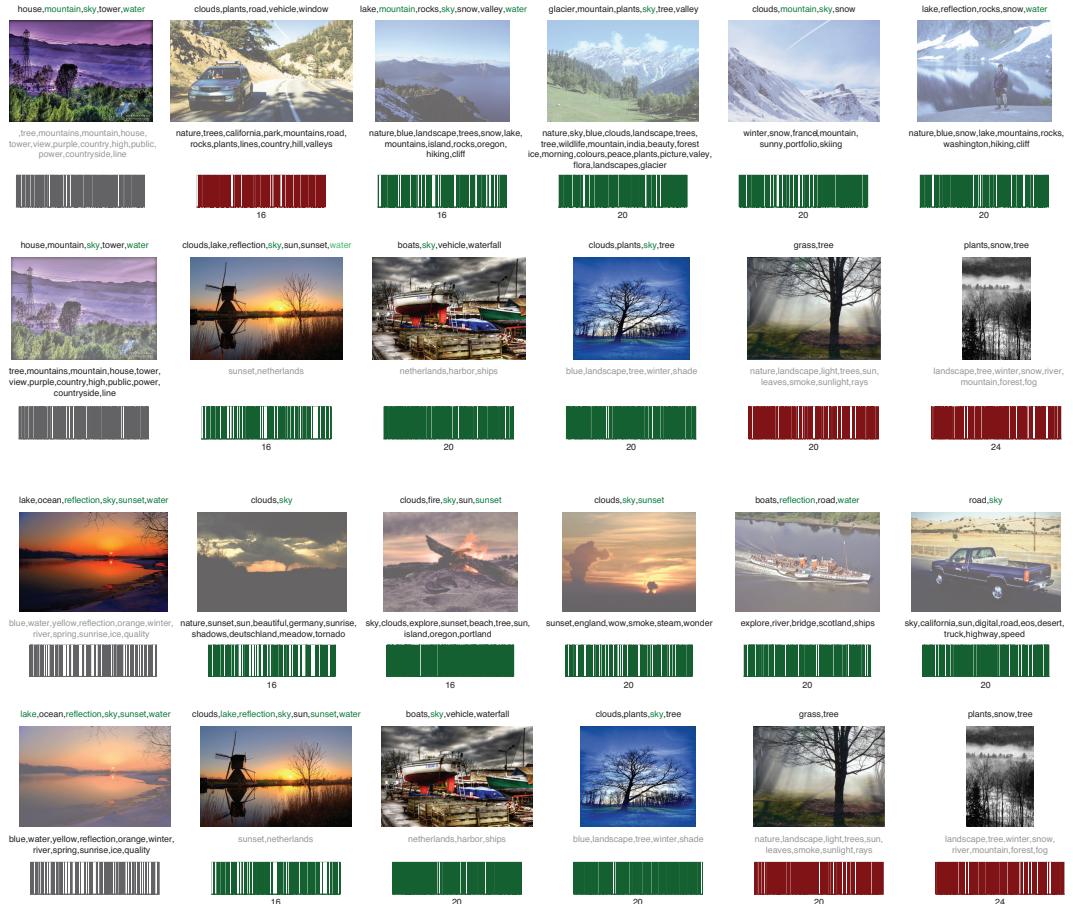


Figure 5: Top 5 matches for MM-NNhash in BoF-Tags (rows 1-5) and Tags-BoF (rows 6-10) cross-modal retrieval. Rows 1, 6: classes; rows 2, 7: images; rows 3, 8: tags (subdued implies the modality was not used); rows 4, 9: resulting hashes; rows 5, 10: Hamming distance from query.



Figure 6: Example of text-based image retrieval on NUS dataset using multimodal hashing. Shown are top five image matches produced by CM-SSH (top) and MM-NN (bottom) in response to two different queries: *cloud* (left) and *red+food* (right). Relevant matches are shown in green.



Figure 7: Example of image annotation on NUS dataset using multimodal hashing. Shown are tags returned for image query using CM-SSH (top) and MM-NN (bottom). Groundtruth tags are shown in green; synonyms are italicized.

our approach is that it produces much smaller compact binary codes (at least $10\times$ smaller) that can be searched very efficiently.

		Image-Text	Text-Image	Average
MM-NN	L1	27.8%	21.2%	24.5%
	L2	28.5%	22.0%	25.3%
CM-NN	L1	26.7%	20.9%	23.8%
	L2	27.1%	21.1%	24.1%
CM-SSH		22.2%	18.4%	20.3%
CM*		24.9%	19.6%	22.3%
SM*		22.5%	22.3%	22.4%
SCM*		27.7%	22.6%	25.2%

Table 5: Performance of different methods in the cross-modal retrieval experiments (Image and Text modalities) on the Wiki dataset. 32-bit hashes were used. Results marked with * are from Rasiwasia et al. (2010) based on Euclidean embedding; these results are not directly comparable with hashing and are brought here for reference only.

5 Conclusions

We introduced a novel learning framework for multimodal similarity-preserving hashing based on the coupled siamese neural network architecture. Our approach is free from assuming linear projections unlike existing cross-modal similarity learning methods; in fact, by increasing the number of layers in the network, mappings of arbitrary complexity can be trained (our experiments showed that using multilayer architecture results in a significant improvement of performance). We also solve the exact optimization problem during training making no approximations like the boosting-based CM-SSH. Our method does not involve semidefinite programming, and is scalable to a very large number of dimensions and training samples. Experimental results on standard multimedia retrieval datasets showed performance superior to state-of-the-art hashing approaches.

References

- Arya, S., Mount, D. M., Netanyahu, N. S., Silverman, R., and Wu, A. Y. An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *J. ACM*, 45(6):891–923, 1998.
- Bach, F. R., Lanckriet, G. R. G., and Jordan, M. I. Multiple kernel learning, conic duality, and the SMO algorithm. In *Proc. ICML*, 2004.
- Belkin, M. and Niyogi, P. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.
- Borg, I. and Groenen, P. J. F. *Modern multidimensional scaling: Theory and applications*. Springer, 2005.
- Bronstein, A. M., Bronstein, M. M., Guibas, L. J., and Ovsjanikov, M. Shape google: Geometric words and expressions for invariant shape retrieval. *TOG*, 30(1):1–20, 2011.
- Bronstein, M. M., Bronstein, A. M., Michel, F., and Paragios, N. Data fusion through cross-modality metric learning using similarity-sensitive hashing. In *Proc. CVPR*, 2010.
- Chua, T.-S., Tang, J., Hong, R., Li, H., Luo, Z., and Zheng, Y.-T. Nus-wide: A real-world web image database from national university of singapore. In *Proc. CIVR*, 2009.
- Coifman, R. R. and Lafon, S. Diffusion maps. *App. Comp. Harmonic Analysis*, 21(1):5–30, 2006.
- Davis, J. V., Kulis, B., Jain, P., Sra, S., and Dhillon, I. S. Information-theoretic metric learning. In *Proc. ICML*, 2007.
- Gionis, A., Indyk, P., and Motwani, R. Similarity search in high dimensions via hashing. In *Proc. Conf. Very Large Data Bases*, 1999.
- Hadsell, R., Chopra, S., and LeCun, Y. Dimensionality reduction by learning an invariant mapping. In *Proc. CVPR*, 2006.
- Johnson, R. A. and Wichern, D. W. *Applied multivariate statistical analysis*, volume 4. Prentice Hall, 2002.

- LeCun, Y. Une procédure d'apprentissage pour réseau à seuil asymétrique. *Proceedings of Cognitiva 85, Paris*, pp. 599–604, 1985.
- Lee, D., Hofmann, M., Steinke, F., Altun, Y., Cahill, N. D., and Scholkopf, B. Learning similarity measure for multi-modal 3d image registration. In *Proc. CVPR*, 2009.
- Masci, J., Migliore, D., Bronstein, M. M., and Schmidhuber, J. Descriptor learning for omnidirectional image matching. Technical Report arXiv:1112.6291, 2011.
- McFee, B. and Lanckriet, G. R. G. Partial order embedding with multiple kernels. In *Proc. ICML*, 2009.
- McFee, B. and Lanckriet, G. R. G. Learning multi-modal similarity. *JMLR*, 12:491–523, 2011.
- Mika, S., Ratsch, G., Weston, J., Schoelkopf, B., and Mueller, K. R. Fisher discriminant analysis with kernels. In *Proc. Neural Networks for Signal Processing*, 1999.
- Rasiwasia, N., Costa Pereira, J., Coviello, E., Doyle, G., Lanckriet, G.R.G., Levy, R., and Vasconcelos, N. A new approach to cross-modal multimedia retrieval. In *Proc. ICM*, pp. 251–260, 2010.
- Roweis, S. T. and Saul, L. K. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323, 2000.
- Rumelhart, D. E., Hinton, G. E., and Williams, R. J. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.
- Schmidhuber, J. and Prelinger, D. Discovering predictable classifications. *Neural Computation*, 5 (4):625–635, 1993.
- Schoelkopf, B., Smola, A., and Mueller, K. R. Kernel principal component analysis. *Artificial Neural Networks*, pp. 583–588, 1997.
- Shakhnarovich, G., Viola, P., and Darrell, T. Fast pose estimation with parameter-sensitive hashing. In *Proc. CVPR*, 2003.
- Shen, C., Kim, J., Wang, L., and Hengel, A. Positive semidefinite metric learning with boosting. In *Proc. NIPS*, 2009.
- Strecha, C., Bronstein, A. M., Bronstein, M. M., and Fua, P. LDAHash: Improved matching with smaller descriptors. *PAMI*, 34(1):66–78, 2012.
- Taylor, G. W., Spiro, I., Bregler, C., and Fergus, R. Learning invariance through imitation. In *Proc. CVPR*, 2011.
- Torralba, A., Fergus, R., and Weiss, Y. Small codes and large image databases for recognition. In *Proc. CVPR*, 2008.
- Weinberger, K. Q. and Saul, L. K. Distance metric learning for large margin nearest neighbor classification. *JMLR*, 10:207–244, 2009.
- Weiss, Y., Torralba, A., and Fergus, R. Spectral hashing. In *Proc. NIPS*, 2008.
- Werbos, P. J. *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. PhD thesis, Harvard University, 1974.
- Weston, J., Bengio, S., and Usunier, N. Large scale image annotation: learning to rank with joint word-image embeddings. *Machine learning*, 81(1):21–35, 2010.
- Xing, E. P., Ng, A. Y., Jordan, M. I., and Russell, S. Distance metric learning with application to clustering with side-information. In *Proc. NIPS*, 2002.