A New Baseline for Image Annotation

Ameesh Makadia[1], Vladimir Pavlovic[2], and Sanjiv Kumar[1]

[1] Google Research, New York, NY
makadia@google.com, sanjivk@google.com
[2] Rutgers University, Piscataway, NJ
vladimir@cs.rutgers.edu

Abstract. Automatically assigning keywords to images is of great interest as it allows one to index, retrieve, and understand large collections of image data. Many techniques have been proposed for image annotation in the last decade that give reasonable performance on standard datasets. However, most of these works fail to compare their methods with simple baseline techniques to justify the need for complex models and subsequent training. In this work, we introduce a new baseline technique for image annotation that treats annotation as a retrieval problem. The proposed technique utilizes low-level image features and a simple combination of basic distances to find nearest neighbors of a given image. The keywords are then assigned using a greedy label transfer mechanism. The proposed baseline outperforms the current state-of-the-art methods on two standard and one large Web dataset. We believe that such a baseline measure will provide a strong platform to compare and better understand future annotation techniques.

1 Introduction

Given an input image, the goal of automatic image annotation is to assign a few relevant text keywords to the image that reflect its visual content. Utilizing image content to assign a richer, more relevant set of keywords would allow one to further exploit the fast indexing and retrieval architecture of Web image search engines for improved image search. This makes the problem of annotating images with relevant text keywords of immense practical interest.

Image annotation is a difficult task for two main reasons: First is the well-known *pixel-to-predicate* or *semantic gap* problem, which points to the fact that it is hard to extract semantically meaningful entities using just low level image features, e.g. color and texture. Doing explicit recognition of thousands of objects or classes reliably is currently an unsolved problem. The second difficulty arises due to the lack of *correspondence* between the keywords and image regions in the training data. For each image, one has access to keywords assigned to the *entire* image and it is not known which regions of the image correspond to these keywords. This makes difficult the direct learning of classifiers by assuming each keyword to be a separate class. Recently, techniques have emerged to circumvent the correspondence problem under a discriminative multiple instance learning paradigm [1] or a generative paradigm [2].

Image annotation has been a topic of on-going research for more than a decade and several interesting techniques have been proposed [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12].

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part III, LNCS 5304, pp. 316–329, 2008.

Most of these techniques define a parametric or non-parametric model to capture the relationship between image features and keywords. Even though some of these techniques have shown impressive results, one thing that is sorely missing in the annotation literature is comparison with very simple 'straw-man' techniques.

The goal of this work is to create a family of baseline measures against which new image annotation methods should be compared to justify the need for more complex models and training procedures. We introduce several simple techniques characterized by minimal training requirements that can efficiently serve this purpose. Surprisingly, we also show that these baseline techniques can outperform more complex state-of-the art annotation methods on several standard datasets, as well as on a large Web dataset.

Arguably, one of the simplest annotation schemes is to treat the problem of annotation as that of image-retrieval. For instance, given a test image, one can find its nearest neighbor (defined in some feature space with a pre-specified distance measure) from the training set, and assign all the keywords of the nearest image to the input test image. One obvious modification of this scheme would be to use K-nearest neighbors to assign the keywords instead of relying on just the nearest one. In the multiple neighbors case, as we discuss in Section 3.3, one can easily assign the appropriate keywords to the input image using a simple greedy approach. As we show in Section 4, some simple distance measures defined on even global image features perform similar to or better than several popular image annotation techniques.

The K-nearest neighbor approach can be extended to incorporate multiple distance measures, possibly defined over distinct feature spaces. Recently, combining different distances or kernels has been shown to yield good performance in object recognition task [13]. In this work, we explore two different ways of linearly combining different distances to create the baseline measures. The first one simply computes the average of different distances after scaling each distance appropriately. The second one is based on selecting relevant distances using a sparse logistic regression method, Lasso [14]. To learn the weights of Lasso, one needs a training set containing *similar* and *dissimilar* images. A typical training set provided for the annotation task does not contain such information directly. We show that one can train Lasso by creating a labeled set from the annotation training data. Even such a weakly trained Lasso outperforms the state-of-the-art methods in most cases. Surprisingly, however, the averaged distance performs better or similar to the noisy Lasso.

The main contributions of our work are that it (1) introduces a simple method to perform image annotation by treating it as a retrieval problem in order to create a new baseline against which annotation algorithms can be measured, and (2) provides exhaustive experimental comparisons of several state-of-the-art annotation methods on three different datasets. These include two standard sets (Corel and IAPR TC-12) and one Web dataset containing about $20K$ images.

2 Prior Work

A large number of techniques have been proposed in the last decade [15]. Most of these treat annotation as translation from image instances to keywords. The translation paradigm is typically based on some model of image and text co-occurrences [3, 16].

The translation approach of [3] was extended to models that ascertain associations indirectly, through latent topic/aspect/context spaces [4, 8] . One such model, the Correspondence Latent Dirichlet Allocation (CorrLDA) [4], considers associations through a latent topic space in a generatively learned model. Despite its appealing structure, this class of models remains sensitive to the choice of topic model, initial parameters, prior image segmentation, and more importantly the inference and learning approximations to handle the typically intractable exact analysis.

Cross Media Relevance Models (CMRM) [5], Continuous Relevance Model (CRM) [7], and Multiple Bernoulli Relevance Model (MBRM) [9] assume different, nonparametric density representations of the joint word-image space. In particular, MBRM achieves robust annotation performance using simple image and text representations: a mixture density model of image appearance that relies on regions extracted from a regular grid, thus avoiding potentially noisy segmentation, and the ability to naturally incorporate complex word annotations using multiple Bernoulli models. However, the complexity of the kernel density representations may hinder MBRM's applicability to large data sets. Alternative approaches based on graph representation of joint queries [11], and cross-language LSI [12], offer means for linking the word-image occurrences, but still do not perform as well as the non-parametric models.

Recent research efforts have focused on extensions of the translation paradigm that exploit additional structure in both visual and textual domains. For instance, [17] utilizes a coherent language model, eliminating independence between keywords. Hierarchical annotations in [18] aim not only to identify specific objects in an image, but also explicitly incorporate concept ontologies. The added complexity, however, makes the models applicable only to limited settings with small-size dictionaries. To address this problem, [19] developed a real-time ALIPR image search engine which uses multiresolution 2D Hidden Markov Models to model concepts determined by a training set. While this method successfully infers higher level semantic concepts based on global features, identification of more specific categories and objects remains a challenge. In an alternative approach, [2] relies on a hierarchical mixture representation of keyword classes, leading to a method that demonstrates both computational efficiency and state-of-the-art performance on several complex annotation tasks. However, the annotation problem is treated as a set of one-vs-all binary classification problems, potentially failing to benefit from competition among models during the learning stage.

Even though promising results have been reported by many sophisticated annotation techniques, they commonly lack a comparison with simple baseline measures across diverse image datasets. In the absence of such a comparison, it is hard to understand the gains and justify the need for complex models and training processes as required by most of the current annotation methods. Our work addresses this issue by suggesting a family of baseline measures, some of which surprisingly outperform the current state-of-the-art in image annotation on several large real-world datasets.

3 Baseline Methods

We propose a family of baseline methods for image annotation that are built on the hypothesis that images similar in appearance are likely to share keywords. We treat

image annotation as a process of transferring keywords from nearest neighbors. The neighborhood structure is constructed using simple low-level image features resulting in a rudimentary baseline model. The details are given below.

3.1 Features and Distances

Color and texture are recognized as two important low-level visual cues for image representation. The most common color descriptors are based on coarse histograms, which are frequently utilized within image matching and indexing schemes, primarily due to their effectiveness and ease of computation. Image texture is commonly captured with Wavelet features. In particular, Gabor and Haar wavelets have been shown to be quite effective in creating sparse yet discriminative image features. To limit the influence and biases of individual features, and to maximize the amount of information extracted, we choose to employ a number of simple and easy to compute features.

Color. We generate features from images in three different color spaces: RGB, HSV, and LAB. While RGB is the default color space for image capturing and display, both HSV and LAB isolate important appearance characteristics not captured by RGB. For example, the HSV (Hue, Saturation, and Value) colorspace encodes the amount of light illuminating a color in the Value channel, and the Luminance channel of LAB is intended to reflect the human perception of brightness. The RGB, HSV, and LAB features are 16-bin-per-channel histograms in their respective colorspaces. To determine the corresponding distance measures, we evaluated four measures commonly used for histograms and distributions (KL-divergence, χ^2 statistic, L_1-distance, and L_2-distance) on the human-labeled training data from the Corel5K dataset. L_1 performed the best for RGB and HSV, while KL-divergence was found suitable for LAB distances. Throughout the remainder of the paper, RGB and HSV distances imply the L_1 measure, and the LAB distance implies KL-divergence.

Texture. We represent the texture with Gabor and Haar Wavelets. Each image is filtered with Gabor wavelets at three scales and four orientations. From each of the twelve response images, a histogram over the response magnitudes is built. The concatenation of these twelve histograms is a feature vector we refer to as 'Gabor'. The second feature captures the quantized Gabor phase. The phase angle at each response pixel is averaged over 16×16 blocks in each of the twelve Gabor response images. These mean phase angles are quantized to 3 bits (eight values), and are concatenated into a feature vector referred to as 'GaborQ'.

Haar Wavelet responses are generated by block-convolution of an image with Haar filters at three different orientations (horizontal, diagonal, and vertical). Responses at different scales were obtained by performing the convolution with a suitably subsampled image. After rescaling an image to 64x64 pixels, a Haar feature is generated by concatenating the Haar response magnitudes (this feature is referred to as 'Haar'). As with the Gabor features, we also consider a quantized version, where the sign of the Haar responses are quantized to three values (either 0, 1, or -1 if the response is zero, positive, or negative, respectively). Throughout the text this quantized feature is referred to as 'HaarQ.' We use L_1 distance for all the texture features.

3.2 Combining Distances

Joint Equal Contribution (JEC). If labeled training data is unavailable, or the labels are extremely noisy, the simplest way to combine distances from different descriptors would be to allow each individual distance to contribute equally (after scaling the individual distances appropriately). Let I_i be the i-th image, and say we have extracted N features $f_i^1, \ldots, f_i^N$. Let us define $d^k_{(i,j)}$ as the distance between f_i^k and f_j^k. We would like to combine the individual distances $d^k_{(i,j)}, k = 1, \ldots, N$ to provide a comprehensive distance between image I_i and I_j. Since, in JEC, each feature contributes equally towards the image distance, we first need to find the appropriate scaling terms for each feature. These scaling terms can be determined easily if the features are normalized in some way (e.g., features that have unit norm), but in practice this is not always the case. We can obtain estimates of the scaling terms by examining the lower and upper bounds on the feature distances computed on some training set. We scale the distances for each feature such that they are bounded by 0 and 1. If we denote the scaled distance as $\tilde{d}^k_{(i,j)}$, we can define the comprehensive image distance between images I_i and I_j as $\sum_{k=1}^{N} \frac{\tilde{d}^k_{(i,j)}}{N}$. We refer to this distance as Joint Equal Contribution (JEC).

L_1-Penalized Logistic Regression (Lasso [14]). Another approach to combining feature distances would be to identify those features that are more relevant for capturing image similarity. This is the well-known problem of feature selection. Since we are using different color (and texture) features that are not completely independent, it is an obvious question to ask: Which of these color (or texture) features are redundant? Logistic regression with L_1 penalty, also known as Lasso [14], provides a simple way to answer this question.

The main challenge in applying Lasso to image annotation lies in creating a training set containing pairs of similar and dissimilar images. Typical training datasets for image annotation contain images and associated text keywords, and there is no direct notion of similarity between images. In this setting, we consider any pair of images that share enough keywords to be a positive training example, and any pair with no keywords in common to be a negative example. Clearly, the quality of such a training set will depend on the number of keywords required to match before an image pair can be called 'similar.' In this work, we obtained training samples from the designated training set of the Corel5K benchmark (Section 4). Images pairs that had at least four common keywords were treated as positive samples for training, and those with no common keywords were used as negative samples (training samples are illustrated in Fig. 1).

Combining basic distances using JEC or Lasso gives us a simple way to compute distances between images. Using such composite distances, one can find the K nearest neighbors of an image. In the next section, we present a label transfer algorithm that assigns keywords to any test image given its nearest neighbors.

3.3 Label Transfer

We propose a simple method to transfer n keywords to a query image $\tilde{I}$ from the query's K nearest neighbors in the training set. Let $I_i, i = 1, \ldots, K$ be these K nearest neighbors, ordered by increasing distance (i.e., I_1 is the most similar image). The number of

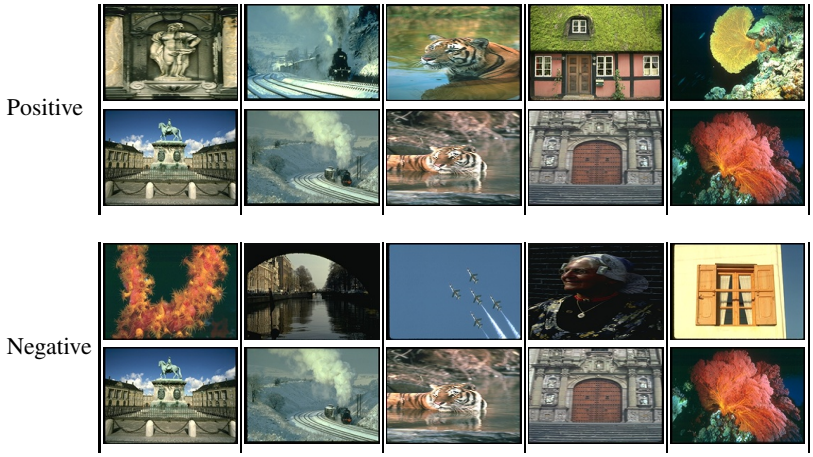


Fig. 1. Pairs of images that were used as positive training examples (top row) and negative training examples (bottom row) for Lasso. In positive pairs the images shared at least 4 keywords, while in negative pairs they shared none.

keywords associated with I_i is denoted by $|I_i|$. Following are the steps of our greedy label transfer algorithm.

1. Rank the keywords of I_1 according to their frequency in the training set.
2. Of the $|I_1|$ keywords of I_1, transfer the n highest ranking keywords to query $\tilde{I}$. If $|I_1| < n$, proceed to step 3.
3. Rank the keywords of neighbors I_2 through I_K according to two factors: 1) co-occurrence in the training set with the keywords transferred in step 2, and 2) local frequency (i.e. how often they appear as keywords of images I_2 through I_K). Select the highest ranking $n - |I_1|$ keywords to transfer to $\tilde{I}$.

This transfer algorithm is somewhat different from other obvious choices. One can imagine simpler algorithms where keywords are selected simultaneously from the entire neighborhood (i.e., all the neighbors are treated equally), or where the neighbors are weighted according to their distance from the test image. However, an initial evaluation showed that these simple approaches underperform in comparison to our two-stage transfer algorithm (see Section 4).

In summary, our baseline annotation methods are comprised of a composite image distance measure (JEC or Lasso) for nearest neighbor ranking, combined with our label transfer algorithm. Is there any hope to achieve reasonable results for image annotation using such simplistic methods? To answer this question, we evaluate our baseline methods on three different datasets as described in the following section.

4 Experiments and Discussion

Our experiments examined the performance and behavior of the proposed baselines for image annotation on three collections of images.

- Corel5K [3] has become a de-facto evaluation benchmark in the image annotation community. It contains 5,000 images collected from the larger Corel CD set, split into 4,500 training and 500 test examples. Each image is annotated with an average of 3.5 keywords, and the dictionary contains 260 words that appear in both the train and the test set.
- IAPR TC-12 is a collection of 19,805 images of natural scenes that include different sports and actions, photographs of people, animals, cities, landscapes and many other aspects of contemporary life[1]. Unlike other similar databases, images in IAPR TC-12 are accompanied by free-flowing text captions. While this set is typically used for cross-language retrieval, we have concentrated on the English captions and extracted keywords (nouns) using the TreeTagger part-of-speech tagger[2]. This resulted in a dictionary size of 291 and an average of 4.7 keywords per image. 17,825 images were used for training, and the remaining 1,980 for testing. Samples from IAPR are depicted in Fig. 2.
- ESP Game consists of a set of 21,844 images collected in the ESP collaborative image labeling task [20][3]. In ESP game, two players assign labels to the same image without communicating. Only common labels are accepted. As an image is shown to more teams, a list of taboo words is accumulated, increasing the difficulty for future players and resulting in a challenging dataset for annotation. The set we obtained[4] contains a wide variety of images annotated by 269 keywords, and is split into 19,659 train and 2,185 test images. Each image is associated with up to 15 keywords, and on average 4.6 keywords. Examples are shown in Fig. 3.

For the IAPR TC-12 and ESP datasets, we have made public the dictionaries, as well as the training and testing image set partitions, used in our evaluations[5]. On all three

Fig. 2. Sample IAPR data. On the left are 25 randomly selected images from the dataset. On the right is a single image and its associated annotation. Noun extraction from the caption provides keywords for annotation.

[1] http://eureka.vu.edu.au/~grubinger/IAPR/TC12_Benchmark.html
[2] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger
[3] http://www.espgame.org
[4] http://hunch.net/~jl/
[5] http://www.cis.upenn.edu/~makadia/annotation/

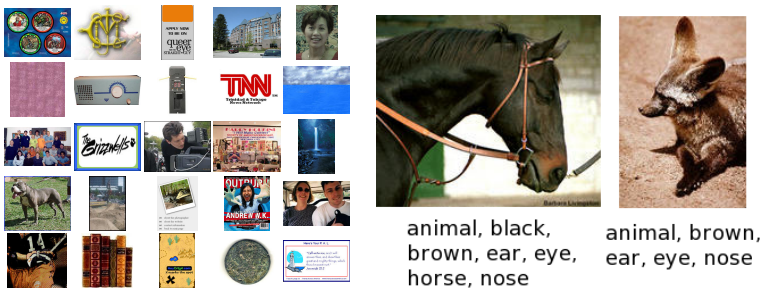


Fig. 3. Sample ESP data. On the left are 25 randomly selected images from the dataset, while on the right are two images and their associated keywords. These images are quite different in appearance and content, but share many of the same keywords.

annotation datasets, we evaluated the performance of a number of baseline methods. For comparisons on Corel5K, we summarized published results of several approaches, including the most popular topic model (i.e. CorrLDA [4]), as well as MBRM [9] and SML [2], which have shown state-of-the-art performance on Corel5K. On the IAPR TC-12 and ESP datasets, where no published results of annotation methods are available, we compared the performance of our baseline methods against MBRM [9] which was relatively easier to implement and had comparable performance to SML [2][6].

When evaluating performance of baseline methods, we focused on three different settings: 1) performance of individual distance measures, 2) performance of the learned weighted distance model (Lasso), and 3) performance of the Joint Equal Contribution (JEC) model, where all features contributed equally to the global distance measure. In the Corel setting, we also examined the impact of leaving-out one distance measure at a time in the JEC model.

Performance of all models was evaluated using five measures following the methodology used in [2, 9]. We report mean precision (P%) and mean recall (R%) rates obtained by different models, as well as the number of total keywords recalled (N^+). Precision and recall are defined in the standard way: the annotation precision for a keyword is defined as the number of images assigned the keyword correctly divided by the total number of images predicted to have the keyword. The annotation recall is defined as the number of images assigned the keyword correctly, divided by the number of images assigned the keyword in the ground-truth annotation. Similar to other approaches, we assign top 5 keywords to each image using label transfer. Additionally, we report two retrieval performance measures based on top 10 images retrieved for each keyword: mean retrieval precision (rP%) and mean retrieval precision for only the recalled keywords (rP^+%) [2].

4.1 Corel

The results of experiments on the Corel set are summarized in Table 1(a). The top portion of the table displays published results of a number of standard and top-performing

[6] No implementation of SML [2] was publicly available.

Table 1. Results on three datasets for different annotation algorithms. Corel5K contains 5,000 images and 260 keywords, IAPR-TC12 has 19,805 images and 291 keywords, and ESP has 21,844 images and 268 keywords. P% and R% denote the mean precision and the mean recall, respectively, over all keywords in percentage points. N^+ denotes the number of recalled keywords. rP%, and rP^+% denote the mean retrieval precision for all keywords and the mean retrieval precision for recalled keywords only, respectively. Note that the proposed simple baseline technique (JEC) outperforms state-of-the-art techniques in all datasets. CorrLDA[1] and JEC[1] correspond to models built on a reduced 168 keyword dictionary, as in [4].

(a) Corel5K

Method	$P_{\%}$	$R_{\%}$	N^+	$rP_{\%}$	$rP^+_{\%}$
CRM[7]	16	19	107	-	-
InfNet [11]	17	24	112	-	-
NPDE [21]	18	21	114	-	-
MBRM [9]	24	25	122	30	35
SML [2]	23	29	137	31	49
CorrLDA[4][1]	6	9	59	27	37
RGB	20	23	110	24	49
HSV	18	21	110	23	45
LAB	20	25	118	25	47
Haar	6	8	53	12	33
HaarQ	11	13	87	16	35
Gabor	8	10	72	11	31
GaborQ	5	6	52	7	26
Lasso	24	29	127	30	51
JEC	**27**	**32**	**139**	**33**	**52**
JEC[1]	**32**	**40**	**113**	**35**	**48**

(b) IAPR-TC12 & ESP

	IAPR-TC12					ESP				
Method	$P_{\%}$	$R_{\%}$	N^+	$rP_{\%}$	$rP^+_{\%}$	$P_{\%}$	$R_{\%}$	N^+	$rP_{\%}$	$rP^+_{\%}$
MBRM	24	23	223	24	30	18	19	209	18	24
RGB	24	24	233	23	29	20	22	212	19	25
HSV	20	20	215	18	24	18	20	212	17	21
LAB	24	25	232	23	29	20	22	221	20	24
Haar	20	11	176	21	**32**	21	18	205	**21**	**27**
HaarQ	19	16	189	18	28	18	19	207	18	24
Gabor	15	15	183	14	22	15	16	186	15	21
GaborQ	8	9	137	9	18	14	15	193	13	19
Lasso	**28**	**29**	246	26	31	21	24	224	**21**	25
JEC	**28**	**29**	**250**	**27**	31	**22**	**25**	**224**	**21**	25

methods that approach the annotation problem from different perspectives, using different image representations: CRM [7], InfNet [11], NPDE [21], MBRM [9], SML [2], and CorrLDA [4]. The middle part of the table shows results of using only the distance measures induced by individual features. Finally, the bottom rows list results of the baseline methods that rely on combinations of distances from multiple features. Individual feature distances show a wide spread in performance scores, ranging from high-scoring LAB and RGB color measures to the potentially less effective Haar and GaborQ. It is interesting to note that some of the best individual measures perform on par or better than several more complex published methods. More surprising, however, is that the measures which arise from combinations of individual distances (Lasso and JEC) perform significantly better than most other published methods. In particular, JEC, which emphasizes equal contribution of all the feature distances, shows domination in all five performance measures. One reason for this exceptional performance may be due to the use of a wide spectrum of different features, contributing along different "orthogonal" factors. This also points to the well-understood inadequacies and limitations of most image representation models that rely on individual or small subsets of features. Figure 4 shows some images annotated using the JEC baseline. Additionally, we show some retrieval examples using the JEC baseline in Fig. 5.

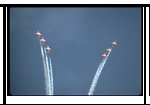
					
Predicted keywords	sky, jet, plane, smoke, formation	grass, rocks, sand, valley, canyon	sun, water, sea, waves, birds	water, tree, grass, deer, white-tailed	bear, snow, wood, deer, white-tailed
Human annotation	sky, jet, plane, smoke	rocks,sand, valley, canyon	sun, water, clouds, birds	tree, forest, deer, white-tailed	tree, snow, wood, fox

Fig. 4. Predicted keywords using JEC versus the human annotations for a sampling of images in the Corel5K dataset (using all 260 keywords)

Fig. 5. Retrieval results using JEC on Corel5K. Each row displays the first seven images retrieved for a query. From top to bottom, the queries are: *sky, street, mare, train*.

It should be noted that most top-performing methods in literature rely on instance-based representations (such as MBRM, CRM, InfNet, and NPDE) which are closely related to our baseline approach. While generative parametric models such as CorrLDA [4] have significant modeling appeal due to the interpretability of the learned models, they fail to stack up to the nonparametric representations on this difficult task. Table 1 confirms that the gap between the two paradigms remains large.

Another interesting result is revealed by comparing JEC with Lasso. One may expect the learned weights through Lasso to perform better than the equal contributions in JEC. However, this is not the case, in part, because of the different requirements posed by the two models. Lasso relies on the existence of sets of positive (similar) and negative (dissimilar) pairs of images, while JEC is a learning-free model. Since the Lasso training set was created artificially from the annotation training set, the effect of noisy labels undoubtedly reflects on the model's performance.

We further contrast the role of individual features and examine their contribution to the combined baseline models in experiments summarized in Tables 2(a) and 2(b). Performance of individual features shown in Table 1 may tempt one to leave out the low-performing features, such as the texture-based Haar and Gabor descriptors. However, Table 2(a) suggests that this is not a wise thing to do. Correlated features, such

Table 2. (a) All-but-one testing of the JEC scheme. In each row, a different feature was left out of JEC. It is clear from these results that all seven features make some positive contribution to the combined distances. The last row shows the JEC results for the full set of features for reference. (b) Texture vs. color results for 260 keywords in Corel5K. The texture feature is a weighted average of all four texture features, and the color feature is a weighted average of all three color features. The third row shows the full JEC results with all the texture and color features.

(a) All-but-one

Feature held out	P%	R%	N^+	rP%	rP^+%
RGB	27	31	134	32	53
HSV	27	31	137	32	52
LAB	27	32	134	33	53
Haar	26	31	133	32	54
HaarQ	26	30	130	31	53
Gabor	25	29	128	30	53
GaborQ	26	31	134	33	53
None	27	32	139	33	52

(b) Texture & Color

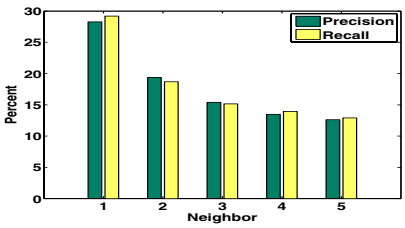
Feature Class	P%	R%	N^+	rP%	rP^+%
Texture	16	19	101	24	45
Color	23	26	120	27	51
Texture + Color	27	32	139	33	52

Table 3. Evaluation of alternative label transfer schemes on Corel5K. In (a), we assess two simple methods. *All neighbors equal* simultaneously selects keywords from all 5 nearest neighbors. Keywords are ranked by their frequency in the neighborhood. *All neighbors weighted* applies an additional weighting relative to the distance of the neighbor from the test image. In (b), we evaluate the individual neighbors in isolation (i.e. all keywords transferred from a single neighbor).

(a) Alternative label transfer methods

	P%	R%	N^+	rP%	rP^+%
All neighbors equal	23	24	113	39	56
All neighbors weighted	25	31	135	32	50
Proposed method (Section 3.3)	27	32	139	33	52

(b) Single-neighbor performance



as HSV and LAB may contribute little jointly and could potentially be left out. While the texture-based descriptors lead to individually inferior annotation performance, they complement the color features. A similar conclusion may be reached when considering joint performance of all color and all texture features separately, as depicted in Table 2(b): either of the two groups alone results in performance inferior to the JEC combined model.

Finally, as mentioned earlier, the greedy label transfer algorithm utilized in JEC is not immediately obvious. One straightforward alternative is to transfer all keywords simultaneously from the entire neighborhood while optionally weighting the neighbors according to their distance from the test image. Additionally, by evaluating the labels transferred from a single neighbor, we can estimate the average "quality" of neighbors in isolation. These results are summarized in Table 3. The simple alternative of selecting

					
Predicted keywords	clothes, jean, man, shop, square	edge, front, glacier, life, tourist	court, player, sky, stadium, tennis	brick, grave, mummy, stone, wall	desert, grass, mountain, sky, slope
Human annotation	clothes, jean, man, pavement, shop, square	glacier, jacket, life, rock, sky, water, woman	court, player, sky, stadium, man, tennis	brick, grave, mummy, wall	desert, grey mountain, round, stone

Fig. 6. Predicted keywords using JEC versus human annotations for sample images in the IAPR dataset

all keywords simultaneously from the entire neighborhood (with and without weighting the neighbors) underperforms our proposed label transfer algorithm. Regarding individual neighbors, the difference in performance between the first two neighbors is greater than the difference between the second and fifth neighbor. This observation led us to treat the first neighbor specially.

4.2 IAPR TC-12

The Corel set has served as a common evaluation platform for many annotation methods. Nevertheless, it if often criticized for its bias due to insufficiently varying appearance and contrived annotations. We therefore measure performance of our baseline models, JEC and Lasso, as well as that of individual features on a more challenging IAPR set. Table 1(b) depicts performance of different methods on this set. Figure 6 shows some examples of annotated images using the JEC baseline.

Trends similar to those observed on the Corel set carry over to the IAPR setting: the JEC baseline leverages multiple, potentially "orthogonal" factors, to retrieve neighboring images most relevant for predicting reasonable annotation of queries. The baseline also shows performance superior to that of the MBRM. While color features contribute consistently more than the texture descriptors, we observe improved individual performance of Gabor and Haar measures. This can be due to the presence of a larger number of images exhibiting textured patterns in IAPR compared to the Corel set. It is also interesting to note that selection of relevant features using Lasso exhibits performance on par with JEC in two out of the five measures. This is a potential indicator that the selection criterion for determining the Lasso training set may be more reflective of the true image similarities in IAPR than in Corel.

4.3 ESP

ESP game set has arisen from an experiment in collaborative human computing—annotation of images in this case [20]. An advantage of this set, compared to Corel and IAPR, lies in the fact that its human annotation elicits a collective semantic agreement among annotators, leading to annotations with less individual bias. Table 1(b) depicts

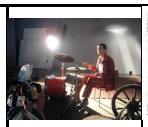
					
Predicted keywords	bikini, girl, grass, hair, woman	bear, black, brown, nose, white	band, light, man, music, play	man, old, picture, red, wall	cloud, grass, green, hill, red
Human annotation	bed, girl, woman	animal, bear, black, brown, head, nose	band, light, man, music, red, wheel	black, man, old, red, sit	cloud, gray, green, mountain, picture, rock, sky, stone

Fig. 7. Predicted keywords using JEC versus human annotations for sample images in the ESP dataset

results of MBRM and our baseline methods on this set. Figure 7 shows some examples of annotated images using JEC.

Even though JEC again gives the best performance, the overall low precision and recall rates for this dataset indicate its difficult nature. Also, more so than in other sets, the texture features play a critical role in the process. For instance, the Haar and Gabor distances fall not far behind the color features.

4.4 Discussion

To be able to solve the image annotation problem at the human level, perhaps one needs to first solve the problem of scene understanding. However, identifying objects, events, and activities in a scene is still a topic of intense research with limited success. The goal of our work was not to develop a new annotation method but to create a family of very simple and intuitive baseline methods. Experiments on three different datasets reaffirm the enormous importance of considering multiple sources of evidence to bridge the gap between the pixel representations of images and the semantic meanings. It is clear that a simple combination of basic distance measures defined over commonly used image features can effectively serve as a baseline method to provide a solid test-bed for developing future annotation methods.

Acknowledgments. Our thanks to Ni Wang for the Lasso training code and Henry Rowley for helpful discussions on feature extraction.

References

1. Yang, C., Dong, M., Hua, J.: Region-based image annotation using asymmetrical support vector machine-based multiple-instance learning. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (2006)
2. Carneiro, G., Chan, A.B., Moreno, P.J., Vasconcelos, N.: Supervised learning of semantic classes for image annotation and retrieval. IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (2007)

3. Duygulu, P., Barnard, K., de Freitas, J.F.G., Forsyth, D.A.: Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In: European Conference on Computer Vision, pp. 97–112 (2002)
4. Blei, D.M., Jordan, M.I.: Modeling annotated data. In: Proc. ACM SIGIR, pp. 127–134 (2003)
5. Jeon, J., Lavrenko, V., Manmatha, R.: Automatic image annotation and retrieval using cross-media relevance models. In: ACM SIGIR Conf. Research and Development in Informaion Retrieval, New York, NY, USA, pp. 119–126 (2003)
6. Wang, L., Liu, L., Khan, L.: Automatic image annotation and retrieval using subspace clustering algorithm. In: ACM Int'l Workshop Multimedia Databases (2004)
7. Lavrenko, V., Manmatha, R., Jeon, J.: A model for learning the semantics of pictures. In: Advances in Neural Information Processing Systems, vol. 16 (2004)
8. Monay, F., Gatica-Perez, D.: On image auto-annotation with latent space models. In: ACM Int'l Conf. Multimedia, pp. 275–278 (2003)
9. Feng, S.L., Manmatha, R., Lavrenko, V.: Multiple bernoulli relevance models for image and video annotation. In: IEEE Conf. Computer Vision and Pattern Recognition (2004)
10. Barnard, K., Johnson, M.: Word sense disambiguation with pictures. Artificial Intelligence 167, 13–30 (2005)
11. Metzler, D., Manmatha, R.: An inference network approach to image retrieval. In: Image and Video Retrieval, pp. 42–50. Springer, Heidelberg (2005)
12. Hare, J.S., Lewisa, P.H., Enserb, P.G.B., Sandomb, C.J.: Mind the gap: Another look at the problem of the semantic gap in image retrieval. Multimedia Content, Analysis, Management and Retrieval (2006)
13. Frome, A., Singer, Y., Sha, F., Malik, J.: Learning globally-consistent local distance functions for shape-based image retrieval and classification. In: Proceedings of the IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil (2007)
14. Tibshirani, R.: Regression shrinkage and selection via the Lasso. J. Royal Statistical Soc. B 58, 267–288 (1996)
15. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image retrieval: Ideas, influences, and trends of the new age. ACM Computing Surveys (2008)
16. Mori, Y., Takahashi, H., Oka, R.: Image-to-word transformation based on dividing and vector quantizing images with words. In: First International Workshop on Multimedia Intel ligent Storage and Retrieval Management (MISRM) (1999)
17. Jin, R., Chai, J.Y., Si, L.: Effective automatic image annotation via a coherent language model and active learning. In: ACM Multimedia Conference, pp. 892–899 (2004)
18. Gao, Y., Fan, J.: Incorporating concept ontology to enable probabilistic concept reasoning for multi-level image annotation. In: 8th ACM International Workshop on Multimedia Information Retrieval, pp. 79–88 (2006)
19. Li, J., Wang, J.: Automatic linguistic indexing of pictures by a statistical modeling approach. IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (2003)
20. von Ahn, L., Dabbish, L.: Labeling images with a computer game. In: ACM CHI 2004 (2004)
21. Yavlinsky, A., Schofield, E., Ruger, S.: Automated Image Annotation Using Global Features and Robust Nonparametric Density Estimation. In: Leow, W.-K., Lew, M., Chua, T.-S., Ma, W.-Y., Chaisorn, L., Bakker, E.M. (eds.) CIVR 2005. LNCS, vol. 3568. Springer, Heidelberg (2005)