6-2002

Using a Probabilistic Source Model for Comparing Images

Rong Jin  
*Carnegie Mellon University*

Alexander Hauptmann  
*Carnegie Mellon University*

Follow this and additional works at: [http://repository.cmu.edu/compsci](http://repository.cmu.edu/compsci)
Using a Probabilistic Source Model for Comparing Images

Rong Jin and Alexander G. Hauptmann

School of Computer Science, Carnegie Mellon University

ABSTRACT

In this paper, we propose a probabilistic model for image retrieval. To obtain the similarity between the query image \( I_Q \) and any image \( I' \) in the collection, this model computes the probability of generating the image \( I' \) given the observation of the query image \( I_Q \). We compared our probabilistic model for image retrieval with a color histogram based image retrieval method and the IBM QBIC image search engine. The evaluation used the 11-hour video retrieval collection (80000 extracted images) and associated queries from the 2001 TREC-10 information retrieval evaluations. The experimental results show that the probabilistic model dramatically outperforms the color histogram based image retrieval method and the IBM QBIC image search engine by 40%.

1. INTRODUCTION

Content-Based image retrieval has been studied for many years [1]. The task requires an image search engine to find the set of images from a given image collection that is similar to the given query image. Traditional methods for content-based image retrieval are based on a vector model [2,3]. These methods represent an image as a set of features and the difference between two images is measured through a (usually Euclidean) distance between their feature vectors. Such vector-based image retrieval methods contain several problems:

- **Independence Assumption.** By using Euclidean-type distances as a similarity measure, the approach implicitly assumes that all features are independent. However, this assumption is quite wrong in many cases. For example, it is well known that the red, green and blue color components of an image are not independent of each other at all.

- **Scale Problem.** To make the value of one feature comparable with the value of another feature, we need to scale the feature value appropriately to avoid the dominance of one feature in the Euclidean distance measurement. One example would be a query image with a large portion of blue sky. Since the blue sky dominates the query picture all top matching images would just be those comprised of mostly blue sky whereas the difference between the real content of the images would be ignored in similarity measure. However, there is no fundamental principle to automatically determine the appropriate scale for each feature in the vector model.

In this paper, we propose a probabilistic model for computing the similarity between images:

Let’s assume that images are generated through some stochastic process. Given the observation of an image \( I \), we can find the underlying probabilistic model \( M \) that generated this image. The optimal probabilistic model for an image \( I \) should maximize the generation probability \( P(I|M) \). By assuming that if two images are similar, their underlying generation model should also be similar, we can compute the similarity of image \( I_1 \) to image \( I_2 \) as \( P(I_1|M_2) \), i.e. the probability of generating image \( I_1 \) from the statistical model \( M_2 \).

One advantage of a probabilistic model for image retrieval is that it avoids the independence assumption for features. By introducing an explicit dependency relationship between some features, our probabilistic model takes the correlation between these features into account.

[Miss Related Work]

2. PROBABILISTIC MODEL FOR IMAGE RETRIEVAL

2.1. Probabilistic Model for Describing an Image

Let’s assume any image is generated by the following simple probabilistic model: The color \( C \) of each pixel around location \( L \) is generated by the joint probability distribution \( P(C,L) \). The key issue of the probabilistic model is to determine the probabilities for applying color \( C \) to location \( L \), or \( P(C,L) \). The problem with directly using the joint probability \( P(C,L) \) is that the number of different joint probabilities \( P(C,L) \) is very large. In the following, we will discuss two techniques to decrease the number of parameters.

2.1.1. Decoupling the Correlation between Different Color components and Location
One reason that number of different probabilities $P(C,L)$ is large is because the color $C$ is not just one value, but has three components. Since we use the Munsell color space for describing color [5], a color $C$ is represented by three components, hue ($h$), value ($v$) and chroma ($c$). Therefore, the joint probability $P(C,L)$ can be expressed as $P(h,v,c,L)$. To decrease the number of parameters, we need to decouple the color components from location.

Using the chain rule, we can rewrite $P(C,L)$ as

$$P(C,L) = P(L)P(C|L) = P(h,v,c|L)P(L) = P(h|L)P(v|h,L)P(c|v,h,L)P(L)$$ (1)

In the above equation, the joint probability $P(C,L)$ is decomposed into four parts: $P(L)$, $P(h|L)$, $P(v|h,L)$ and $P(c|v,h,L)$. We treat the location probability $P(L)$ as a uniform distribution, i.e. $P(L) = P_0$ where $P_0$ is a constant. Since the value ($v$) of a color represents the lightness of that color, we assume that it only depends on the location $L$ and has little to do with its hue. Therefore, the conditional probability $P(v|h,L)$ can be approximated as $P(v|L)$. The chroma ($c$) of a color represents the degree of departure from the neutral color. Thus, we can assume that it only depends on the hue of that color and is independent of location ($L$) and the value ($v$) of that color. This approximates probability $P(c|v,L)$ as $P(c|h)$. With these two simplifications, Equation (1) can be simplified to

$$P(C,L) = P(h,v,c,L) \approx P_0 P(h|L)P(v|L)P(c|h)$$ (2)

In Equation (2), the simplified probabilistic model contains three probabilities, i.e. $P(h|L)$, $P(v|L)$ and $P(c|h)$.  

2.1.2. Quantization of Location Space and Color Space

The second way to decrease the number of probabilities is to decrease the number different values for location $L$, color component hue ($h$), color component value ($v$) and color component chroma ($c$). One way of quantizing a continuous range is to divide it into bins of equal size.

This simple idea is fine for quantizing the location space but it is a bad idea for color space. The problem is caused by the skewed distribution of the values and chroma for a color. To show this more clearly, we did a simple experiment: In the collection used for testing our probabilistic model, there are approximately 80,000 images. For every pixel in image, we decompose its color into the three the Munsell components and in each components we add the total number of pixels that have the same value. Figure 1 shows the relationship between the number of pixels and the values of chroma. As seen from the figure, the distribution for chroma values is very narrow and far from uniform. Therefore, dividing the whole range for the chroma component into bins of equal length would result in most pixels having the same chroma value. A better solution is to divide the whole value range for each color component in such a way that each bin contains an equal number of pixels. By quantizing the color space into this way, we avoid the case where most pixels have the same value for a color component.

![Figure 1: Plot of the number of pixels versus different values of the chroma color component for 80000 images.](image1.png)

To find the optimal probabilistic model $M$ for an image $I$, we must maximize the probability $P(I|M)$, i.e. the probability of applying the model $M$ to generate image $I$. With Equation (2) and assuming that each pixel is generated independently, $P(I|M)$ can be expanded as

$$P(I|M) = \prod_L P(h(v,c,L)|I)P(v|L)P(c|h|L)P(L)$$ (5)

where $n(h,v,c,L)$ stands for the number of pixels in location $L$ with hue $h$, value $v$ and chroma $c$.

To maximize the probability $P(I|M)$, we obtain as parameters for the optimal model

$$P(h|L) = \sum_{k} n(h,k,v,c,L) \quad P(v|L) = \sum_{k} n(h,k,v,c,L) \quad P(c|h|L) = \sum_{k} n(h,k,v,c,L)$$ (6)

2.2. Probability Smoothing

If we directly use Equation (6) to estimate the probability $P(h|L)$, $P(v|L)$ and $P(c|h)$, many probabilities will be zero because in some locations, some color didn’t appear at all. Zero probabilities cause serious problems in computing the image similarity described below because of the
To compute the similarity of an image, we define the similarity between images as:

\[ \text{Sim}(I, I_0) = \log(P(I | M_{I_0})) = \log(\sum_{h,v,c,L} P_O(h,v,c,L)^{n(h,v,c,L)}) \]

where we use the logarithm of probability \( P(I|M) \) to avoid problems representing very small numbers.

To avoid disparities for different sizes of images, we normalize the similarity function by dividing it by the total number of pixels in image \( I \). Then, Equation (9) can be simplified as:

\[ \text{Sim}(I, I_0) = R_h + R_l \sum_{v,c} P(h | L) \log(P(h | L)) + R_l \sum_{c} P(c | h) \log(P(c | h)) \]

where probability \( P(\cdot) \) and \( P(\cdot | \cdot) \) stand for the estimated parameters for image \( I \) and \( I_0 \). Equation (9') was used in our experiment to find images similar to the query images. Note that the similarity function is asymmetric for the two images \( I \) and \( I_0 \), which means that even if image \( I \) is similar to image \( I_0 \), image \( I_0 \) may not be similar to image \( I \).

### 3. EXPERIMENT

To evaluate the effectiveness of our probabilistic image retrieval model, we compared our image retrieval model against two other vector-based image retrieval algorithms using video retrieval task.

#### 3.1. The Video Collection Used for Testing

The video collection used in our experiment originated from the video retrieval evaluation track in TREC10 [6].

It contains about 11 hours of video with approximate 80,000 I-frame images. In the video retrieval task of TREC10, there are 32 “known-item” queries, where a complete set of target answers to the queries had been manually marked. Each video query is consisted of a video sample and a text description of the topic. Systems used the images within the query video example to retrieve up to 100 video shots believed to be similar to the query video sample. By comparing against the manually marked relevant video shots, we can compute the retrieval accuracy of our video retrieval system.

#### 3.2. Other Image Retrieval Algorithms for Comparison

In this experiment, we compared our probabilistic image retrieval model against two other vector-based image retrieval algorithms, namely the well-known QBIC image search engine [7] and a Munsell-color histogram based image retrieval algorithm [8]. Both of these two algorithms represent an image as a vector of features and compute the similarity between images based on the Euclidean distance between their representation vectors.

#### 3.3. Applying a Still Image Retrieval Algorithm to the Video Retrieval Task

To accomplish the video retrieval task using still image retrieval methods, we need to compute the similarity between video shots by using the similarity between images. Let \( V_Q \) be a query video example and be represented as a set of I-frame images, i.e. \( V_Q = \{V_Q^1, V_Q^2, ..., V_Q^n\} \). Let \( V_S \) be a video shot from the collection and be represented as another set of I-frame images, i.e. \( V_S = \{V_S^1, V_S^2, ..., V_S^n\} \). The similarity of video shot \( V_S \) with respect to query video example \( V_Q \) is defined as

\[ \text{Sim}(V_S, V_Q) = \arg \max_{i \in [1..n]} \left( \sum_{j=1}^{m} \text{Sim}(V_S^j, V_Q^j) \right) \]

where \( \text{Sim}(V_S^j, V_Q^j) \) is the similarity of image \( V_S^j \) with respect to image \( V_Q^j \).

#### 3.4. Evaluation

There are two aspects involved in the evaluation:

- **Recall.** A good retrieval system should retrieve as many relevant items as possible.
- **Precision.** A good retrieval system should only retrieve relevant items.

Many evaluation metrics have been used in information retrieval [9] to balance these two aspects. We adopt the average reciprocal rank (ARR) [10] as our evaluation metric, which is defined as follows:
For a given query, there are a total of \( N_r \) items in the collection that are relevant to this query. Assume that the system only retrieves \( k \) relevant items and they are ranked as \( r_1, r_2, \ldots, r_k \). Then, the average reciprocal rank is computed as

\[
ARR = \frac{\sum_{i=1}^{k} 1/r_i}{N_r}
\]  

(11)

As shown in Equation (11), there are two interesting aspects of the metric: first, it rewards the systems that put the relevant items near the top of the retrieval list and punish those that add relevant items near the bottom of the list. Secondly, the score is divided by the total number of relevant items for a given query. Since queries with more answer items are much easier than those with only a few answer items, this factor will balance the difficulty of queries and avoid the predominance of easy queries.

3.5. Results and Discussion

<table>
<thead>
<tr>
<th></th>
<th>QBIC (Histogram)</th>
<th>QBIC (Draw)</th>
<th>Color Histogram</th>
<th>Probabilistic Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARR</td>
<td>6.6%</td>
<td>10.1%</td>
<td>8.6%</td>
<td>14%</td>
</tr>
</tbody>
</table>

Table 1: Average reciprocal rank of image retrieval results obtained from two modes of the QBIC image search engine, from a color histogram based image retrieval method and from our probabilistic model.

Table 1 lists the average reciprocal rank (ARR) score for probabilistic model for image retrieval, QBIC image search engine and color histogram based image retrieval method.

The probabilistic model outperforms the other image retrieval methods with an average reciprocal rank as 14%. The ‘Draw’ mode of the QBIC engine is ranked second with an average reciprocal rank equal to 10.1% and the color histogram search is ranked as third with an average reciprocal rank of 8.6%.

4. DISCUSSIONS AND CONCLUSIONS

In this paper, we proposed a probabilistic model for image retrieval. Our experiment clearly shows the advantage of the probabilistic model compared with two other vector-based image retrieval systems in terms of average reciprocal rank. In addition to providing a radically new approach to image comparisons based generative models, the benefit of the probabilistic model for image retrieval is in breaking the independence assumption. In our probabilistic model for image retrieval, the three color components are not independent. A correlation between them is captured in our probabilistic model through the probability \( P(c|h) \). Finally, we also introduce a better partitioning of color space. Unlike a uniform partitioning of color space, we use the criteria that each bin should have an equal number of pixels. This effectively avoids the problem caused by skewed distributions.

11. REFERENCES


