Hidden Markov Models for Image and Video Retrieval Using Textual Queries

Arnab Ghoshal[1], Pavel Ircing[2], and Sanjeev Khudanpur[1]

[1] Johns Hopkins University, 3400 N Charles Street, Baltimore MD 21218, USA.
{ag,khudanpur}@jhu.edu
[2] University of West Bohemia, Univerzitiní 22, 306 14 Pilsen, Czech Republic.
ircing@kky.zcu.cz

Abstract. A novel method is introduced for automatic annotation of images with keywords from a generic vocabulary of *concepts* or objects for the purpose of content-based image retrieval. An image, represented as sequence of feature-vectors characterizing low-level visual features such as color, texture or oriented-edges, is modeled as having been *stochastically generated* by a hidden Markov model, whose states represent concepts. The parameters of the model are estimated from a set of manually annotated training-images. Each image in a large test-collection is then automatically labeled with the *a posteriori* probability of concepts present in it. This annotation supports content-based search of the test-collection via keywords. Various aspects of model parameterization, parameter estimation, and image annotation are dicussed, and empirical retrieval results are presented on the COREL and TRECVID data-sets. Comparisons are made with two other recently developed techniques on the same data-sets.

1 Introduction

The content of communications in the digital age is increasingly multi-modal in nature, with text, images and even speech or video being used in a single "document." Content-based indexing and retrieval of multimedia is therefore becoming an increasingly important issue. Unlike text retrieval, where the modality in which the user usually specifies her information need is the same as the modality of the search collection, there is relatively little work in image retrieval based on textual queries. The major reason, of course, is that open domain image understanding remains a largely unsolved problem, and even detecting a single predetermined object in an image, such as a human face, is a fairly difficult problem (cf e.g. [1]). Important progress has been made in the last few years in content-based image retrieval, as reported by [2–5] and others.

We are not the first to observe that while the "pure" image understanding problem, *i.e.* the problem of recognizing all the objects in a given image, is very difficult due to several invariance issues, there are two aspects of the image indexing and retrieval problem which make it relatively more tractable. One is the availability of *side information* in the accompanying text: images in multimedia documents are often accompanied by descriptive text that a model may use to *learn* the content of an image. The other is that the search problem is fairly tolerant of weak inferences on individual images: the system need not recognize with *high* probability that a particular image in a collection

contains, say, a tiger; it suffices if the likelihood assigned to *tiger* in images containing tigers is merely *higher* than it is in images *not* containing tigers, *no matter how small either likelihood is*. The images only need to be rank-ordered correctly.

We therefore develop a joint stochastic model for images and their accompanying captions, whose parameters can be estimated from a manually annotated (training) collection of image+caption pairs. We measure the efficacy of this model by annotating a (test) collection of images using the words seen in captions of the training images, and performing image retrieval experiments using textual queries. We report performance metrics for *ranked retrieval* standardized by the information retrieval community.

This paper is organized as follows. Section 2 formally describes the hidden Markov model (HMM) used for image annotation, and compares it with two other recently developed techniques for the same task. Section 3 delves into some issues in the design and parameter estimation for the model. Section 4 presents a series of experimental results on two image-collections, namely the COREL and the TRECVID data-sets. We conclude with some short remarks in Section 5.

2 Hidden Markov Models for Image Annotation

Let a collection $\{(I, C)\}$ of image+caption pairs be given. Let $I \equiv \{i_1, \ldots, i_T\}$ denote image-segments (image-regions), and $C \equiv \{c_1, \ldots, c_N\}$ denote the objects (concepts) present in that image, as specified by the label (caption). The T segments may be object based, with each region corresponding to one semantically distinct object, or they may be a simple rectangular partition of the image into fixed-size blocks. For each image-region i_t, $t = 1, \ldots, T$, let $x_t \in \mathbb{R}^d$ represent color, texture, edges, shape and other salient visual features of the region. Finally, let $\mathcal{V}$ denote the total vocabulary of the caption-words c_n across the entire collection of images.

We propose to model the $\{x_t\}$-sequence as a hidden Markov process [6], generated by an underlying unobserved Markov chain $\{s_t\}$ whose states take values in C. Specifically, given the states s_t, each x_t is generated according to a probability density function $f(\cdot|s_t)$, and $\{s_t\}$ itself is a Markov chain with a known initial state s_0 and transition probabilities $p(s_t|s_{t-1})$. Figure 1 illustrates an image with $T = 24$ rectangular regions with the caption {`horse`, `foal`, `fence`}, and the state-diagram of the underlying Markov chain that generates these regions. Formally, the joint likelihood of a state-sequence $s_1^T \equiv \{s_1, \ldots, s_T\} \in C^T$ and features $x_1^T \equiv \{x_1, \ldots, x_t\}$ is

$$f(x_1^T, s_1^T|s_0) = f(x_1, \ldots, x_T, s_1, \ldots, s_T|s_0) = \prod_{t=1}^{T} f(x_t|s_t)\, p(s_t|s_{t-1}).$$

Note that knowing the state sequence s_t is equivalent to being given the *alignment* of each image-region i_t with one of the words in the caption. Since this level of detail is generally not provided in captions, a hidden Markov model (HMM) may be used for computing the joint likelihood of an image+caption pair as

$$f(I, C|s_0) = f(x_1^T, C|s_0) = \sum_{s_1^T \in C^T} \prod_{t=1}^{T} f(x_t|s_t)\, p(s_t|s_{t-1}). \tag{1}$$

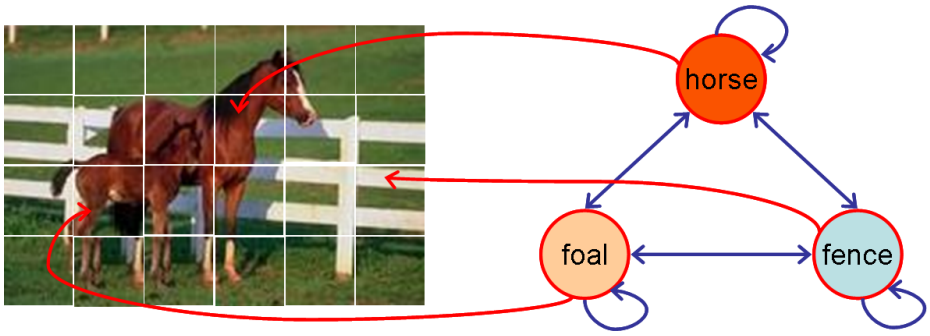

Fig. 1. State transitions diagram and output for an image+caption HMM.

We model the output density $f(\cdot|c)$ for each state $c \in \mathcal{V}$ as a mixture of multivariate Gaussian densities on $\mathbb{R}^d$:

$$f(x|c) = \sum_{m=1}^{M} w_{m,c} \times \frac{1}{\sqrt{(2\pi)^d |\Sigma_{m,c}|}} e^{-\frac{1}{2}(x-\mu_{m,c})^T \Sigma_{m,c}^{-1}(x-\mu_{m,c})}, \tag{2}$$

where $w_{m,c}$'s are the mixture weights, $\mu_{m,c}$ the mean-vector and $\Sigma_{m,c}$ the diagonal covariance-matrix for the of the m-th mixture component of the state c. The transition probabilities $p(\cdot|\cdot)$ are initially set to be uniform for all permissible states in C.

The *emission* densities $f(\cdot|\cdot)$ and *transition* probabilities $p(\cdot|\cdot)$ of the HMM may be estimated, given a training collection of image+caption pairs, to maximize the joint likelihood (1). Details of this maximum likelihood estimation procedure are standard (cf [6]) and therefore omitted.

For indexing a new image I, the HMM provides the conditional probability, given all the visual evidence x_1^T in I, that a particular image-region i_t was generated by a particular concept $c \in \mathcal{V}$, as

$$p(s_t = c|x_1^T, s_0) = \frac{f(x_1^T, s_t = c|s_0)}{f(x_1^T|s_0)} = \frac{\sum_{s_1^T : s_t = c} \prod_{t=1}^{T} f(x_t|s_t)\, p(s_t|s_{t-1})}{\sum_{s_1^T \in \mathcal{V}^T} \prod_{t=1}^{T} f(x_t|s_t)\, p(s_t|s_{t-1})}.$$

Therefore, the probability of a particular concept $c \in \mathcal{V}$ being present (somewhere) in an image may be calculated as

$$p(c \in C|I, s_0) = \sum_{t=1}^{T} p(s_t = c|x_1^T, s_0). \tag{3}$$

A freely available toolkit, HTK [7], efficiently implements all the basic parameter estimation and probability calculation algorithms needed for working with the model described above. We have used HTK in the experiments described in Section 4 below.

2.1 HMMs vs Statistical Machine Translation

Duygulu et al [2] have presented the image annotation problem as a variant on the statistical machine translation (MT) problem. In their formulation, the representations x_t

of the segmented image-regions are discretized and then treated as "words," replacing $\mathbb{R}^d$ with a vocabulary $\mathcal{X}$ of visual expressions (visterms). The training corpus of image+caption pairs is then treated as a corpus of aligned bi-lingual text, and statistical machine translation techniques are employed to infer a stochastic translation lexicon $p(x|c)$ between $c \in \mathcal{V}$ and $x \in \mathcal{X}$. Given a test image I with visterms x_1^T, a probability is calculated for each concept $c \in \mathcal{V}$ as

$$p(c \in C|I) = \sum_{x \in \mathcal{X}} p(c|x)\hat{p}(x|I) = \sum_{t=1}^{T} p(c|x_t)\frac{1}{T}.$$

Note that the estimation of $p(x|c)$ in their case strongly resembles the embedded estimation of $f(x|c)$ in the HMM case, provided we set the transition probabilities $p(\cdot|\cdot)$ to be uniform. Thus the primary difference between our model and theirs is that the image-features are modeled here as continuous-valued vectors, avoiding the need for quantization. Empirical evidence in Section 4 will show that preserving the continuous-valued representation of the x_t's results in significant improvement in indexing and retrieval performance.

2.2 HMMs vs Continuous Relevance Models

Manmatha et al [5, 8] have used a continuous relevance model (CRM) to perform image annotation and retrieval. In their case, continuous-valued visual features $\{x_t\}$ are extracted from rectangular regions of the (test) image I and compared with the visual features extracted from each training image J using a Gaussian kernel function. In particular, for a training image J with visual features $\{r_1, \ldots, r_T\}$ and caption $\{y_1, \ldots, y_N\}$,

$$k(x|J) = \frac{1}{T}\sum_{t=1}^{T} \frac{e^{-\frac{1}{2}(x-r_t)^T \Sigma^{-1}(x-r_t)}}{\sqrt{(2\pi)^d|\Sigma|}}.$$

The kernel bandwidth $\Sigma = \sigma I$ is estimated from held-out data. This yields the probability of a concept $c \in \mathcal{V}$ being present in a test image I as

$$p(c|I) \propto \sum_J \hat{P}(c|J)\, k(I, J) = \sum_J \hat{P}(c|J) \left[\prod_{t=1}^{T} k(x_t|J)\right], \tag{4}$$

where $\hat{P}(c|J)$ is a smoothed estimate of relative frequency of c in the caption of J.

While, upon first glance, the CRM seems to differ fundamentally from the HMM paradigm proposed here, deeper analysis reveals remarkable similarities. In particular, if each state $c \in \mathcal{V}$ of the HMM has exactly as many Gaussian densities in the mixture as the number of training images-regions that contain c in their caption, the densities have mean $\mu_{m,c} = r_t$, the variances $\Sigma_{m,c}$ are constant, and the weights $w_{m,c}$ are uniform, the probability calculations of the HMM and the CRM are nearly identical.

The HMM, by modeling each c with a small mixture of Gaussians, performs an *abstraction* of the information presented in the paired image+caption training data. For each concept $c \in \mathcal{V}$, only the sufficient statistics (mean and covariance) of the

image-features are retained. Consequently, the HMM performs the test annotations very fast, since each image-feature vector x_t needs to be compared only with the Gaussian-mixture representation of each concept, not each training image. *This results in tremendous computational speed-ups compared to the CRM*, which in principle needs to compare the test-image with every training image. The annotation performance of the HMM and CRM are comparable, as will be shown in Section 4.

3 HMM Design, Training and Indexing Issues

We next discuss several design choices that need to be made while using HMMs: model topology selection, parameterization, estimation and decoding (inference).

- HMM topology and permitted transitions: Given a set of image+caption pairs, we construct, for each image, an HMM with as many states as the number of words in its caption. The states of the HMM are *fully connected*, amounting to permitting any object to be present adjacent to any other object. While we have not yet done so, one could investigate the exploitation of the spatial propensities of objects such as `sky` and `grass` to appear in certain positions t in the image, and the propensity of other objects such as `tiger` and `sky` (not to) appear in adjacent blocks.
- Shared-state HMMs: A caption $c \in \mathcal{V}$ appearing in two different images is modeled by the same state. The HMM for each image "shares" states from a common *pool* of $|\mathcal{V}|$ states, *i.e.,* the states of the HMM of each training image are "tied" to the corresponding states of other HMMs. $|\mathcal{V}| = 375$ for COREL; $|\mathcal{V}| = 75$ for TRECVID.
- Accounting for unlabeled objects: In several images, the annotators do not label some obvious concept in-spite of having a word in $\mathcal{V}$ corresponding to it. For a far-from-the-most-egregious example of such omissions, note that grass and the fence occupy roughly equal areas in the image of Figure 1, and yet the caption includes `fence` but not `grass`. To account for such omissions, an additional concept, which we call `null`, is introduced into the vocabulary and is added forcibly to the caption of each image. Since an HMM is a generative model, the entire image needs to be accounted for by the model, and adding a `null` to the set of permissible states results in better modeling of the data.

3.1 HMM Training Issues

Maximum likelihood estimation of the HMM parameters is performed in a standard manner as prescribed in [7]. We initially use a single Gaussian probability density function (pdf) for each state $c \in \mathcal{V}$, initialized with the common mean and variance computed from all the images in the training data. Several iterations of the EM algorithm are carried out to update the means and variances of the HMMs. Transition probabilities are not updated in the experiments reported below.

- Mixture splitting: Once the single Gaussian pdf's have been estimated, the pdf for each state $c \in \mathcal{V}$ is replaced by a mixture of a pair of Gaussian pdf obtained by minor perturbation. Further iterations of the EM algorithm are then carried out

to (re)estimate the Gaussian mixture pdf's. This procedure is standard in training speech recognition systems and is well described in [7]. As long as the data-likelihood increases (substantially) by increasing the number of mixture components in $f(x|c)$, we continue to increase the mixture size of the output pdf's of each state until a certain pre-specified maximum mixture size is attained.

– Variance-floor: It is standard in estimation of Gaussian pdf's to impose a "variance floor" — a minimum value for the estimate of the variance. Typically, this is set to be a fraction of the total empirical variance of all the image data. This prevents overfitting of the model, as well as avoids numerical (divide-by-zero) issues.

3.2 Image Annotation Issues

For indexing test images, a fully connected $|\mathcal{V}|$ state HMM is used, one state corresponding to each $c \in \mathcal{V}$, with $f(\cdot|c)$ as estimated above. Transition probabilities of the HMM are set to be uniform. Given a test image I, the Balm-Welch algorithm is used to compute the posterior probability of (3), which in turn is used to rank all images in response to a single-word query comprised of c.

The transition probabilities of the decoding HMM need not be uniform. A simple variation is to compute the co-occurrence statistics of the words in the concept vocabulary, and use such statistics to adjust the transition probabilities. For instance, `plane` and `sky` tend to co-occur much more often than `plane` and `water`, suggesting that $p(\texttt{sky}|\texttt{plane})$ could be set higher than $p(\texttt{water}|\texttt{plane})$. In a contrastive experiment, we set these probabilities to be proportional to the corresponding relative frequencies.

Specifically, for each concept $c \in \mathcal{V}$, consider all training images whose captions contain c. For each such caption, enumerate *co-occurrence* pairs (c, c'), where $c' \neq c$ denotes other concepts in the same caption. For example, if c is `plane`, and an image is captioned $C \equiv \{\texttt{sky}, \texttt{plane}, \texttt{clouds}\}$, then the co-occurrences enumerated are $(\texttt{plane}, \texttt{sky})$ and $(\texttt{plane}, \texttt{clouds})$. Then we set

$$p(c'|c) = \begin{cases} 0.8 \times \frac{\text{co-occurrence count of } (c,c')}{\sum_{c'' \in \mathcal{V}} \text{co-occurrence count of } (c,c'')}, & c' \neq c \\ 0.2, & c' = c. \end{cases} \tag{5}$$

Since the frequent concepts often co-occur with each other, e.g. `sky` and `water` in the COREL data-set, and the rare concepts co-occur with one of the freuent concepts, the HMM remains fully connected. The choice of the self-loop probability, 0.2, is ad hoc.

4 Experimental Results

We were kindly provided the processed training and test data for two image collections described below, one of still images and another of key frames extracted from video, by the 2004 Johns Hopkins University Summer Workshop team [9]. We use the image-segmentation and features — color moments, texture coefficients and oriented-edge histograms — extracted by [9]. State-of-the-art image indexing and retrieval performance has been published on these data-sets, and provides us a way to make controlled comparisons with related techniques.

1. The COREL data-set consists of 5000 images from 50 Corel Stock Photo CDs provided to us by [2]. Each CD contains 100 images, of which 90 are allocated to the training set and 10 to the test set, resulting in 4500 training images and a balanced 500-image test collection. Each image is divided into 6×4=24 blocks, and 30-dimensional visual features are extracted from each block. The caption vocabulary has $|\mathcal{V}| = 375$ words. This training and test partition is also used by [5, 10].
2. The TRECVID feature-detection data-set [11] consists of key-frames extracted from news video from several broadcast sources. A community-wide effort has resulted in 44,100 images from the TRECVID 2002 corpus being annotated with a set of 137 concepts. This collection was divided into four parts by [12]: *concept-train*, *concept-validate*, *concept-fusion-1* and *concept-fusion-2*. We use the first three subsets (35K images) to train the HMMs, and hold out *concept-fusion-2* (9K images) for image retrieval experiments. Each image is divided into 7×5=35 blocks, and 76-dimensional image-features are extracted from each. The concept vocabulary for these experiments has been culled to $|\mathcal{V}| = 75$ words (cf [9]).

To evaluate image retrieval performance, we follow [5] and construct single-word queries for each word c in the concept vocabulary $\mathcal{V}$. An image from the corresponding test collection is deemed "relevant" to a query if its true caption contains the query-word c; other images are deemed irrelevant. We compute non-interpolated *mean average precision* (mAP) over all queries, a standard measure for ranked retrieval (cf [11]).

While $|\mathcal{V}| = 375$ for COREL, only 260 of the concepts have at least one relevant image in the 500-image test collection. The mAP can therefore be computed only on this subset of single-word queries for COREL; mAP is computed over all $|\mathcal{V}| = 75$ single-word queries for TRECVID.

4.1 Increasing the Number of Gaussian Mixture Components

One expects that increasing model complexity results initially in better models and improved indexing, until overtraining sets in. We investigate this by varying M, the number of Gaussian pdfs in the mixture of (2), and measuring mAP on the two test collections described above. The variance-floor for each pdf is 1% of the global variance of all image-features. The decoding HMM has uniform transition probabilities: $\frac{1}{\mathcal{V}}$.

Table 1 reports performance on the COREL and TRECVID collections. Note that the models improve as expected. The improvement in mAP, for instance, from 2 to 10 mixture components for the COREL collection is statistically significant at a p-value of 0.002, while that from 16 to 20 on TRECVID at a p-value of 0.006.

Recall that the COREL training set is almost an order of magnitude smaller than the TRECVID training set. Consequently, while performance on COREL begins to level off at $M = 10$, TRECVID exhibits little overtraining. Retrieval results with $M = 100$ for TRECVID will be presented in Section 4.5.

4.2 Using Word Co-occurrence Probabilities

One also expects to see some benefit from using the word co-occurrence probabilities of (5) instead of uniform transition probabilities in the decoding HMM. The results of

Table 1. Effect of mixture-size on image retrieval performance.

Mixture-size	1	2	8	10	12
COREL mAP	0.132	0.140	0.161	**0.169**	0.167

Mixture size	1	10	12	16	20
TRECVID mAP	0.095	0.141	0.145	0.157	**0.163**

Table 2. Effect of co-occurrence probabilities on image retrieval performance.

Transition Probabilities	Uniform	From (5)
COREL mAP	0.169	**0.178**
TRECVID mAP	0.163	**0.165**

this investigation are reported in Table 2. The variance floor is held at 1% of the global variance, with $M = 10$ for COREL and $M = 20$ for TRECVID.

While persistent across data-sets, the mAP improvements are not statistically significant: on COREL, it is significant only at a p-value of 0.09, and on TRECVID at 0.19. However, they are easy to estimate and entail no additional computation during indexing. So we retain the use of co-occurrence probabilities in subsequent experiments.

4.3 Lowering the Variance Floor During Density Estimation

Another way of controlling overfitting of Gaussian mixture models is to prevent the estimate of variance from being too small. We start with a (conservative) floor of 1% of the total variance of all the image-features in the training data on the variance of each mixture component of every state emission density $f(\cdot|c)$. Alternatively, we also estimate HMMs with lower variance floors — 0.2%, 0.1%, 0.05% and 0.02% of the global variance — and evaluate image retrieval performance. The number M of mixture components is held at 10 for COREL and 20 for TRECVID, and the corresponding word co-occurrence probabilities are used in the decoding HMM.

Table 3 reports the mAP on the two collections. The results indicate that 1% is indeed a conservative limit, and lower settings improve retrieval performance. The mAP improvement from 0.179 to 0.192 on COREL is statistically significant at a p-value of 0.03, but from 0.165 to 0.169 on TRECVID is not significant (p=0.24).

4.4 Comparison with the Normalized CRM and Statistical MT Models

On the same COREL training/test partition, the same 260 single-word queries and the same 30-dim features, the normalized CRM of [10] has a considerably higher mAP of 0.26, while the MT model of [2] has a lower mAP of 0.15 (cf [9]). For the 75 single-word TRECVID queries, with identical 76-dim visual features, the normalized CRM has a mAP of 0.16 and the MT model 0.13 (again see [9]). Table 3 therefore suggests that the HMM approach is already competitive with alternatives proposed in the literature, with a high potential to be significantly better as discussed in Section 5.

Table 3. Effect of the variance-floor on image retrieval performance.

Variance-Floor	0.01	0.002	0.001	0.0005	0.0002
COREL mAP	0.178	0.179	**0.192**	0.185	—
TRECVID mAP	0.165	—	**0.169**	—	0.164

4.5 Larger HMMs for the TRECVID Data-Set

Since Table 1 suggests that overtraining has not yet set in at $M = 20$ for the TRECVID data-set, we gradually increase the number of mixture components up to 100, enforcing a variance floor of 0.001. Word co-occurrence probabilities are used during indexing.

The mAP of the resulting model is 0.186, which significantly better (p=0.002) than 0.169, the best result of Table 3. It is also significantly better than the CRM or MT model for the same visual features. The authors of [10] report in a private communication that the CRM attains a mAP of 0.182 using different (30-dim) visual features.

Figure 2 illustrates the top 5 retrieved images for 3 out of the 75 queries, and their top caption words according to (3). Words present in the manual annotation are underlined. The probability (3) of the query-word given the test image is used to *rank* images in the test collection. Note that in the ranked-retrieval paradigm adopted here, many of the top-ranked images are relevant even when their automatic captions contain errors.

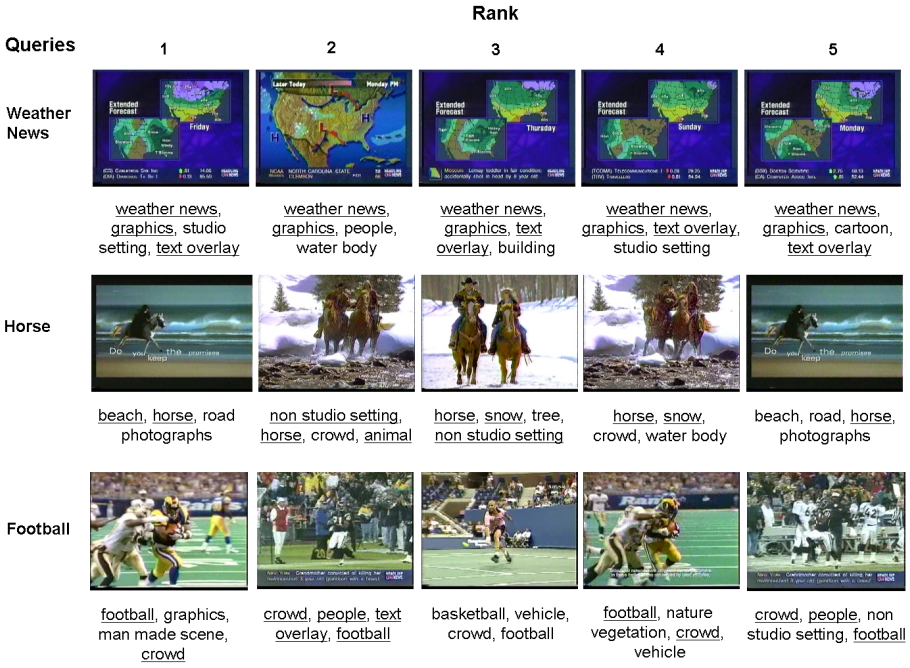

Fig. 2. Retrieved shots for a few TRECVID queries, and their top-4 automatic captions.

5 Concluding Remarks

We have presented a novel method for image annotation for the purpose of content based image retrieval, and evaluated it in several ways to establish that it is indeed competitive with or better than the state of the arton standard data-sets.

Much more is known among practitioners of automatic speech recognition about training and decoding using HMMs, including context-dependent modeling, unsupervised adaptation, adaptive and discriminative training, and minimum error decoding, which remains to be applied to the image annotation and retrieval problem. We expect that significant improvements in image annotation and retrieval performance will be obtained by intelligently importing such ideas from the speech recognition literature.

6 Acknowledgments

The authors gratefully acknowledge the considerable assistance, particularly standardized image data-sets and visual features, provided by Pinar Duygulu, Giri Iyengar, R. Manmatha and the 2004 Johns Hopkins Summer Workshop team. This research is partially support by National Science Foundation grant number ITR-IIS-0121285.

References

1. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition. (2001) I–511–I–518
2. Duygulu, P., Barnard, K., de Freitas, N., Forsyth, D.: Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary. In: Seventh European Conference on Computer Vision. Volume 4. (2002) 97–112
3. Barnard, K., Duygulu, P., de Freitas, N., Forsyth, D., Blei, D.M., Jordan, M.I.: Matching words and pictures. Journal of Machine Learning Research **3** (2003) 1107–1135
4. Blei, D.M., Jordan, M.I.: Modeling Annotated Data. In: 26th Annual International ACM SIGIR Conference. (2003) 127–134
5. Jeon, J., Lavrenko, V., Manmatha, R.: Automatic Image Annotation and Retrieval using Cross-Media Relevance Models. In: 26th Annual International ACM SIGIR COnference. (2003) 119–126
6. Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE **77** (1989) 257–286
7. S. Young, e.a.: The HTK Book. (2002)
8. Lavrenko, V., Feng, S.L., Manmatha, R.: Statistical models for automatic video annotation and retrieval. In: Proc. IEEE International Conf. on Acoustics, Speech and Signal Processing. Volume 3. (2003) 17–21
9. G. Iyengar, e.a.: Joint Visual-Test Modeling for Multimedia Retrieval. In: Available at: http://www.clsp.jhu.edu/ws2004/groups/ws04vstxt/. (2004)
10. Feng, S.L., Manmatha, R., Lavrenko, V.: Multiple Bernoulli relevance models for image and video annotation. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition. Volume 2. (2004) II–1002–II–1009
11. NIST. In: Proceedings of the TREC Video Retrieval Evaluation Conference (TRECVID2003). (2003)
12. A. Amir, e.a.: IBM Research TRECVID-2003 Video Retrieval System. In: Proc. TRECVID2003. (2003)