Automatic Image Annotation and Retrieval using Cross-Media Relevance Models

J. Jeon, V. Lavrenko and R. Manmatha
Center for Intelligent Information Retrieval
Computer Science Department
University of Massachusetts
Amherst, MA 01003
[jeon,lavrenko,manmatha]@cs.umass.edu

ABSTRACT

Libraries have traditionally used manual image annotation for indexing and then later retrieving their image collections. However, manual image annotation is an expensive and labor intensive procedure and hence there has been great interest in coming up with automatic ways to retrieve images based on content. Here, we propose an automatic approach to annotating and retrieving images based on a training set of images. We assume that regions in an image can be described using a small vocabulary of blobs. Blobs are generated from image features using clustering. Given a training set of images with annotations, we show that probabilistic models allow us to predict the probability of generating a word given the blobs in an image. This may be used to automatically annotate and retrieve images given a word as a query. We show that relevance models. allow us to derive these probabilities in a natural way. Experiments show that the annotation performance of this cross-media relevance model is almost six times as good (in terms of mean precision) than a model based on word-blob co-occurrence model and twice as good as a state of the art model derived from machine translation. Our approach shows the usefulness of using formal information retrieval models for the task of image annotation and retrieval.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Retrieval Models; I.4.8 [**Image Processing and Computer Vision**]: Scene Analysis—*Object Recognition*

General Terms

Algorithms, Measurement, Experimentation

Keywords

Image annotation, image retrieval, relevance models

SIGIR'03, July 28–August 1, 2003, Toronto, Canada.

1. INTRODUCTION

Efficient access to multimedia information requires the ability to search and organize the information. While, the technology to search text has been available for some time - and in the form of web search engines is familiar to many people - the technology to search images and videos, is much more challenging. Several researchers (see [8] for a review) have investigated techniques to retrieve images based on their content but many of these approaches require the user to query based on image concepts like color or texture which most people are not familiar with. In general, people would like to pose semantic queries using textual descriptions and find images relevant to those semantic queries. For example, one should be able to pose a query like "find me all images of tigers in grass". This is difficult if not impossible with many of these image retrieval systems and hence has not led to widespread adoption of these systems. The traditional solution to this problem, used by libraries and other organizations is to annotate such images manually and then search those annotations. Although this allows semantic image retrieval manual annotations are expensive and do not always capture the content of images and videos well.

One approach to automatically annotating images is to look at the probability of associating words with image regions. Mori *et al.* [15] used a *Co-occurrence Model* in which they looked at the co-occurrence of words with image regions created using a regular grid. More recently, a few other researchers [2, 3] have also examined the problem using machine learning approaches. In particular Duygulu *et al* [9] proposed to describe images using a vocabulary of blobs. Each image is generated by using a certain number of these blobs. Their *Translation Model* - a substantial improvement on the Co-occurence Model - assumes that image annotation can be viewed as the task of translating from a vocabulary of blobs to a vocabulary of words. Given a set of annotated training images, they show how one can use one of the classical machine translation models suggested by Brown *et al.* [5] to annotate a test set of images.

Isolated pixels or even regions in an image are often hard to interpret. It is the context in which an image region is placed that gives it meaning. Query expansion is a standard technique for reducing ambiguity in information retrieval. One approach to doing this is to perform an initial query and then expand queries using terms from the top relevant documents (often approximated by the top documents). This expanded query when used for retrieval increases the perfor-

mance substantially. In the image context, tigers are more often associated with grass, water, trees or sky and less often with objects like cars or computers and we want to take advantage of this context.

Relevance-based language models [13, 14] were introduced to allow query expansion to be performed in a more formal manner. These models have been successfully used for both ad-hoc retrieval and cross-language retrieval. Here, we investigate the problem of automatically annotating images as well as the ranked retrieval of images using a modification of the relevance model. As in Duygulu *et al* [9] we assume that every image may be described using a small vocabulary of blobs. Using a training set of annotated images, we learn the joint distribution of blobs and words which we call a cross-media relevance model (CMRM) for images. There are two ways this model can be used. In the first case, which corresponds to document based expansion, the blobs corresponding to each test image are used to generate words and associated probabilities from the joint distribution of blobs and words. Each test image can, therefore, be annotated with a vector of probabilities for all the words in the vocabulary. We call this the probabilistic annotation-based cross-media relevance model (PACMRM). Given a query word, this model can be used to rank the images using a language modeling approach [12, 11, 4, 18]. While this model is useful for *ranked* retrieval, it is less useful for people to look at. Fixed length annotations can be generated by using the top N (N = 3, 4 or 5) words (without their probabilities) to annotate the images. This model is called the fixed annotation-based cross-media relevance model (FACMRM). FACMRM is not useful for *ranked* retrieval (since there are no probabilities associated with the annotations) but is easy for people to use when the number of annotations is small.

In the second case, which corresponds to query expansion, the query word(s) is used to generate a set of blob probabilities from the joint distribution of blobs and words. This vector of blob probabilities is compared with the vector of blobs for each test image using Kullback-Liebler (KL) divergence and the resulting KL distance is used to rank the images. We call this model the direct-retrieval cross-media relevance model (DRCMRM).

We should point out that cross-media relevance models are not translation models in the sense of translating words to blobs. Instead, these models take advantage of the joint distribution of words and blobs - that is the fact that an image can be described both using image features (blobs) and text (words). As Duygulu *et al.*[9] point out the problem of object recognition can be viewed as one of assigning "names" or words to images or image regions. In our model, we assign words to entire images and not to specific blobs because the blob vocabulary can give rise to many errors.

Our annotation-based model performs much better than either the Co-occurrence Model or the Translation Model on the same dataset (same training and test images and the same features). Specifically, for the top 49 annotations, we show that FACMRM gives a mean precision of 0.41 compared to 0.20 (obtained from published results [9]) for the Translation Model. This is twice as good as the Translation Model. At the same time the FACMRM also has a much higher recall than the Translation Model. Both models perform substantially better than the Co-occurrence Model. PACMRM and DRCMRM cannot be directly compared to the other systems since the Translation Model and the Co-occurrence model have not been used for ranked retrieval.

Figure 1: Images automatically annotated as "sunset" (FACMRM) but not manually annotated as "sunset". The color of sunset may not show up clearly in black and white versions of this figure.

Figure 1 illustrates the power of the relevance model. The figure shows three images (from the test set) which were annotated with "sunset" by FACMRM. Although the three are clearly pictures of sunset (the last picture shows both a sun and a sunset), the word "sunset" was missing from the manual annotations. In these cases, the model allows us to catch errors in manual annotation.

This paper is organized as follows. We discuss related work in section 2. This is followed by a brief discussion of how the blob features are constructed. Section 4 has a discussion of the cross-media language model and how it can be used for image annotation and retrieval. Section 5 shows experimental results for the different models and compares them to those for the Translation and Co-occurrence Model. The section also shows example results to illustrate the different aspects of the model. Finally, the last section concludes with a discussion of future work in this area.

2. RELATED WORK

While there has been some work on statistical models for object recognition and image retrieval [10], there has been little work on automatically annotating images. We have already mentioned the Co-occurrence [15] and Translation Models [9]. The Co-occurrence model tends to require large numbers of training samples to estimate the correct probability and also tends to map frequent words to every blob. Duygulu et al [9] also try to use their Translation Model to label individual regions in the image. Picard and Minka [16] describe a tool for users to semi-automatically annotate image regions by selecting positive and negative examples manually and then using texture similarity to propagate annotations. Barnard and Forsyth[2] extended Hofmann's Hierarchical Aspect Model for text and proposed a multimodal hierarchical aspect model for hierarchical clustering of images and words. The results of this model are not available in a form which can be directly compared to our present model. Blei and Jordan [3] extended the Latent Dirichlet Allocation (LDA) Model and proposed a Correlation LDA model which relates words and images. This model assumes that a Dirichlet distribution can be used to generate a mixture of latent factors. This mixture of latent factors is then used to generate words and regions. EM is again used to estimate this model. Blei and Jordan show a few examples for labeling specific regions in an image. They also report recall-precision graphs for retrieval performance based on one word queries but the results are not directly comparable since the datasets are different.

3. DISCRETE FEATURES IN IMAGES

An important question is how can one obtain an image

vocabulary. In other words, how does one represent every image in the collection using a subset of items from a finite set of items. An intuitive answer to this question is to segment the image into regions, cluster similar regions and then use the regions as a vocabulary. The hope is that this will produce semantic regions and hence a good vocabulary. In general, image segmentation is a very fragile and erroneous process and so the results are usually not very good.

Barnard and Forsyth[2] and Duygulu *et al.* [9] used general purpose segmentation algorithms like Blobworld[7] and Normalized-cuts[17] to extract regions. These algorithms do not always produce good segmentations (see Figure 2) but are useful for building and testing models. For each segmented region, features such as color, texture, position and shape information are computed. Duygulu et al [9] used Normalized-cuts to segment images and then extracted 33 features from the images. They ignored regions which were smaller than a threshold size. Given a set of training images, a K-means clustering algorithm (K = 500) is applied to cluster the regions on the basis of these features. These 500 clusters which they call "blobs" compose the vocabulary for the set of images. Each blob is assigned a unique integer to serve as its identifier (analogous to a word's ASCII representation). All images in the training set can now be represented as a set of blobs from this vocabulary. Figure 2 shows the segmentation and the clustering process for some training images. The resulting blobs produced by this approach still leave a lot to be desired (see for example section 5.4). However, given the complexity of images, this is a good first start. Given a new test image, it can be segmented into regions and region features can be computed. The blob which is closest to it in cluster space is assigned to it. Our primary purpose in this paper is to show that relevance models are a powerful tool for solving the problem of image annotation and retrieval. In order to make a fair comparison with other models we choose to use their [9] data and feature sets.

4. CROSS-MEDIA RELEVANCE MODELS

Suppose we are given a collection $\mathcal{C}$ of un-annotated images. Each image $I \in \mathcal{C}$ is represented by a discrete set of blob numbers, generated as described in Section 3: $I = \{b_1 \ \ldots \ b_m\}$. In this section we develop a formal model that allows us to answer the following questions:

(i) Given an un-annotated image $I \in \mathcal{C}$, how can we automatically assign meaningful keywords to that image?

(ii) Given a text query $w_1 \ \ldots \ w_k$, how can we retrieve images $I \in \mathcal{C}$ that contain objects mentioned in the query?

We assume there exists a training collection $\mathcal{T}$, of annotated images, where each image $J \in \mathcal{T}$ has a dual representation in terms of both words and blobs: $J = \{b_1 \ \ldots \ b_m; \ w_1 \ \ldots \ w_n\}$. Here $\{b_1 \ \ldots \ b_m\}$ represents the blobs corresponding to regions of the image and $\{w_1 \ \ldots \ w_n\}$ represents the words in the image caption [1]. The number of blobs and words in each image (m and n) may be different from image to image. In contrast to the translation model, we do not assume that there is an underlying one-to-one correspondence (alignment) between the blobs and the words in an image, we only assume that a set of keywords $\{w_1 \ \ldots \ w_n\}$ is related to the set of objects represented by blobs $\{b_1 \ \ldots \ b_m\}$.

[1]The word caption is used to denote keyword annotations in this paper except in section 6

4.1 A Model of Image Annotation

Suppose we are given an un-annotated image $I \in \mathcal{C}$. We have the blob representation of that image $I = \{b_1 \ \ldots \ b_m\}$, and want to automatically select a set of words $\{w_1 \ \ldots \ w_n\}$ that accurately reflects the content of the image.

We adopt a generative language modeling approach [12, 11, 13]. Assume that for each image I there exists some underlying probability distribution $P(\cdot|I)$. We refer to this distribution as the *relevance model* of I (see [13, 14]). The relevance model can be thought of as an urn that contains all possible blobs that could appear in image I, as well as all words that could appear in the caption of I. We assume that the observed image representation $\{b_1 \ \ldots \ b_m\}$ is the result of m random samples from $P(\cdot|I)$.

A natural way to annotate an image I would be to sample n words $w_1 \ \ldots \ w_n$ from its relevance model $P(\cdot|I)$. In order to do that, we need to know the probability of observing any given word w when sampling from $P(\cdot|I)$. That is, we need to estimate the probability $P(w|I)$ for every word w in the vocabulary. Given that $P(\cdot|I)$ itself is unknown, the probability of drawing the word w is best approximated by the conditional probability of observing w given that we previously observed $b_1 \ \ldots \ b_m$ as a random sample from the same distribution:

$$P(w|I) \approx P(w|b_1 \ \ldots \ b_m) \tag{1}$$

We cannot use the prevalent maximum-likelihood estimator for that probability because the image representation $b_1 \ldots b_k$ does not contain any words. However, we can use the training set $\mathcal{T}$ of annotated images to estimate the joint probability of observing the word w and the blobs $b_1 \ \ldots \ b_m$ in the same image, and then marginalizing the distribution with respect to w. The joint distribution can be computed as the expectation over the images J in the training set:

$$P(w, b_1, \ \ldots \ , b_m) = \sum_{J \in \mathcal{T}} P(J) P(w, b_1, \ \ldots \ , b_m|J) \tag{2}$$

We assume that the events of observing w and $b_1, \ \ldots \ , b_m$ are mutually independent once we pick the image J, and identically distributed according to the underlying distribution $P(\cdot|J)$. This assumption follows directly from our earlier decision to model each image as an urn containing both words and blobs. Since the events are independent, we can rewrite equation (2) as follows:

$$P(w, b_1, \ \ldots \ , b_m) = \sum_{J \in \mathcal{T}} P(J) P(w|J) \prod_{i=1}^{m} P(b_i|J) \tag{3}$$

The prior probabilities $P(J)$ can be kept uniform over all images in $\mathcal{T}$. Since the images J in the training set contain both words and blobs, we can use smoothed maximum-likelihood estimates for the probabilities in equation (3). Specifically, the probability of drawing the word w or a blob b from the model of image J is given by:

$$P(w|J) = (1 - \alpha_J) \frac{\#(w, J)}{|J|} + \alpha_J \frac{\#(w, \mathcal{T})}{|\mathcal{T}|} \tag{4}$$

$$P(b|J) = (1 - \beta_J) \frac{\#(b, J)}{|J|} + \beta_J \frac{\#(b, \mathcal{T})}{|\mathcal{T}|} \tag{5}$$

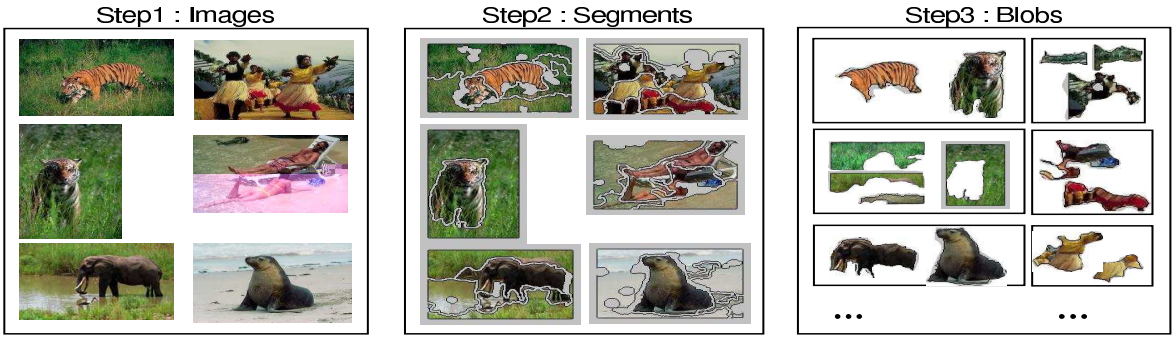

Figure 2: Image preprocessing: Step 2 shows the segmentation results from a typical segmentation algorithm (Blobworld) The clusters in step 3 are manually constructed to show the concept of blobs. Both the segmentation and the clustering often produce semantically inconsistent segments (breaking up the tiger) and blobs (seals and elephants in the same blob) .

Here, $\#(w, J)$ denotes the actual number of times the word w occurs in the caption of image J (usually 0 or 1, since the same word is rarely used multiple times in a caption). $\#(w, \mathcal{T})$ is the total number of times w occurs in all captions in the training set $\mathcal{T}$. Similarly, $\#(b, J)$ reflects the actual number of times some region of the image J is labeled with blob b, and $\#(b, \mathcal{T})$ is the cumulative number of occurrences of blob b in the training set. $|J|$ stands for the aggregate count of all words and blobs occurring in image J, and $|\mathcal{T}|$ denotes the total size of the training set. The smoothing parameters α_J and β_J determine the degree of interpolation between the maximum likelihood estimates and the background probabilities for the words and the blobs respectively. We use different smoothing parameters for words and blobs because they have very different occurrence patterns: words generally follow a Zipfian distribution, whereas blobs are distributed much more uniformly, due in part to the nature of the clustering algorithm that generates them. The values of these parameters are selected by tuning system performance on the held-out portion of the training set.

4.1.1 Using the model for Image Annotation

Equations (1) - (5) provide the machinery for approximating the probability distribution $P(w|I)$ underlying some given image I. We can produce automatic annotations for new images by first estimating the distribution $P(w|I)$ and then sampling from it repeatedly, until we produce a caption of desired length. Or we could simply pick a desired number n of words that have the highest probability under $P(w|I)$ and use those words for the annotation.

4.2 Two Models of Image Retrieval

The task of image retrieval is similar to the general ad-hoc retrieval problem. We are given a text query $Q = w_1 \ldots w_k$ and a collection $\mathcal{C}$ of images. The goal is to retrieve the images that contain objects described by the keywords $w_1 \ldots w_k$, or more generally rank the images I by the likelihood that they are relevant to the query. We cannot simply use a text retrieval systems because the images $I \in \mathcal{C}$ are assumed to have no captions. In the remainder of this section we develop two models of image retrieval. The first model makes extensive use of the annotation model developed in the previous section. The second model does not rely on annotations and instead "translates" the query into the language of blobs.

4.2.1 Annotation-based Retrieval Model

A simple approach to retrieving images is to annotate each image in $\mathcal{C}$ using the techniques proposed in section 4.1 with a small number of keywords. We could then index the annotations and perform text retrieval in the usual manner. This approach is very straightforward, and, as we will show in section 5, is quite effective for single-word queries. However, there are several disadvantages. First, the approach does not allow us to perform ranked retrieval (other than retrieval by coordination-level matching). This is due to the binary nature of word occurrence in automatic annotations: a word either is or is not assigned to the image, it is rarely assigned multiple times. In addition, all annotations are likely to contain the same number of words, so document-length normalization will not differentiate between images. As a result, all images containing some fixed number of the query words are likely to receive the same score. The second problem with indexing annotations is that we must a-priori decide what annotation length is appropriate. The number of words in the annotation has a direct influence on the recall and precision of this system. In general, shorter annotations will lead to higher precision and lower recall, since fewer images will be annotated with any given word. Short annotations are more appropriate for a casual user, who is interested in finding a few relevant images without looking at too much junk. On the other hand, a professional user may be interested in higher recall and thus may need longer annotations. Consequently, it would be challenging to field the retrieval system in a way that would suit diverse users.

An alternative to fixed-length annotation is to use probabilistic annotation. In section 4.1 we developed a technique that assigns a probability $P(w|I)$ to every word w in the vocabulary. Rather than matching the query against the few top words, we could use the entire probability distribution $P(\cdot|I)$ to score images using a language-modeling approach [12, 11, 4, 18]. In a language modeling approach we score the documents (images) by the probability that a query would be observed during i.i.d. random sampling from a document (image) language model. Given the query $Q = w_1 \ldots w_k$, and the image $I = \{b_1 \ldots b_m\}$, the probability of drawing Q from the model of I is:

$$P(Q|I) = \prod_{j=1}^{k} P(w_j|I) \tag{6}$$

where $P(w_j|I)$ is computed according to equations (1) - (5). This model of retrieval does not suffer from the drawbacks of fixed-length annotation and allows us to produce ranked lists of images that are more likely to satisfy diverse users.

4.2.2 Direct Retrieval Model (DRCMRM)

The annotation-based model outlined in section 4.2.1 in effect converts the images in $\mathcal{C}$ from the blob-language to the language of words. It is equally reasonable to reverse the direction and convert the query into the language of blobs. Then we can directly retrieve images from the collection $\mathcal{C}$ by measuring how similar they are to the blob-representation of the query. The approach we describe was originally proposed by [14] for the task of cross-language information retrieval. We start with a text query $Q = w_1 \ \ldots \ w_k$. We assume that there exists an underlying relevance model $P(\cdot|Q)$, such that the query itself is a random sample from that model. We also assume that images relevant to Q are random samples from $P(\cdot|Q)$ (hence the name relevance model). In the remainder of this section we describe: (i) how to estimate the parameters $P(b|Q)$ of this underlying relevance model, and (ii) how we could rank the images with respect to this model.

Estimation of the unknown parameters of the query model is performed using the same techniques used in section 4.1. The probability of observing a given blob b from the query model can be expressed in terms of the joint probability of observing b from the same distribution as the query words $w_1 \ \ldots \ w_k$:

$$P(b|Q) \approx P(b|w_1 \ \ldots \ w_k) = \frac{P(b, w_1 \ \ldots \ w_k)}{P(w_1 \ \ldots \ w_k)} \qquad (7)$$

The joint probability $P(b, w_1 \ \ldots \ w_k)$ can be estimated as an expectation over the annotated images in the training set, by assuming independent sampling from each image $J \in \mathcal{T}$:

$$P(b, w_1, \ \ldots \ , w_k) = \sum_{J \in \mathcal{T}} P(J) P(b|J) \prod_{i=1}^{k} P(w_i|J) \qquad (8)$$

The probabilities $P(b|J)$ and $P(w_i|J)$ can be estimated from equation (5). The prior probabilities $P(J)$ can be kept uniform, or they can be set to reflect query-independent user preferences for a particular type of image, if such information is available.

Ranking. Together, equations (7) and (8) allow us to "translate" the query Q into a distribution $P(\cdot|Q)$ over the blob vocabulary. What remains is to specify how this distribution can be used for effective ranking of images $I \in \mathcal{C}$. One possibility would be to rank the images by the probability that they are a random sample from $P(\cdot|Q)$, as was suggested for the task of ad-hoc retrieval in [13]. In this paper we opt for a specific case [6] of the more general risk minimization framework for retrieval proposed and developed by [18]. In this approach, documents (images) are ranked according to the negative Kullback-Liebler divergence between the query model $P(\cdot|Q)$ and the document (image) model $P(\cdot|I)$:

$$-KL(Q||I) = \sum_{b \in \mathcal{B}} P(b|Q) \log \frac{P(b|I)}{P(b|Q)} \qquad (9)$$

Here $P(b|Q)$ is estimated using equations (7) and (8), while $P(b|I)$ can be computed directly from equation (5), since every image $I \in \mathcal{C}$ has a blob representation.

5. EXPERIMENTAL RESULTS

In this section we will discuss details of the dataset used and also show experimental results using the different models. Section 5.2 compares the results of the fixed length annotation model FACMRM with the Co-occurrence and Translation Models. This is followed by results on the two retrieval models PACMRM and DRCMRM. Finally, we show some examples to illustrate different aspects of the models.

5.1 Dataset

Since our focus in this paper is on models and not features we use the dataset in Duygulu *et al.*[9] [2]. This also allows us to compare the performance of models in a strictly controlled manner. The dataset consists of 5,000 images from 50 Corel Stock Photo cds. Each cd includes 100 images on the same topic. Segmentation using normalized cuts followed by quantization ensures that there are 1-10 blobs for each image. Each image was also assigned 1-5 keywords. Overall there are 371 words and 500 blobs in the dataset. Details of the above process are contained in Duygulu *et al* [9]. We divided the dataset into 3 parts - with 4,000 training set images, 500 evaluation set images and 500 images in the test set. The evaluation set is used to find system parameters. After fixing the parameters, we merged the 4,000 training set and 500 evaluation set images to make a new training set. This corresponds to the training set of 4500 images and the test set of 500 images used by Duygulu *et al* [9].

5.2 Automatic Image Annotation

The FACMRM model uses a fixed number of words to annotate the images. To evaluate the annotation performance, we retrieve images using keywords from the vocabulary (note that this is not ranked retrieval). We can easily judge the relevance of the retrieved images by looking at the real (manual) annotations of the images. The recall is the number of correctly retrieved images divided by the number of relevant images in the test dataset. The precision is the number of correctly retrieved images divided by the number of retrieved images. We calculate the mean of precisions and recalls for a given query set. To combine recall and precision in a single efficiency measure, we use the F-measure, $F = \frac{2*recall*precision}{recall+precision}$.

5.2.1 Finding model parameters

Our model requires the estimation of two smoothing parameters, α_J for word smoothing and β_J for blob smoothing. These parameters were estimated by training on a 4000 image set and testing on the 500 image evaluation set. One can trade-off recall for precision in this task by varying the smoothing parameters. We optimize on the F-measure (which is a function of both recall and precision) to pick the best smoothing parameters. It turns out $\alpha_J = 0.1$ and $\beta_J = 0.9$. These parameters were used to compare FACMRM with other models.

5.2.2 Model Comparison

We compare the annotation performance of the three models - the Co-occurrence Model, the Translation Model and FACMRM. We annotate each test image with 5 keywords for both the Co-occurrence Model and FACMRM. The Trans-

[2]Available at http://www.cs.arizona.edu/people/kobus/ research/data/eccv_2002

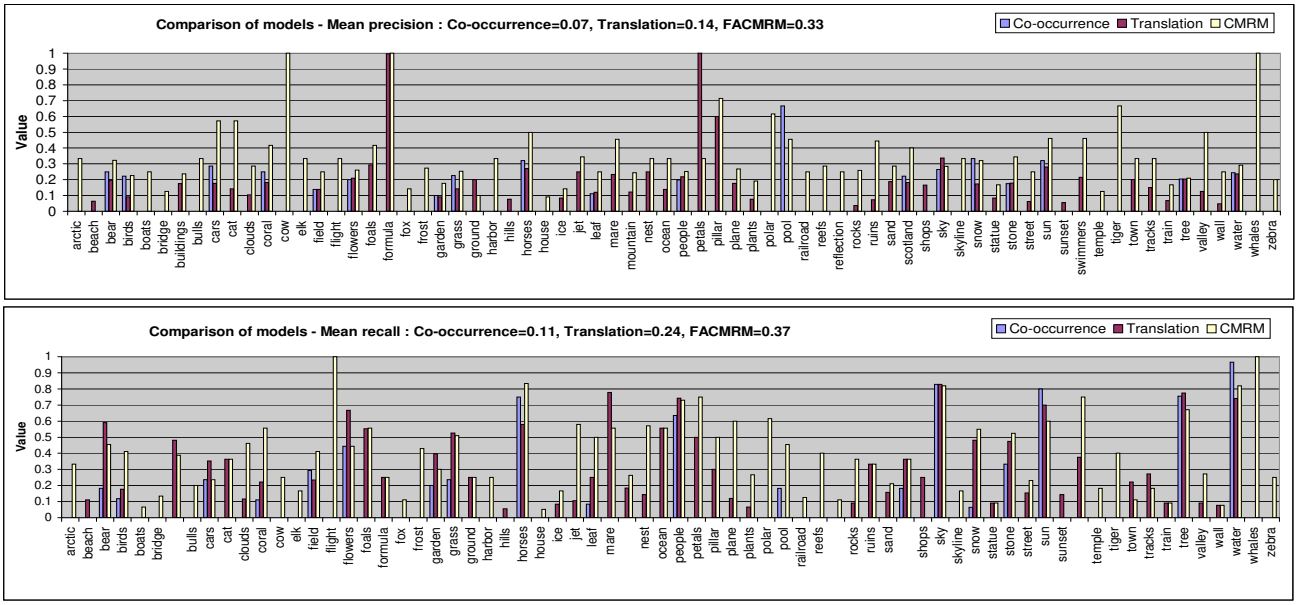

Figure 3: Comparison of 3 models: The graph shows mean precisions and recall for 3 different models for 70 queries (one word queries). FACMRM significantly outperforms other models.

lation Model annotates different images with different numbers of keywords. A total of 263 one word queries are possible in the dataset. The number of queries which retrieve at least one relevant image vary depending on the models - the Co-occurrence Model has 19, Translation Model has 49 and FACMRM has 66 queries. The union of these three query sets gives us a new 70 query set. Figure 3 shows the precision and recall of each model for this query set. The Co-occurrence Model has 0.07 mean precision and 0.11 mean recall, the Translation Model has 0.14 mean precision and 0.24 recall and the FACMRM has 0.33 mean precision and 0.37 mean recall. If we compare numbers for only the top 49 queries (since the Translation Model only has 49 queries which have at least one relevant document), the Translation Model has 0.20 mean precision and 0.34 mean recall and FACMRM has 0.41 mean precision and 0.49 mean recall. FACMRM is significantly better in terms of precision, recall and the number of keywords which are used to annotate images.

Discussion The Translation Model uses model 2 of Brown *et al.* [5] which requires that we sum over all the possible assignments of words to blobs.

$$p(w|b) = \prod_{n=1}^{N} \prod_{j=1}^{M_n} \sum_{i=1}^{L_n} p(a_{nj} = i) t(w = w_{nj} | b = b_{ni}) \quad (10)$$

where N is the #images, M_n is the #words in the n-th image and L_n is the #blobs in the n-th image. $p(a_{nj} = i)$ is the assignment probability that in image n, a particular blob b_i is associated with a specific word w_j, $t(w = w_{nj}|b = b_{ni})$ (i.e. the transition probability of word w given blob b). They use the EM algorithm to maximize this likelihood. Since EM requires constraints on the probabilities, they assume that the assignment probabilities when summed over the same number of words and images add up to one. Thus, for example, the assignment probabilities for all images having exactly 4 words and 8 blobs is equal to one. Unfortunately this does not seem reasonable in this context and maybe one reason why the model does not perform as well as ours.

5.3 Evaluation of Ranked Retrieval

Query length	1 word	2 words	3 words	4 words
Number of qrys	179	386	178	24
Relevant images	1675	1647	542	67
AveP (PACMRM)	0.1501	0.1419	0.1730	0.2364
AveP (DRCMRM)	0.1697	0.1642	0.2030	0.2765

Table 1: Details of the different query sets and relative performance of the two retrieval models in terms of average precision.

We use the standard framework for evaluating the ranking effectiveness of the two retrieval models proposed in section 4.2. . The collection $\mathcal{C}$ is composed of the testing set of 500 images. As a set of queries, we take all combinations of 1, 2, 3 and 4 words in the vocabulary. Since the number of all 3- and 4-word combinations is prohibitively large, we discard any queries that occur only once in the testing set. For a given query Q, the relevant images are the ones that contain all query words in the manual annotation (we use the manual annotations available for the test images purely for this evaluation). As evaluation metrics, we use the standard 11-point recall-precision graphs, along with non-interpolated average precision.

Table 1 shows the details of the four subsets of our query set, along with average precision for the two retrieval models on each of the subsets. We achieve average precision of above 0.2 on 3-4 word queries. This is particularly encouraging because the results are obtained over a large number of queries. As expected, performance is generally higher for longer queries. The direct retrieval model (DRCMRM) outperforms the annotation-based model (PACMRM) on all query subsets. The differences are statistically significant

Image				
Original Annotation	people pool swimmers water	cars formula tracks wall	clouds mountain sky water	field foals horses mare
Automatic Annotation	water people swimmers pool	cars tracks wall formula	sky mountain clouds park	field horses foals mare

Figure 5: Automatic annotations (best four words) compared with the original manual annotations. In images 1, 2 and 4 the annotations are identical while in image 3 the annotations differ on one word "Park" (not unreasonable as in national park) instead of "water".

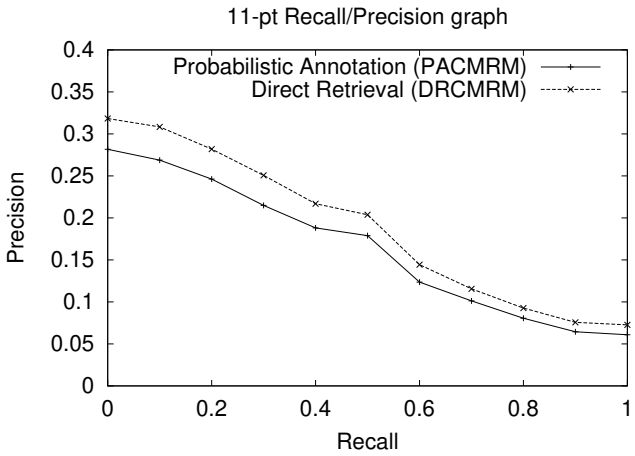

Figure 4: Performance of the two retrieval models on the task of ranked retrieval. Direct Retrieval model consistently outperforms the annotation model by a small margin.

		Sun	Sunset
Original Annotation	Recall	0.60	0.00
	Precision	0.46	0.00
Modified Annotation	Recall	0.5	0.57
	Precision	0.71	0.21

Table 2: Recall/precision of "sun" and "sunset" keywords before and after correcting erroneous manual annotations (only "sun" and "sunset" keywords).

according to the Wilcoxon test at the 5% confidence level. The exception is the 4-word query set, where it is not possible to achieve statistical significance because of the small number of queries. Figure 4 shows a recall-precision graph of the two models on the combined query set.

5.4 Illustrative Examples

This section shows some illustrative examples of the annotations generated by our models. Figure 5 shows that for images 1, 2 and 4 the automatic and manual annotations are identical while for image 3, the automatic annotation generates "park" (which is a reasonable annotation) while it is manually annotated as "water".

Test image	**Training image**
	
Blobs : 163 43 147 451 117 282 88 79 147 360	Blobs : 147 147 88 79 282 88 79 451 147 117
Real-annotation: pillar sculpture stone temple Auto-annotation: tree house landscape roof	Real-annotation: house landscape roofs tree

Figure 6: Example of Bad Annotations: Although semantically different, the training and test image have 6 blobs in common. This points out the need for better blob descriptors.

Figure 6 shows that blobs are not always good descriptors of images. Since the test and training images have 6 blobs in common the model tends to annotate the test image with the words used to manually annotate the training image. However, this produces incorrect results showing that the blobs are at best an imperfect vocabulary.

As Figure 1 illustrates, manual annotations can often be wrong. Table 2 shows the the recall and precision obtained for the words "Sun" and "Sunset" used as queries. With the original manual annotations supplied with the dataset, the recall and precision for "Sunset" are 0. A significant factor is the incorrect human labeling of many of these images in both the test and training images. After correctly relabeling (by hand) the sun and sunset images in the dataset and re-training the model we get a much higher precision (0.21) and recall (0.57) for the "sunset" keyword. The precision for the "sun" keyword also improves dramatically while the recall drops slightly. Automatic annotation may be useful in checking the accuracy of manual annotations.

Figures 7 and 8 show example retrievals using DRCMRM in response to the text queries "tiger" and "pillar" respectively.

Figure 7: Retrieval (DRCMRM) in response to the text query "tiger".

Figure 8: Retrieval (DRCMRM) in response to the text query "pillar". Note the pillar(s) in each image

6. CONCLUSIONS AND FUTURE WORK

We have shown that Cross-Media Relevance Models are a good choice for annotating and retrieving images. Three different models were suggested and tested. The FACMRM model is more than twice as good (in terms of mean precision) as a state of the art Translation Model in annotating images. We also showed how to perform ranked retrieval using some of our models.

Obtaining large amounts of labeled training and test data is difficult but we believe this is needed for improvements in both performance and evaluation of the algorithms proposed here. Better feature extraction or the use of continuous features will probably improve the results. Other areas of possible research include the use of actual captions (instead of keywords) We believe that this is a fruitful area of research for applying formal models of information retrieval.

7. ACKNOWLEDGMENTS

We thank Kobus Barnard and P. Duygulu for making their dataset [9] available. This work was supported in part by the Center for Intelligent Information Retrieval, by the National Science Foundation under grant NSF IIS-9909073 and by SPAWARSYSCEN-SD under grants N66001-99-1-8912 and N66001-02-1-8903. Jiwoon Jeon is partially supported by the Government of Korea. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsor.

8. REFERENCES

[1] K. Barnard, P. Duygulu, N. de Freitas, D. Forsyth, D. Blei, and M. I. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135, 2003.

[2] K. Barnard and D. Forsyth. Learning the semantics of words and pictures. In *International Conference on Computer Vision*, Vol.2, pages 408-415, 2001.

[3] D. Blei, Michael, and M. I. Jordan. Modeling annotated data. To appear in the *Proceedings of the 26th annual international ACM SIGIR conference*

[4] Berger, A. and Lafferty, J. Information retrieval as statistical translation. In *Proceedings of the 22nd annual international ACM SIGIR conference*, pages 222–229, 1999.

[5] P. Brown, S. D. Pietra, V. D. Pietra, and R. Mercer. The mathematics of statistical machine translation: Parameter estimation. In *Computational Linguistics*, 19(2):263-311, 1993.

[6] W. B. Croft. Combining Approaches to Information Retrieval, in *Advances in Information Retrieval* ed. W. B. Croft, Kluwer Academic Publishers, Boston, MA.

[7] C. Carson, M. Thomas, S. Belongie, J. M. Hellerstein, and J. Malik. Blobworld: A system for region-based image indexing and retrieval. In *Third International Conference on Visual Information Systems*, Lecture Notes in Computer Science, 1614, pages 509-516, 1999.

[8] M. Das and R. Manmatha and E. M. Riseman, Indexing Flowers by Color Names using Domain Knowledge-driven Segmentation, *IEEE Intelligent Systems*, 14(5):24–33, 1999.

[9] P. Duygulu, K. Barnard, N. de Freitas, and D. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Seventh European Conference on Computer Vision*, pages 97-112, 2002.

[10] D. Forsyth and J. Ponce, *Computer Vision: A Modern Approach* Prentice Hall, 2003

[11] D. Hiemstra *Using Language Models for Information Retrieval*. PhD dissertation, University of Twente, Enschede, The Netherlands, 2001.

[12] J. M. Ponte, and W. B. Croft, A language modeling approach to information retrieval. *Proceedings of the 21st annual international ACM SIGIR Conference*, pages 275–281, 1998.

[13] V. Lavrenko and W. Croft. Relevance-based language models. *Proceedings of the 24th annual international ACM SIGIR conference*, pages 120-127, 2001.

[14] V. Lavrenko, M. Choquette, and W. Croft. Cross-lingual relevance models. *Proceedings of the 25th annual international ACM SIGIR conference*, pages 175–182, 2002.

[15] Y. Mori, H. Takahashi, and R. Oka. Image-to-word transformation based on dividing and vector quantizing images with words. In *MISRM'99 First International Workshop on Multimedia Intelligent Storage and Retrieval Management*, 1999.

[16] R. W. Picard and T. P. Minka", Vision Texture for Annotation, In *Multimedia Systems*, 3(1):3–14, 1995.

[17] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[18] J. Lafferty and C. Zhai. Document language models, query models, and risk minimization for information retrieval, Proceedings of the 24th annual international ACM SIGIR Conference, pages 111-119, 2001.