

A Multi-View Embedding Space for Modeling Internet Images, Tags, and their Semantics

Yunchao Gong · Qifa Ke · Michael Isard · Svetlana Lazebnik

Abstract This paper investigates the problem of modeling Internet images and associated text or tags for tasks such as image-to-image search, tag-to-image search, and image-to-tag search (image annotation). We start with *canonical correlation analysis (CCA)*, a popular and successful approach for mapping visual and textual features to the same latent space, and incorporate a third view capturing high-level image semantics, represented either by a single category or multiple non-mutually-exclusive concepts. We present two ways to train the three-view embedding: supervised, with the third view coming from ground-truth labels or search keywords; and unsupervised, with semantic themes automatically obtained by clustering the tags. To ensure high accuracy for retrieval tasks while keeping the learning process scalable, we combine multiple strong visual features and use explicit nonlinear kernel mappings to efficiently approximate kernel CCA. To perform retrieval, we use a specially designed similarity function in the embedded space, which substantially outperforms the Euclidean distance. The resulting system produces compelling qualitative results and outperforms a number of two-view baselines on retrieval tasks on three large-scale Internet image datasets.

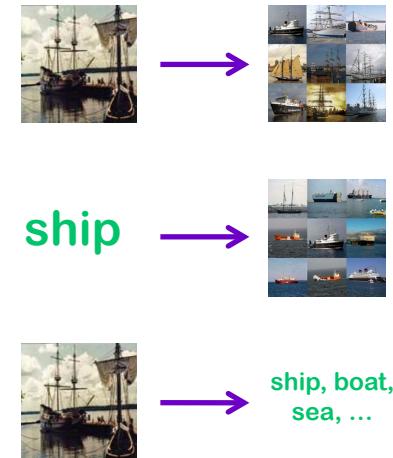


Fig. 1 Retrieval scenarios considered in this paper. Top: Given a query image, retrieve similar images from the database. Middle: Given a search keyword or tag, retrieve relevant images. Bottom: Given a query image, retrieve keywords or tags describing this image (automatic image annotation).

1 Introduction

The goal of this work is modeling the statistics of images and associated textual data in large-scale Internet photo collections in order to enable a variety of retrieval scenarios: similarity-based image search, keyword-based image search, and automatic image annotation (Figure 1). Practical models for these tasks must meet several requirements. First, they must be *accurate*, which is a big challenge given that the imagery is extremely heterogeneous and user-provided annotations are noisy. Second, they must be *scalable* to millions of images. Third, they must be *flexible*, accommodating *cross-modal* retrieval tasks such as tag-to-image or image-to-tag search in the same framework, and enabling, for example, tag-based search of images without any tags.

Y. Gong
Department of Computer Science
University of North Carolina at Chapel Hill
E-mail: yunchao@cs.unc.edu
Project website: <http://www.unc.edu/~yunchao/crossmodal.htm>

Q. Ke, M. Isard
Microsoft Research Silicon Valley, Mountain View, CA
E-mail: qke@microsoft.com, misard@microsoft.com

S. Lazebnik
Department of Computer Science
University of Illinois at Urbana-Champaign
E-mail: slazebni@illinois.edu

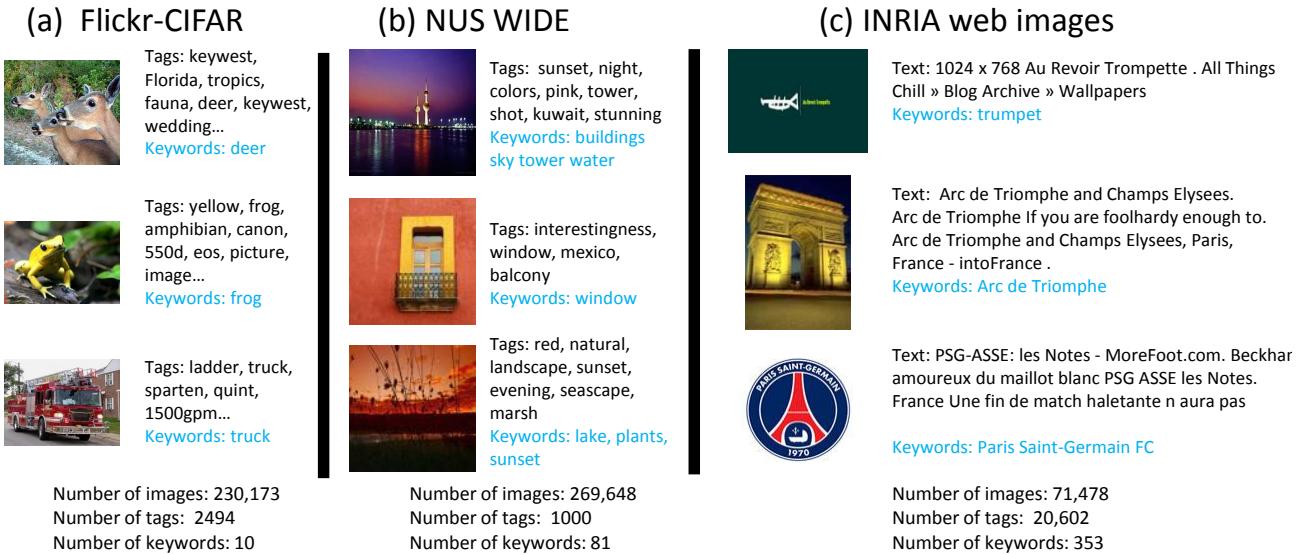


Fig. 2 An overview of the Internet image datasets used in this paper. Each image has three views associated with it: the visual features; the text or tags; and the semantics or ground-truth keywords. For Flickr-CIFAR dataset (collected by ourselves as described in Section 5) and INRIA-Websearch dataset (Krapac *et al.*, 2010), each image only has one ground-truth keyword. For the NUS-WIDE dataset (Chua *et al.*, 2009), each image has multiple keywords.

Several promising recent approaches for modeling images and associated text (Gong and Lazebnik, 2011; Hardoon *et al.*, 2004; Hwang and Grauman, 2010, 2011; Rasiwasia *et al.*, 2010) rely on *canonical correlation analysis* (CCA), a classic technique that maps two views, given by visual and textual features, into a common latent space where the correlation between the two views is maximized (Hotelling, 1936). This space is *cross-modal*, in the sense that embedded vectors representing visual and textual information are treated as the same class of citizens, and thus image-to-image, text-to-image, and image-to-text retrieval tasks can in principle all be handled in exactly the same way.

While CCA is very attractive in its simplicity and flexibility, existing CCA-based approaches have several shortcomings. In particular, the works cited above use classic two-view CCA, which only considers the direct correlations between images and corresponding textual feature vectors. However, as we will show in this paper, significant improvements can be obtained by considering a third view with which the first two are correlated – that of the underlying semantics of the image.

In this work, we use the term “semantics” to refer to high-level labels or topics that characterize the content of an image for the sake of a given application. For concrete examples, consider Figure 2, which illustrates the three-view datasets used in our experiments. In these datasets, the semantic view of an image consists of one or more ground-truth *keywords*. Even though our approach does not rely on a probabilistic generative model, we can think of the other two views, i.e., *visual features* and *tags/text*, as being stochastically generated based on the keywords. In particular, the tags

tend to come from a larger vocabulary than the keywords and they tend to be noisier. As in Figure 2 (a), the semantics of an image may be given by a single object category (“deer”), while the user-provided tags may include a number of additional terms correlated with that category (“keywest, Florida, tropics, fauna, wedding” etc.). Alternatively, the semantics might be given by multiple keywords corresponding to objects, scene types, or attributes. Thus, as in Figure 2 (b), an image may be annotated by multiple ground-truth keywords “buildings, sky, tower, water” and tags “sunset, night, colors, pink, tower, shot, kuwait, stunning.” Or, as in Figure 2 (c), the semantics may be given by the name of a logo or landmark, and the text may be taken from the surrounding webpage, and may or may not explicitly mention the ground-truth keyword.

In this paper, we present a three-view CCA model that explicitly incorporates the high-level semantic information as a third view. The difference between the standard two-view CCA and our proposed three-view embedding is visualized in Figure 3. In the two-view embedding space (Figure 3 (a)), which is produced by maximizing the correlations between visual features and the corresponding tag features, images from different classes are very mixed. On the other hand, the three-view embedding (Figure 3 (b)) provides a much better separation between the classes. As our experiments will confirm, a third semantic view – which may be derived from a variety of sources – is capable of considerably increasing the accuracy of retrieval on very diverse datasets.

In all the examples of Figure 2, the ground-truth semantic keywords are defined ahead of time and accurately antic-

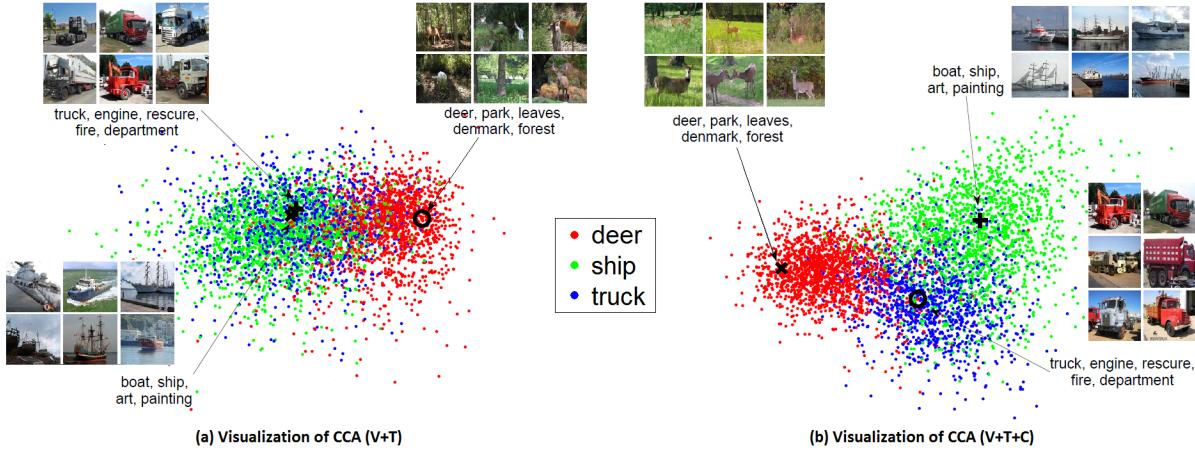


Fig. 3 A visualization of the first two directions of the common latent space for (a) standard two-view CCA and (b) our proposed three-view CCA model. Different colors indicate different image categories (though note that category information is not used in learning the three-view embedding). Black points indicate sample tag queries, and the corresponding images are their nearest neighbors in the latent space.

notated for the express purpose of training recognition algorithms. However, in most realistic situations, it is easy to gather noisy text and tags, but not so easy to get at the underlying semantics. Fortunately, we will show that even in cases when clean ground-truth annotation for the third view is unavailable, it is still possible to learn a better embedding for the photo collection by representing the semantics explicitly. In some cases, we can get an informative additional signal from search keywords. For example, if we retrieve a number of images together with their tags from Flickr using a search for “frog,” then knowing the original search keyword gives us additional modeling power even if many of these images do not actually depict frogs. Furthermore, if ground truth category or search keyword information is absent completely, we will demonstrate that an effective third view can be derived in an unsupervised way by clustering the noisy tag vectors constituting the second view. This approach is inspired by cluster-based information retrieval (Wei *et al.*, 2011) and the “cluster assumption” in semi-supervised learning (Chapelle *et al.*, 2003). In effect, the tag clustering can be thought of as “reconstructing” or “recovering” the absent topics or distinct types of image content.

To obtain high retrieval accuracy, most modern methods have found it necessary to combine multiple high-dimensional visual features, each of which may come with a different similarity or kernel function. Retrieval approaches of Hwang and Grauman (2010, 2011); Yakhnenko and Honavar (2009) accomplish this combination using nonlinear kernel CCA (KCCA) (Bach and Jordan, 2002; Hardoon *et al.*, 2004), but the standard KCCA formulation scales *cubically* in the number of images in the dataset. Instead of KCCA, we use a scalable approximation scheme based on efficient explicit kernel mapping followed by linear dimensionality reduction

and linear CCA. Finally, we specifically design a similarity function suitable for our learned latent embedding, and show that it achieves significant improvement over the Euclidean distance. Experiments on the three large-scale datasets of Figure 2 show the promise of the proposed approach.

The following is a preview of the structure and main contributions of this paper:

- A novel *three-view* CCA framework that explicitly incorporates the dependence of visual features and text on the underlying image semantics (Section 3.1).
- A similarity function specially adapted to CCA that improves the accuracy of retrieval in the embedded space (Section 3.2).
- Scalable yet discriminative representations for the visual and textual views based on multiple feature combination, explicit kernel mappings, and linear dimensionality reduction (Sections 4.1 and 4.2).
- Two methods for instantiating the third semantic view: *supervised*, or derived from ground-truth annotations by unsupervised clustering; and *unsupervised*, or derived by clustering the tag vectors from the second (textual) view. In both cases, our experiments confirm that adding the third view helps to improve retrieval accuracy. For the unsupervised case, we perform a comparative evaluation of several tag clustering methods from the literature (Section 4.3).
- Extensive evaluation of the proposed three-view models on three tasks – image-to-image, tag-to-image, and image-to-tag search. Section 5 will give an overview of our experimental protocol, and Sections 6–8 will present results on the three large-scale datasets introduced in Figure 2.

2 Related Work

In the vision and multimedia communities, jointly modeling images and text has been an active research area. This section gives a non-exhaustive survey of several important lines of research related to our work.

Some of the earliest research on images and text (Barnard and Forsyth, 2001; Blei and Jordan, 2003; Blei *et al.*, 2003; Duygulu *et al.*, 2002; Lavrenko *et al.*, 2003) has focused on learning the co-occurrences between image regions and tags using a generative model. Since most datasets used for training such models lack image annotation at the region level, learning to associate tags with image regions is a very challenging problem, especially for contaminated Internet photo collections with very large tag vocabularies. Moreover, image tags frequently refer to global properties or characteristics that cannot be easily localized. Therefore, we focus on establishing relationships between whole images and words.

Conceptually, our three-view formulation may be compared to the generative model that attempts to capture the relationships between the image class, annotation tags, and image features. One example of such a model in the literature is Wang *et al.* (2009a). Unlike Wang *et al.* (2009a), though, we do not concern ourselves with the exact generative nature of the dependencies between the three views, but simply assign symmetric roles to them and model the pairwise correlations between them. Also, while Wang *et al.* (2009a) tie annotation tags to image regions following Blei and Jordan (2003); Blei *et al.* (2003), we treat both the image appearance and all the tags assigned to the image as global feature vectors. This allows for much more scalable learning and inference (the approach of Wang *et al.* (2009a) is only tested on datasets of under 2,000 images and eight classes each).

The major goal of our work is learning a joint latent space for images and tags, in which corresponding images and tags are mapped to nearby locations, so that simple nearest-neighbor methods can be used to perform cross-modal tasks, including image-to-image, tag-to-image, and image-to-tag search. A number of successful recent approaches to learning such an embedding rely on Canonical Correlation Analysis (CCA) (Hotelling, 1936). Hardoon *et al.* (2004) and Rasiwasia *et al.* (2010) have applied CCA to map images and text to the same space for cross-modal retrieval tasks. Hwang and Grauman (2010, 2011) have presented a cross-modal retrieval approach that models the relative importance of words based on the order in which they appear in user-provided annotations. Blaschko and Lampert (2008) have used KCCA to develop a cross-view spectral clustering approach that can be applied to images and associated text. CCA embeddings have also been used in other domains, such as cross-language retrieval (Udupa and Khapra, 2010; Vinokourov *et al.*, 2002). Unlike all the other CCA-based

image retrieval and annotation approaches, ours adds a third view that explicitly represents the latent image semantics.

Our approach also has connections to supervised *multi-view learning*, in which images are characterized by visual and textual views, both of which are linked to the underlying semantic labels. The literature contains a number of sophisticated methods for multi-view learning, including generalizations of CCA/KCCA (Rai and Daumé, 2009; Sharma *et al.*, 2012; Yakhnenko and Honavar, 2009), metric learning (Quadrianto and Lampert, 2011) and large-margin formulations (Chen *et al.*, 2012). Fortunately, we have found that our basic CCA formulation already gives very promising results without having to pay the price of increased complexity for learning and inference.

Since learning a projection for the data is equivalent to learning a Mahalanobis metric in the original feature space, our work is related to *metric learning* (Globerson *et al.*, 2005; Goldberg *et al.*, 2004; Weinberger *et al.*, 2005). For example, the large-margin nearest neighbor (LMNN) approach (Weinberger *et al.*, 2005) learns a distance metric that is optimized for nearest neighbor classification, and neighborhood component analysis (NCA) (Goldberg *et al.*, 2004) optimizes leave-one-out loss for nearest neighbor classification. Metric learning has been used for image classification and annotation (Guillaumin *et al.*, 2009; Mensink *et al.*, 2012; Verma and Jawahar, 2012). However, all of these approaches learn an embedding or a metric for visual features only, so they cannot be used to perform cross-modal retrieval.

The two main tasks we use for evaluating our system are image-to-image search, which has been traditionally studied as *content-based image retrieval* (Datta *et al.*, 2008; Smeulders *et al.*, 2000), and tag-to-image search, or image retrieval using text-based queries (Grangier and Bengio, 2008; Krpac *et al.*, 2010; Liu *et al.*, 2009; Lucchi and Weston, 2012). A task related to tag-to-image search, though one we do not consider directly, is re-ranking of contaminated image search results for the purpose of dataset collection (Berg and Forsyth, 2006; Fan *et al.*, 2010; Frankel *et al.*, 1997; Schroff *et al.*, 2007).

The third task we are interested in evaluating is image-to-tag search or automatic image annotation (Carneiro *et al.*, 2007; Li and Wang, 2008; Monay and Gatica-Perez, 2004). This task has traditionally been addressed with the help of sophisticated generative models such as Blei and Jordan (2003); Carneiro *et al.* (2007); Lavrenko *et al.* (2003). More recently, a number of publications have reported better results with simple data-driven schemes based on retrieving database images similar to a query and transferring the annotations from those images (Chua *et al.*, 2009; Guillaumin *et al.*, 2009; Makadia *et al.*, 2008; Verma and Jawahar, 2012; Wang *et al.*, 2008). We will adopt this strategy in our experiments and

demonstrate that retrieving similar images in our embedded latent space can improve the accuracy of tag transfer.

The data-driven image annotation approaches of Guillaumin *et al.* (2009); Makadia *et al.* (2008); Verma and Jawahar (2012) use discriminative learning to obtain a metric or a weighting of different features to improve the relevance of database images retrieved for a query. Unfortunately, the learning stage is very computationally expensive – for example, in the TagProp method of Guillaumin *et al.* (2009), it scales quadratically with the number of images. In fact, the standard datasets used for image annotation by Makadia *et al.* (2008); Guillaumin *et al.* (2009); Verma and Jawahar (2012) consist of 5K-20K images and have 260-290 tags each. By contrast, our datasets (shown in Figure 2) range in size from 71K to 270K and have tag vocabularies of size 1K-20K. While it is possible to develop scalable metric learning algorithms using stochastic gradient descent (e.g., Mensink *et al.* (2012)), our work shows that learning a linear embedding using CCA can serve as a simpler attractive alternative.

Perhaps the largest-scale image annotation system in the literature is the *Wsabie* (Web Scale Annotation by Image Embedding) system by Weston *et al.* (2011). It uses stochastic gradient descent to optimize a ranking objective function and is evaluated on datasets with ten million training examples. Like our approach, *Wsabie* learns a common embedding for visual and tag features. Unlike ours, however, it has only a two-view model and thus does not explicitly represent the distinction between the tags used to describe the image and the underlying image content. Also, *Wsabie* is not explicitly designed for multi-label annotation, and evaluated on datasets whose images come with single labels (or single paths in a label hierarchy).

One of the shortcomings of data-driven annotation approaches (Guillaumin *et al.*, 2009; Makadia *et al.*, 2008; Verma and Jawahar, 2012) as well as *Wsabie* is that they not account for co-occurrence and mutual exclusion constraints between different tags for the same image. If the retrieved nearest neighbors of an image belong to incompatible semantic categories (e.g., “bird” and “plane”), then the tags transferred from them to the query may be incoherent as well (see Figure 13 (a) for an example). To better exploit constraints between multiple tags, it is possible to treat image annotation as a multi-label classification problem (Chen *et al.*, 2011; Zhu *et al.*, 2005). In the present work, we limit ourselves to learning the joint visual-textual embedding. It would be interesting to impose multi-label prediction constraints in the joint latent space – in fact, Zhang and Schneider (2011) have recently proposed an approach combining CCA with multi-label decoding – but doing so is outside the scope of our paper.

Finally, our work has connections to approaches that use Internet images and accompanying text as auxiliary training data to improve performance on tasks such as image clas-

sification, for which cleanly labeled training data may be scarce (Guillaumin *et al.*, 2010; Quattoni *et al.*, 2007; Wang *et al.*, 2009b). In particular, Quattoni *et al.* (2007) use the multi-task learning framework of Ando and Zhang (2005) to learn a discriminative latent space from Web images and associated captions. We will use this embedding method as one of our baselines, though, unlike our approach, it can only be applied to images, not to tag vectors. Apart from multi-task learning, another popular way to obtain an intermediate embedding space for images is by mapping them to outputs of a bank of concept or attribute classifiers (Rasiwasia and Vasconcelos, 2007; Wang *et al.*, 2009c). Once again, unlike our method, this produces an embedding for images only; also, training of a large number of concept classifiers tends to require more supervision and be more computationally intensive than training of a CCA model.

3 Modeling Images, Tags, and High-Level Semantics

3.1 Scalable three-view CCA formulation

In this section, we introduce a three-view kernel CCA formulation for learning a joint space for visual, textual, and semantic information. Then we show how to obtain a scalable approximation using explicit kernel embeddings and linear CCA.

We assume we have n training images each of which is associated with a v -dimensional visual feature vector and a t -dimensional tag feature vector (our specific feature representations for both views will be discussed in Section 4). The respective vectors are stacked as rows in matrices $V \in \mathbb{R}^{n \times v}$ and $T \in \mathbb{R}^{n \times t}$. In addition, each training image is also associated with semantic class or topic information, which is encoded in a matrix $C \in \mathbb{R}^{n \times c}$, where c is the number of classes or topics. Each image may be labeled with exactly one of the c classes (in which case only one entry in each row of C is 1 and the rest are 0); alternatively, each image may be described by several of the c keywords (in which case, multiple entries in each row of C may be 1). Another possibility is that C is a soft indication matrix, where the i, j th entry indicates the degree (or posterior probability) with which image i belongs to the j th class or topic. In the supervised learning scenario, C is obtained from (possibly noisy) annotations that come with the training data. In the unsupervised scenario (where only images and tags are initially given), C is “latent” and must be obtained by clustering the tags, as will be discussed in Section 4.3. To simplify the notation in the following, we will also use X_1, X_2, X_3 to denote V, T, C respectively.

Let \mathbf{x}, \mathbf{y} denote two points from the i th view. The similarity between these points is defined by a kernel function K_i such that $K_i(\mathbf{x}, \mathbf{y}) = \varphi_i(\mathbf{x})\varphi_i(\mathbf{y})^\top$, where $\varphi_i(\cdot)$ is a

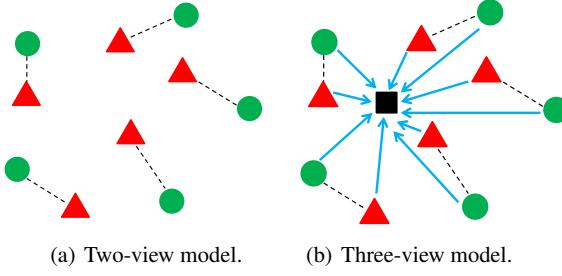


Fig. 4 (a) Traditional two-view CCA minimizes the distance (equivalently, maximizes the correlation) between images (triangles) and their corresponding tags (circles). (b) Our proposed approach is to incorporate semantic classes or topics (black squares) as a third view. Images and tags belonging to the same semantic cluster are forced to be close to each other, imposing additional high-level structure. See also Figure 3 for a visualization of two embeddings on real data.

function embedding the original feature vector into a nonlinear kernel space. Practical kernel-based learning schemes do not work in the embedded space directly, relying on the kernel function instead. However, we will formulate KCCA as solving for a linear projection from the kernel space, because this leads directly to our scalable approximation scheme based on explicit embeddings.

In KCCA, we want to find matrices W_i that project the embedded vectors $\varphi_i(\mathbf{x})$ from each view into a low-dimensional common space such that the distances in the resulting space between each pair of views for the same data item are minimized. The objective function for this formulation is given by

$$\min_{W_1, W_2, W_3} \sum_{i,j=1}^3 \|\varphi_i(X_i)W_i - \varphi_j(X_j)W_j\|_F^2 \quad (1)$$

$$\text{subject to } W_i^\top \Sigma_{ii} W_i = I, \quad w_{ik}^\top \Sigma_{ij} w_{jl} = 0, \\ i, j = 1, \dots, 3, \quad i \neq j, \quad k, l = 1, \dots, d, \quad k \neq l,$$

where Σ_{ij} is the covariance matrix between $\varphi(X_i)$ and $\varphi(X_j)$, and w_{ik} is the k th column of W_i (the number of columns in each W_i is equal to the dimensionality of the resulting common space). To better understand this objective function, let us consider its three terms:

$$\min_{W_1, W_2, W_3} \|\varphi_1(V)W_1 - \varphi_2(T)W_2\|_F^2 + \\ \|\varphi_1(V)W_1 - \varphi_3(C)W_3\|_F^2 + \|\varphi_2(T)W_2 - \varphi_3(C)W_3\|_F^2.$$

The first term tries to align corresponding images and tags, and it is the sole term in the standard two-view CCA objective (Hardoon *et al.*, 2004). The remaining two terms, which are introduced in our three-view model, try to align images (resp. tags) with their semantic topic. Figure 4 illustrates the difference between the two- and three-view formulations graphically.

In the standard KCCA formulation, instead of directly solving for linear projections of data explicitly mapped into the kernel space by φ_i , one applies the ‘‘kernel trick’’ and

expresses the coordinates of a data point in the CCA space as linear combinations of kernel values of that point and several training points. To find the weights in this combination, one must solve a $3n \times 3n$ generalized eigenvalue problem (see Bach and Jordan (2002); Hardoon *et al.* (2004) for details), which is infeasible for large-scale data.

To handle large numbers of images and high-dimensional features, we propose a scalable approach based on the idea of approximate kernel maps (Maji and Berg, 2009; Perronnin *et al.*, 2010; Rahimi and Recht, 2007; Vedaldi and Zisserman, 2010). Let $\hat{\varphi}(\mathbf{x})$ denote an approximate kernel mapping such that $K_i(\mathbf{x}, \mathbf{x}') \simeq \hat{\varphi}_i(\mathbf{x})\hat{\varphi}_i(\mathbf{x}')^\top$. The dimensionality of $\hat{\varphi}(\mathbf{x})$ needs to be much lower than n to reduce the complexity of the problem. The specific kernel mappings used in our implementation will be described in Section 4.1. Then, instead of using the kernel trick, we can directly substitute $\hat{\varphi}(\mathbf{x})$ into the linear CCA objective function (1). The solution is given by the following generalized eigenvalue problem:

$$\begin{pmatrix} S_{11} & S_{12} & S_{13} \\ S_{21} & S_{22} & S_{23} \\ S_{31} & S_{32} & S_{33} \end{pmatrix} \begin{pmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \mathbf{w}_3 \end{pmatrix} = \lambda \begin{pmatrix} S_{11} & 0 & 0 \\ 0 & S_{22} & 0 \\ 0 & 0 & S_{33} \end{pmatrix} \begin{pmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \mathbf{w}_3 \end{pmatrix},$$

where $S_{ij} = \hat{\varphi}_i(X_i)^\top \hat{\varphi}_j(X_j)$ is the covariance matrix between the i th and j th views, and \mathbf{w}_i is a column of W_i . The size of this problem is $(d_1 + d_2 + d_3) \times (d_1 + d_2 + d_3)$, where the d_i are the dimensionalities of the respective explicit mappings $\hat{\varphi}_i(\cdot)$. This is independent of training set size, and much smaller than $3n \times 3n$. To regularize the problem, we add a small constant (10^{-4} in the experiments) to the diagonal of the covariance matrix.

In order to obtain a d -dimensional embedding for different views, we form projection matrices $W_i \in \mathbb{R}^{d_i \times d}$ from the top d eigenvectors corresponding to each \mathbf{w}_i . Then the projection of a data point \mathbf{x} from the i th view into the latent CCA space is given by $\hat{\varphi}_i(\mathbf{x})W_i$. Note that once they are learned, the respective projection matrices are applied to each view individually, which means that at test time, we can compute the embedding for data for which one or two views are missing (e.g., an image without tags or ground-truth semantic labels). In the latent CCA space, points from different views are directly comparable, so we can do image-to-image, image-to-tag, and tag-to-image retrieval by nearest neighbor search.

In the implementation, we select the embedding dimensionality d by measuring the retrieval accuracy in embedded spaces with different values of d on validation images set aside from each of our datasets (details will be given in Sections 6-8). We search a range from 16 to 1,024, doubling the dimensionality each time, and the resulting values typically fall around 128-256 on all our datasets.

3.2 Similarity function in the latent space

In the CCA-projected latent space, the function used to measure the similarity between data points is important. An obvious choice is the Euclidean distance between embedded data points, as used in Foster *et al.* (2010); Hwang and Grauman (2010); Rasiwasia *et al.* (2010). However, for our learned embedding, we were able to find a similarity function that produces better empirical results. In particular, we scale the dimensions in the common latent space by the magnitude of the corresponding eigenvalues (Chapelle *et al.*, 2003), and then compute normalized correlation between projected vectors. Indeed, the CCA objective can be reformulated as maximizing the normalized correlation between different views (Hardoon *et al.*, 2004).

Let \mathbf{x} and \mathbf{y} be points from the i th and j th views, respectively (we can have $i = j$). Then we define the similarity function between \mathbf{x} and \mathbf{y} as

$$\text{sim}(\mathbf{x}, \mathbf{y}) = \frac{(\hat{\varphi}_i(\mathbf{x})W_iD_i)(\hat{\varphi}_j(\mathbf{y})W_jD_j)^\top}{\|\hat{\varphi}_i(\mathbf{x})W_iD_i\|_2\|\hat{\varphi}_j(\mathbf{y})W_jD_j\|_2}, \quad (2)$$

where W_i and W_j are the CCA projections for data points \mathbf{x} and \mathbf{y} , and D_i and D_j are diagonal matrices whose diagonal elements are given by the p -th power of the corresponding eigenvalues (Chapelle *et al.*, 2003). We fix $p = 4$ in all our experiments as we have found this leads to the best performance. Section 6.4 will experimentally confirm that the above similarity measure leads to much higher retrieval accuracy than Euclidean distance.

4 Representations of the Three Views

In Sections 4.1 and 4.2, we will present our visual and text features with their respective kernel mappings. Next, in Section 4.3, we will discuss different text clustering approaches that can be used to extract semantic topics in the unsupervised scenario, where the third view is not given for the training data.

4.1 Visual feature representation

We represent image appearance using a combination of nine different visual cues:

GIST (Oliva and Torralba, 2001): We resize each image to 200×200 and use three different scales [8, 8, 4] to filter each RGB channel, resulting in 960-dimensional (320×3) GIST feature vectors.

SIFT: We extract six different texture features based on two different patch sampling schemes: dense sampling and Harris corner detection. For each local patch, we extract **SIFT** (Lowe, 2004), **CSIFT** (van de Sande *et al.*, 2010), and **RGSIFT** (van de Sande *et al.*, 2010). For each feature, we

form a codebook of size 1,000 using k-means clustering and build a two-level spatial pyramid (Lazebnik *et al.*, 2006), resulting in a 5000-dimensional vector. We will refer to these six features as D-SIFT, D-CSIFT, D-RGBSIFT, H-SIFT, H-CSIFT, and H-RGBSIFT.

HOG (Dalal and Triggs, 2005): To represent texture and edge information on a larger scale, we use 2×2 overlapping HOG as described in Xiao *et al.* (2010). We quantize the HOG features to a codebook of size 1,000 and use the same spatial pyramid scheme as above, once again resulting in 5,000-dimensional feature vectors.

Color: We use a joint RGB color histogram of 8 bins per dimension, for a 512-dimensional feature.

Recall from Section 3.1 that we transform all the features by nonlinear kernel maps $\hat{\varphi}(\mathbf{x})$ and then apply linear CCA to the result. We discuss the specific feature maps we use here. For GIST features, we use the random Fourier feature mapping (Rahimi and Recht, 2007) that approximates the Gaussian kernel. We compute this mapping with 3,000 dimensions and set its standard deviation equal to the average distance to the 50th nearest neighbor in each dataset. All the other descriptors above are histograms, and for them we adopt the exact Bhattacharyya kernel mapping given by term-wise square root (Perronnin *et al.*, 2010). To combine different features, we simply average the respective kernels, which has been proven to be quite effective in Gehler and Nowozin (2009). This corresponds to concatenating all the different visual features after putting them through their respective explicit kernel mappings. However, the resulting concatenated feature has 38,512 dimensions, necessitating additional dimensionality reduction. To do this, we perform PCA on top of each kernel-mapped feature $\hat{\varphi}_i(\cdot)$. This is essentially using the low-rank approximation of kernel PCA (KPCA) (Scholkopf *et al.*, 1997) to approximate the combined multiple feature kernel matrix.

In our experiments, we reduce each kernel-mapped feature to 500 dimensions and the final concatenated feature is a 4,500-dimensional vector. As validated in Section 6.2, this dimensionality achieves good balance between efficiency and accuracy. Note that for multiple feature combination, we have found it important to center all feature dimensions at the origin.

4.2 Tag feature representation

For the tags associated with the images, we construct a dictionary consisting of t most frequent tags (the vocabulary sizes used for the different datasets are summarized in Figure 2 and will be further detailed in Section 5) and manually remove a small set of stop words. These include camera brands (e.g., “canon,” “nikon,” etc.), lens characteristics (e.g., “eos,” “70-200mm,” etc.), and words like “geo.” The

tag feature matrix T is binary: $T_{ij} = 1$ if image i is tagged with tag j and 0 otherwise. Note that even though the dimensionality of the tag feature may be high (our vocabularies range from 1,000 to over 20,000 on the different datasets), this representation is highly sparse.

Like Guillaumin *et al.* (2010), we use the linear kernel for T , which corresponds to counting the number of common tags between two images. However, because of the high dimensionality of the tag features, additional compression is required, just as with the concatenated visual features. We apply sparse SVD (Larsen, 1998) to the tag feature T to obtain a low-rank decomposition as $T = U_1 S U_2^\top$. It is easy to show that $U_1 S$ is actually the PCA embedding for T , but directly applying sparse SVD to T is more efficient. In our implementation, the compressed representation of T is given by the top 500 columns of $U_1 S$.

We have also investigated more sophisticated tag features such as the ranking-based representation of Hwang and Grauman (2010), which is based on the idea that tags listed earlier by users are more salient to the image content. However, we have found almost no improvement from our chosen representation.

4.3 Semantic view representation

As initially discussed in Section 3, the third view of our CCA model is given by the class or topic indicator matrix $C \in \mathbb{R}^{n \times c}$ for n images and c topics. In the supervised training scenario, C is straightforwardly given by ground-truth annotations or noisy search keywords used to download the data. In the more interesting unsupervised scenario, training images come with noisy text or tags, but no additional semantic annotations. In this case, we choose to obtain C by clustering the tags. Given the raw tag feature T (*prior* to the application of sparse SVD), our goal is to find c semantic clusters. For this purpose, we investigate several models that have proven successful for text clustering. We briefly describe these models below; quantitative and qualitative evaluation results will be presented in Section 6.3.

K-means clustering: The simplest baseline approach is k-means clustering on raw tag feature T using c centers. The resulting matrix C has a 1 in the i, j th position if the i th tag feature vector belongs to the j th cluster.

Normalized cut (NC) (Ng *et al.*, 2001; Shi and Malik, 2000): For text clustering, the normalized cut model is usually formulated as computing the eigenvectors of

$$L = I - D^{-1/2} T T^\top D^{-1/2},$$

in which $D = \text{diag}(T(T^\top \mathbf{1}))$. This is equivalent to computing the first c singular vectors of the sparse matrix $D^{-1/2} T$. Following Ng *et al.* (2001), we normalize each row of the matrix of top c eigenvectors to have unit norm and perform

k-means clustering of rows of the resulting matrix \tilde{U} . Directly using \tilde{U} as C would represent a “soft” version of NC, but we have found that the “hard” version obtained by k-means produces better results.

Nonnegative matrix factorization (NMF) (Xu *et al.*, 2003): The data matrix is normalized as $D^{-1/2} T$, where D is defined the same way as for NC, and then factorized into two nonnegative matrices U and V such that $T = U^\top V$ (if T is $n \times t$, then U is $c \times n$ and V is $c \times t$). Then, as in Xu *et al.* (2003), we obtain a normalized matrix \tilde{U} with entries $U_{ij} / \sqrt{\sum_j V_{ij}^2}$. Finally, we do hard cluster assignment based on the highest value of \tilde{U} in each row. Just as with NC, this produces better results than using \tilde{U} as a “soft” cluster indicator matrix directly.

Probabilistic latent semantic analysis (pLSA) (Hofmann, 1999): This approach models each document (vector of tags for an image) as a mixture of topics. The output of pLSA is the posterior probability of each topic given each document. Directly using this matrix of posterior probabilities as C leads to “soft” pLSA clustering. However, once again, we get better performance with “hard” pLSA where we map each document to the cluster index with the highest posterior probability. We have also investigated latent Dirichlet allocation (LDA) (Blei *et al.*, 2003) and found the performance to be similar to pLSA, so we omit it.

Because the number of topics used in this work is not very high (from 10 to 100), we simply use a linear kernel on C with no further dimensionality reduction.

5 Overview of Experimental Evaluation

This section will present the components of our experimental evaluation, including datasets, retrieval tasks, multi-view models being compared, and baselines. Subsequently, Sections 6-8 will present results on our three datasets.

5.1 Datasets

Our selection of datasets is motivated by two considerations. First, we want datasets that are as large as possible, both in the number of images and in the number of tags. Second, we want datasets that have the right kind of annotations for evaluating our method – specifically, images that are accompanied both by noisy text or tags, and ground-truth labels.

We have considered a number of datasets used in recent related papers, but unfortunately, most of them are unsuitable for our goals. In particular, standard image annotation datasets used by Makadia *et al.* (2008); Guillaumin *et al.* (2009); Rasiwasia and Vasconcelos (2007); Verma and Jawahar (2012) – namely, Corel5K (Duygulu *et al.*, 2002), ESP Game (von Ahn and Dabbish, 2004), and IAPR-TC

(Grubinger *et al.*, 2006) – have only two views and are rather small-scale (5K-20K images and 260-290 tags). Rasiwasia *et al.* (2010), who have first proposed a two-view CCA model for cross-modal retrieval of Internet images, perform experiments on a Wikipedia dataset that has rich textual views as well as ground-truth labels, but it consists of only 2,866 documents. Weston *et al.* (2011) evaluate their Wsabie annotation system on millions of images. However, one of their datasets is drawn from ImageNet (Deng *et al.*, 2009), which is more appropriate for image classification, and the other one is proprietary; neither has the three-view structure we are looking for.

The three datasets we have chosen are shown in Figure 2. The first one is collected by ourselves, while the other two are publicly available.

Flickr-CIFAR dataset: We have downloaded 230,173 images from Flickr by running queries for categories from the CIFAR10 dataset (Krizhevsky, 2009): airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. We keep tags that appear at least 150 times, resulting in a tag dictionary with dimensionality 2,494. On average, there are 6.84 tags per image. The Flickr images come with search keywords and user-provided tags, but no ground-truth labels. To quantitatively evaluate retrieval performance we need another set of cleanly labeled test images. We get this set by collecting the same ten categories from ImageNet (Deng *et al.*, 2009), resulting in 15,167 test images with no tags but ground-truth class labels.

NUS-WIDE dataset: This dataset (Chua *et al.*, 2009) was collected at the National University of Singapore. It also originates from Flickr, and contains 269,648 images. The dataset is manually annotated with 81 ground truth concept labels, e.g., animal, snow, dog, reflection, city, storm, fog, etc. One important difference between NUS-WIDE and other datasets is that NUS-wide images may be associated with multiple ground truth labels. For the tags, we use the list of 1,000 words provided by Chua *et al.* (2009); on average, each image has 5.78 tags and 1.86 ground truth annotations. Each ground truth concept is also in the tag dictionary.

INRIA-Websearch dataset: Finally, we use the INRIA Web query dataset (Krapac *et al.*, 2010), which contains 71,478 web images and 353 different concepts or categories, which include famous landmarks, actors, films, logos, etc. Each concept comes with a number of images retrieved via Internet search, and each image is marked as either relevant or irrelevant to its query concept. This dataset is especially challenging in that it contains a very large number of concepts relative to the total number of images. The second view for this dataset consists of text surrounding images on web pages, not tags. We keep words that appear more than 20 times and remove stop words using a standard list for text document analysis, which gives us a tag dictionary of

size 20,602. On this dataset, we also apply *tf-idf* weighting to the tag feature.

The above three datasets have different characteristics and present different challenges for our method. Flickr-CIFAR has the fewest classes but the largest number of images per class. It is also the only dataset whose training images have no ground-truth semantic annotation, and whose test images come from a different distribution than the training images. We use this dataset for detailed comparative validation of the different implementation choices of our method (Section 6). NUS-WIDE images are fully manually annotated and come with multiple ground truth concepts per image. INRIA-Websearch is the only one not collected from Flickr, and its images are the most inconsistent in quality. It has the largest number of classes but the smallest number of images per class. It also has by far the largest vocabulary for the second view and the noisiest statistics for this view.

5.2 Retrieval tasks

For evaluation, we consider the following tasks.

Image-to-image search (I2I): Given a query image, project its visual feature vector into the CCA space, and use it to retrieve the most similar visual features from the database. Recall that our similarity function in the CCA space is given by eq. (2).

Tag-to-image search (T2I): Given a search tag or combination of tags, project the corresponding feature vector into the CCA space and retrieve the most similar database images. This is a cross-modal task, in that the CCA-embedded tag query is used to directly search CCA-embedded visual features. Note that with our method, we can use tags to search database images that do not initially come with any tags. In scenarios where ground-truth labels or keywords are available for the database images, we also consider a variant of this task where we use the keywords as queries, which we refer to as **keyword-to-image search (K2I)**.

Image-to-tag search (I2T): Given an image, retrieve a set of tags that accurately describe it. This task is more challenging than the other two because going from a feature vector in CCA space to a coherent set of tags requires a sophisticated reconstruction or decoding algorithm (see, e.g., Hsu *et al.* (2009); Zhang and Schneider (2011)). The design of such an algorithm is beyond the scope of our present work, but to get a preliminary idea of the promise of our latent space representation for this task, we evaluate a simple data-driven scheme similar to that of Makadia *et al.* (2008). Namely, given a query image, we first find the fifty nearest neighbor tag vectors in CCA space, and then return the five tags with the highest frequencies in the corresponding database images. Note that Makadia *et al.* (2008) return tags according to their global frequencies, while for our larger

and more diverse datasets, we have found that local frequency works better. Because our method for I2T is somewhat preliminary and because proper evaluation of this task requires human annotators (see Section 6.7), our experiments on this task will be smaller-scale and more limited than on the other two.

The precise evaluation protocols and performance metrics used for each task are dataset-specific, and will be described in Sections 6-8.

5.3 Multi-view models

In the subsequent presentation, we will denote visual features as \mathbf{V} , tag features as \mathbf{T} , the keyword or ground truth annotations as \mathbf{K} , and the automatically computed topics as \mathbf{C} . **CCA (V+T)** will refer to the two-view baseline model based on visual and tag features; **CCA (V+T+K)** to the three-view model based on visual features, tags, and supervised semantic information (ground truth labels or search keywords); and **CCA (V+T+C)** to the three-view model with the unsupervised third view (automatically computed tag clusters). All of these models will be evaluated for both I2I and T2I retrieval. For completeness, we will also evaluate the two-view **CCA (V+C)** and **CCA (V+K)**¹ models for I2I retrieval. However, because these models do not give an embedding for the tags, they cannot be used for cross-modal retrieval (i.e., T2I). In addition, we will evaluate K2I and I2T on subsets of models as appropriate (see Sections 6-8 for details).

5.4 Baselines

It is important to evaluate how CCA compares to alternative methods for obtaining intermediate embeddings for visual features supervised by tag or keyword information. For this, we have implemented two embedding methods from the recent literature, as described below.

Structural learning. Our first baseline is given by the structural or multi-task learning method of Ando and Zhang (2005); Quattoni *et al.* (2007). In their formulation, the tag matrix T is treated as supervisory information for the visual features V and a matrix of image-to-tag predictors W is obtained by ridge regression: $\|T - VW\|^2 + \rho\|W\|^2$. Next, since the tasks of predicting multiple tags are assumed to be correlated, we look for low-rank structure in W by computing its SVD. If $W = U_1 S U_2^\top$, then we use U_1 (or more precisely, its top d columns) as the embedding matrix for the visual features: $E = VU_1$. We select d by validation just as with our CCA-based methods. Note that the structural learning method does not produce an embedding for tags, so unlike

¹ It can be shown that CCA with labels as one of the views is equivalent to Linear Discriminant Analysis (LDA) (Bartlett, 1938).

CCA (V+T) and our three-view models, it is not suitable for cross-modal retrieval.

Wsabie. As a second baseline, we use Wsabie (Weston *et al.*, 2011). This method learns a discriminative model of the form $f(\mathbf{x}) = \mathbf{x}UW$ where U is the embedding and W is the matrix of weights for a set of classifiers, which in our case correspond to keywords (the K view). Once we obtain the embedding matrix U for the visual features, we evaluate the accuracy of I2I in the embedded space. Note that Wsabie, just as the structural learning method described above, cannot be used for cross-modal retrieval.

We have implemented Wsabie as described in Weston *et al.* (2011), using stochastic gradient descent (SGD) to optimize a ranking-based loss function. We use random initialization and a fixed learning rate of 0.01. The only difference from Weston *et al.* (2011) is that instead of explicit regularization, we use early stopping as suggested in Gordo *et al.* (2012). Namely, we run the SGD training for $20n$ iterations, where n is the number of training points, and validate the performance on the I2I task after processing every 1,000 points. At the end, the parameters with the highest validation accuracy are picked. For the dimensionality of the embedding U , we use a value of 128, which is relatively efficient and achieves good performance (it is also comparable to the typical dimensionalities that get selected by validation for our CCA model). Overall, the training of Wsabie is slower and more complicated than our approach, as it involves learning rate tuning and validation for early stopping. By contrast, the only parameter in our approach is the regularization constant in the covariance matrix (see Section 3.1) and it is set to a fixed value in all our experiments.

6 In-depth Analysis on Flickr-CIFAR

In this section, we use the Flickr-CIFAR dataset to conduct a detailed study of the various components of our proposed multi-view models, including feature combination and compression, different methods for tag clustering, and the proposed similarity function. For this purpose, we use the I2I, T2I, and K2I tasks. At the end, in Section 6.7 we perform a smaller-scale evaluation of I2T or image tagging.

6.1 Experimental protocol

Remember from Section 5 that the Flickr-CIFAR dataset consists of 230,173 Flickr images that are used to learn the CCA embedding and 15,167 ImageNet images that are used for quantitative evaluation. The ImageNet images are split into 13,167 “database” images against which retrieval is performed, 1,000 validation images, and 1,000 test images. One fixed split is used for all experiments. The validation images

	D-SIFT	D-CSIFT	D-RGBSIFT	H-SIFT	H-CSIFT	H-RGBSIFT	HOG	GIST	RGB Hist.	All
Image-to-image search										
No PCA	42.51	42.58	41.54	43.57	41.27	43.02	43.13	40.48	24.32	—
PCA (500D)	42.03	42.01	40.86	42.51	40.98	42.21	42.41	39.96	24.38	54.90
Tag-to-image search										
No PCA	53.68	50.34	52.02	55.67	52.18	54.84	54.06	48.75	26.88	—
PCA (500D)	55.24	53.71	53.83	58.42	54.18	57.29	56.12	51.41	28.52	64.07

Table 1 Precision@50 for full-dimensional vs. PCA-reduced data for **image-to-image** (top) and **tag-to-image** (bottom) retrieval for CCA (V+T). We did not obtain the result for combined features without PCA due to excessive computational cost (see text).

are run as queries against the database in order to select the dimensionality d of the embedding.

For I2I search, we use the test images as queries and report precision at top p retrieved images (Precision@ p) – that is, the fraction of the p returned images having the same ImageNet label as the query. For T2I search, we need a different set of queries as the ImageNet images are not tagged. For this, we take the tag feature vectors of 1,000 randomly chosen Flickr images (which are excluded from the set used to learn the embedding). The ground truth label of each query is given by the search keyword used to download the corresponding image. We have manually examined 50 of these queries and found that about 75% percent of the tags are closely related to the ground-truth keyword – for example, for the “car” keyword, the tags contain “car,” “cars,” and most of the other tags are also loosely related to the ground-truth keyword (“auto”). Thus, we expect this evaluation scheme to be reasonably accurate. Just as for image-to-image search, the evaluation metric for tag-to-image search is Precision@ p .

6.2 Evaluation of features and training set size

Recall from Section 4.1 that we apply nonlinear kernel maps to nine different visual features, reduce each of them to 500 PCA dimensions, and concatenate them together. As for the tag features (Section 4.2), we use sparse SVD to compress them to 500 dimensions. In this section, we evaluate these transformations. Since no dimensionality reduction is involved in the third view (K or C), for simplicity, we perform the evaluation with the standard two-view CCA (V+T) model.

Table 1 reports the effect of PCA on image-to-image and tag-to-image search for individual and combined visual features. For image-to-image retrieval, applying PCA to the original feature vectors may slightly hurt performance, but the decrease is less than 1%. More importantly, combined features significantly outperform each individual feature. On the other hand, for tag-to-image retrieval, PCA consistently helps to improve performance. This is possibly because Flickr tags are noisy, and reducing the dimensionality smooths the data.

To motivate our use of dimensionality reduction, it is instructive to give some running times on our platform, a 4-core Xeon 3.33GHz workstation with 48GB RAM. For a

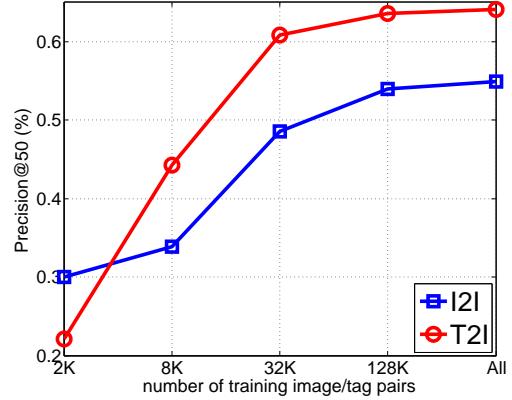


Fig. 5 Performance with increasing number of training images on the Flickr-CIFAR dataset.

single feature, it takes 2.5 seconds to obtain the CCA solution for approximated data (500 V + 500 T dimensions), versus 20 minutes for the full-dimensional kernel-mapped data (5,000 V + 2,494 T). For all nine combined visual features, it took around five minutes to get the approximated solution (4,500 V + 500 T); because the computation scales cubically with the number of dimensions, we have not tried to obtain the full-dimensional solution for all features (38,512 V + 2,494 T). Likewise, while one would ideally want to compare our results to an exact KCCA solution using kernel matrices, for hundreds of thousands of training points this is completely infeasible on our platform (for n training points, exact two-view KCCA involves solving a $2n \times 2n$ eigenvalue problem).

We conclude that combining multiple visual cues is indeed necessary to get the highest absolute levels of accuracy, and that dimensionality reduction of kernel-mapped features can satisfactorily address memory and computational issues with negligible loss of accuracy.

Finally, Figure 5 reports the retrieval accuracy of the two-view CCA (V+T) as a function of training dataset size. We can see that a fairly large amount of data, above 100K, is needed to converge to good performance.

# clusters	Image-to-image search						Tag-to-image search					
	10	20	30	40	50	100	10	20	30	40	50	100
visual k-means	49.12	48.73	48.73	48.52	48.50	47.58	57.35	56.67	56.47	56.07	56.16	56.20
k-means	51.60	56.18	57.36	57.45	57.34	57.40	63.23	65.23	67.36	66.76	68.29	70.29
NC	54.86	62.33	61.90	61.21	60.65	56.65	63.75	76.05	74.90	72.58	72.45	67.71
NMF	54.01	55.45	57.51	56.63	55.98	53.07	64.44	66.58	67.03	67.41	66.33	62.54
pLSA	55.40	56.43	57.33	57.62	56.89	54.88	62.45	64.71	64.92	65.76	67.32	66.83

Table 2 Precision@50 for different clustering methods for **image-to-image** and **tag-to-image** retrieval with the CCA (V+T+C) model. The second row shows results for visual clusters and the remaining rows for semantic (tag-based) clusters. The results are averaged over five different random initializations of the clustering methods. The performance of the CCA (V+T) baseline is 54.90% for image-to-image search, and 64.02% for tag-to-image search.

6.3 Comparison of tag clustering methods

In Section 4.3, we have presented a number of tag clustering methods that can be used to obtain the semantic topic matrix C for the third view when the training images do not come with ground-truth semantic information. Table 2 compares the performance of these methods. We use our proposed CCA (V+T+C) model, where V and T are low-dimensional approximations to the visual and tag features as discussed in Sections 4.1 and 4.2, and C is generated by the different clustering methods being compared. As a baseline, we also include results for k-means clustering based solely on visual features. It is clear that visual clusters have worse performance than all of the tag-based clustering methods, thus confirming that the visual features are too ambiguous for unsupervised extraction of high-level structure (this will also be demonstrated qualitatively in Figure 8). In fact, the performance of the three-view CCA (V+T+C) model with visual clusters is even worse than that of the CCA (V+T) baseline.

Among the tag-based clustering methods, normalized cut (NC) method achieves the best performance across a wide range of cluster sizes, followed by NMF, k-means and pLSA in decreasing order of accuracy. Thus, NC will be our method of choice for computing the CCA (V+T+C) embedding. As a function of the number of clusters, accuracy tends to increase up to a certain point, after which overfitting sets in. The best number of clusters to use depends on the breadth of coverage and the semantic structure of the dataset; we select it using validation data.

Figure 6 visualizes a few tag clusters in the joint CCA space for the NUS-WIDE dataset. For each example cluster, it shows the most frequent tags associated with the images in that cluster, as well as the sixteen images closest to the center of the cluster in the CCA (V+T+C) space. We can see that the clusters have a good degree of both visual and semantic coherence. For comparison, Figure 7 shows some clusters in just the tag view, i.e., before the joint CCA projection. We can see the images in each cluster correspond to the same semantic concept, though they are not visually similar. Fi-

method	I2I	T2I
CCA (V+T) (Eucl)	45.69	51.32
CCA (V+T) (scale+Eucl)	48.60	54.43
CCA (V+T) (scale+corr)	54.90	64.07
CCA (V+T+C) (Eucl)	53.61	69.69
CCA (V+T+C) (scale+Eucl)	57.06	72.42
CCA (V+T+C) (scale+corr)	62.44	75.92
CCA (V+T+K) (Eucl)	52.41	71.47
CCA (V+T+K) (scale+Eucl)	57.47	75.23
CCA (V+T+K) (scale+corr)	62.56	78.88

Table 3 Evaluation of different components of our proposed similarity function (eq. 2) on three multi-view setups. “Eucl” denotes Euclidean distance, “scale” denotes scaling of the feature dimensions by the CCA eigenvalues, and “corr” denotes normalized correlation. CCA (V+T+C) is computed using 20 NC clusters and CCA (V+T+K) is the supervised three-view model with K given by search keywords of the Flickr images.

nally, for completeness, Figure 8 shows k-means clusters on just the visual features. Because our visual features are relatively powerful, the cluster images are still perceptually similar, but they are no longer semantically consistent (in particular, note the poor correspondence between the most frequent tags for the entire clusters and the top sixteen images in the clusters). Figure 8 confirms the difficulty of finding good semantic themes by visual clustering alone, and helps to explain why visual clusters decrease the retrieval performance when incorporated as a third view in the learned CCA model (Table 2).

6.4 Comparison of similarity functions

Section 3.2 has presented our similarity function for nearest neighbor retrieval in the CCA space. Table 3 compares this function (eq. 2) to plain Euclidean distance for three different multi-view setups. We separately evaluate the effects of its two main components: eigenvalue scaling and normalized correlation. From the table, we can find that both these components give significant improvements over the Euclidean distance. We have consistently observed similar pat-

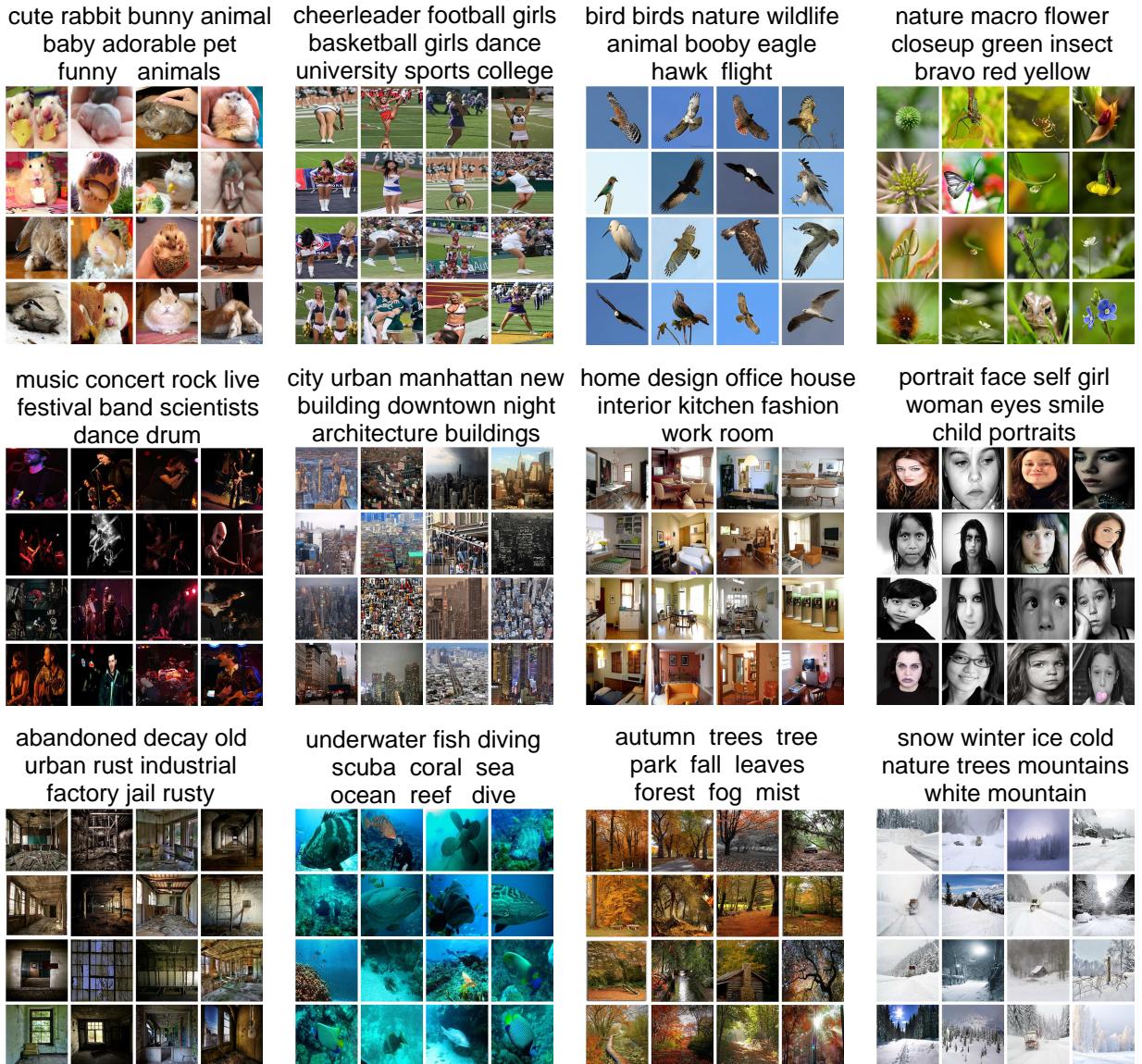


Fig. 6 Example semantic clusters on the NUS-WIDE dataset. For each cluster, we show the most frequent tags and the images closest to the cluster center in the CCA (V+T+C) space.

tern on other datasets, so we adopt the proposed similarity function in all subsequent experiments.

6.5 Comparison of multi-view models

Table 4 evaluates the performance of several multi-view models on three tasks: image-to-image (I2I), tag-to-image (T2I), and keyword-to-image (K2I) retrieval. As explained in Section 6.1, our performance metric for all tasks is class label (keyword) precision at top 50 images.

The most naive baselines for our approach are given by the single-view representations consisting only of visual features – either raw 38,512-dimensional ones (V-full) or PCA-compressed 4,500-dimensional ones (V). Both of these rep-

resentations can only be used directly for image-to-image similarity search (I2I). As can be seen from Table 4, the PCA-compressed feature gets higher precision for this task, but in absolute terms, both perform poorly.

A stronger baseline for our three-view models is given by the two-view CCA (V+T) representation, which can be used for all three retrieval tasks we are interested in (it can be used for K2I because the ten class labels or keywords in this dataset are a subset of the tag vocabulary). For I2I, the CCA (V+T) embedding improves the precision over non-embedded image features (V) from 41.65% to 54.9%. Thus, projecting visual features into a space that maximizes correlation with Flickr tags greatly improves the semantic coherence of similarity-based image matches (i.e., in the CCA



Fig. 7 Example tag clusters produced by k-means clustering on top of normalized cuts (Section 4.3). For each cluster center, the sixteen images with the closest tag vectors are shown. The most frequent tags in the cluster are shown above the central cluster images.

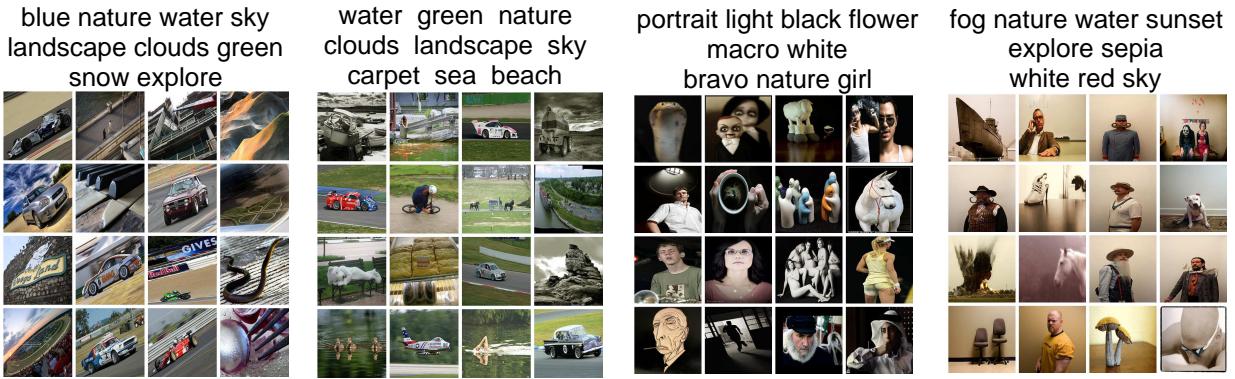


Fig. 8 Example visual k-means clusters for the NUS-WIDE dataset.

method	I2I	T2I	K2I
V-full	33.68	–	–
V	41.65	–	–
CCA (V+T)	54.90	64.07	95.60
CCA (V+K)	61.77	–	92.60
CCA (V+T+K)	62.56	78.88	97.20
CCA (V+C)	61.69	–	–
CCA (V+T+C)	62.44	75.92	97.80
Structural learning	57.77	–	–
Wsabie	57.15	–	–

Table 4 Results on Flickr-CIFAR for image-to-image (**I2I**), tag-to-image (**T2I**), and keyword-to-image(**K2I**) retrieval. The protocols for I2I and T2I are described in Section 6.1. For K2I, each of the 10 ground truth classes is used as a query once. The evaluation metric is average precision (%) at top 50 retrieved images. **V-full** refers to the concatenated 38,512-dimensional visual features. In all the other approaches, **V** refers to the 4,500-dimensional PCA-reduced features, **T** to the 500-dimensional sparse SVD-reduced tag features, and **C** is computed based on 20 NC clusters. **Structural learning** refers to the method of Ando and Zhang (2005); Quattoni *et al.* (2007) and **Wsabie** refers to the method of Weston *et al.* (2011). We have obtained standard deviations from five random database/query splits, and they are around 0.25% - 1%.

space, “truck” query images are much more likely to have top matches that are also “truck” images).

Next, we consider our supervised three-view model, CCA (V+T+K), where the third view is given by the search keywords used to retrieve the Flickr images. Even though this supervisory information is noisy (not all images retrieved by Flickr search for “truck” will actually contain trucks), we can see that incorporating it as a third view improves the precision of all three of our target retrieval tasks. The unsupervised version of our three-view model, CCA (V+T+C), where the third view is given by 20 NC clusters, performs almost identically to CCA (V+T+K) on I2I and K2I, and has slightly lower precision for T2I. This is a very encouraging result, since it shows that semantic information that is automatically recovered from noisy tags still provides a very powerful form of supervision.

Table 4 also lists the performance of two-view models CCA (V+K) and CCA (V+C) given by replacing the lower-level tag-based view T by the higher-level but lower-dimensional semantic view (K or C). Compared to CCA (V+T), both models have significantly higher I2I precision (though it is a bit lower than that of the respective three-view models). Thus, replacing noisy tags with the cleaner semantic views can help to improve performance. However, the two-view V+K and V+C models are not suitable for tag-to-image search, while the three-view models can be used for all the tasks we care about.

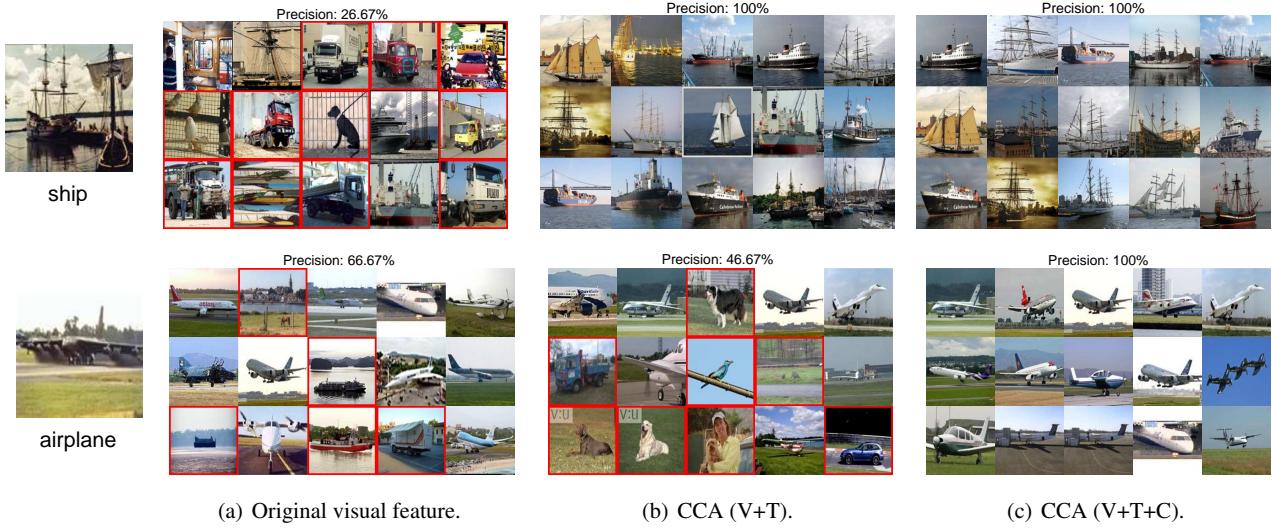


Fig. 9 Image-to-image retrieval results for two sample queries. The leftmost image is the query. Red border indicates a false positive.

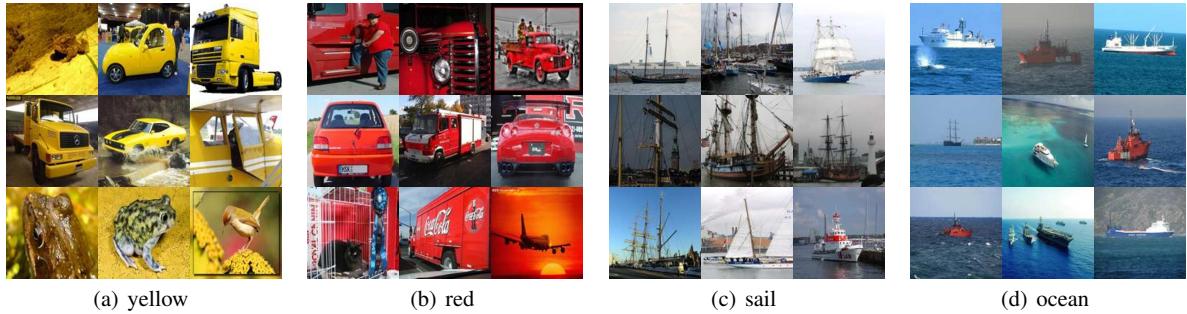


Fig. 10 Examples of tag-based image search on the Flickr-CIFAR dataset.

The last two lines of Table 4 report baseline comparisons to structural learning (Ando and Zhang, 2005; Quattoni *et al.*, 2007) and Wsabie (Weston *et al.*, 2011). Both perform better than CCA (V+T) but worse than all our other multi-view CCA models. One reason for this is because these models were designed for discrimination, not retrieval. In the case of Wsabie, it is possible that a batch learning approach (for example, second-order batch optimization) can give better performance than first-order SGD. However, a batch implementation of Wsabie is beyond the scope of our baseline comparisons (as described in the original paper, Wsabie is a sampling method specifically designed for SGD). Furthermore, neither structural learning nor Wsabie produces an embedding for tags, so unlike CCA (V+T) and our three-view models, these baselines are not suitable for tag-to-image retrieval.

6.6 Qualitative results

Figure 9 shows image-to-image search results for two example queries, and Figures 10 and 11 show examples of tag-to-

image search results. As noted earlier, one advantage of our system over traditional tag-based search approaches is that once our multi-view embedding is learned, we can use it to perform tag-to-image search on databases of images *without any accompanying text*. In fact, for the Flickr-CIFAR dataset, recall that we are using an embedding trained on tagged Flickr images to embed and search ImageNet images that lack tags. Figure 10 shows top retrieved images for four tags that do not correspond to the main ten keywords that were used to download the dataset. In particular, we are able to learn colors, common background classes like “ocean,” and sub-classes of the main keywords like “sail.”

Figure 11 shows images retrieved for more complex queries consisting of multiple tags such as “deer, snow.” Note that “deer” is one of our ten main keywords, and “snow” is a much less common tag. To get good retrieval results in such cases, we have found that we need to give higher weights to the more minor concepts when forming the query tag vector. Intuitively, the tag projection matrix found by minimizing the CCA objective function (eq. 1) is much more influenced by the distortion due to the common tags rather than the rare ones. We have empirically observed that we

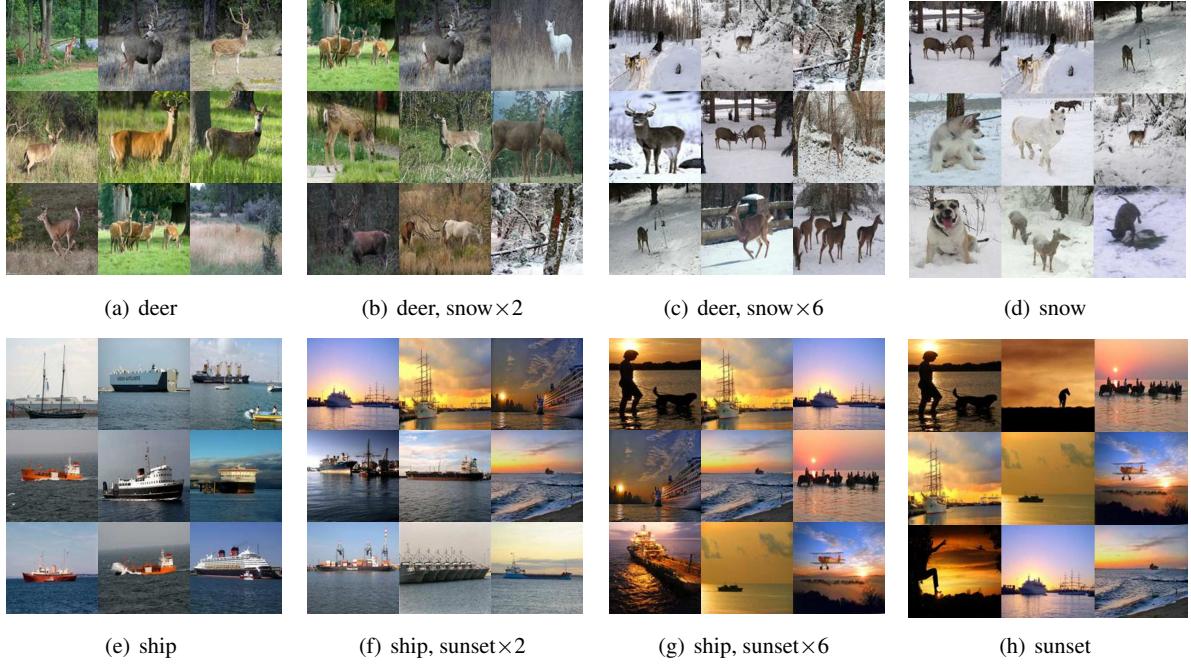


Fig. 11 Examples of tag-to-image search on Flickr-CIFAR with multiple query tags and adjustable weights (see text).

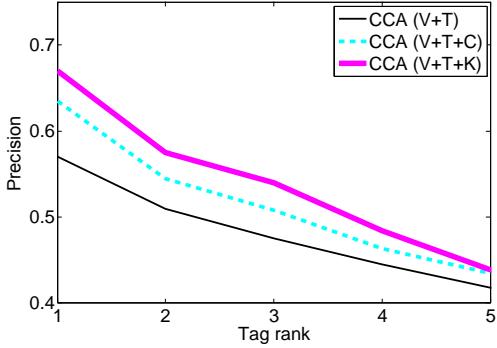


Fig. 12 Tagging results on the Flickr-CIFAR dataset: Average precision of retrieved tags vs. tag rank based on manual evaluation (see text).

can counteract this effect and obtain more accurate results for less frequent tags by increasing their weights in the tag vector at query time. To date, we have not designed a way to tune the weights automatically. However, in an interactive image search system, it would be very natural for users to adjust the weights on the fly to modulate the importance of different concepts in their query. For example, in Figure 11 (a)-(c), when we increase the weight for “snow,” snow becomes more and more prominent in the retrieved images.

6.7 Tagging results

This section presents a quantitative evaluation of our method for image tagging or annotation. As described in Section 5, we use the data-driven annotation scheme of Makadia *et al.*

(2008), where tags are transferred from top fifty neighbors to the query in the latent space. We randomly sample 200 query images from our ImageNet test set and use CCA (V+T), CCA (V+T+C), and CCA (V+T+K) spaces to transfer tags. To evaluate the results, we ask four individuals (members of the research group not directly involved with this project) to verify the tags suggested by different methods, that is, mark each tag as either relevant or irrelevant to the image. To avoid bias, our evaluation interface does not tell the evaluators which set of tags was produced by which method, and presents the sets of tags corresponding to different methods in random order for each test image. Our reasons for using human evaluation are twofold: first, our test images do not have any ground truth annotations; second, it is hard to provide ground truth consisting of a complete set of tags that could be relevant to an image. We combine the results of the human evaluators by voting: each tag that gets marked as relevant by three or more evaluators is considered correct.

Figure 12 reports average precision as a function of tag rank (which is determined by frequency of the tag in the top fifty closest images to the query in the CCA space). We can find that our proposed three-view models, CCA (V+T+K) and CCA (V+T+C), lead to better accuracy than the baseline CCA (V+T) method. Figure 13 shows the tagging results for CCA (V+T) vs. CCA (V+T+C) on a few example test images.

image	CCA (V+T)	CCA (V+T+C)	image	CCA (V+T)	CCA (V+T+C)
(a) 	airplane plane airshow Wisconsin eaa	bird flying airplane plane air	(b) 	ship Canada bay sea steel	ship boat water harbor Canada
(c) 	turbo cars automobiles Subaru legacy	aircraft airplane aeroplane 10millionphotos aviation	(d) 	frog green nature garden treefrog	frog bird nature florida wildlife
(e) 	deer zoo may garden game	zoo deer animals Chester wildlife	(f) 	cat kitty kitten cute katze	cat dog kent animals rebel

Fig. 13 Example image tagging results on Flickr-CIFAR dataset for CCA (V+T) and CCA (V+T+C). Tags in red have been marked as irrelevant by human annotators.

7 Results on the NUS-WIDE Dataset

In this section, we compare different multi-view embeddings on the NUS-WIDE dataset. We randomly split the dataset into 219,648 training and 50,000 test images. As before, we learn the joint embedding using the training images, and test retrieval accuracy on testing dataset. In the test set, we randomly sample and fix 1,000 images as the queries, 1,000 images as the validation set, and retrieve the remaining images. The validation set is used to find the number of clusters for NC. For this dataset, this number ends up being 100, vs. 20 for Flickr-CIFAR. The larger number of clusters for NUS-WIDE is not surprising, since this dataset has a much larger number of underlying semantic concepts than Flickr-CIFAR (81 vs. ten). Since there are relatively fewer images per class, we report Precision@20 instead of Precision@50. Also, since the images in this dataset may contain multiple ground truth keywords, we compute *average per-keyword* precision. That is, if q is the number of keywords for a given query image and a is the number of relevant keywords retrieved in the top p images, we define Precision@ p as $\frac{a}{pq}$.

Table 5 reports results for different multi-view models on I2I, T2I, and K2I search. For the supervised K view, we directly use the ground truth annotations (which may contain multiple nonzero entries per image). On this dataset, the best performance is achieved by the supervised CCA (V+T+K) and CCA (V+K) models. The unsupervised three-view model CCA (V+T+C) still improves over CCA (V+T) for all three tasks, but not as much as CCA (V+T+K). By contrast, on the Flickr-CIFAR dataset (Table 4), we found that CCA (V+T+C) and CCA (V+T+K) were very close together. The weaker performance of the unsupervised three-

method	I2I	T2I	K2I
V-full	25.25	–	–
V	32.23	–	–
CCA (V+T)	42.44	42.37	60.87
CCA (V+K)	48.53	–	74.39
CCA (V+T+K)	48.06	50.46	68.25
CCA (V+C)	41.72	–	–
CCA (V+T+C)	44.03	43.11	64.02
Structural learning	41.21	–	–
Wsabie	43.65	–	–

Table 5 Comparison of multi-view models and baselines on the NUS-WIDE dataset. For K2I, since images may have multiple ground truth keywords, we do not generate the keyword queries directly but use the keyword vectors of the 1,000 query images used for I2I. The performance metric is Precision@20 averaged over the number of keywords per query, as described in the text. **Structural learning** refers to the method of Ando and Zhang (2005); Quattoni *et al.* (2007) and **Wsabie** refers to the method of Weston *et al.* (2011). We have obtained standard deviations from five random database/query splits, and they are around 0.66% - 1.06%.

view model on NUS-WIDE is not entirely surprising, however, since the tag clusters for NUS-WIDE are likely much more mixed than for Flickr-CIFAR, whose concepts were fewer and better separated. Intuitively, for richer and more diverse datasets, ground truth annotations are likely to be the strongest source of semantic information. Also, unlike in Table 4, the two-view supervised model CCA (V+K) appears to have stronger results than the three-view CCA (V+T+K) for I2I and especially K2I. This may be due to the T view adding noise to the K view. Despite this, the two-view CCA (V+K) model is not as useful or flexible as the three-view



Fig. 14 Image-to-image retrieval results on the NUS-WIDE dataset. The query image is shown on the left, together with its ground truth concepts. Red borders indicate false positive retrieval results. We consider an image to be a false positive if its ground truth annotation does not share any concepts with the query. Please note, however, that the ground truth is noisy, so some false (resp. true) positives are labeled inaccurately.

CCA (V+T+K) one – in particular, the former is not suitable for T2I retrieval.

Figure 14 shows example image-to-image search results and Figure 15 shows example tag-to-image search results for the CCA (V+T+C) model. As can be seen from the latter figure, our system can return appropriate images for compound queries consisting of combinations of as many as three tags, e.g., “mountain, river, waterfalls” or “beach, people, red.” Figure 16 compares tag-to-image retrieval results for the two-view model, CCA (V+T), and the three-view one, CCA (V+T+C). The three-view model tends to retrieve more relevant images, especially for compound queries.

Figure 17 compares image annotation results for CCA (V+T), CCA (V+T+C), and CCA (V+T+K) using the same human evaluation protocol as in Section 6.7. Unlike the Flickr-CIFAR results in Figure 12, where the three-view models produced higher precision than CCA (V+T), all three models work comparably for image tagging on NUS-WIDE. The example results shown in Figure 18 confirm that the subjective quality of the tags produced by two- and three-view models is similar. We believe that the explanation for this result has to do, at least in part, with the statistics of images and tags in NUS-WIDE. Specifically, many images in this dataset are either abstract or are natural landscape scenes

with no distinctive objects. For such images, all our embeddings tend to suggest generic tags. Also, suggesting tags such as “landscape,” “night,” “light,” etc., appears to be somewhat easier than trying to suggest object-specific tags, which are much more important for Flickr-CIFAR – indeed, in terms of absolute performance, the precision curves for NUS-WIDE (Figure 18) are higher than for Flickr-CIFAR (Figure 12). Furthermore, as discussed in Section 5, our embedding does not provide a complete solution to the image annotation problem, as it does not include a decoding step exploiting multi-label constraints. Developing such a solution is an important subject for our future work.

8 Results on the INRIA-Websearch Dataset

Finally, we report results on the INRIA web search dataset. As explained in Section 5, ground-truth semantic information for each image in this dataset is in the form of a binary label saying whether or not that image is relevant to a particular query concept. This information directly gives us our third view for the supervised CCA (V+T+K) model. Since this dataset, just as NUS-WIDE, has relatively few images per concept, we evaluate performance using Precision@20. We randomly split the dataset into 51,478 training

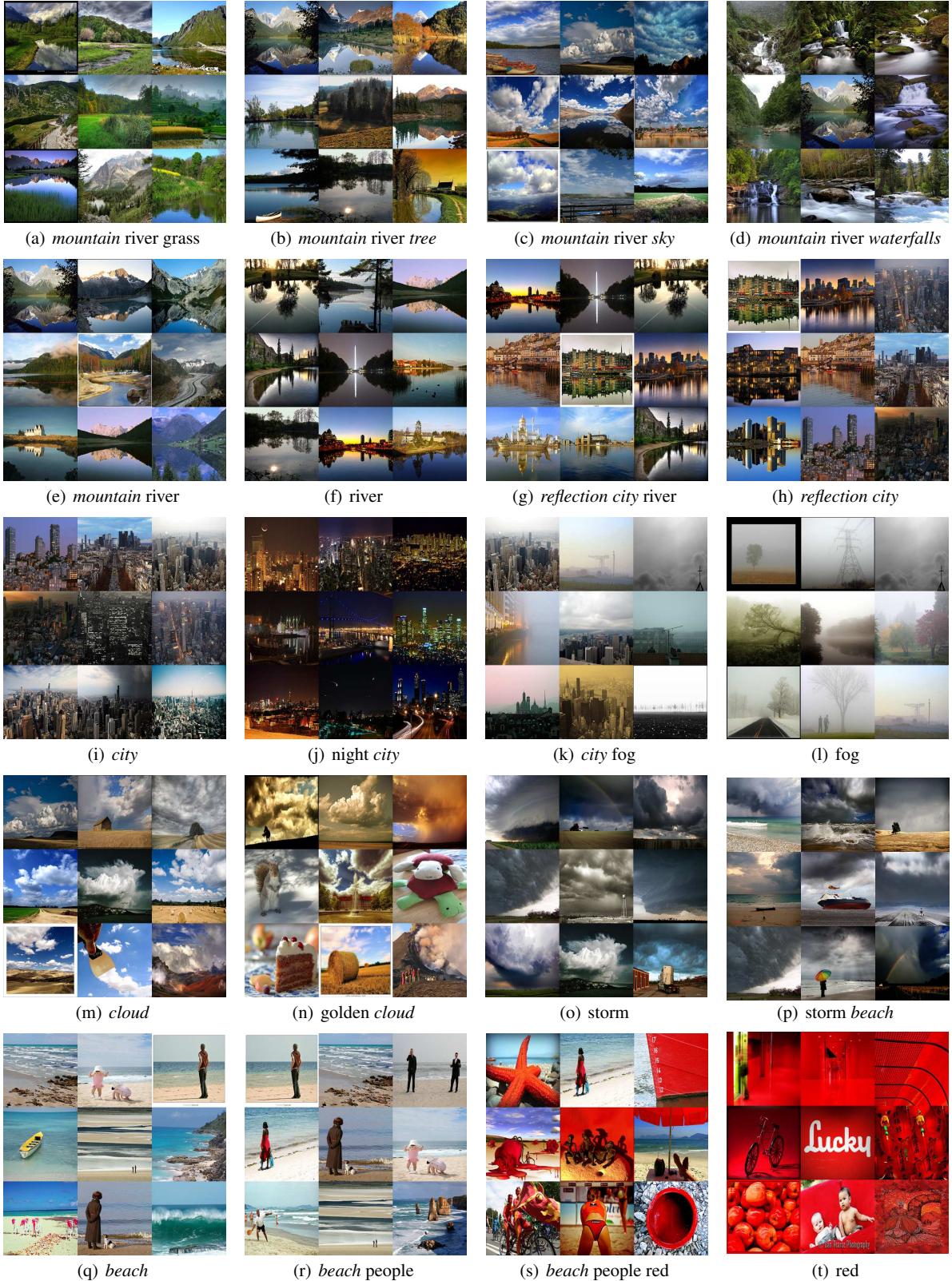


Fig. 15 Examples of tag-to-image search on the NUS-WIDE dataset with CCA (V+T+C). Tags in italic are also part of the 81-member semantic concept vocabulary. Notice that the three-view model can return appropriate images for combinations of up to three query tags.

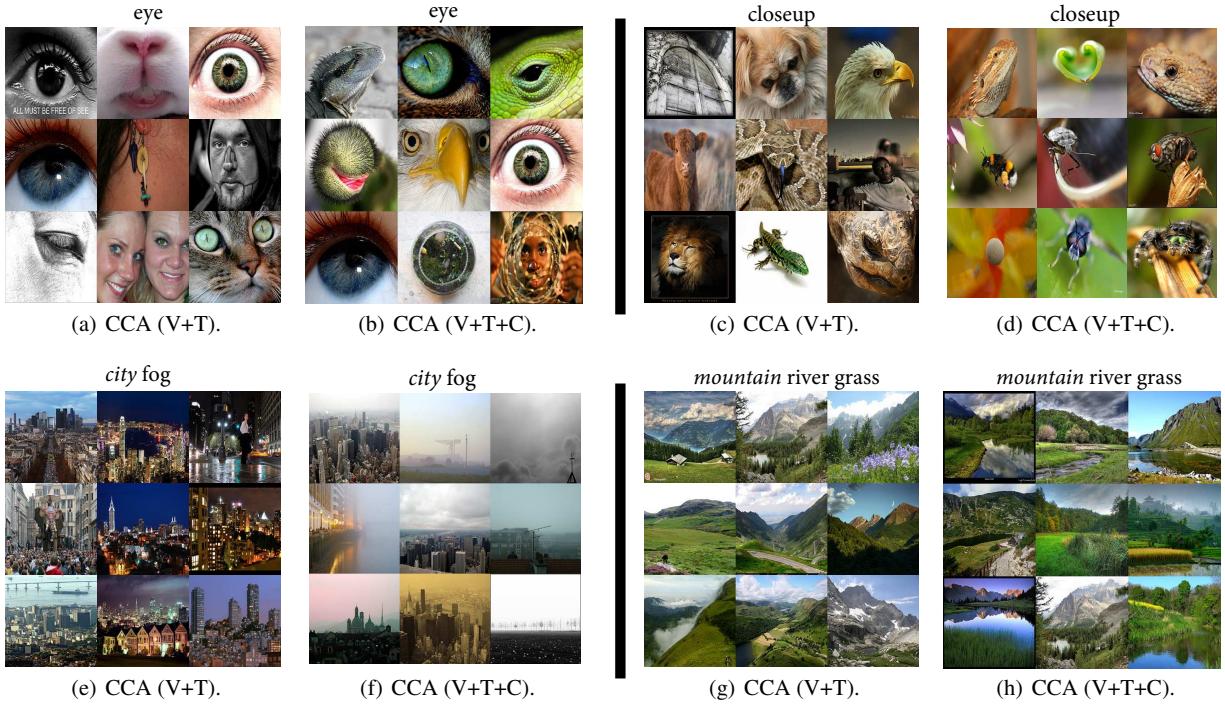


Fig. 16 A qualitative comparison of tag-to-image search for CCA (V+T) and CCA (V+T+C) on the NUS-WIDE dataset. Qualitatively, CCA (V+T+C) works better. For “city, fog,” the three-view model successfully finds city images with fog, while CCA (V+T) only finds city images. For “mountain, river, grass,” almost all images found by the three-view model contain some river, while the images found by CCA (V+T) do not contain river.

and 20,000 test images. In the test set, we use 18,000 images as the database, 1,000 images as validation queries, and 1,000 as test queries. Note that the database includes images marked as “irrelevant,” but not the validation or test queries. For CCA (V+T+C), we tune the number of NC clusters on the validation dataset to obtain 450 clusters.

Table 6 reports image-to-image and tag-to-image search results. As this dataset is extremely noisy and diverse, the absolute accuracy for all the methods is low. Precision may be further lowered by the fact that each database image is annotated with its relevance to just a single query concept – thus, if a retrieved image is relevant for more than one query, this may not show up in the quantitative evaluation. Nevertheless, CCA (V+T+C) still consistently works better than the CCA (V+T) baseline. As on the NUS-WIDE dataset, the supervised CCA (V+T+K) model works better than CCA (V+T+C). Also, as on NUS-WIDE, CCA (V+K) works slightly better than CCA (V+T+K) for I2I. Once again, this may be because the tag view (T) is adding noise to the embedding. Figure 19 shows some qualitative image-to-image search results.

Finally, since the second view of this dataset consists not of tags, but of text mined from webpages, we do not evaluate image-to-tag search.

9 Discussion and Future Work

This paper has presented a multi-view embedding approach for Internet images, tags, and their semantics. We have started with the two-view visual-textual CCA model popular in several recent works (Gong and Lazebnik, 2011; Hardoon *et al.*, 2004; Hwang and Grauman, 2010, 2011; Rasiwasia *et al.*, 2010) and shown that its performance can be significantly improved by adding a third view based on semantic ground truth labels, image search keywords, or even topics obtained by unsupervised tag clustering. In terms of quantitative results, this is our most significant finding – both the supervised and unsupervised three-view models, CCA (V+T+K) and CCA (V+T+C), have consistently outperformed the two-view CCA (V+T) model on all three datasets, despite the extremely diverse characteristics shown by these datasets.

For the unsupervised three-view model, CCA (V+T+C), it may appear somewhat unintuitive that the third cluster-based view, which is completely derived from the second textual one, can add any useful information to improve the embedding. There are several ways to understand what the unsupervised third view is doing. Especially in simpler datasets with a few well-separated concepts, such as our Flickr-CIFAR dataset, tag clustering is actually capable of “recovering” the underlying class labels. Even in more diverse and ambiguous datasets with overlapping concepts, tag clustering can still find sensible concepts that impose useful high-level

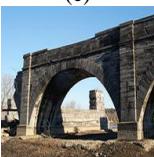
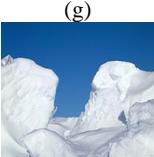
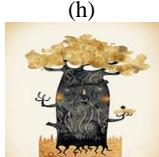
image	CCA (V+T)	CCA (V+T+C)	image	CCA (V+T)	CCA (V+T+C)
(a) 	orange dog girls costume sport	party cheerleader girls costume game	(b) 	wall door yellow red orange	red wall door handmade art
(c) 	architecture church spain building abandoned	abandoned architecture military building decay	(d) 	night lights skyline sunset water	sunset night lights skyline city
(e) 	storm weather clouds sky night	blue clouds sky night storm	(f) 	black white portrait woman hands	portrait black white woman girl
(g) 	snow white skiing winter blue	snow winter blue white sky	(h) 	illustration art handmade drawing vintage	art illustration handmade drawing design

Fig. 17 Example tagging results on the NUS-WIDE dataset (see text for discussion).

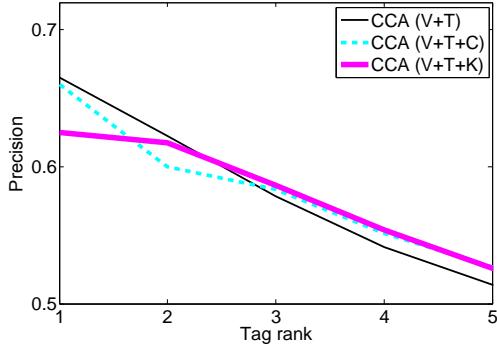


Fig. 18 Tagging results on the NUS-WIDE dataset: average precision of retrieved tags vs. tag rank.

structure (Figure 6). In its attempt to discover this structure, our embedding space may be likened to a non-generative version of a model that connects visual features and noisy tags to latent image-level semantics (Wang *et al.*, 2009a). From another point of view, we can observe that the output of the clustering process, given by the cluster indicator matrix C , is a highly nonlinear transformation of the second view T that either regularizes the embedding or improves its expressive power.

The quantitative and qualitative results presented in this paper demonstrate that our proposed multi-view embedding space, together with the similarity function specially designed for it, successfully captures visual and semantic consistency

method	I2I	T2I	K2I
V-full	5.42	–	–
V	7.29	–	–
CCA (V+T)	12.66	25.67	–
CCA (V+K)	16.84	–	44.43
CCA (V+T+K)	15.36	32.76	41.75
CCA (V+C)	13.25	–	–
CCA (V+T+C)	13.61	29.57	–
Structural learning	8.35	–	–
Wsabie	10.01	–	–

Table 6 Precision@20 for different multi-view models on the INRIA-Websearch dataset. For K2I, the queries correspond to the 353 ground truth concepts. Note that these concepts are no longer necessarily part of the tag vocabulary, so we cannot report K2I results for any embedding that does not include the K view. We have obtained standard deviations from five random database/query splits, and they are around 0.6% - 1.1%.

in diverse, large-scale datasets. This space can form a good basis for a scalable and flexible retrieval system capable of simultaneously accommodating multiple usage scenarios. The visual and semantic clusters discovered by tag clustering and subsequent CCA projection can be used to summarize and browse the content of Internet photo collections (Berg and Berg, 2009; Raguram and Lazebnik, 2008). Figure 6 has shown an example of what such a summary could look like. Furthermore, users can search with images for similar im-

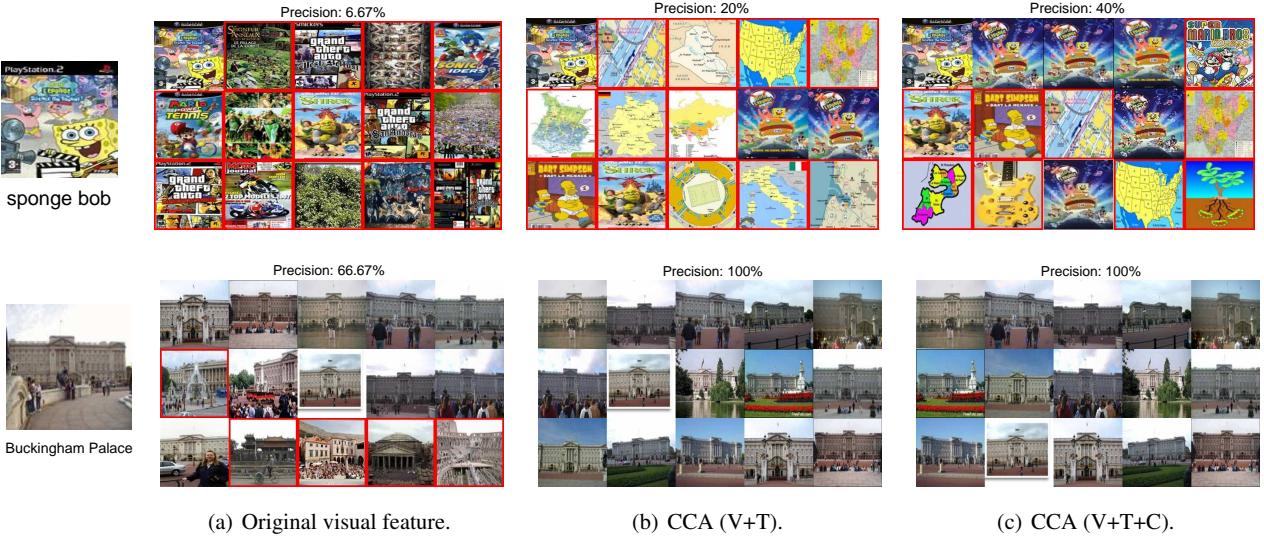


Fig. 19 Sample image-to-image retrieval results on the INRIA-Websearch dataset. The query is on the left. Red border means false positive.

ages, or retrieve images based on queries consisting of multiple tags or keywords. As illustrated in Figure 11, they can also manually adjust weights corresponding to different keywords according to the importance of those keywords. Finally, our embedding space can also serve as a basis for an automatic image annotation system. However, as discussed in Section 7, in order to achieve satisfactory results on this task, we need to develop more sophisticated decoding methods incorporating multi-label consistency constraints.

Besides the application scenarios named above, we are also interested in using our learned latent space as an intermediate representation for recognition tasks. One of these is nonparametric image parsing (Liu *et al.*, 2010; Tighe and Lazebnik, 2010) where, given a query image, a small number of similar training images is retrieved and labels are transferred from these images to the query. With a better embedding for images and tags, this retrieval step may be able to return training images more consistent with the query and lead to improved accuracy for image parsing. Another problem of interest to us is describing images with sentences (Farhadi *et al.*, 2010; Kulkarni *et al.*, 2011; Ordonez *et al.*, 2011). Once again, with a good intermediate embedding space linking images and tags, the subsequent step of sentence generation may become easier.

Acknowledgements We would like to thank the anonymous reviewers for their constructive comments; Jason Weston for advice on implementing the Wsabie method; Albert Gordo and Florent Perronnin for useful discussions; and Joseph Tighe, Hongtao Huang, Juan Caicedo, and Mariyam Khalid for helping with manual evaluation of the auto-tagging experiments. Gong and Lazebnik were supported by NSF grant IIS 1228082, DARPA Computer Science Study Group (D12AP00305), and Microsoft Research Faculty Fellowship.

References

- Ando, R. K. and Zhang, T. (2005). A framework for learning predictive structures from multiple tasks and unlabeled data. *JMLR*.
- Bach, F. R. and Jordan, M. I. (2002). Kernel independent component analysis. *Journal of Machine Learning Research*, **3**, 1–48.
- Barnard, K. and Forsyth, D. (2001). Learning the semantics of words and pictures. In *ICCV*.
- Bartlett, M. S. (1938). Further aspects of the theory of multiple regression. *Mathematical Proceedings of the Cambridge Philosophical Society*, **34**(1), 33–40.
- Berg, T. and Forsyth, D. (2006). Animals on the web. *CVPR*.
- Berg, T. L. and Berg, A. C. (2009). Finding iconic images. In *Second Workshop on Internet Vision at CVPR*.
- Blaschko, M. and Lampert, C. (2008). Correlational spectral clustering. *CVPR*.
- Blei, D. and Jordan, M. (2003). Modeling annotated data. In *ACM SIGIR*, pages 127–134.
- Blei, D., Ng, A. Y., and Jordan, M. I. (2003). Latent Dirichlet allocation. *JMLR*.
- Carneiro, G., Chan, A., Moreno, P., and Vasconcelos, N. (2007). Supervised learning of semantic classes for image annotation and retrieval. In *PAMI*.
- Chapelle, O., Weston, J., and Scholkopf, B. (2003). Cluster kernels for semi-supervised learning. *NIPS*.
- Chen, N., Zhu, J., Sun, F., and Xing, E. P. (2012). Large-margin predictive latent subspace learning for multi-view data analysis. In *PAMI*.
- Chen, X., Yuan, X.-T., Chen, Q., Yan, S., and Chua, T.-S. (2011). Multi-label visual classification with label exclusive context. In *ICCV*.

- Chua, T.-S., Tang, J., Hong, R., Li, H., Luo, Z., and Zheng, Y.-T. (2009). NUS-WIDE: A real-world web image database from National University of Singapore. In *Proc. of ACM Conf. on Image and Video Retrieval (CIVR'09)*, Santorini, Greece.
- Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *CVPR*.
- Datta, R., Joshi, D., Li, J., and Wang, J. Z. (2008). Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, **40**(2), 1–60.
- Deng, J., Dong, W., Socher, R., Li, L., Li, K., and Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. *CVPR*.
- Duygulu, P., Barnard, K., de Freitas, N., and Forsyth, D. (2002). Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *ECCV*.
- Fan, J., Shen, Y., Zhou, N., and Gao, Y. (2010). Harvesting large-scale weakly-tagged image databases from the web. In *CVPR*, pages 802–809.
- Farhadi, A., Hejrati, M., Sadeghi, A., Young, P., Rashtchian, C., Hockenmaier, J., and Forsyth, D. A. (2010). Every picture tells a story: generating sentences for images. In *ECCV*.
- Foster, D. P., Johnson, R., Kakade, S. M., and Zhang, T. (2010). Multi-view dimensionality reduction via canonical correlation analysis. *Tech Report. Rutgers University*.
- Frankel, C., Swain, M. J., and Athitsos, V. (1997). Webseer: An image search engine for the World Wide Web. In *CVPR*.
- Gehler, P. and Nowozin, S. (2009). On feature combination for multiclass object classification. *ICCV*.
- Gong, Y. and Lazebnik, S. (2011). Iterative quantization: An procrustean approach to learning binary codes. *CVPR*.
- Globerson A. and Roweis S. (2005). Metric Learning by Collapsing Classes. *NIPS*.
- Goldberger J., Roweis S., Hinton G., Salakhutdinov R. (2004). Neighbourhood Components Analysis. *NIPS*.
- Grangier, D. and Bengio, S. (2008). A discriminative kernel-based model to rank images from text queries. *PAMI*.
- Grubinger, M., Clough, P. D., Müller, H., and Deselaers, T. (2006). The IAPR TC-12 benchmark - a new evaluation resource for visual information systems. In *Proceedings of the International Workshop OntoImage'2006 Language Resources for Content-Based Image Retrieval*, pages 13 – 23.
- Gordo, A., Rodriguez-Serrano, J., Perronnin, F., and Valveny, E. (2012). Leveraging category-level labels for instance-level image retrieval. *CVPR*.
- Guillaumin, M., Mensink, T., Verbeek, J., and Schmid, C. (2009). TagProp: Discriminative metric learning in nearest neighbor models for image auto-annotation. *ICCV*.
- Guillaumin, M., Verbeek, J., and Schmid, C. (2010). Multi-modal semi-supervised learning for image classification. *CVPR*.
- Hardoon, D., Szedmak, S., and Shawe-Taylor, J. (2004). Canonical correlation analysis; an overview with application to learning methods. *Neural Computation*.
- Hofmann, T. (1999). Probabilistic latent semantic indexing. *SIGIR*.
- Hotelling, H. (1936). Relations between two sets of variables. *Biometrika*, **28**, 312–377.
- Hsu, D., Kakade, S., Langford, J., and Zhang, T. (2009). Multi-label prediction via compressed sensing. In *NIPS*.
- Hwang, S. J. and Grauman, K. (2010). Accounting for the relative importance of objects in image retrieval. *BMVC*.
- Hwang, S. J. and Grauman, K. (2011). Learning the relative importance of objects from tagged images for retrieval and cross-modal search. *IJCV*.
- Weinberger, K., Blitzer, J., and Saul, L. (2005). Distance metric learning for large margin nearest neighbor classification. *NIPS*.
- Krapac, J., Allan, M., Verbeek, J., and Jurie, F. (2010). Improving web-image search results using query-relative classifiers. *CVPR*.
- Krizhevsky, A. (2009). Learning multiple layers of features from tiny images. *Tech Report. University of Toronto*.
- Kulkarni, G., Premraj, V., Dhar, S., Li, S., Choi, Y., Berg, A. C., and Berg, T. L. (2011). Babytalk: Understanding and generating simple image descriptions. In *CVPR*.
- Larsen, R. M. (1998). Lanczos bidiagonalization with partial reorthogonalization,. *Department of Computer Science, Aarhus University, Technical report*.
- Lavrenko, V., Manmatha, R., , and Jeon, J. (2003). A model for learning the semantics of pictures. In *NIPS*.
- Lazebnik, S., Schmid, S., and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. *CVPR*.
- Li, J. and Wang, J. (2008). Real-time computerized annotation of pictures. In *PAMI*.
- Liu, C., Yuen, J., and Torralba, A. (2010). Sift flow: dense correspondence across difference scenes. In *PAMI*.
- Liu, Y., Xu, D., Tsang, I., and Luo, J. (2009). Using large-scale web data to facilitate textual query based retrieval of consumer photos. In *ACM MM*.
- Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *IJCV*.
- Lucchi, A. and Weston, J. (2012). Joint image and word sense discrimination for image retrieval. In *ECCV*.
- Maji, S. and Berg, A. (2009). Max-margin additive classifiers for detection. *CVPR*.
- Makadia, A., Pavlovic, V., and Kumar, S. (2008). A new baseline for image annotation. In *ECCV*.
- Mensink, T., Verbeek, J., Csurka, G., and Perronnin, F. (2012). Metric learning for large scale image classification: Generalizing to new classes at near-zero cost. In *ECCV*.

- Monay, F. and Gatica-Perez, D. (2004). PLSA-based image auto-annotation: Constraining the latent space. In *ACM Multimedia*.
- Ng, A., Jordan, M., and Weiss, Y. (2001). On spectral clustering: analysis and an algorithm. *NIPS*.
- Oliva, A. and Torralba, A. (2001). Modeling the shape of the scene: a holistic representation of the spatial envelope. *IJCV*.
- Ordonez, V., Kulkarni, G., and Berg, T. L. (2011). Im2text: Describing images using 1 million captioned photographs. *NIPS*.
- Perronnin, F., Sanchez, J., , and Liu, Y. (2010). Large-scale image categorization with explicit data embedding. *CVPR*.
- Quadrianto, N. and Lampert, C. H. (2011). Learning multi-view neighborhood preserving projections. In *ICML*.
- Quattoni, A., Collins, M., and Darrell, T. (2007). Learning visual representations using images with captions. *CVPR*.
- Raguram, R. and Lazebnik, S. (2008). Computing iconic summaries for general visual concepts. In *First Workshop on Internet Vision at CVPR*.
- Rahimi, A. and Recht, B. (2007). Random features for large-scale kernel machines. *NIPS*.
- Rai, P. and Daumé, H. (2009). Multi-label prediction via sparse infinite CCA. *NIPS*.
- Rasiwasia, N. and Vasconcelos, N. (2007). Bridging the gap: Query by semantic example. *IEEE Transactions on Multimedia*.
- Rasiwasia, N., Pereira, J. C., Covello, E., Doyle, G., Lanckriet, G., Levy, R., and Vasconcelos, N. (2010). A new approach to cross-modal multimedia retrieval. *ACM MM*.
- Scholkopf, B., Smola, A., and Muller, K.-R. (1997). Kernel principal component analysis. *ICANN*.
- Schroff, F., Criminisi, A., and Zisserman, A. (2007). Harvesting image databases from the Web. In *ICCV*.
- Sharma, A., Kumar, A., Daumé, H., and Jacobs, D. (2012). Generalized multiview analysis: A discriminative latent space. In *CVPR*.
- Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. *PAMI*.
- Smeulders, A. W., Worring, M., Santini, S., Gupta, A., and Jain, R. (2000). Content-based image retrieval at the end of the early years. *PAMI*, **22**(12), 1349–1380.
- Tighe, J. and Lazebnik, S. (2010). Superparsing: Scalable nonparametric image parsing with superpixels. In *ECCV*.
- Udupa, R. and Khapra, M. (2010). Improving the multilingual user experience of Wikipedia using cross-language name search. In *NAACL*.
- van de Sande, K. E. A., Gevers, T., and Snoek, C. G. M. (2010). Evaluating color descriptors for object and scene recognition. *PAMI*.
- Vedaldi, A. and Zisserman, A. (2010). Efficient additive kernels via explicit feature maps. In *CVPR*.
- Verma, Y. and Jawahar, C. V. (2012). Image annotation using metric learning in semantic neighbourhoods. In *ECCV*.
- Vinokourov, A., Shawe-Taylor, J., and Cristianini, N. (2002). Inferring a semantic representation of text via cross-language correlation analysis. In *NIPS*.
- von Ahn, L. and Dabbish, L. (2004). Labeling images with a computer game. In *ACM SIGCHI*.
- Wang, C., Blei, D., and Li, F. (2009a). Simultaneous image classification and annotation. In *CVPR*, pages 1903–1910.
- Wang, G., Hoiem, D., and Forsyth, D. (2009b). Building text features for object image classification. *CVPR*.
- Wang, G., Hoiem, D., and Forsyth, D. (2009c). Learning image similarity from Flickr groups using stochastic intersection kernel machines. *ICCV*.
- Wang, X.-J., Zhang, L., Li, X., and Ma, W.-Y. (2008). Annotating images by mining image search results. *PAMI*, **30**(11), 1919–1932.
- Weston, J., Bengio, S., and Usunier, N. (2011). Wsabie: Scaling up to large vocabulary image annotation. In *IJCAI*.
- X. Wei and W. B. Croft (2006). LDA-Based Document Models for Ad-hoc Retrieval. In *SIGIR*.
- Xiao, J., Hays, J., Ehinger, K., Oliva, A., and Torralba, A. (2010). SUN database: Large-scale scene recognition from abbey to zoo. In *CVPR*.
- Xu, W., Liu, X., and Gong, Y. (2003). Document clustering based on non-negative matrix factorization. *SIGIR*.
- Yakhnenko, O. and Honavar, V. (2009). Multiple label prediction for image annotation with multiple kernel correlation models. In *Workshop on Visual Context Learning (in conjunction with CVPR)*.
- Zhang, Y. and Schneider, J. (2011). Multi-label output codes using canonical correlation analysis. In *AISTATS*.
- Zhu, S., Ji, X., Xu, W., and Gong, Y. (2005). Multi-labelled classification using maximum entropy method. In *ACM SIGIR*.