Accounting for the Relative Importance of Objects in Image Retrieval

Sung Ju Hwang
sjhwang@cs.utexas.edu
Kristen Grauman
grauman@cs.utexas.edu
The University of Texas
Austin, TX, USA

Abstract

We introduce a method for image retrieval that leverages the implicit information about object *importance* conveyed by the list of keyword tags a person supplies for an image. We propose an unsupervised learning procedure based on Kernel Canonical Correlation Analysis that discovers the relationship between how humans tag images (e.g., the order in which words are mentioned) and the relative importance of objects and their layout in the scene. Using this discovered connection, we show how to boost accuracy for novel queries, such that the search results may more closely match the user's mental image of the scene being sought. We evaluate our approach on two datasets, and show clear improvements over both an approach relying on image features alone, as well as a baseline that uses words and image features, but ignores the implied importance cues.

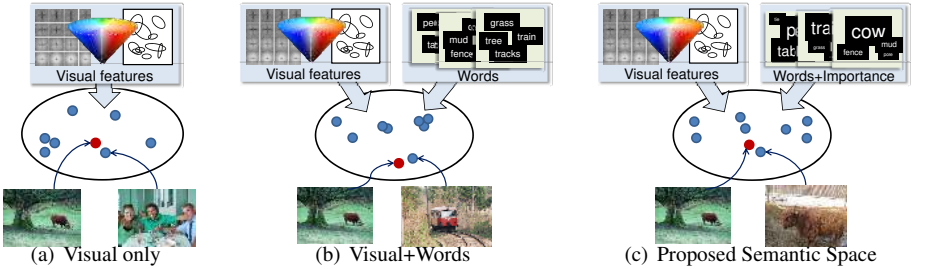
1 Introduction

Images tagged with human-provided keywords are a valuable source of data, and are increasingly available thanks to community photo sharing sites such as Flickr, collaborative annotation games [39], online captioned news photo collections, and various labeling projects in the vision community [7, 32]. While a user's intentions when tagging may vary, often the keywords reflect the objects and events of significance and can thus be exploited as a loose form of labels and context.

Accordingly, researchers have explored a variety of ways to leverage images with associated text—including learning their correspondence for auto-annotation of regions, objects, and scenes [2, 4, 8, 14, 18, 23], using keyword search for inexpensive dataset creation [12, 22, 33, 38], and building richer image representations based on the two simultaneous "views" for retrieval or clustering [3, 5, 15, 28, 29, 30, 31, 40]. Such methods have shown that learning with words and images together can yield stronger models.

However, existing approaches largely assume that image tags' value is purely in indicating the presence of certain objects. As such, for retrieval applications, one scores the query results according to the number of shared keywords among the ground truth tags; similarly, for recognition applications, one scores the label predictions according to their per-class accuracy. The problem with this assumption is that it ignores the relative *importance* of different objects composing a scene, and the impact that this importance can have on a user's perception of relevance.

To appear, Proceedings of the British Machine Vision Conference, Sept 2010.



(a) Visual only (b) Visual+Words (c) Proposed Semantic Space

Figure 1: Main idea. (a) Using only visual cues allows one to retrieve similar-looking examples for a query (bottom left image / denoted with red dots), but can result in semantically irrelevant retrievals (e.g., similar colors and edges, but different objects). (b) Learning a Visual+Words "semantic space" (e.g., with KCCA) helps narrow retrievals to images with a similar distribution of objects. However, the representation still lacks knowledge as to which objects are more important. (c) Our idea is to learn the semantic space using the order and relative ranks of the human-provided tag-lists. As a result, we retrieve not only images with similar distributions of objects, but more specifically, images where the objects have similar degrees of "importance" as in the query image. For example, the retrieved image on the bottom right, though lacking a 'tree', is a better match due to its focus on the cow.

For example, consider the bottom right image in Figure 1(c): if a system were to auto-tag this image with *either* mud or fence or pole or cow, would all responses be equally useful? We argue they would not; it is more critical to name those objects that appear more prominent or best define the scene (arguably, the cow in this example). Likewise, given a query image, we should prefer to retrieve images that are similar not only in terms of their total object composition, but also in terms of those objects' relative importance to the scene.

How can we learn the relative importance of objects and use this knowledge to improve image retrieval? Our approach rests on the assumption that humans name the most prominent or interesting items first when asked to summarize an image. Thus, rather than treat tags as simply a set of names, we consider them as an ordered list conveying useful cues about what is most notable to the human viewer. Specifically, we record a tag-list's nouns, their absolute ordering, and their relative ranking compared to their typical placement across all images. We propose an unsupervised approach based on Kernel Canonical Correlation Analysis (KCCA) [1, 13, 16] to discover a "semantic space" that captures the relationship between those tag cues and the image content itself, and show how it can be used to more effectively process novel text- or image-based queries or auto-tagging requests.

Unlike traditional "appearance only" retrieval systems, we expect to find images that have more semantic (object-level) relevance to the query. Unlike existing approaches that learn from both words and images (including prior uses of KCCA [5, 15, 40]), we expect to find images that better capture the human's concept of importance and layout. See Figure 1.

Our main contributions are: (1) an approach to learn the relative importance or configuration of objects that requires no manual supervision beyond gathering tagged images, and (2) experiments demonstrating that the learned semantic space enhances retrieval, as judged by quantitative measures of object prominence derived from human-provided data.

2 Related Work

Space constraints prohibit a full review of the content-based image retrieval literature; please see [6, 34] and references therein. Our approach most relates to work in image auto-

annotation, learning from multiple input channels (text and imagery), and models of human attention and saliency, as we briefly overview in the following.

One way to exploit tagged or captioned photos is to recover the correspondence (or "translation") between region descriptors and the keywords that appear alongside them [4, 8], or to model the joint distribution of words and regions [2, 21, 28] and scenes [23]. Recent work further shows how exploiting predicates (e.g., isAbove(A,B)) provides even stronger cues tying the two data views together [14]. Having learned such a model, one can predict the labels for new image examples.

Another general strategy is to learn a new image representation (or similarly, distance function) that is bolstered by having observed many examples together with relevant text. To this end, variants of metric learning [27, 29], transfer learning [31], matrix factorization [25], and random field models [3] have all been explored. Previous work has also specifically considered KCCA for this purpose [5, 15, 40]. However, because they limit the text description to a straight bag-of-words, the semantic space returned by KCCA can only associate image content with an appropriate distribution of words; in contrast to the proposed approach, the relative significance of the objects present is ignored.

What objects do people notice most in an image, and what do they tag first? Recent work suggests that there is a close relationship between the relative semantic importance or saliency of an object, and in what order a user tags the image [9, 10, 35]. A specific definition for *importance* is defined in [35]: an object's importance in an image is the probability that it would be named first by a viewer. The authors devise a model for this concept, and demonstrate that one can *predict* the importance of named objects in an image via regression on some intuitive image cues (e.g., scale, saliency, etc.). However, a major restriction of their system is the requirement that all examples (including the novel test inputs) be hand-segmented and labeled by object category.

In our recent work we addressed the inverse problem: given a novel test image, its tags are used to prime an object detector [18]. In that approach, we train with images for which both tags and ground truth bounding boxes are available, learning a prior for the scales and positions of the named objects given a set of rank and proximity cues extracted from the tag words. We directly adapt one of the tag-list features proposed in that work (see Section 3.1).

While these methods help motivate our idea, our target application is distinct: our goal is to provide more accurate image retrieval by virtue of learning about object importance, whereas [18] is concerned with faster object detection and [35] aims to prioritize a list of pre-recognized objects. In addition, our approach requires significantly less supervision than either [35] or [18]. During training it only needs access to tagged images, and at query time it requires only an image *or* a set of tags (depending on whether one performs image-to-image retrieval or a tag-to-image retrieval). To our knowledge, no previous work attempts to improve image retrieval based on importance-based semantics automatically gleaned from tagged images.

3 Approach

Our goal is to provide an image retrieval and auto-tagging system that accounts not only for the objects present in the images, but also their relative significance within the scene. If it works well, then a user's query should map to images where the most "important" components of the scene are preserved, and similarly, auto-generated tags should name the most defining objects first.

A naive approach might be to run a bank of object detectors on all database images, and then filter retrieval results according to the predicted objects' sizes and positions. However, such a technique would entail manually specifying global rules about preferred scales and knowing what preset list of detectors is relevant, and would quickly become unaffordable for large databases.

Instead, we propose a lightweight approach that directly learns the implicit cues about importance from human-tagged image data. First, we collect images of the sort an ordinary user might want to search and organize based on content.[1] Then, we obtain tags for these photos via online annotators, whom are simply asked to name the objects in the scene, but with no requirements as to the number or order in which those tags should be provided. We treat the ordered tag words and the image's visual descriptors (color, local features, etc.) as two views stemming from the common semantics of the image content. To learn a representation that exploits both views and allows us to compute similarities *across* the two views (i.e., so that we may support not only image-to-image retrieval, but also tag-to-image retrieval or image-to-tag auto-annotation), we employ Kernel Canonical Correlation Analysis (KCCA). This algorithm essentially learns two sets of basis functions, one per view, such that correlation is maximized for projections of either view of the same instance. Finally, at test time, we project the novel image or tag query onto this learned semantic space, and rank the database images according to their semantic feature similarity.

Our approach makes two key assumptions: (1) people tend to agree about which objects most define a scene, and (2) the significance and prominence of those objects in turn influence the order in which a person provides image tags. Though difficult to state in the absolute, a number of previous studies lend support for both points [9, 10, 18, 35, 36, 39], as does our own experimental data.

We next briefly define the features and KCCA algorithm, and then describe how we use the combined semantic representation to process three types of queries.

3.1 Tag and Image Features

We examine three types of tag-based features, which together capture the objects present as well as an indirect signal about their inter-relationships in the scene.

Word Frequency. This feature is a traditional bag-of-words. It records which objects are named, and how many times. Supposing V total possible words in the text vocabulary, each tag-list is mapped to a V-dimensional vector $W = [w_1, \ldots, w_V]$, where w_i denotes the number of times the i-th word is mentioned in the list. For tag lists, usually counts are simply 0 or 1. This feature serves to help learn the connection the low-level image features and the objects they refer to (and has been used previously in applications of KCCA [5, 15, 40]).

Relative Tag Rank. This feature encodes the relative rank of each word compared to its typical rank: $R = [r_1, \ldots, r_V]$, where r_i denotes the percentile of the i-th word's rank relative to all its previous ranks observed in the training data. The higher the value, the more this word tops the list relative to where it typically occurs in any other tag-list; if absent, the percentile is 0. This feature captures the order of mention while emphasizing those words that may be atypically prominent, which we assume hints at the relative importance. We used this feature previously in [18], albeit in a strongly supervised learning procedure for the purpose of priming an object detector.

Absolute Tag Rank. This feature encodes the absolute rank of each word: $A = [\frac{1}{\log_2(1+a_1)}, \ldots,$

[1] In particular, we use the PASCAL VOC dataset, which originates from Flickr photo albums.

$\frac{1}{\log_2(1+a_V)}]$, where a_i denotes the average absolute rank of the i-th word in the tag-list. Note that a_i will be 1 if the object is mentioned first, and will drop exponentially towards 0 for lower ranked or absent words. In contrast to the relative rank, this feature more directly captures the importance of each object compared to the others in the *same* scene.

We extract three image features: Gist, color histograms, and bag-of-visual-words histograms. The Gist is a 512-dimensional vector recording the pooled steerable filter responses within a grid of spatial cells across the image [37]. It captures the total scene structure. We use 64-dimensional HSV color histograms, with 8, 4, and 2 bins for hue, saturation, and value, following [5]. The bag-of-words (BOW) summarizes the frequency with which each of a set of prototypical local appearance patches occurs; we use DoG interest point selection and SIFT descriptors [26], and form 200 words with k-means. These local features are useful to capture the appearance of component objects, without the spatial rigidity of Gist.

3.2 Kernel Canonical Correlation Analysis

Given the two views of the data, we are ready to construct their common representation. Canonical Correlation Analysis (CCA) uses data consisting of paired views to simultaneously find projections from each feature space such that correlation between projected features originating from the same instance is maximized [17]. Formally, given samples of paired data $\{(x_1,y_1),\ldots,(x_N,y_N)\}$, where each $x \in \Re^m$ and $y \in \Re^n$ denote the two views, the goal is to select directions $w_x \in \Re^m$ and $w_y \in \Re^n$ so as to maximize the canonical correlation:

$$\max_{w_x,w_y} \frac{\hat{E}[\langle x,w_x\rangle\langle y,w_y\rangle]}{\sqrt{\hat{E}[\langle x,w_x\rangle^2]\hat{E}[\langle y,w_y\rangle^2]}} = \max_{w_x,w_y} \frac{w_x^T C_{xy} w_y}{\sqrt{w_x^T C_{xx} w_x w_y^T C_{yy} w_y}}, \quad (1)$$

where $\hat{E}$ denotes the empirical expectation, C_{xy} denotes the between-sets covariance matrix, and C_{xx} and C_{yy} denote the auto-covariance matrices for x and y data, respectively. Note that in our case, the x view is the visual cue, and the y view is the tag-list cue. The solution can be found via a generalized eigenvalue problem. The CCA method has often been used in cross-language information retrieval, where one queries a document in one language to retrieve relevant documents in another language [24].

Kernel CCA is a kernelized version of CCA. Given kernel functions for both feature spaces, $k_x(x_i,x_j) = \phi_x(x_i)^T\phi_x(x_j)$ and $k_y(y_i,y_j) = \phi_y(y_i)^T\phi_y(y_j)$, one seeks projection vectors in the kernels' implicit feature spaces, which may only be accessed through kernel function evaluations. The solution for w_x and w_y must lie in the span of the data points $\phi_x(x_i)$ and $\phi_y(y_i)$, that is, $w_x = \sum_i \alpha_i\phi_x(x_i)$ and $w_y = \sum_i \beta_i\phi_y(y_i)$. The objective in the kernelized form is then to identify the $\alpha,\beta \in \Re^N$ that maximize

$$\max_{\alpha,\beta} \frac{\alpha^T K_x K_y \beta}{\sqrt{\alpha^T K_x^2 \alpha \beta^T K_y^2 \beta}}, \quad (2)$$

where K_x and K_y denote the $N \times N$ kernel matrices over a sample of N pairs. This can also be formulated as an eigenvalue problem (with additional regularization modifying the above to avoid degenerate solutions), and the top T eigenvectors yield a series of bases $(\alpha^{(1)},\beta^{(1)}),\ldots,(\alpha^{(T)},\beta^{(T)})$ with which to compute the T-dimensional projections for an input x or y. See [16] for details.

We use χ^2 kernels for all component features: $K_{\chi^2}(h_i,h_j) = \exp\left(-\frac{1}{2\Omega}\sum_{k=1}^{d}\frac{(h_i(k)-h_j(k))^2}{h_i(k)+h_j(k)}\right)$, where Ω denotes the mean of the χ^2 distances among the training examples, and d denotes

the dimensionality of the descriptor. We average the K_{χ^2} kernels for the image features, i.e., $k_x(x_i, x_j)$ is an average of the respective K_{χ^2} kernels for Gist, color, and BOW; for $k_y(y_i, y_j)$ we average the K_{χ^2} kernels for our tag features (R and A), and use only W for the baseline.

3.3 Processing Queries and Auto-Annotating Novel Images

KCCA provides a common representation for the visual and tag features. Given a novel visual input q_x, we project onto a single basis specified by α as:

$$w_x^T \phi_x(q_x) = \sum_{i=1}^{N} \alpha_i \phi_x(x_i)^T \phi_x(q_x) = \sum_{i=1}^{N} \alpha_i k_x(x_i, q_x), \tag{3}$$

that is, by evaluating the weighted kernel function between the input and the N sampled training points. Similarly, the projection of a novel tag-list input q_y is given by $w_y^T \phi_y(q_y) = \sum_{i=1}^{N} \beta_i k_y(y_i, q_y)$. The final projection of q_x or q_y onto the T-dimensional semantic space is formed by the vector of these values for $\alpha^{(1)}, \ldots, \alpha^{(T)}$ or $\beta^{(1)}, \ldots, \beta^{(T)}$, respectively.

After projecting all database images onto this learned semantic space, there are three tasks we can perform:

- **Image-to-Image Retrieval:** Given a novel query image, we use its image features only ($\phi_x(x)$) to project onto the semantic space, and then sort all database images relative to it based on their normalized correlation in that space. Compared to the results we'd expect using the database's image features *alone* to rank the data, the results should be favorably skewed towards showing scenes with similarly relevant objects when using our approach.

- **Tag-to-Image Retrieval:** Given a novel tag-list query ($\phi_y(y)$), we project onto the semantic space, and again sort the database images by correlation. Compared to traditional keyword-based image search, we can now expect to retrieve images where not only are objects shared with the query, but they are also of similar relative importance.

- **Image-to-Tag Auto-Annotation:** Given a novel query image, we project onto the semantic space, identify its nearest example among those that are tagged, and return the top keywords on that tag-list. Compared to existing approaches that auto-annotate by predicting labels for each blob in the image, this strategy attempts to provide the most *important* tags based on all available image features.

The primary offline cost is solving the eigenvalue problem for KCCA. Note that the image database may be a superset of the N images used to train KCCA, and need not be fully tagged, since any novel untagged image can be projected onto the learned semantic space. To retrieve the nearest neighbors given a query, we compute a linear scan of the database; a faster sub-linear time implementation could also easily be incorporated—for example, with kernel LSH [20].

4 Experimental Results

In this section, we apply our method for each of the three scenarios outlined above.

Baselines. We compare to three baselines: **Visual-only**, which ranks images relative to an image query according to their visual descriptors only, **Word-only**, which ranks tagged images relative to a keyword query according to tag similarity only, and **Words+Visual**,

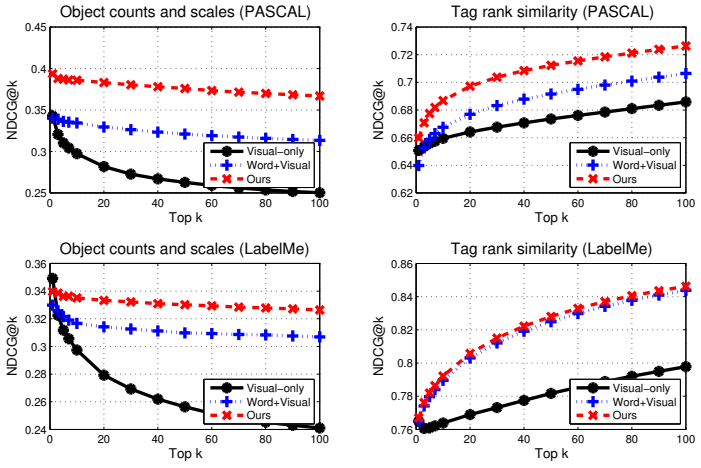


Figure 2: Image-to-image retrieval results for PASCAL (top row) and LabelMe (bottom row). Higher curves are better. By modeling importance cues from the tag-lists when building the semantic space, our method outperforms both a method using image features alone, as well as a semantic space that looks only at unordered keywords.

a strong baseline that builds a KCCA semantic space using the image cues plus a bag-of-keywords W, as in [15, 40]. Our method uses R and A together with the visual cues. All methods use the same visual features and kernels.

Datasets. We consider the PASCAL VOC 2007 [11] and LabelMe datasets. The PASCAL contains more object-centric images, in that each contains few tagged objects—5.5 tags on average among $V = 399$ total words. We use the VOC train and test splits as our database (5011 images) and query examples (4952 images). We use the tags collected in [18] from 758 Mechanical Turkers. LabelMe contains broader scenes (often offices and streets), with many more objects per image. We use the 3825 images compiled in [18] for which there are at least 10 tags (on average each has 23). We run five 50-50 random database-query splits.

Implementation details. We use the KCCA code provided by [16].[2] We fix the learning rate $\mu = 0.5$ and set the regularization parameter by maximizing the difference in correlation between samples of actual and random image-tag pairs (see [16]). We fix $T = 20$ for all KCCA projections. In initial informal tests, we did not find the overall results to be very sensitive to the semantic space dimensionality.

Evaluation metrics. We score the methods using Normalized Discounted Cumulative Gain at top K (NDCG@K) [19], a measure commonly used in information retrieval. It reflects how well a computed ranking agrees with the ideal (ground truth) ranking, and more strongly emphasizes the accuracy of the higher ranked items. Specifically, NDCG@$K = \frac{1}{Z}\sum_{p=1}^{K}\frac{2^{s(p)}-1}{\log(1+p)}$, where Z is a query-specific normalization term, p cycles over the top K ranks, and $s(p)$ denotes the *reward* at rank position p. The score ranges from 0 to 1, where 1 indicates perfect agreement.

We define reward functions for two variants of "ideal" rankings, both intended to reveal how well the retrievals exhibit the query's important objects. The first is *object counts and scales*, where the ideal ranking would sort the images based on the correlation of the presence and relative scales of all objects named in the query's ground truth tag-list; the

[2] http://www.davidroihardoon.com/Research/Code.html

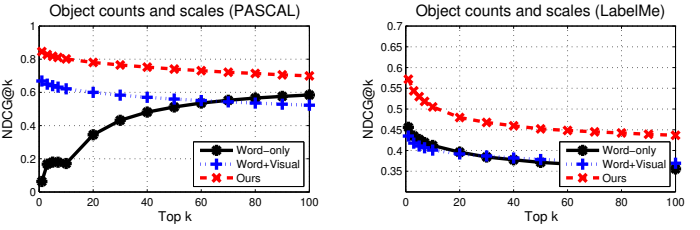


Figure 3: Tag-to-image retrieval results. Given a user's ordered tag-list, our method retrieves images better respecting the objects' relative importance.

Dataset	PASCAL VOC 2007			
Method	k=1	k=3	k=5	k=10
Visual-only	0.0826	0.1765	0.2022	0.2095
Word+Visual	0.0818	0.1712	0.1992	0.2097
Ours	**0.0901**	**0.1936**	**0.2230**	**0.2335**

Dataset	LabelMe			
Method	k=1	k=3	k=5	k=10
Visual-only	0.3940	0.4153	0.4297	0.4486
Word+Visual	0.3996	0.4242	0.4407	0.4534
Ours	**0.4053**	**0.4325**	**0.4481**	**0.4585**

Figure 4: Top: Image-to-tag auto annotation accuracy (F1 score). Bottom: Examples of annotated images. Tags are quite accurate for images that depict typical scenes. Last example is a failure case.

reward function is $s(p) = \frac{1}{2}\left(\frac{\langle S_p, S_q\rangle}{\|S_p\|\,\|S_q\|} + \frac{\langle W_p, W_q\rangle}{\|W_p\|\,\|W_q\|}\right)$, where S_p and W_p are V-dimensional vectors recording the scale (normalized by image size) and count of each object within the p-th retrieved image, respectively, and S_q and W_q are the same for the query. For the second, *tag rank similarity*, the ideal ranking would sort the images based on the agreement between the (withheld) ground truth ordered tag-list features: $s(p) = \frac{1}{2}\left(\frac{\langle R_p, R_q\rangle}{\|R_p\|\,\|R_q\|} + \frac{\langle A_p, A_q\rangle}{\|A_p\|\,\|A_q\|}\right)$, where the subscripts again refer to the p-th ranked and $q$uery images.

Image-to-image retrieval results. First we show that our approach better retrieves images matching the important objects in a query. Figure 2 shows the results for our method and the two baselines. While the Word+Visual semantic space improves over the Visual-only retrievals, our method outperforms both methods, in terms of the object presence and scales (left plots), and the tag-list agreement (right plots). Our method's gains are quite significant on the challenging PASCAL data, which makes sense given that the images contain a clearer mix of more and less prominent objects, and there is wider variance in a given category's importance across images. Looking at the $K = 30$ top-ranked images (the number of images one might fit in a Web page search return), we see our method yields a 39% improvement over the Visual-only result, and about a 17% gain over the Word+Visual semantic space. Figure 5 shows example results illustrating our method's advantages.

In comparison, LabelMe's broad scenes make it less apparent which objects are most important, and instances of the same scene type contain less variance in composition (e.g., office images tend to have the computer in a similarly prominent role). This allows the traditional semantic space using unordered words to be almost as effective—especially under the tag rank similarity scoring. Note that for very small values of K the three methods give similar results, since the first couple matches tend to be very close visually *and* semantically.

Tag-to-image retrieval results. Figure 3 shows the results when we query with a human-provided ordered list of keywords, and return relevant images from the database. Again, the learned semantic space allows us to find the relevant content, this time in a cross-modal

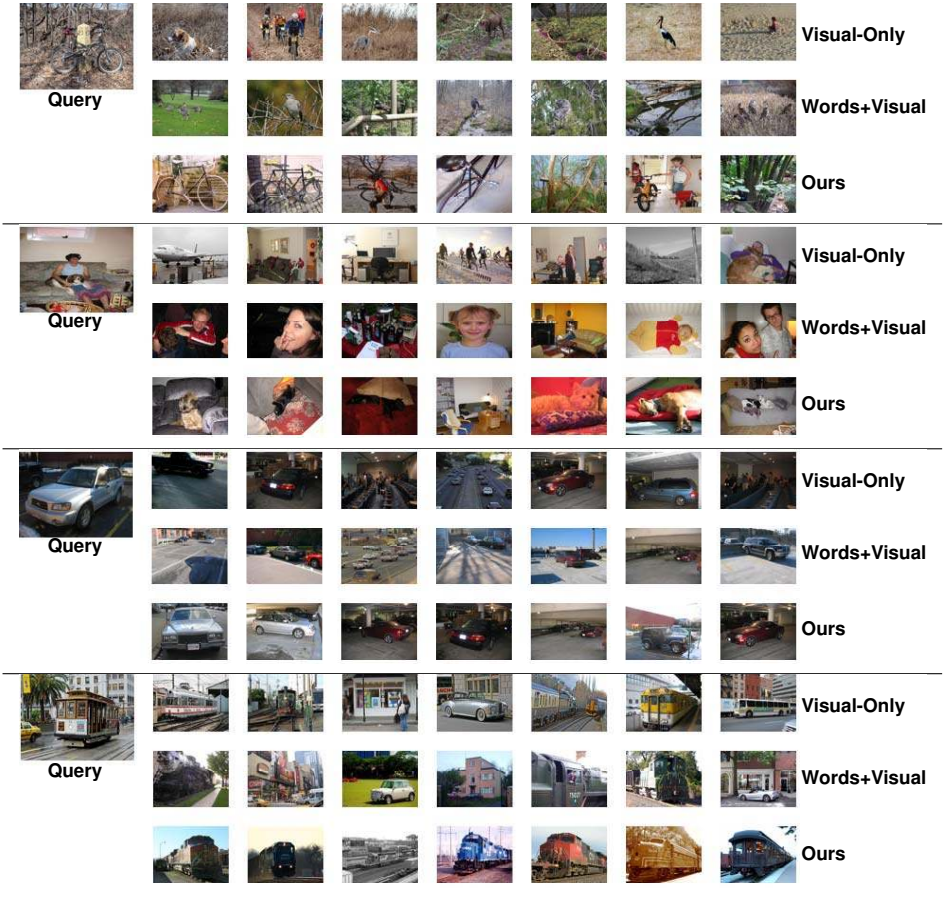


Figure 5: Example image-to-image retrievals for our method and the baselines; leftmost image is query, three rows show top ranked results per method. While a baseline that builds the semantic space from Words+Visual features can often retrieve images with an object set overlapping the query's, ours often better captures the important objects that perceptually define the scene.

format, where the objects emphasized by the human are more likely to be prominent in the results. Our approach makes dramatic improvements over the baselines—31% better than the Words+Visual baseline for $K = 30$ on the PASCAL. (Note that we omit scoring with tag rank similarity for this case, since it would be trivial for the Word-only baseline.)

Image-to-tag auto-annotation results. Finally, we explore auto-annotation, where our method takes an image and generates a list of tags. In contrast to previous work, however, we account for the importance of the tags when scoring the outputs: listing all the objects present is less valuable than listing those that most define the scene. We take the average ranks from the top k neighbors, and sort the keywords accordingly. Figure 4 shows the results as a function of k. Our method outperforms the baselines noticeably on the PASCAL images; differences are more modest on LabelMe, again likely due to the minor variation of importance per object occurrence. Given that PASCAL stems from real Flickr images, it is more realistic for the target setting where a user uploads photos and would like them auto-tagged and indexed. The fact that our results are strongest for this challenging set is therefore quite promising.

5 Conclusions

We proposed an unsupervised approach to learn the connections between human-provided tags and visual features, and showed the impact of accounting for importance in several retrieval and auto-tagging tasks. Our results show our method makes consistent improvements over pure content-based search as well as a method that also exploits tags, but disregards their implicit importance cues. We next plan to explore how richer textual cues from natural language captions might also shape our model of object importance.

References

[1] S. Akaho. A Kernel Method for Canonical Correlation Analysis. In *International Meeting of Psychometric Society*, 2001.

[2] K. Barnard, P. Duygulu, N. de Freitas, D. Forsyth, D. Blei, and M. Jordan. Matching Words and Pictures. *JMLR*, 3:1107–1135, 2003.

[3] R. Bekkerman and J. Jeon. Multi-modal Clustering for Multimedia Collections. In *CVPR*, 2007.

[4] T. Berg, A. Berg, J. Edwards, and D. Forsyth. Who's in the Picture? In *NIPS*, 2004.

[5] M. B. Blaschko and C. H. Lampert. Correlational Spectral Clustering. In *CVPR*, 2008.

[6] R. Datta, D. Joshi, J. Li, and J. Wang. Image Retrieval: Ideas, Influences, and Trends of the New Age. *ACM Computing Surveys*, 40(2):1–60, 2008.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A Large-Scale Hierarchical Image Database. In *CVPR*, 2009.

[8] P. Duygulu, K. Barnard, N. de Freitas, and D. Forsyth. Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary. In *ECCV*, 2002.

[9] W. Einhauser, M. Spain, and P. Perona. Objects Predict Fixations Better than Early Saliency. *Journal of Vision*, 8(14):1–26, 2008.

[10] L. Elazary and L. Itti. Interesting Objects are Visually Salient. *Journal of Vision*, 8(3): 1–15, 2008.

[11] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

[12] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning Object Categories from Google's Image Search. In *ICCV*, 2005.

[13] C. Fyfe and P. Lai. Kernel and Nonlinear Canonical Correlation Analysis. *International Journal of Neural Systems*, 10:365–374, 2001.

[14] A. Gupta and L. Davis. Beyond Nouns: Exploiting Prepositions and Comparative Adjectives for Learning Visual Classifiers. In *ECCV*, 2008.

[15] D. Hardoon and J. Shawe-Taylor. KCCA for Different Level Precision in Content-Based Image Retrieval. In *Third International Workshop on Content-Based Multimedia Indexing*, 2003.

[16] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor. Canonical Correlation Analysis: An Overview with Application to Learning Methods. *Neural Computation*, 16(12), 2004.

[17] H. Hotelling. Relations Between Two Sets of Variants. *Biometrika*, 28:321–377, 1936.

[18] S. J. Hwang and K. Grauman. Reading Between the Lines: Object Localization Using Implicit Cues from Image Tags. In *CVPR*, 2010.

[19] J. Kekalainen K. Jarvelin. Cumulated Gain-based Evaluation of ir Techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

[20] B. Kulis and K. Grauman. Kernelized Locality-Sensitive Hashing for Scalable Image Search. In *ICCV*, 2009.

[21] V. Lavrenko, R. Manmatha, and J. Jeon. A Model for Learning the Semantics of Pictures. In *NIPS*, 2003.

[22] L. Li, G. Wang, and L. Fei-Fei. Optimol: Automatic Online Picture Collection via Incremental Model Learning. In *CVPR*, 2007.

[23] L-J. Li, R. Socher, and L. Fei-Fei. Towards Total Scene Understanding: Classification, Annotation and Segmentation in an Automatic Framework. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, 2009.

[24] Y. Li and J. Shawe-Taylor. Using KCCA for Japenese-English Cross-Language Information Retrieval and Document Classification. *Journal of Intelligent Information Systems*, 27(2), September 2006.

[25] N. Loeff and A. Farhadi. Scene Discovery by Matrix Factorization. In *ECCV*, 2008.

[26] D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *IJCV*, 60(2), 2004.

[27] A. Makadia, V. Pavlovic, and S. Kumar. A New Baseline for Image Annotation. In *ECCV*, 2008.

[28] F. Monay and D. Gatica-Perez. On Image Auto-Annotation with Latent Space Models. In *ACM Multimedia*, 2003.

[29] G. J. Qi, X. S. Hua, and H. J. Zhang. Learning Semantic Distance from Community-Tagged Media Collection. In *Proceedings of the seventeen ACM international conference on Multimedia*, 2009.

[30] T. Quack, B. Leibe, and L. Van Gool. World-scale Mining of Objects and Events from Community Photo Collections. In *CIVR*, 2008.

[31] A. Quattoni, M. Collins, and T. Darrell. Learning Visual Representations Using Images with Captions. In *CVPR*, 2007.

[32] B. Russell, A. Torralba, K. Murphy, and W. Freeman. Labelme: a Database and Web-Based Tool for Image Annotation. Technical report, MIT, 2005.

[33] F. Schroff, A. Criminisi, and A. Zisserman. Harvesting Image Databases from the Web. In *ICCV*, 2007.

[34] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-Based Image Retrieval at the End of the Early Years. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 22(12):1349–1380, 2000.

[35] M. Spain and P. Perona. Some Objects Are More Equal Than Others: Measuring and Predicting Importance. In *ECCV*, 2008.

[36] B. Tatler, R. Baddeley, and I. Gilchrist. Visual Correlates of Fixation Selection: Effects of Scale and Time. *Vision Research*, 45:643–659, 2005.

[37] A. Torralba. Contextual Priming for Object Detection. *IJCV*, 53(2):169–191, 2003.

[38] S. Vijayanarasimhan and K. Grauman. Keywords to Visual Categories: Multiple-Instance Learning for Weakly Supervised Object Categorization. In *CVPR*, 2008.

[39] L. von Ahn and L. Dabbish. Labeling Images with a Computer Game. In *CHI*, 2004.

[40] O. Yakhnenko and V. Honavar. Multiple Label Prediction for Image Annotation with Multiple Kernel Correlation Models. In *Workshop on Visual Context Learning, in conjunction with CVPR*, 2009.