Automatic Linguistic Indexing of Pictures by a Statistical Modeling Approach

Jia Li, *Member*, *IEEE*, and James Z. Wang, *Member*, *IEEE*

Abstract—Automatic linguistic indexing of pictures is an important but highly challenging problem for researchers in computer vision and content-based image retrieval. In this paper, we introduce a statistical modeling approach to this problem. Categorized images are used to train a dictionary of hundreds of statistical models each representing a concept. Images of any given concept are regarded as instances of a stochastic process that characterizes the concept. To measure the extent of association between an image and the textual description of a concept, the likelihood of the occurrence of the image based on the characterizing stochastic process is computed. A high likelihood indicates a strong association. In our experimental implementation, we focus on a particular group of stochastic processes, that is, the two-dimensional multiresolution hidden Markov models (2D MHMMs). We implemented and tested our ALIP (Automatic Linguistic Indexing of Pictures) system on a photographic image database of 600 different concepts, each with about 40 training images. The system is evaluated quantitatively using more than 4,600 images outside the training database and compared with a random annotation scheme. Experiments have demonstrated the good accuracy of the system and its high potential in linguistic indexing of photographic images.

Index Terms—Content-based image retrieval, image classification, hidden Markov model, computer vision, statistical learning, wavelets.

✦

---

1 INTRODUCTION

A picture is worth a thousand words. As human beings, we are able to tell a story from a picture based on what we have seen and what we have been taught. A 3-year old child is capable of building models of a substantial number of concepts and recognizing them using the learned models stored in her brain. Can a computer program learn a large collection of semantic concepts from 2D or 3D images, build models about these concepts, and recognize them based on these models? This is the question we attempt to address in this work.

Automatic linguistic indexing of pictures is essentially important to content-based image retrieval and computer object recognition. It can potentially be applied to many areas, including biomedicine, commerce, the military, education, digital libraries, and Web searching. Decades of research have shown that designing a generic computer algorithm that can learn concepts from images and automatically translate the content of images to linguistic terms is highly difficult.

Much success has been achieved in recognizing a relatively small set of objects or concepts within specific domains. There is a rich resource of prior work in the fields of computer vision, pattern recognition, and their applications [9]. Space limitation does not allow us to present a broad survey. Instead, we try to emphasize some work most related to what we propose. The references below are to be taken as examples of related work, not as the complete list of work in the cited areas.

1.1 Related Work on Indexing Images

Many content-based image retrieval (CBIR) systems have been developed since the early 1990s. A recent article published by Smeulders et al. reviewed more than 200 references in this ever changing field [20]. Readers are referred to that article and some additional references [2], [17], [18], [25], [23], [4], [11], [26] for more information.

Most of the CBIR projects aimed at general-purpose image indexing and retrieval systems focus on searching images visually similar to the query image or a query sketch. They do not have the capability of assigning comprehensive textual description automatically to pictures, i.e., linguistic indexing, because of the great difficulty in recognizing a large number of objects. However, this function is essential for linking images to text and consequently broadening the possible usages of an image database.

Many researchers have attempted to use machine-learning techniques for image indexing and retrieval [16], [24]. In 1997, a system developed by Minka and Picard included a learning component. The system internally generated many segmentations or groupings of each image's regions based on different combinations of features, then learned which combinations best represented the semantic categories given as examples by the user. The system requires the supervised training of various parts of the image.

A growing trend in the field of image retrieval is to automate linguistic indexing of images by statistical classification methods. The Stanford SIMPLIcity system [22] uses statistical classification methods to group images into rough semantic classes, such as textured-nontextured, graph-photograph. Potentially, the categorization enhances retrieval by permitting semantically adaptive searching methods and by narrowing down the searching range in a database. The

● *J. Li is with the Department of Statistics, The Pennsylvania State University, University Park, PA 16802. E-mail: jiali@stat.psu.edu.*
● *J.Z. Wang is with the School of Information Sciences and Technology, The Pennsylvania State University, University Park, PA 16802. E-mail: jwang@ist.psu.edu.*

approach is limited because these classification methods are problem specific and do not extend straightforwardly.

Recent work in associating images explicitly with words was done at the University of California at Berkeley by Barnard and Forsyth [1] and Duygulu et al. [8]. Using region segmentation, Barnard and Forsyth [1] explored automatically annotating entire images and Duygulu et al. [8] focused on annotating specific regions. The work has achieved some success for certain image types. But, as pointed out by the authors in [1], one major limitation is that the algorithm relies on semantically meaningful segmentation which is, in general, unavailable to image databases. Automatic segmentation is still an open problem in computer vision [27], [19].

1.2 Our Approach

In our work, categories of images, each corresponding to a concept, are profiled by statistical models, in particular, the two-dimensional multiresolution hidden Markov model (2D MHMM) [13]. The pictorial information of each image is summarized by a collection of feature vectors extracted at multiple resolutions and spatially arranged on a pyramid grid. The 2D MHMM fitted to each image category plays the role of extracting representative information about the category. In particular, a 2D MHMM summarizes two types of information: clusters of feature vectors at multiple resolutions and the spatial relation between the clusters, both across and within resolutions. As a 2D MHMM is estimated separately for each category, a new category of images added to the database can be profiled without repeating computation involved with learning from the existing categories. Since each image category in the training set is manually annotated, a mapping between profiling 2D MHMMs and sets of words can be established. For a test image, feature vectors on the pyramid grid are computed. Consider the collection of the feature vectors as an instance of a spatial statistical model. The likelihood of this instance being generated by each profiling 2D MHMM is computed. To annotate the image, words are selected from those in the text description of the categories yielding highest likelihoods.

Readers are referred to Li and Gray [14] for details on 2D MHMM. Many other statistical image models have been developed for various tasks in image processing and computer vision. Theories and methodologies related to Markov random fields (MRFs) [6], [10], [12], [3] have played important roles in the construction of many statistical image models. For a thorough introduction to MRFs and their applications, see Kindermann and Snell [12] and Chellappa and Jain [3]. Given its modeling efficiency and computational convenience, we consider 2D MHMMs an appropriate starting point for exploring the statistical modeling approach to linguistic indexing.

1.3 Outline of the Paper

The remainder of the paper is organized as follows: The architecture of the ALIP (Automatic Linguistic Indexing of Pictures) system is introduced in Section 2. The model-learning algorithm is described in Section 3. Linguistic indexing methods are described in Section 4. In Section 5, experiments and results are presented. We conclude and suggest future research in Section 6.
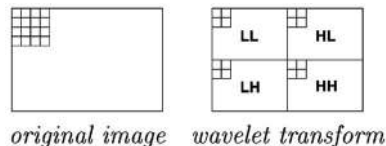

Fig. 1. Decomposition of images into frequency bands by wavelet transforms.

2 SYSTEM ARCHITECTURE

The system has three major components, the feature extraction process, the multiresolution statistical modeling process, and the statistical linguistic indexing process. In this section, we introduce the basics about these individual components and their relationships.

2.1 Feature Extraction

The system characterizes localized features of training images using wavelets. In this process, an image is partitioned into small pixel blocks. For our experiments, the block size is chosen to be 4×4 as a compromise between the texture detail and the computation time. Other similar block sizes can also be used. The system extracts a feature vector of six dimensions for each block. Three of these features are the average color components of pixels in the block. The other three are texture features representing energy in high-frequency bands of wavelet transforms [5]. Specifically, each of the three features is the square root of the second order moment of wavelet coefficients in one of the three high-frequency bands. The features are extracted using the LUV color space, where L encodes luminance and U and V encode color information (chrominance). The LUV color space is chosen because of its good perception correlation properties.

To extract the three texture features, we apply either the Daubechies-4 wavelet transform or the Haar transform to the L component of the image. These two wavelet transforms have better localization properties and require less computation compared to Daubechies' wavelets with longer filters. After a one-level wavelet transform, a 4×4 block is decomposed into four frequency bands, as shown in Fig. 1. Each band contains 2×2 coefficients. Without loss of generality, suppose the coefficients in the HL band are $\{c_{k,l}, c_{k,l+1}, c_{k+1,l}, c_{k+1,l+1}\}$. One feature is then computed as

$$f = \frac{1}{2} \sqrt{\sum_{i=0}^{1} \sum_{j=0}^{1} c_{k+i,l+j}^2}.$$

The other two texture features are computed in a similar manner using the LH and HH bands, respectively.

These wavelet-based texture features provide a good compromise between computational complexity and effectiveness. Unser [21] has shown that moments of wavelet coefficients in various frequency bands can effectively discern local texture. Wavelet coefficients in different frequency bands signal variation in different directions. For example, the HL band reflects activities in the horizontal direction. A local texture of vertical strips thus has high energy in the HL band and low energy in the LH band.
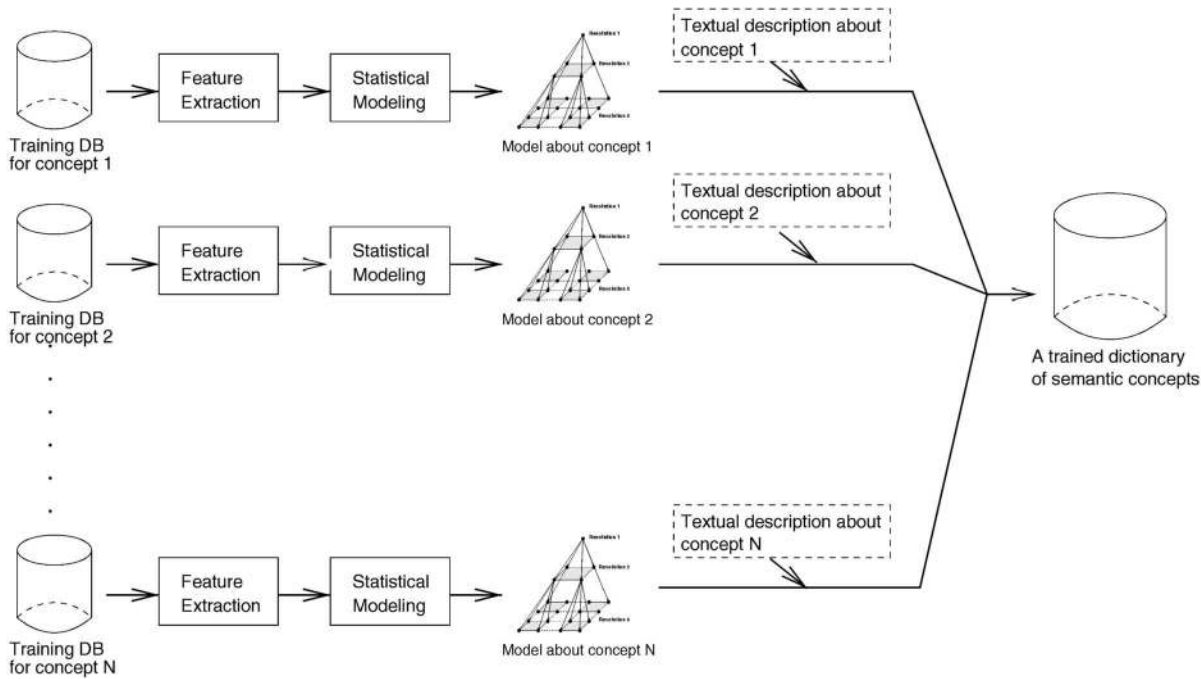
Fig. 2. The architecture of the statistical modeling process.

2.2 Multiresolution Statistical Modeling

Fig. 2 illustrates the flow of the statistical modeling process of the system. We first manually develop a series of concepts to be trained for inclusion in the *dictionary* of concepts. For each concept in this dictionary, we prepare a training set containing images capturing the concept. Hence, at the data level, a concept corresponds to a particular category of images. These images do not have to be visually similar. We also manually prepare a short but informative description about any given concept in this dictionary. Therefore, our approach has the potential to train a large collection of concepts because we do not need to manually create a description about each image in the training database.

Block-based features are extracted from each training image at several resolutions. The statistical modeling process does not depend on a specific feature extraction algorithm. The same feature dimensionality is assumed for all blocks of pixels. A cross-scale statistical model about a concept is built using training images belonging to this concept, each characterized by a collection of multiresolution features. This model is then associated with the textual description of the concept and stored in the concept dictionary.

In the current work, we focus on building statistical models using images that are precategorized and annotated at a categorical level. Many databases contain images not initially categorized, for example, those discussed in [7], [8]. If each image is annotated separately, there are a number of possible approaches to generating profiling models. A clustering procedure can be applied to the collection of annotation words. A cluster of words can be considered as a concept. Images annotated with words in the same cluster will be pooled to train a model. A detailed discussion on word clustering for the purpose of autoannotation is provided in [8]. A more sophisticated approach involves clustering images and estimating a model using images in the same cluster. The clustering of images and the estimation of models can be optimized in an overall manner based on a certain

higher-level statistical model of which the image clusters and profiling 2D MHMMs are components. We have not experimented with these approaches.

2.3 Statistical Linguistic Indexing

The system automatically indexes images with linguistic terms based on statistical model comparison. Fig. 3 shows the statistical linguistic indexing process of the system. For a given image to be indexed, we first extract multiresolution block-based features by the same procedure used to extract features for the training images.

To quantify the statistical similarity between an image and a concept, the likelihood of the collection of feature vectors extracted from the image is computed under the trained model for the concept. All the likelihoods, along with the stored textual descriptions about the concepts, are analyzed by the significance processor to find a small set of statistically significant index terms about the image. These index terms are then stored with the image in the image database for future keyword-based query processing.

2.4 Major Advantages

Our system architecture has several major advantages:

1. If images representing new concepts or new images in existing concepts are added into the training database, only the statistical models for the involved concepts need to be trained or retrained. Hence, the system naturally has good scalability without invoking any extra mechanism to address the issue. The scalability enables us to train a relatively large number of concepts at once.

2. In our statistical model, spatial relations among image pixels and across image resolutions are both taken into consideration. This property is especially useful for images with special texture patterns. Moreover, the modeling approach enables us to
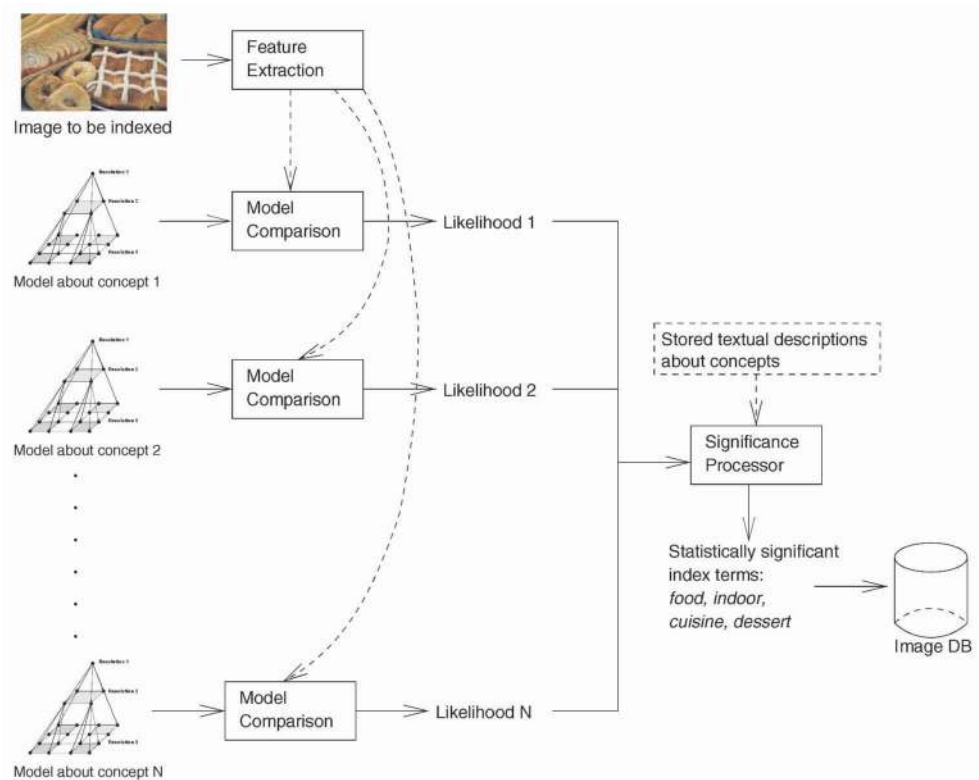
Fig. 3. The architecture of the statistical linguistic indexing process.

avoid segmenting images and defining a similarity distance for any particular set of features. Likelihood can be used as a universal measure of similarity.

3 THE MODEL-BASED LEARNING OF CONCEPTS

In this section, we present in detail a statistical image modeling process that learns a dictionary of a large number of concepts automatically. We describe here assumptions of the 2D MHMM modified from a model originally developed for the purpose of image segmentation [13]. The model is aimed at characterizing the collection of training images, each in their entireties, within a concept.

3.1 Image Modeling

For the purpose of training the multiresolution model, multiple versions of an image at different resolutions are obtained first. The original image corresponds to the highest resolution. Lower resolutions are generated by successively filtering out high-frequency information. Wavelet transforms [5] naturally provide low-resolution images in the low-frequency band (the LL band).

To save computation, features are often extracted from nonoverlapping blocks in an image. An element in an image is therefore a block rather than a pixel. Features computed from one block at a particular resolution form a feature vector and are treated as multivariate data in the 2D MHMM. The 2D MHMM aims at describing statistical properties of the feature vectors and their spatial dependence. The numbers of blocks in both rows and columns reduce by half successively at each lower resolution. Obviously, a block at a lower resolution covers a spatially more global region of the image. As indicated by Fig. 4, a block at a lower resolution is referred to as a parent block and the four blocks at the same spatial

location at the higher resolution are referred to as child blocks. We will always assume such a "quad-tree" split in the sequel since the extension to other hierarchical structures is straightforward.

We first review the basic assumptions of the single resolution 2D HMM as presented in [15]. In the 2D HMM, feature vectors are generated by a Markov model that may change state once every block. Suppose there are M states, the state of block (i,j) being denoted by $s_{i,j}$. The feature vector of block (i,j) is $u_{i,j}$. We use $P(\cdot)$ to represent the probability of an event. We denote $(I',j') < (i,j)$ if $i' < i$ or $i' = i, j' < j$, in which case we say that block (i',j') is before block (i,j). The first assumption is that

$$P(s_{i,j} \mid \text{context}) = a_{m,n,l},$$
$$\text{context} = \{s_{i',j'}, u_{i',j'} : (i',j') < (i,j)\},$$

where $m = s_{i-1,j}$, $n = s_{i,j-1}$, and $l = s_{i,j}$. The second assumption is that, given every state, the feature vectors follow a Gaussian distribution. Once the state of a block is known, the feature vector is conditionally independent of information on other blocks. The covariance matrix Σ_s and the mean vector μ_s of the Gaussian distribution vary with state s.
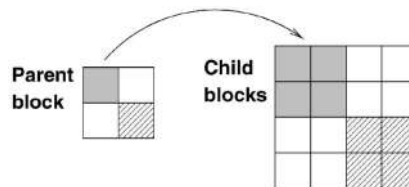

Fig. 4. The image hierarchy across resolutions.

The fact that only feature vectors are observable in a given image accounts for the name "Hidden" Markov Model. The state of a feature vector is conceptually similar to the cluster identity of a vector in unsupervised clustering. As with clustering, the state of a vector is not provided directly by the training data and, hence, needs to be estimated. In clustering, feature vectors are considered as independent samples from a given distribution. In the 2D HMM, feature vectors are statistically dependent through the underlying states modeled by a Markov chain.

For the MHMM, denote the set of resolutions by $\mathcal{R} = \{1, \ldots, R\}$, with $r = R$ being the finest resolution. Let the collection of block indices at resolution r be

$$\mathbb{N}^{(r)} = \{(i, j) : 0 \le i < w/2^{R-r}, 0 \le j < z/2^{R-r}\}.$$

Images are represented by feature vectors at all the resolutions, denoted by $u_{i,j}^{(r)}$, $r \in \mathcal{R}$, $(i, j) \in \mathbb{N}^{(r)}$. The underlying state of a feature vector is $s_{i,j}^{(r)}$. At each resolution r, the set of states is $\{1^{(r)}, 2^{(r)}, \ldots, M_r^{(r)}\}$. Note that, as states vary across resolutions, different resolutions do not share states.

To structure statistical dependence among resolutions, a first-order Markov chain is assumed across the resolutions. In particular, given the states at the parent resolution, the states at the current resolution are conditionally independent of the other preceding resolutions, so that

$$P\left\{s_{i,j}^{(r)} : r \in \mathcal{R}, (i, j) \in \mathbb{N}^{(r)}\right\} = P\left\{s_{i,j}^{(1)} : (i, j) \in \mathbb{N}^{(1)}\right\}$$
$$\prod_{r=2}^{R} P\left\{s_{i,j}^{(r)} : (i, j) \in \mathbb{N}^{(r)} \mid s_{k,l}^{(r-1)} : (k, l) \in \mathbb{N}^{(r-1)}\right\}.$$

In addition, given its state $s_{i,j}^{(r)}$, a feature vector $u_{i,j}^{(r)}$ at any resolution is conditionally independent of any other states and feature vectors. As the states are unobservable, during model estimation, different combinations of states need to be considered. An important quantity to compute is the joint probability of a particular set of states and the feature vectors. Based on the assumptions, we can compute this probability by the following chain rule:

$$P\left\{s_{i,j}^{(r)}, u_{i,j}^{(r)} : r \in \mathcal{R}, (i, j) \in \mathbb{N}^{(r)}\right\} =$$
$$P\left\{s_{i,j}^{(1)}, u_{i,j}^{(1)} : (i, j) \in \mathbb{N}^{(1)}\right\} \times$$
$$P\left\{s_{i,j}^{(2)}, u_{i,j}^{(2)} : (i, j) \in \mathbb{N}^{(2)} \mid s_{k,l}^{(1)} : (k, l) \in \mathbb{N}^{(1)}\right\} \times \cdots \times \quad (1)$$
$$P\left\{s_{i,j}^{(R)}, u_{i,j}^{(R)} : (i, j) \in \mathbb{N}^{(R)} \mid s_{k,l}^{(R-1)} : (k, l) \in \mathbb{N}^{(R-1)}\right\}.$$

At the coarsest resolution, $r = 1$, feature vectors are assumed to be generated by a single resolution 2D HMM. At a higher resolution, the conditional distribution of a feature vector given its state is assumed to be Gaussian. The parameters of the Gaussian distribution depend upon the state at the particular resolution.

Given the states at resolution $r - 1$, statistical dependence among blocks at the finer resolution r is constrained to sibling blocks (child blocks descended from the same parent block). Specifically, child blocks descended from different parent blocks are conditionally independent. In addition, given the state of a parent block, the states of its child blocks are independent of the states of their "uncle" blocks (nonparent blocks at the parent resolution). State transitions among sibling blocks are governed by the same Markovian property
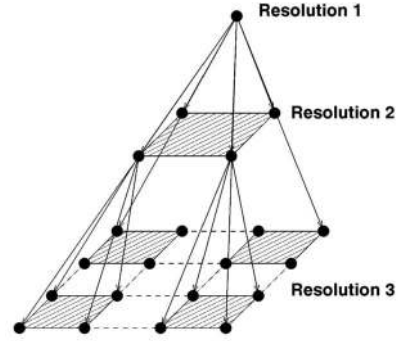

Fig. 5. The hierarchical statistical dependence across resolutions.

assumed for a single resolution 2D HMM. The state transition probabilities, however, depend on the state of their parent block. To formulate these assumptions, denote the child blocks at resolution r of block (k, l) at resolution $r - 1$ by

$$\mathbb{D}(k, l) = \{(2k, 2l), (2k+1, 2l), (2k, 2l+1), (2k+1, 2l+1)\}.$$

According to the assumptions,

$$P\left\{s_{i,j}^{(r)} : (i, j) \in \mathbb{N}^{(r)} \mid s_{k,l}^{(r-1)} : (k, l) \in \mathbb{N}^{(r-1)}\right\} =$$
$$\prod_{(k,l) \in \mathbb{N}^{(r-1)}} P\left\{s_{i,j}^{(r)} : (i, j) \in \mathbb{D}(k, l) \mid s_{k,l}^{(r-1)}\right\},$$

where $P\{s_{i,j}^{(r)} : (i, j) \in \mathbb{D}(k, l) \mid s_{k,l}^{(r-1)}\}$ can be evaluated by transition probabilities conditioned on $s_{k,l}^{(r-1)}$, denoted by $a_{m,n,l}(s_{k,l}^{(r-1)})$. We thus have a different set of transition probabilities $a_{m,n,l}$ for every possible state in the parent resolution. The influence of previous resolutions is exerted hierarchically through the probabilities of the states, which can be visualized in Fig. 5. The joint probability of states and feature vectors at all the resolutions in (1) is then derived as

$$P\left\{s_{i,j}^{(r)}, u_{i,j}^{(r)} : r \in \mathcal{R}, (i, j) \in \mathbb{N}^{(r)}\right\} =$$
$$P\left\{s_{i,j}^{(1)}, u_{i,j}^{(1)} : (i, j) \in \mathbb{N}^{(1)}\right\} \times \prod_{r=2}^{R} \prod_{(k,l) \in \mathbb{N}^{(r-1)}}$$
$$\left(P\left\{s_{i,j}^{(r)} : (i, j) \in \mathbb{D}(k, l) \mid s_{k,l}^{(r-1)}\right\} \prod_{(i,j) \in \mathbb{D}(k,l)} P\left\{u_{i,j}^{(r)} \mid s_{i,j}^{(r)}\right\}\right).$$

To summarize, a 2D MHMM captures both the interscale and intrascale statistical dependence. The interscale dependence is modeled by the Markov chain over resolutions. The intra-scale dependence is modeled by the HMM. At the coarsest resolution, feature vectors are assumed to be generated by an 2D HMM. Fig. 6 illustrates the interscale and intrascale dependencies modeled. At all the higher resolutions, feature vectors of sibling blocks are also assumed to be generated by 2D HMMs. The HMMs vary according to the states of parent blocks. Therefore, if the next coarser resolution has M states, then there are, correspondingly, M HMMs at the current resolution.

The 2D MHMM can be estimated by the maximum likelihood criterion using the EM algorithm. The computational complexity of estimating the model depends on the number of states at each resolution and the size of the pyramid grid. In our experiments, the number of resolutions
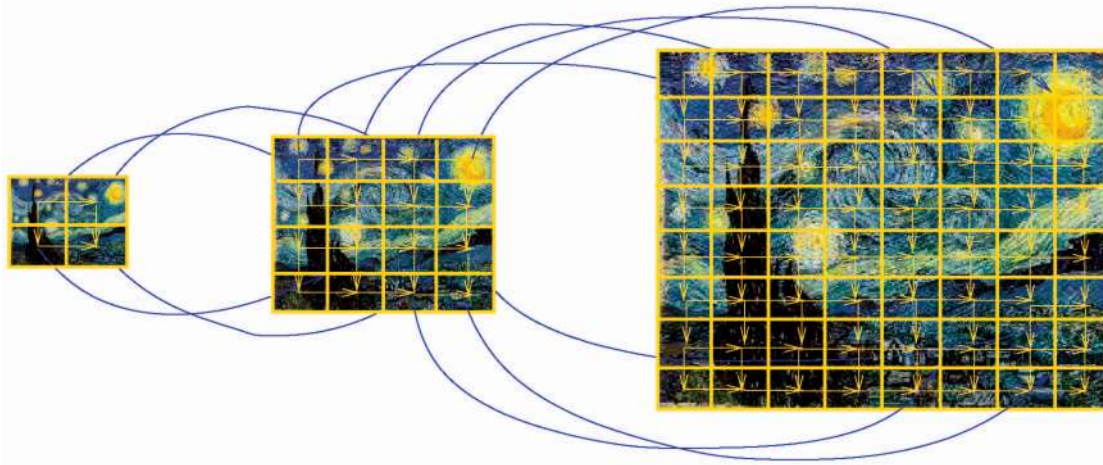
Fig. 6. In the statistical modeling process, spatial relations among image pixels and across image resolutions are both taken into consideration. Arrows, not all drawn, indicate the transition probabilities captured in the statistical model.

is 3; the number of states at the lowest resolution is 3; and those at the two higher resolutions are 4. Details about the estimation algorithm, the computation of the likelihood of an image given a 2D MHMM, and computational complexity are referred to [13].

4   THE AUTOMATIC LINGUISTIC INDEXING OF PICTURES

In this section, we describe the component of the system that automatically indexes pictures with linguistic terms. For a given image, the system compares the image statistically with the trained models in the concept dictionary and extracts the most statistically significant index terms to describe the image. For any given image, a collection of feature vectors at multiple resolutions $\{u_{i,j}^{(r)}, r \in \mathcal{R}, (i,j) \in \mathbb{N}^{(r)}\}$ is computed as described in Section 3. We regard $\{u_{i,j}^{(r)}, r \in \mathcal{R}, (i,j) \in \mathbb{N}^{(r)}\}$ as an instance of a stochastic process defined on a multiresolution grid. The similarity between the image and a category of images in the database is assessed by the log likelihood of this instance under the model $\mathcal{M}$ trained from images in the category, that is,

$$\log P\left\{u_{i,j}^{(r)}, r \in \mathcal{R}, (i,j) \in \mathbb{N}^{(r)} \mid \mathcal{M}\right\}.$$

A recursive algorithm [13] is used to compute the above log likelihood. After determining the log likelihood of the image belonging to any category, we sort the log likelihoods to find the few categories with the highest likelihoods. Suppose k top-ranked categories are used to generate annotation words for the query. The selection of k is somewhat arbitrary. An adaptive way to decide k is to use categories with likelihoods exceeding a threshold. However, it is found that the range of likelihoods computed from a query image varies greatly depending on the category the image belongs to. A fixed threshold is not useful. When there are a large number of categories in the database, it is observed that choosing a fixed number of top-ranked categories tends to yield relatively robust annotation.

Words in the description of the selected k categories are candidates for annotating the query image. If a short description for the query is desired, a certain mechanism

needs to be used to choose a subset of words. There are many possibilities. A system can provide multiple choices for selecting words with only negligible increase of computational load, especially in comparison with the amount of computation needed to obtain likelihoods and rank them. Inspired by hypothesis testing, we explore in detail a particular scheme to choose words. Suppose, in the annotation of the k categories, a word appears j times. If we can reject the hypothesis that the k categories are chosen randomly based on the number of times the word arises, we gain confidence in that the k categories are chosen because of similarity with the query. To reject the hypothesis, we compute the probability of the word appearing at least j times in the annotation of k randomly selected categories. A small probability indicates it is unlikely that the word has appeared simply by chance. Denote this probability by $P(j,k)$. It is given by

$$P(j,k) = \sum_{i=j}^{k} I(i \leq m) \frac{\binom{m}{i}\binom{n-m}{k-i}}{\binom{n}{k}}$$
$$= \sum_{i=j}^{k} I(i \leq m) \frac{m! \ (n-m)! \ k! \ (n-k)!}{i! \ (m-i)! \ (k-i)! \ (n-m-k+i)! \ n!},$$

where $I(\cdot)$ is the indicator function that equals 1 when the argument is true and 0 otherwise, n is the total number of image categories in the database, and m is the number of image categories that are annotated with the given word. The probability $P(j,k)$ can be approximated as follows, using the binomial distribution if $n, m >> k$:

$$P(j,k) = \sum_{i=j}^{k} \binom{k}{i} p^i (1-p)^{k-i} = \sum_{i=j}^{k} \frac{k!}{i!(k-i)!} p^i (1-p)^{k-i},$$

where $p = m/n$ is the percentage of image categories in the database that are annotated with this word or, equivalently, the frequency of the word being used in annotation. A small value of $P(j,k)$ indicates a high level of significance for a given word. We rank the words within the description of the most likely categories according to their statistical significance. Most significant words are used to index the image.

Intuitively, assessing the significance of a word by $P(j,k)$ is attempting to quantify how surprising it is to see the
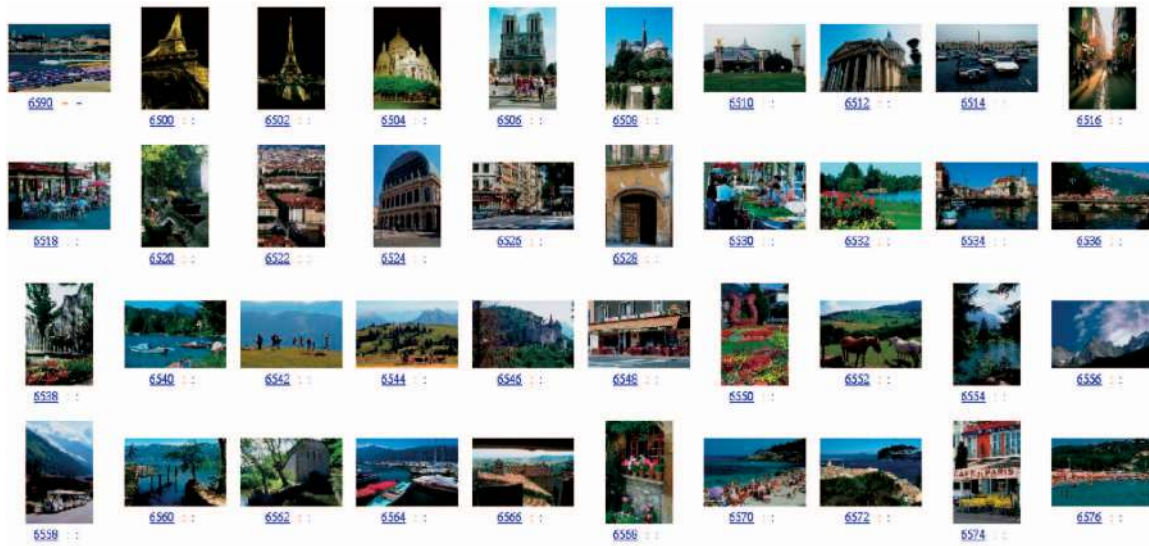
Fig. 7. Training images used to learn a given concept are not necessarily all visually similar. For example, these 40 images were used to train the concept of *Paris/France* with the category description: "Paris, European, historical building, beach, landscape, water."

Fig. 8. Training images used to learn the concept of *male* with the category description: "man, male, people, cloth, face."

word. Words may have vastly different frequencies of being used to annotate image categories in a database. For instance, many more categories may be described by "landscape" than by "dessert." Therefore, obtaining the word "dessert" in the top ranked categories matched to an image is in a sense more surprising than obtaining "landscape" since the word "landscape" may have a good chance of being selected even by random matching.

The proposed scheme of choosing words favors "rare" words. Hence, if the annotation is correct, it tends to provide relatively specific or interesting information about the query. On the other hand, the scheme is risky since it avoids, to a certain extent, using words that fit a large number of image categories.

5 EXPERIMENTS

To validate the methods we have described, we implemented the components of the ALIP system and tested with a general-purpose image database including about 60,000 photographs.

These images are stored in JPEG format with size 384×256 or 256×384. The system was written in the C programming language and compiled on two Unix platforms: Linux and Solaris. In this section, we describe the training concepts and show indexing results.

5.1 Training Concepts

We conducted experiments on learning-based linguistic indexing with a large number of concepts. The system was trained using a subset of 60,000 photographs based on 600 CD-ROMs published by COREL Corporation. Typically, each COREL CD-ROM of about 100 images represents one distinct topic of interest. Images in the same CD-ROM are often not all visually similar. Fig. 7 shows the those images used to train the concept of *Paris/France* with the description: "Paris, European, historical building, beach, landscape, water." Images used to train the concept *male* are shown in Fig. 8. For our experiment, the dictionary of concepts contains all 600 concepts, each associated with one CD-ROM of images.

TABLE 1
Examples of the 600 Categories and Their Descriptions

ID	Category Descriptions
0	Africa, people, landscape, animal
10	England, landscape, mountain, lake, European, people, historical building
20	Monaco, ocean, historical building, food, European, people
30	royal guard, England, European, people
40	vegetable
50	wild life, young animal, animal, grass
60	European, historical building, church
70	animal, wild life, grass, snow, rock
80	plant, landscape, flower, ocean
90	European, historical building, grass, people
100	painting, European
110	flower
120	decoration, man-made
130	Alaska, landscape, house, snow, mountain, lake
140	Berlin, historical building, European, landscape
150	Canada, game, sport, people, snow, ice
160	castle, historical building, sky
170	cuisine, food, indoor
180	England, landscape, mountain, lake, tree
190	fitness, sport, indoor, people, cloth
200	fractal, man-made, texture
210	holiday, poster, drawing, man-made, indoor
220	Japan, historical building, garden, tree
230	man, male, people, cloth, face
240	wild, landscape, north, lake, mountain, sky
250	old, poster, man-made, indoor
260	plant, art, flower, indoor
270	recreation, sport, water, ocean, people
280	ruin, historical building, landmark
290	sculpture, man-made

Every category has 40 training images.

We manually assigned a set of keywords to describe each CD-ROM collection of 100 photographs. The descriptions of these image collections range from as simple or low-level as "mushrooms" and "flowers" to as complex or high-level as "England, landscape, mountain, lake, European, people, historical building" and "battle, rural, people, guard, fight, grass." On average, 3.6 keywords are used to describe the content of each of the 600 image categories. It took the authors approximately 10 hours to annotate these categories. In Tables 1 and 2, example category descriptions are provided.

While manually annotating categories, the authors made efforts to use words that properly describe nearly all if not all images in one category. It is possible that a small number of images are not described accurately by all words assigned to their category. We view them as "outliers" introduced into training for the purpose of estimating the 2D MHMM. In practice, outliers often exist for various reasons. There are ample statistical methods to suppress the adverse effect of them. On the other hand, keeping outliers in training will testify to the robustness of a method. For the model we use, the number of parameters is small relative to the amount of training data. Hence, the model estimation is not anticipated to be affected considerably by inaccurately annotated images. We therefore simply use those images as normal ones.

5.2 Categorization Performance in a Controlled Database

To provide numerical results on the performance, we evaluated the system based on a controlled subset of the COREL database, formed by 10 image categories (African people and villages, beach, buildings, buses, dinosaurs, elephants, flowers, horses, mountains and glaciers, food), each containing 100 pictures. In the next section, we provide categorization and annotation results with 600 categories. Because many of the 600 categories share semantic meanings, the categorization accuracy is conservative for evaluating the annotation performance. For example, if an image of the category with scenes in France is categorized wrongly into the category with European scenes, the system is still useful in many applications. Within this controlled database, we can assess annotation performance reliably by categorization accuracy because the tested categories are distinct and share no description words.

We trained each concept using 40 images and tested the models using 500 images outside the training set. Instead of annotating the images, the program was used to select the category with the highest likelihood for each test image. That is, we use the classification power of the system as an indication of the annotation accuracy. An image is considered to be annotated correctly if the computer predicts the true category the image belongs to. Although these image categories do not share annotation words, they may be

TABLE 2
Examples of the 600 Categories and Their Descriptions (cont.)

ID	Category Descriptions
300	Stmoritz, ski, snow, ice, people
310	texture, man-made, painting
320	texture, natural
330	train, landscape, man-made
340	Virginia, historical building, landscape, rural
350	wild life, art, animal
360	work, people, cloth
370	architecture, building, historical building
380	Canada, British Columbia, landscape, mountain
390	blue
400	Canada, landscape, historical building
410	city, life, people, modern
420	Czech Republic, landscape, historical building
430	Easter egg, decoration, indoor, man-made
440	fashion, people, cloth, female
450	food, man-made, indoor
460	green
470	interior, indoor, man-made
480	marine time, water, ocean, building
490	museum, old, building
500	owl, wild life, bird
510	plant, flower
520	reptile, animal, rock
530	sail, boat, ocean
540	Asia, historical building, people
550	skin, texture, natural
560	summer, people, water, sport
570	car, man-made, landscape, plane, transportation
580	US, landmark, historical building, landscape
590	women, face, female, people

Every category has 40 training images.

semantically related. For example, both the "beach" and the "mountains and glaciers" categories contain images with rocks, sky, and trees. Therefore, the evaluation method we use here only provides a lower bound for the annotation accuracy of the system. Table 3 shows the automatic classification result. Each row lists the percentage of images in one category classified to each of the 10 categories by the computer. Numbers on the diagonal show the classification accuracy for every category.
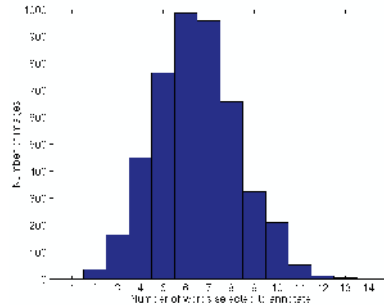

Fig. 9. The histogram of the numbers of words assigned to the test images by our system. For each word in the annotation of the five matched categories, a value indicating its significance is computed and thresholded by 0.0649. A word with a value below the threshold is selected.

5.3 Categorization and Annotation Results

A statistical model is trained for each of the 600 categories of images. Depending on the complexity of a category, the training process takes 15 to 40 minutes of CPU time, with an average of 30 minutes, on an 800 MHz Pentium III PC to converge to a model. These models are stored in a fashion similar to a dictionary or encyclopedia. The training process is entirely parallelizable because the model for each concept is estimated separately.

We randomly selected 4,630 test images outside the training image database and processed these images by the linguistic indexing component of the system. For each of these test images, the computer program selected five concepts in the dictionary with the highest likelihoods of generating the image. For every word in the annotation of the five concepts, the value indicating its significance, as described in Section 4, is computed. The median of all these values is 0.0649. We use the median as a threshold to select annotation words from those assigned to the five matched concepts. Recall that a small value implies high significance. Hence, a word with a value below the threshold is selected.

The histogram of the numbers of words assigned to the test images is provided in Fig. 9. These numbers range from 1 to 14 with median 6. The unique image with only one word assigned to it is shown in Fig. 10a. This image is automatically annotated by "fractal," while the manual description of its category contains two words: "fractal" and "texture." There are two images annotated with as many as 14 words, which are shown in Figs. 10b and 10c. For the first image, the manual

TABLE 3
Results of the Automatic Image Categorization Experiments

%	Africa	beach	build-ings	buses	dino-saurs	ele-phants	flowers	horses	moun-tains	food
Africa	52	2	4	0	8	16	10	0	6	2
beach	0	32	6	0	0	0	2	2	58	0
buildings	8	4	64	0	8	6	0	0	6	4
buses	0	18	6	46	2	8	0	0	16	4
dinosaurs	0	0	0	0	100	0	0	0	0	0
elephants	8	0	2	0	8	40	0	8	34	0
flowers	0	0	2	0	0	0	90	0	2	6
horses	0	2	0	0	0	4	24	60	4	6
mountains	0	6	6	0	2	2	0	0	84	0
food	6	4	0	2	6	0	8	0	6	68

Each row lists the percentage of images in one category classified to each of the 10 categories by the computer. Numbers on the diagonal show the classification accuracy for each category.
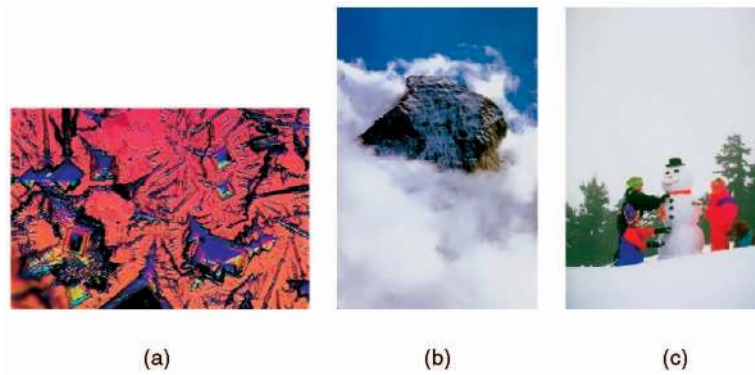
Fig. 10. Three test images. (a) This image is annotated with one word by ALIP. (b) and (c) These two images are annotated with 14 words by ALIP.

TABLE 4
Comparison between the Image Categorization Performance of ALIP and that of a Random Selection Scheme

Accuracy	11.88%	17.06%	20.76%	23.24%	26.05%
Number of top-ranked categories required by ALIP	1	2	3	4	5
Number of categories required by a *random selection* scheme	72	103	125	140	151

Accuracy is the percentage of test images whose true categories are included in top-ranked categories. ALIP requires substantially fewer categories to achieve the same accuracy.

annotation contains "mountain," "snow," "landscape," and the automatically assigned words are "mountain," "rockies," "snow," "ice," "glacier," "sky," "ski," "winter," "water," "surf," "up," "boat," "ship," and "no-fear." The only word discarded by thresholding is "cloud," which would be a good description of the image although not included in the manual annotation. The value indicating its significance is 0.073, quite close to the threshold. Several words outside the manual annotation in fact describe the image quite accurately, e.g., "rockies," "glacier," "sky." This example shows that the computer annotation can sometimes be more specific than the manual annotation that tends to stay at a general level in order to summarize all the images in the category. For the second image, the manual annotation includes "season," "landscape," "autumn," "people," and "plant." The word "autumn" used to annotate the category is not very appropriate for this particular image. The automatically annotated words have no overlap with the manual annotation. The word "people" is marginally discarded by thresholding. Other words assigned to this image include "sport," "fitness," and "fight."

To quantitatively assess the performance, we first compute the accuracy of categorization for the randomly selected test images and then compare the annotation system with a random annotation scheme. Although the ultimate goal of ALIP is to annotate images linguistically, presenting the accuracy of image categorization helps to understand how the categorization supports this goal. Due to the overlap of semantics among categories, it is important to evaluate the linguistic indexing capability. Because ALIP's linguistic indexing capability depends on a categorized training database and a categorization process, the choice of annotation words for the training image categories may improve the usefulness of the training database. The experimental results we are to present here show that both ALIP's image categorization process and linguistic indexing process are of good accuracy.

The accuracy of categorization is evaluated in the same manner as described in Section 5.2. In particular, for each test image, the category yielding the highest likelihood is identified. If the test image is included in this category, we call it a "match." The total number of matches for the 4,630 test images is 550. That is, an accuracy of 11.88 percent is achieved. In contrast, if random drawing is used to categorize the images, the accuracy is only 0.17 percent. If the condition of a "match" is relaxed to having the true category covered by the highest ranked *two* categories, the accuracy of ALIP increases to 17.06 percent, while the accuracy for the random scheme increases to 0.34 percent.

In Table 4, we list the percentage of images whose true categories are included in their corresponding top-ranked k ($k = 1, 2, \ldots, 5$) categories in terms of likelihoods computed by ALIP. As a comparison, we computed the number of categories required to cover the true category at the same accuracy using random selection. When m categories are randomly selected from 600 categories, the probability that the true category is included in the m categories is $\frac{m}{600}$ (derived from sampling without replacement). Therefore, to achieve an accuracy of 11.88 percent by the random scheme, 72 categories must be selected. Table 4 shows details about the comparison.

To compare with the random annotation scheme, all the words in the annotation of the 600 categories are pooled to compute their frequencies of being used. The random scheme selects words independently according to the marginal distribution specified by the frequencies. To compare with words selected by our system using the 0.0649 threshold, six words are randomly generated for each image. The number 6 is the median of the numbers of words selected for all the images by our system, hence considered as a fair value to use.

ALIP with thresholding at 0.0649

Random annotation with 6 words

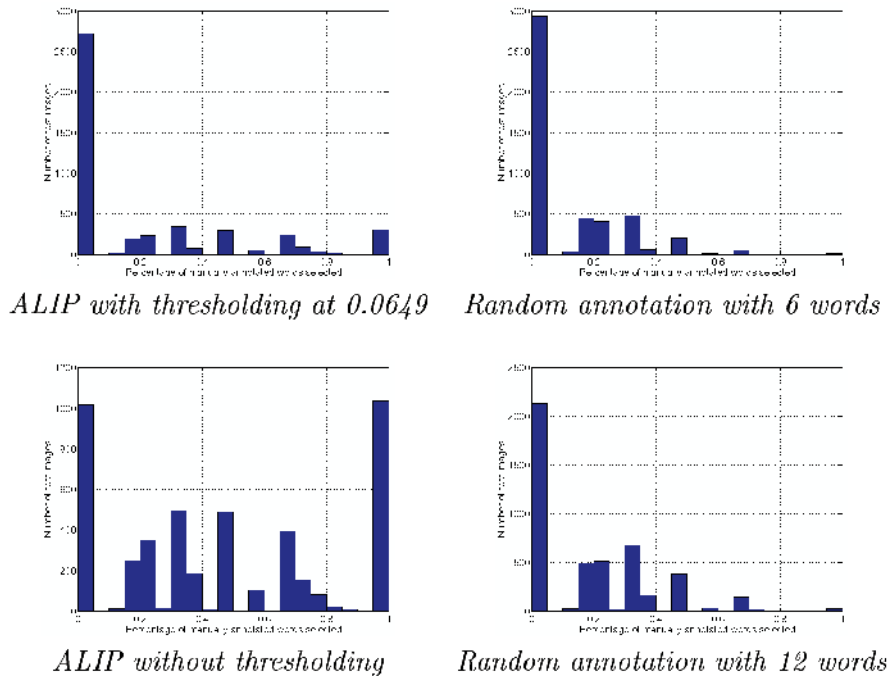ALIP without thresholding

Random annotation with 12 words

Fig. 11. The histograms of the coverage percentages obtained by the ALIP system with and without thresholding and the random scheme based on a test set of 4,630 images.

The quality of a set of annotation words for a particular image is evaluated by the percentage of manually annotated words that are included in the set, referred to as the coverage percentage. It is worth pointing out that this way of evaluating the annotation performance is "pessimistic" because the system may provide accurate words that are not included in the manual annotation, as shown by previous examples. An intelligent system tends to be punished more by the criterion in comparison with a random scheme because, among the words not matched with manually assigned ones, some may well be proper annotation. For our system, the mean coverage percentage is 21.63 percent, while that of the random scheme is 9.93 percent. If all the words in the annotation of the five matched concepts are assigned to a query image, the median of the numbers of words assigned to the test images is 12. The mean coverage percentage is 47.48 percent, while that obtained from assigning 12 words by the random scheme is 17.67 percent. The histograms of the coverage percentages obtained by our system with and without thresholding and the random scheme are compared in Fig. 11.

One may suspect that the 4,630 test images, despite being outside the training set, are rather similar to training images in the same categories and, hence, are unrealistically well annotated. We thus examine the annotation of 250 images taken from five categories in the COREL database using only models trained from the other 595 categories, i.e., no image in the same category as any of the 250 images is used in training. The mean coverage percentages obtained for these images by our system with and without thresholding at 0.0649 are 23.20 percent and 48.50 percent, both slightly higher than the corresponding average values for the previous 4,630 test images. The mean coverage percentages achieved by randomly assigning six and 12 words to each image are 10.67 percent and 17.27 percent. It is thus demonstrated that,

for these 250 images, relying merely on models trained for other categories, the annotation result is at least as good as that of the large test set.

It takes an average of 20 minutes of CPU time to compute all the likelihoods of a test image under the models of the 600 concepts. The computation is highly parallelizable because processes to evaluate likelihoods given different models are independent. The average amount of CPU time to compute the likelihood under one model is only two seconds. We are planning to implement the algorithms on massively parallel computers and provide real-time online demonstrations in the future.

Automatic and manual annotation of the over 4,600 test images can be viewed on the Web.[1] Fig. 12 shows some computer indexing results of 21 randomly selected images outside the training database. Annotation results on four photos taken by the authors and hence not in the COREL database are reported in Fig. 13. The method appears to be highly promising for automatic learning and linguistic indexing of images. Some of the computer predictions seem to suggest that one can control what is to be learned and what is not by adjusting the training database of individual concepts.

6 CONCLUSIONS AND FUTURE WORK

In this paper, we demonstrated our statistical modeling approach to the problem of automatic linguistic indexing of pictures for the purpose of image retrieval. We used categorized images to train a dictionary of hundreds of concepts automatically. Wavelet-based features are used to describe local color and texture in the images. After analyzing all training images for a concept, a two-dimensional multi-resolution hidden Markov model (2D MHMM) is created and

---

1. http://wang.ist.psu.edu/IMAGE/alip.html.

Image	Computer predictions	Image	Computer predictions	Image	Computer predictions
	building,sky,lake, landscape, European,tree		snow,animal, wildlife,sky, cloth,ice,people		people,European, female
	food,indoor, cuisine,dessert		people, European, man-made, water		lake,Portugal, glacier,mountain, water
	skyline, sky, New York, landmark		plant,flower, garden		modern,parade, people
	pattern,flower, red,dining		ocean,paradise, San Diego, Thailand, beach,fish		elephant,Berlin, Alaska
	San Diego, ocean side, beach,Florida, Thailand,building		relic,Belgium, Portugal,art		fitness,indoor, Christmas, cloth,holiday
	flower,flora, plant,fruit, natural,texture		travel,fountain, European, Florida, beach,building		Africa,Kenya, Zimbabuwe, animal,cave
	ancestor, drawing, fitness, history, indoor		hair style, occupation,face, female,cloth		night,cyber, fashion,female

Fig. 12. Some annotations automatically generated by our computer-based linguistic indexing algorithm. The dictionary with 600 concepts was created automatically using statistical modeling and learning. Test images were randomly selected outside the training database.

P: building, snow, sky, tree, landscape

P: flower

P: historical building, (bridge), river, Italy, sky, Europe

P: animal, grass (squirrel)

Fig. 13. Test results using photos not in the COREL collection. Statistical models learned from the COREL collection can be used to index other photographic images. These photos were taken by the authors. **P:** Photographer annotation. Words appeared in the annotation of the five matched categories are underlined. Words that in parentheses are not included in the annotation of any of the 600 training categories.

stored in a concept dictionary. Images in one category are regarded as instances of a stochastic process that characterizes the category. To measure the extent of association between an image and the textual description of an image category, we compute the likelihood of the occurrence of the image based on the stochastic process derived from the category. We have demonstrated that the proposed methods can be used to train models for 600 different semantic concepts and these models can be used to index images linguistically.

The major advantages of our approach are 1) models for different concepts can be independently trained and retrained, 2) a relatively large number of concepts can be trained and stored, and 3) spatial relation among image pixels within and across resolutions is taken into consideration with probabilistic likelihood as a universal measure.

The current system implementation and the evaluation methodology have several limitations.

- We train the concept dictionary using only 2D images without a sense of object size. It is believed that the object recognizer of human beings is usually trained using 3D stereo with motion and a sense of object sizes. Training with 2D still images potentially limits the ability of accurately learning concepts.
- As pointed out by one of the anonymous reviewers, the COREL image database is not ideal for training the system because of its biases. For instance, images in some categories, e.g., "tigers," are much more alike than a general sampling of photographs depicting the concept. On the other hand, images in some categories, e.g., "Asia," are widely distributed visually, making it impossible to train such a concept using only a small collection of such images. Until this limitation is thoroughly investigated, the evaluation results reported should be interpreted cautiously.
- For a very complex concept, i.e., when images representing it are visually diverse, it seems that 40 training images are insufficient for the computer program to build a reliable model. The more complex the concept is, the more training images and CPU time are needed. This is similar to the learning process of a person, who, in general, needs more experience and longer time to comprehend more complex concepts.

In future work, we may improve the indexing speed of the system by using approximation in the likelihood computation. A rule-based system may be used to process the words annotated automatically to eliminate conflicting semantics. Moreover, besides assigning words to an image, weights can be given to the words in the mean time to indicate the believed extent of description appropriateness. Experiments with different applications such as biomedicine and art could be interesting.

ACKNOWLEDGMENTS

REFERENCES

[1] K. Barnard and D. Forsyth, "Learning The Semantics of Words and Pictures," *Proc. Int'l Conf. Computer Vision,* vol 2, pp. 408-415, 2001.

[2] A. Berman and L.G. Shapiro, "Efficient Image Retrieval with Multiple Distance Masures," *Proc. SPIE,* vol. 3022, pp. 12-21, Feb. 1997.

[3] R. Chellappa and A.K. Jain, *Markov Random Fields: Theory and Applications.* Academic Press, 1993.

[4] Y. Chen and J.Z. Wang, "A Region-Based Fuzzy Feature Matching Approach to Content-Based Image Retrieval," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 9, Sept. 2002.

[5] I. Daubechies, *Ten Lectures on Wavelets.* Capital City Press, 1992.

[6] R.L. Dobrushin, "The Description of a Random Field by Means of Conditional Probabilities and Conditions of Its Regularity," *Theory Probability Application,* vol. 13, pp. 197-224, 1968.

[7] P. Duygulu, K. Barnard, and D. Forsyth, "Clustering Art," *Computer Vision and Pattern Recognition,* vol. 2, pp. 434-439, 2001.

[8] P. Duygulu, K. Barnard, N. de Freitas, and D. Forsyth, "Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary," *Proc. European Conf. Computer Vision,* vol. 4, pp. 97-112, 2002.

[9] D.A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach.* Prentice Hall, 2002.

[10] S. Geman and D. Geman, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 6, no. 6, pp. 721-741, Nov. 1984.

[11] Q. Iqbal and J.K. Aggarwal, "Retrieval by Classification of Images Containing Large Manmade Objects Using Perceptual Grouping," *Pattern Recognition J.,* vol. 35, no. 7, pp. 1463-1479, 2002.

[12] R. Kindermann and L. Snell, *Markov Random Fields and Their Applications.* Am. Math. Soc. 1980.

[13] J. Li, R.M. Gray, and R.A. Olshen, "Multiresolution Image Classification by Hierarchical Modeling with Two Dimensional Hidden Markov Models," *IEEE Trans. Information Theory,* vol. 46, no. 5, pp. 1826-41, Aug. 2000.

[14] J. Li and R.M. Gray, *Image Segmentation and Compression Using Hidden Markov Models.* Kluwer Academic, 2000.

[15] J. Li, A. Najmi, and R.M. Gray, "Image Classification by a Two Dimensional Hidden Markov Model," *IEEE Trans. Signal Processing,* vol. 48, no. 2, pp. 517-33, Feb. 2000.

[16] T.P. Minka and R.W. Picard, "Interactive Learning Using a Society of Models," *Pattern Recognition,* vol. 30, no. 3, p. 565, 1997.

[17] S. Ravela and R. Manmatha, "Image Retrieval by Appearance," *Proc. SIGIR,* pp. 278-285, July 1997.

[18] G. Sheikholeslami, S. Chatterjee, and A. Zhang, "WaveCluster: A Multi-Resolution Clustering Approach for Very Large Spatial Databases," *Proc. Very Large Date Bases Conf.,* pp. 428-439, Aug. 1998.

[19] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 8, pp. 888-905, Aug. 2000.

[20] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-Based Image Retrieval at the End of the Early Years," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 12, pp. 1349-1380, Dec. 2000.

[21] M. Unser, "Texture Classification and Segmentation Using Wavelet Frames," *IEEE Trans. Image Processing,* vol. 4, no. 11, pp. 1549-1560, Nov. 1995.

[22] J.Z. Wang, *Integrated Region-Based Image Retrieval.* Dordrecht: Kluwer Academic, 2001.

[23] J.Z. Wang, J. Li, and G. Wiederhold, "SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture LIbraries," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, no. 9, pp. 947-963, Sept. 2001.

[24] J.Z. Wang and M.A. Fischler, "Visual Similarity, Judgmental Certainty and Stereo Correspondence," *Proc. DARPA Image Understanding Workshop,* G. Lukes, ed., vol. 2, pp. 1237-1248, Nov. 1998.

[25] J.Z. Wang, G. Wiederhold, O. Firschein, and X.W. Sha, "Content-Based Image Indexing and Searching Using Daubechies' Wavelets," *Int'l J. Digital Libraries (IJODL),* vol. 1, no. 4, pp. 311-328, 1998.

[26] Q. Zhang, S.A. Goldman, W. Yu, and J.E. Fritts, "Content-Based Image Retrieval Using Multiple-Instance Learning," *Proc. Int'l Conf. Machine Learning,* 2002.

[27] S. Zhu and A.L. Yuille, "Region Competition: Unifying Snakes, Region Growing, and Bayes/MDL for Multiband Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 18, no. 9, pp. 884-900, Sept. 1996.

Jia Li received the BS degree in electrical engineering from Xi'an Jiao Tong University in 1993, the MSc degree in electrical engineering in 1995, the MSc degree in statistics in 1998, and the PhD degree in electrical engineering in 1999, all from Stanford University. She is an assistant professor of statistics and by courtesy appointment in computer science and engineering at The Pennsylvania State University. She worked as a research associate in the Computer Science Department at Stanford University in 1999. She was a researcher at the Xerox Palo Alto Research Center from 1999 to 2000. Her research interests include statistical classification and modeling, data mining, bioinformatics, image processing, and image retrieval. She is a member of the IEEE.

James Z. Wang received the summa cum laude bachelor's degree in mathematics and computer science from the University of Minnesota (1994), the MSc degree in mathematics and the MSc degree in computer science, both from Stanford University (1997), and the PhD degree in medical information sciences from the Stanford University Biomedical Informatics Program and Computer Science Database Group (2000). Since 2000, he has been the holder of the endowed PNC Technologies Career Development Professorship and an assistant professor at the School of Information Sciences and Technology at The Pennsylvania State University. He is also affiliated with the Department of Computer Science and Engineering. He has been a visiting scholar at Uppsala University in Sweden, SRI International, IBM Almaden Research Center, and NEC Computer and Communications Research Lab. He is a member of the IEEE.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** http://computer.org/publications/dlib.