A Bayesian framework for content-based indexing and retrieval

Nuno Vasconcelos and Andrew Lippman
MIT Media Lab, {nuno,lip}@media.mit.edu

Abstract

We pose the content-based retrieval problem as a problem of statistical inference and develop a Bayesian framework for indexing and retrieval in the context of large multimedia databases. All the indexing is carried out in the compressed domain and leads to efficient retrieval without compromise of the compression efficiency. The framework allows the integration of information from different sources and modalities as a way to eliminate some of the most significant limitations of the *query by example* search paradigm, and all the model parameters can be learned from training examples.

1 Introduction

The advent of a fully digital communications landscape, characterized by fast networking, ubiquitous computing and storage, and absence of barriers to the publication and access to information, poses new challenges to communication devices. In particular, it is no longer enough to guarantee robust and bandwidth efficient communication links between source and decoder which, under the new communications paradigm, becomes much more of an information seeking device than the traditional information sink. Given the massive amounts of choice made available by the new modes of communication, it is also necessary to equip decoders with tools to filter, sort, summarize, retrieve, and manipulate content.

While relatively sophisticated architectures are already available to perform these operations over text, little is known on how to extend their principles to visual information, i.e. to images or video. In this area, a significant amount of work has recently started to appear in the context of content-based retrieval, under the *query by example* search paradigm [5, 7, 8, 3]. Here, the user supplies the retrieval system with some sort of example of what he/she is looking for (e.g. an image) and the system then retrieves, from an image or video database, the items that are closest to the submitted example.

One of the important requirements for practical retrieval systems is the ability to jointly address the issues of indexing and compression. By formulating query by example as a problem of Bayesian inference [2] and establishing a link between probability density estimation and vector quantization, we have recently introduced a representation that leads to very efficient procedures for indexing and retrieval directly in the compressed domain without compromise of the coding efficiency [10].

In this paper, we build on the potential of the Bayesian formulation to support sophisticated inference, to incorporate this representation in a very flexible indexing and retrieval framework that 1) leads to intuitive retrieval procedures, 2) can integrate different content modalities and, therefore, eliminate some of the strongest limitations of the query by example paradigm, and 3) supports statistical learning of all the model parameters and can, therefore, be trained automatically.

2 A Bayesian retrieval framework

In order to formulate retrieval as a problem of Bayesian inference, we assume that the items in the content database are a set of observations drawn from a set of M content sources. We next assume that the query $\mathbf{Q}$ submitted to the retrieval system is also an observation from one of the M sources, and define an indicator variable $\mathbf{S} = (S_1, \ldots, S_M)^T$, where $\mathbf{S} = \mathbf{e}_i$ if $\mathbf{Q}$ was drawn from the i^th source, and $\mathbf{e}_i$ is the i^{th} vector of the standard basis of $\mathcal{R}^M$ (i.e. contains a one in the i^{th} position and zeros in the remaining ones).

We then define a Bayesian criteria for retrieval, where finding the closest match to the query $\mathbf{Q}$ in the database corresponds to finding the source S^* such that

$$S^* = \arg\max_i P(S_i = 1|\mathbf{Q}), \tag{1}$$

which by simple application of Bayes rule is equivalent to

$$\begin{aligned} S^* &= \arg\max_i P(\mathbf{Q}|S_i = 1)P(S_i = 1) & (2) \\ &= \arg\max_i \{\log P(\mathbf{Q}|S_i = 1) + \log P(S_i = 1)\}. & (3) \end{aligned}$$

2.1 Probabilistic model

To define a probabilistic model for the observed data, we assume that each observation $\mathbf{X}$ from a given source is composed by K attributes $\mathbf{X} = \{X^{(1)}, \ldots, X^{(K)}\}$ which, although marginally dependent, are independent given the knowledge of which source generated the query, i.e.

$$P(\mathbf{X}|S_i) = \prod_k P(X^{(k)}|S_i). \tag{4}$$

Each attribute is simply a unit of information that contributes to the characterization of the source. Possible examples include image features, audio samples, or text annotations.

This probabilistic model can be expressed graphically as a Bayesian network [6], according to Figure 1. The source state variable $\mathbf{S}$ takes the value of any of the vectors in the standard basis of $\mathcal{R}^M$ according to the prior probabilities $P(S_i = 1)$

of equation (2). Associated to each of the links from the source state variable to the indicator variables S_i, is a conditional probability density

$$P(S_i = 1|\mathbf{S}) = \delta(\mathbf{S} - \mathbf{e}_i), \quad P(S_i = 0|\mathbf{S}) = 1 - P(S_i = 1|\mathbf{S});$$

where

$$\delta(x) = \begin{cases} 1, & \text{if } x = 0 \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

Finally, associated with the links from each of the indicator variables S_i to each of the attribute variables $X^{(k)}$ is the conditional density $P(X^{(k)}|S_i)$.

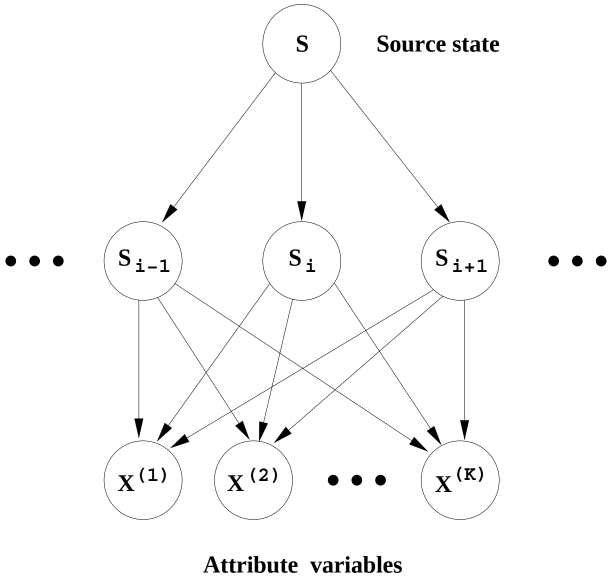

Figure 1: A graphical representation of the probabilistic model for the observations in the database.

2.2 Retrieval as Bayesian inference

In the Bayesian context, retrieval corresponds to finding the source which maximizes equation (2) in response to the instantiation, by the user, of a subset of K content attributes. This instantiation depends on the nature of the attributes themselves. While a keyword attribute is instantiated by the specification of that keyword to the search engine (as is common in text retrieval systems), pictorial attributes can be instantiated by example.

Borrowing the terminology from the Bayesian network literature, we define, for a given query, a set of *observed attributes* $\mathbf{O} = \{X^{(k)}|X^{(k)} = Q^{(k)}\}$ and a set of *hidden attributes* $\mathbf{H} = \{X^{(k)}|X^{(k)}$ is not instantiated by the user$\}$, where $\mathbf{Q} = \{Q^{(k)}|k$ is instantiated$\}$ is the query provided by the user. The likelihood of this query is then given by

$$P(\mathbf{Q}|S_i) = \sum_{\mathbf{H}} P(\mathbf{O}, \mathbf{H}|S_i), \tag{6}$$

where the summation is over all possible configurations of the hidden attributes[1]. Using equation (4) and the fact that $\sum_X P(X|S_i) = 1$,

$$\begin{aligned} P(\mathbf{Q}|S_i) &= P(\mathbf{O}|S_i) \sum_{\mathbf{H}} \prod_{k|X^{(k)} \in \mathbf{H}} P(X^{(k)}|S_i) \\ &= P(\mathbf{O}|S_i) \prod_{k|X^{(k)} \in \mathbf{H}} \sum_{X^{(k)}} P(X^{(k)}|S_i) \\ &= P(\mathbf{O}|S_i), \end{aligned} \tag{7}$$

i.e. the likelihood of the query is simply the likelihood of the instantiated attributes. Or, in terms of the graphical model of Figure 1, the non-observed attribute nodes of the network can simply be disregarded during retrieval. In addition to being intuitively correct, this result is also of considerable practical significance. In practice, it means that the complexity of retrieval grows with the number of attributes specified by the user and not with the number of attributes known to the system, which can therefore be made arbitrarily large.

3 Combining different content modalities

One requirement of sophisticated indexing systems is the ability to integrate information from different content modalities, namely text, audio, and video. This is a consequence not only of the fact that, for most modern multimedia applications, the content is, in fact, multimodal[2], but also of the fact that the integration of information from different sources is a way to eliminate some of the strongest limitations of the query by example paradigm. For example, text annotations enhance the performance of image retrieval systems by allowing the user to express queries in terms of semantic attributes that cannot be easily inferred from the visual properties of the images themselves, e.g. “images of patriotism” or “pictures of vacation spots”.

Because there is no constraint for the attributes $X^{(k)}$ in the model of the previous section to be of the same type, the Bayesian framework can naturally integrate different modalities. Consider, for example, a database of HTML pages containing both text and images. For such a database, some of the attributes in the model could be textual and the remainder visual. Suppose that keywords are used to characterize the text and some compact visual description used to characterize the images. Then, assuming that those visual descriptions can be expressed probabilistically (an issue to which we will get back in section 4), the set of attributes $\mathbf{X}$ in the model of Figure 1 can be decomposed into two subsets $\mathbf{X} = \mathbf{T} \cup \mathbf{V}$, the subset $\mathbf{T}$ containing the keywords understood by the retrieval system, and the subset $\mathbf{V}$ containing the visual attributes.

A natural probabilistic representation for the text attributes is to rely on a

[1] The formulation is also valid in the case of continuous variables with summation replaced by integration.

[2] The World Wide Web being the most proeminent example of this phenomena.

Bernoulli distribution for the conditional probabilities $P(T^{(k)}|S_i)$, i.e. to choose

$$P(T^{(k)}|S_i = 1) = \theta_{ik}^{(1-\delta(T^{(k)}))}(1 - \theta_{ik})^{\delta(T^{(k)})}, \tag{8}$$

where $\delta(x)$ is defined in equation (5) and θ_{ik} is the prior probability that the user will specify the keyword $T^{(k)}$ given that he/she is looking for the database entry drawn from source S_i.

Assuming a query instantiating both text and visual attributes, $\mathbf{Q} = \{\mathbf{Q_t}, \mathbf{Q_v}\}$, and using equations (4) and (7)

$$P(\mathbf{Q}|S_i) = P(\mathbf{Q_v}|S_i) \prod_k P(\mathbf{Q_t}^{(k)}|S_i), \tag{9}$$

from which equation (3) becomes

$$S^* = \arg\max_i \{\log P(\mathbf{Q_v}|S_i = 1) + \sum_k \log P(\mathbf{Q_t}^{(k)}|S_i = 1) + \log P(S_i = 1)\}. \tag{10}$$

The comparison of this equation with equation (3) reveals an alternative interpretation for the Bayesian integration of the information from several sources: that the optimal source is the one which would result from the visual query alone but with a prior consisting of the combination of the second and third terms in the equation. I.e. the text attributes instantiated in the query simply reflect the prior belief, by the user, of which source is most likely to originate the best match to the visual query submitted to the retrieval system. Or, in other words, the text attributes provide a means to constrain the visual search.

Similarly, the second term in the equation can be considered the likelihood function, with the combination of the first and the third forming the prior. In this interpretation, the visual attributes constrain what would be predominantly a text-based search. Both interpretations illustrate the power of the Bayesian framework to integrate information from different sources in a natural and meaningful way.

Consider the first interpretation and assume, for simplicity, that $P(S_i = 1)$ is equal for all sources and can, therefore, be dropped from the summation. Then, using equation (8), the optimal source becomes

$$S^* = \arg\max_i \{\log P(\mathbf{Q_v}|S_i = 1) + \sum_k (1 - \delta(T^{(k)})) \log \theta_{ik} + \sum_k \delta(T^{(k)}) \log(1 - \theta_{ik})\}. \tag{11}$$

The prior (second and third terms) can be seen as a sum of weights, to which each text attribute instantiated with *yes* ($T^{(k)} = 1$) contributes with $\log(\theta_{ik})$ and each instantiated with *no* ($T^{(k)} = 0$) contributes with $\log(1 - \theta_{ik})$.

These functions are plotted in Figure 2, showing that the process is, indeed, very intuitive. In particular, for a given source, attributes which were a priori expected to be set to *yes* during retrieval (θ large), originate a small penalty if instantiated with *yes* and a large penalty if instantiated with *no*. Similarly, attributes which were a priori expected to be set to *no* (θ small), originate a large penalty if instantiated with *yes* and a small penalty if instantiated with *no*. The exact amount of the penalty is a function, for each source i and each attribute k, of the Bernoulli parameter θ_{ik}. We will see in section 5 that this is an interesting property in itself.

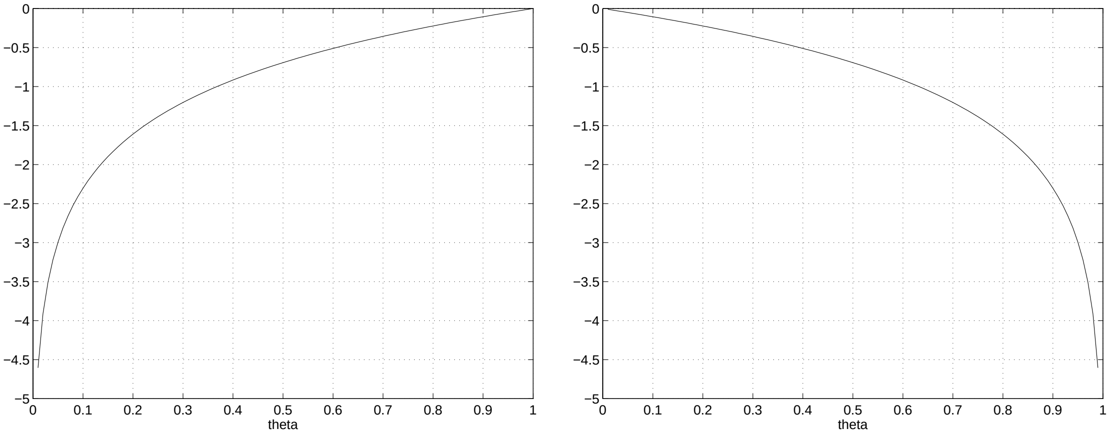

Figure 2: Contribution to the overall prior of attributes instantiated with *yes* (left) and *no* (right) as a function of the Bernoulli parameter θ.

4 Characterizing audio-visual content

One of the challenges of characterizing audio-visual content for the purposes of indexing and retrieval is the sheer amount of data originated by content sources in this category. Due to the magnitude of this problem, virtually every picture or audio sample produced digitally is, sooner or later, stored in some form of compressed format. While compression provides significant savings in terms of storage requirements, it poses new challenges to indexing systems.

In particular, if the compression algorithm does not generate a bitstream already formated in a way that is suitable for retrieval, there is a need for an indexing mechanism that will, for each entry to archive: 1) decode it, 2) reconstruct the compressed images or audio, 3) compute the set of features or content descriptors on which the retrieval system relies, and 4) store those features. This is clearly an inefficient process as it implies a significant amount of duplication of resources in terms of both computation and storage. There is, therefore, a need for coding representations capable of providing support, directly in the compressed domain, for indexing and retrieval without sacrifice of the compression efficiency.

From the point of view of the Bayesian retrieval framework discussed in this paper, this support consists of incorporating in the compressed bit-stream a compact and explicit description of the likelihood function, $P(\mathbf{X_{av}}|S_i = 1)$, for the audio-visual attributes used for retrieval. In [10] we have introduced one such representation, *Library-based Coding*. The main idea is to model the probability density of each source S_i as a mixture of Gaussians

$$P(\mathbf{V}|S_i = 1) = \sum_{k=1}^{C} p_i^{(k)} e^{-\frac{1}{2}(\mathbf{V}-\mu_i^{(k)})^T [\mathbf{\Sigma}_i^{(k)}]^{-1} (\mathbf{V}-\mu_i^{(k)})}, \qquad (12)$$

which is compactly described by the set of parameters $\mu_i = \{\mu_i^{(1)}, \ldots, \mu_i^{(C)}\}$, $\mathbf{\Sigma}_i =$

$\{\mathbf{\Sigma}_i^{(1)}, \ldots, \mathbf{\Sigma}_i^{(C)}\}$, and $\mathbf{p}_i = \{p_i^{(1)}, \ldots, p_i^{(C)}\}$, and where $\mathbf{V}$ is a vector of audio-visual features and C is the number of components in each mixture. These parameters can then be estimated from the images or audio observations drawn from S_i, which constitute the entries to archive in the database, through the Expectation-Maximization (EM) algorithm [1].

Even though the representation is generic and applicable to any feature space, we have worked mostly in the space of image blocks (where $\mathbf{V}$ is a vector of pixel intensities) to achieve compatibility with current coding standards such as JPEG or MPEG. In fact, it turns out, that the EM algorithm is very close to the generalized Loyd algorithm [4] for vector quantizer (VQ) design, and the parameters above can be seen as those of an entropy-constrained VQ under the Mahanolabis distance [10]. Therefore, in addition to providing support for retrieval, the representation is also nearly optimal from a compression standpoint. In practice, the model of equation (12) can be simplified by assuming equally likely Gaussians with identity covariance, in which case, the remaining parameter μ_i becomes a standard VQ codebook.

Each of the entries in this codebook can thus be seen as a probabilistic annotation of the content, and the codebook (or block library) is all that must be decoded for purposes of indexing and retrieval. Indexing and retrieval can, therefore, be performed very efficiently and embedded in hierarchical structures that have various interesting properties [9].

5 Learning the model parameters

One of the interesting properties of Bayesian inference is that it only requires the specification of local probabilities (the conditional probabilities associated with the links in Figure 1), global (or joint) probabilities being inferred by the integration of information from all the variables in the model (e.g. equation (11)). This opens up the possibility for the manual specification of the model parameters, which would be infeasible at the global level.

Consider, for example, the text attributes of equation (8). For a given source i and a given keyword k, the optimal value for the parameter θ_{ik} is nothing more than the answer to the question "given that the user is looking for picture i what is the likelihood that he/she will specify keyword k?". This, in addition to being intuitive, provides a great flexibility with respect to the categorization of the content in the database, by allowing keywords with variable weights.

Imagine a picture of a beach in the Caribbean. A user interested in such a picture is, with high likelihood, looking from pictures of "vacation spots", "beach", "Caribbean" or "water sports". It can also be the case, even though much less likely, that he/she is simply looking for images of "palm trees", "water", or "boats". Under the Bayesian framework, these different levels of relevance can be easily incorporated in the model by simply assigning a larger θ to the former attributes and a smaller θ to the latter ones.

In this case, as seen in section 3 (see Figure 2), the picture would be more penalized in response to the instantiation with *yes*, during retrieval, of the "water" keyword than in response to the instantiation with *yes* of the "Caribbean" keyword. Assuming both θ parameters to be larger than 0.5, the penalty would be even larger for the instantiation of "water" with *no*, and largest for that instantiation of the "Caribbean" attribute. I.e. the image is very unlikely to be returned by the retrieval system if either "Caribbean" or "water" are instantiated with *no*, has a better chance if "water" is instantiated with *yes*, and an even better chance if the user is looking for pictures taken in the "Caribbean". This is a much more flexible indexing paradigm than keyword search with equally weighted attributes.

However, because the framework is probabilistic, it also supports probabilistic learning of the model parameters. Suppose that, instead of setting parameters manually, one would like to estimate them from a training set composed of E examples, each example e consisting of an image (video sequence) $\mathbf{V}_e$ and a set of U instantiations of the text attributes understood by the system $\mathbf{T}_e = \{\mathbf{T}_{e,1}, \ldots, \mathbf{T}_{e,U}\}$. For a system with K such attributes, $\mathbf{T}_{e,i}$ would be the binary vector of length K obtained by asking one of U test subjects to categorize the i^{th} image (sequence) in the training set.

The likelihood of the test set would thus be given by

$$P(\mathbf{V}, \mathbf{T}, \mathbf{S}|\mathbf{\Theta}, \pi, \mu, \mathbf{\Sigma}, \tau), \tag{13}$$

where $\mathbf{V} = \{\mathbf{V}_1, \ldots, \mathbf{V}_E\}$, $\mathbf{T} = \{\mathbf{T}_1, \ldots, \mathbf{T}_E\}$, $\mathbf{S} = \{\mathbf{S}_1, \ldots, \mathbf{S}_E\}$ is the set of indicator vectors assigning each of the examples to each of the M sources, $\mathbf{\Theta} = \{\theta_1, \ldots, \theta_M\}$ the set of Bernoulli parameter vectors associated with equation (8) for each of the sources, $\pi = \{\pi_1, \ldots, \pi_M\}$, $\mu = \{\mu_1, \ldots, \mu_M\}$, $\mathbf{\Sigma} = \{\mathbf{\Sigma}_1, \ldots, \mathbf{\Sigma}_M\}$ the sets of parameters of the mixture models for the image sources (equation (12)), and $\tau = \{P(S_1 = 1), \ldots, P(S_M = 1)\}$ the set of prior source probabilities. Usually, $E = M$ and $\mathbf{S}_e = \mathbf{e}_e$, i.e. each of the images (sequences) would be considered a sample from a different source, but it is equally possible to have multiple examples drawn from the same source.

Defining $\mathbf{\Phi} = \{\mathbf{\Theta}, \pi, \mu, \mathbf{\Sigma}, \tau\}$, using the independence relations in the model and assuming that the examples in the training set are independent

$$\begin{aligned} P(\mathbf{V}, \mathbf{T}, \mathbf{S}|\mathbf{\Phi}) &= P(\mathbf{V}, \mathbf{T}|\mathbf{S}, \mathbf{\Theta}, \pi, \mu, \mathbf{\Sigma})P(\mathbf{S}|\tau) \\ &= P(\mathbf{V}|\mathbf{S}, \pi, \mu, \mathbf{\Sigma})P(\mathbf{T}|\mathbf{S}, \mathbf{\Theta})P(\mathbf{S}|\tau), \\ &= \prod_e P(\mathbf{V}_e|\mathbf{S}_e, \pi, \mu, \mathbf{\Sigma}) \prod_e P(\mathbf{T}_e|\mathbf{S}_e, \mathbf{\Theta}) \prod_e P(\mathbf{S}_e|\tau), \\ &= \prod_e \prod_b P(\mathbf{V}_{e,b}|\mathbf{S}_e, \pi, \mu, \mathbf{\Sigma}) \prod_e \prod_u P(\mathbf{T}_{e,u}|\mathbf{S}_e, \mathbf{\Theta}) \prod_e P(\mathbf{S}_e|\tau), \end{aligned}$$

where the second product in the first term is over all image blocks in example $\mathbf{V}_e$, and the second product in the second term is over all U subjects. The maximum likelihood estimates of the model parameters are then

$$\mathbf{\Phi}^* = \arg\max_{\mathbf{\Phi}}\{\log P(\mathbf{V}, \mathbf{T}, \mathbf{S}|\mathbf{\Phi})\},$$

and can be obtained by decomposing the maximization into the following sub-problems,

$$\{\pi^*, \mu^*, \mathbf{\Sigma}^*\} = \arg\max_{\{\pi, \mu, \mathbf{\Sigma}\}} \sum_e \sum_b \log P(\mathbf{V}_{e,b}|\mathbf{S}_e, \pi, \mu, \mathbf{\Sigma})$$

$$
\begin{aligned}
\mathbf{\Theta}^* &= \arg\max_{\mathbf{\Theta}} \sum_e \sum_u \log P(\mathbf{T}_{e,u}|\mathbf{S}_e, \mathbf{\Theta}) \\
\tau^* &= \arg\max_{\tau} \sum_e \log P(\mathbf{S}_e|\tau).
\end{aligned}
$$

Using

$$
\begin{aligned}
P(\mathbf{V}_{e,b}|\mathbf{S}_e, \pi, \mu, \mathbf{\Sigma}) &= \prod_i [P(\mathbf{V}_{e,b}|\mathbf{S}_e = \mathbf{e}_{\mathbf{i}}, \pi_i, \mu_i, \mathbf{\Sigma}_i)]^{\delta(\mathbf{S}_e - \mathbf{e}_i)} \\
P(\mathbf{T}_{e,u}|\mathbf{S}_e, \mathbf{\Theta}) &= \prod_i [P(\mathbf{T}_{e,u}|\mathbf{S}_e = \mathbf{e}_i, \mathbf{\Theta}_i)]^{\delta(\mathbf{S}_e - \mathbf{e}_i)},
\end{aligned}
$$

the optimal parameters for the characterization of the i^{th} source, S_i, become

$$
\begin{aligned}
\{\pi_i^*, \mu_i^*, \mathbf{\Sigma}_i^*\} &= \arg \max_{\{\pi_i, \mu_i, \mathbf{\Sigma}_i\}} \sum_{e|\mathbf{S}_e = \mathbf{e}_i} \sum_b \log P(\mathbf{V}_{e,b}|\mathbf{S}_e = \mathbf{e}_i, \pi_i, \mu_i, \mathbf{\Sigma}_i) \qquad (14) \\
\mathbf{\Theta}_i^* &= \arg\max_{\mathbf{\Theta}_i} \sum_{e|\mathbf{S}_e = \mathbf{e}_i} \sum_u \log P(\mathbf{T}_{e,u}|\mathbf{S}_e = \mathbf{e}_i, \mathbf{\Theta}_i) \qquad (15) \\
\tau_i^* &= \arg\max_{\tau_i} \sum_e \log P(\mathbf{S}_e = \mathbf{e}_i|\tau_i). \qquad (16)
\end{aligned}
$$

The first sub-problem is simply the density estimation discussed in section 4 and can be solved by EM. Using equations (4) and (8), the second sub-problem becomes

$$
\begin{aligned}
\mathbf{\Theta}_i^* &= \arg\max_{\mathbf{\Theta}_i} \sum_e \sum_u \sum_k (1 - \delta(T_{eu}^{(k)})) \log \theta_{ik} + \sum_e \sum_u \sum_k \delta(T_{eu}^{(k)}) \log(1 - \theta_{ik}) \\
&= \sum_k \log \theta_{ik} \sum_e \sum_u (1 - \delta(T_{eu}^{(k)})) + \sum_k \log(1 - \theta_{ik}) \sum_e \sum_u \delta(T_{eu}^{(k)}),
\end{aligned}
$$

where the summation over e is restricted to the set $\{e|\mathbf{S}_e = \mathbf{e}_i\}$. Taking derivatives with respect to θ_{ik} and setting them to zero, we obtain

$$
\theta_{ik}^* = \frac{\sum_{e|\mathbf{S}_e = \mathbf{e}_i} \sum_u (1 - \delta(T_{eu}^{(k)}))}{\sum_{e|\mathbf{S}_e = \mathbf{e}_i} \sum_u 1}, \qquad (17)
$$

i.e. the optimal θ is simply the ratio between the number of times the corresponding attribute was instantiated with *yes* and the total number of instantiations it received. Similarly

$$
\tau_i^* = \frac{1}{E} \sum_e \delta(\mathbf{S}_e - \mathbf{e}_i), \qquad (18)
$$

i.e. the optimal estimate for the prior probability of source i is simply the ratio between the number of examples from that source and the total number of examples.

These equations have important practical implications. First, because the parameters associated with the different attributes are learned independently, learning is efficient and the model can be easily updated. Consider for example an HTML page. The fact that the keywords need to be recomputed whenever the text is modified does not imply that the probabilistic description of the images also has to be re-estimated. In fact, the keywords do not even need to be defined in the page itself by the content provider, but can be defined by some other entity located elsewhere on the network.

For maximum efficiency, the provider simply needs to include the probabilistic image description in the encoded bitstreams available from the page, a task which is automatically carried out by the encoder during the process of compressing those images. The Bayesian framework thus allows efficient indexing over distributed networks such as the Internet.

References

[1] A. Dempster, N. Laird, and D. Rubin. Maximum-likelihood from Incomplete Data via the EM Algorithm. *J. of the Royal Statistical Society*, B-39, 1977.

[2] A. Gelman, J. Carlin, H. Stern, and D. Rubin. *Bayesian Data Analysis.* Chapman and Hall, 1995.

[3] Y. Gong, H. Zhang, H. Chuan, and M. Sakauchi. An Image Database System with Content Capturing and Fast Image Indexing Abilities. In *Proc. Int. Conf. on Multimedia Computing and Systems*, May 1994, Boston, USA.

[4] Y. Linde, A. Buzo, and R. Gray. An Algorithm for Vector Quantizer Design. *IEEE Trans. on Communications*, Vol. 28, January 1980.

[5] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Pektovic, P. Yanker, C. Faloutsos, and G. Taubin. The QBIC project: Querying images by content using color, texture, and shape. In *Storage and Retrieval for Image and Video Databases*, pages 173–181, SPIE, Feb. 1993, San Jose, CA.

[6] J. Pearl, editor. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann, 1988.

[7] A. Pentland, R. Picard, and S. Sclaroff. Photobook: Content-based Manipulation of Image Databases. In *SPIE Storage and Retrieval for Image and Video Databases II*, number 2185, Feb. 1994, San Jose, CA.

[8] J. Smith and S. Chang. Visually Searching the Web for Content. *IEEE Multimedia*, 4(3):12–20, July-September 1997.

[9] N. Vasconcelos and A. Lippman. Content-based Pre-Indexed Video. In *Proc. Int. Conf. Image Processing*, Santa Barbara, California, 1997.

[10] N. Vasconcelos and A. Lippman. Library-based Coding: a Representation for Efficient Video Compression and Retrieval. In *Proc. Data Compression Conference*, Snowbird, Utah, 1997.