Textural Features for Image Database Retrieval

Selim Aksoy and Robert M. Haralick
Intelligent Systems Laboratory
Department of Electrical Engineering
University of Washington
Seattle, WA 98195-2500
{aksoy,haralick}@@isl.ee.washington.edu

Abstract

This paper presents two feature extraction methods and two decision methods to retrieve images having some section in them that is like the user input image. The features used are variances of gray level co-occurrences and line-angle-ratio statistics constituted by a 2-D histogram of angles between two intersecting lines and ratio of mean gray levels inside and outside the regions spanned by those angles.

The decision method involves associating with any pair of images either the class "relevant" or "irrelevant". A Gaussian classifier and nearest neighbor classifier are used. A protocol that translates a frame throughout every image to automatically define for any pair of images whether they are in the relevance class or the irrelevance class is discussed. Experiments on a database of 300 gray scale images with 9,600 groundtruth image pairs showed that the classifier assigned 80% of the image pairs we were sure were relevant, to the relevance class correctly. The actual retrieval accuracy is greater than this lower bound of 80%.

1. Introduction

In recent years image database retrieval has received significant attention due to the advances in computation power, storage devices, scanning, networking, and the World Wide Web. The image retrieval scenario addressed here begins with a query expressed by an image. The user inputs an image or a section of an image and desires to retrieve images from the database having some section in them that is like the user input image.

Texture has been one of the most important characteristics which have been used to classify and recognize objects and scenes. Also, many researchers [5, 8, 3, 7] used texture in finding similarities between images in a database. In this paper, we discuss two textural feature extraction methods to represent images for content-based retrieval. In the first one, texture is defined as being specified by the statistical distribution of the spatial dependencies of gray level properties. Variances of gray level co-occurrence matrices are used to extract this information. This is consistent with [6], where Haralick *et al.* used co-occurrence matrices to classify sandstone photomicrographs, panchromatic aerial photographs, and ERTS multispectral satellite images. Comparative studies [10, 4] showed that gray level spatial dependencies are more powerful than many other methods.

The second method uses spatial relationships between lines as well as the properties of their surroundings and is motivated by the fact that line content of an image can be used to represent texture of the image [9]. An easy to compute texture histogram method with the only assumption that images have some line content is introduced. Also a protocol to automatically construct groundtruth image pairs to evaluate the performance of the algorithm is discussed.

The paper is organized as follows. First, textural features are discussed in Section 2. Then, decision methods for similarity measurement are described in Section 3. Experiments and results are presented in Sections 4 and 5 respectively. Finally, conclusions are discussed in Section 6.

2. Feature extraction

2.1. Variances of gray level spatial dependencies

We define texture as being specified by the statistical distribution of the spatial relationships of gray level properties. Coarse textures are ones for which the distribution changes slightly with distance, whereas for fine textures the distribution changes rapidly with distance. This information can be summarized in gray level co-occurrence matrices that are

matrices of relative frequencies $P(i, j; d, \theta)$ with which two neighboring pixels separated by distance d at orientation θ occur in the image, one with gray level i and the other with gray level j. Resulting matrices are symmetric and can be normalized by dividing each entry in a matrix by the number of neighboring pixels used in computing that matrix.

In order to use the information contained in the gray level co-occurrence matrices Haralick [6] defined 14 statistical measures. Since many distances and orientations result in a very large number of values, computation of co-occurrence matrices and extraction of many features from them become infeasible for an image retrieval application which requires fast computation. We decided to use only the variance

$$v(d, \theta) = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} (i-j)^2 P(i, j; d, \theta) \qquad (1)$$

which is a difference moment of P that measures the contrast in the image. Rosenfeld [9] called this feature the moment of inertia. It will have a large value for images which have a large amount of local variation in gray levels and a smaller value for images with uniform gray level distributions. We compute this feature for five distances, and for $0°$, $45°$, $90°$, and $135°$ orientations to constitute a 20-dimensional feature vector. Details of this work can be found in [1].

2.2. Line-angle-ratio statistics

Experiments on various types of images showed us that one of the strongest spatial features of an image is the relationship between its line segments. Therefore, an image can be roughly represented by the lines extracted from it.

Before feature extraction, each image is processed off-line by an edge detector, edge linker, line selection operator and line grouping operator to detect line pairs. The goal of the line selection operator is to perform hypothesis tests to eliminate lines that do not have significant difference between gray level distributions on both sides, and the goal of the line grouping operator is to find intersecting and/or near-intersecting lines.

The features for each pair of intersecting and near-intersecting line segments consist of the angle between two lines and the ratio of mean gray level inside the region spanned by that angle to the mean gray level outside that region. An example for this region convention is given in Figure 1.

The features that are extracted from the image form a two-dimensional space of angles and corresponding ratios. This feature space is partitioned into a fixed set of Q non-uniformly spaced cells. The feature vector is then the Q-dimensional vector which has for its q'th component the number of angle-ratio pairs that fall into that q'th cell. This

(a) Pairs of intersecting lines. (b) Regions used for mean calculation.
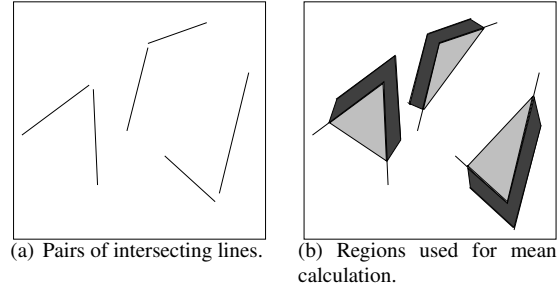
Figure 1. Examples of region convention for mean calculation. Light and dark shaded regions show the in and out regions respectively.

forms the texture histogram. Details of this work can be found in [2].

Since our goal is to find a section in the database which is relevant to the input query, before retrieval, each image in the database is divided into overlapping sub-images using the protocol which will be discussed in Section 3.1. Then textural features are computed for each sub-image in the database.

3. Decision methods

Given an image, we have to decide which images in the database are relevant to it, and we have to retrieve the most relevant ones as the results of the query. In our experiments we use two different types of decision methods; a likelihood ratio approach which is a Gaussian classifier, and a nearest neighbor rule based approach.

3.1. Likelihood ratio

In the likelihood ratio approach, we define two classes, namely the relevance class $\mathcal{A}$ and the irrelevance class $\mathcal{B}$. Given feature vectors of a pair of images, if these images are similar, they should be assigned to the relevance class, if not, they should be assigned to the irrelevance class.

Determining the parameters: The protocol for constructing groundtruths to determine the parameters of the likelihood ratio classifier involves making up two different sets of sub-images for each image in the database. The first set of sub-images begins in row 0 column 0 and partitions each image into K×K sub-images. These sub-images are partitioned such that they overlap by half the area. Partial sub-images on the last group of columns and rows which cannot make up the K×K sub-images are ignored. The second set of sub-images are shifted versions of the ones in the first set. They begin in row K/4 and column K/4 and partition the image into K×K sub-images.

To construct the groundtruths, we record the relationships of shifted sub-images with non-shifted sub-images that were computed from the same image. Each shifted sub-image is strongly related to four non-shifted sub-images in which the overlap is 9/16 of the sub-image area. These pairs calculated for all shifted sub-images constitute the relevance class $\mathcal{A}$. We assume that, in an image, two sub-images that do not overlap are usually not relevant. From this assumption, for each shifted sub-image, four non-shifted sub-images that have no overlap with it are randomly selected. These pairs constitute the irrelevance class $\mathcal{B}$.

An example for the overlapping concept is given in Figure 2. Note that for any sub-image which is not shifted by (K/4,K/4), there is a sub-image which overlaps more than half the area. We will use this property to evaluate the performance of our algorithm in Section 4.
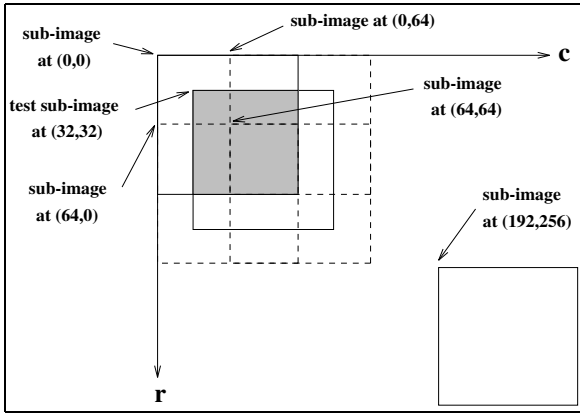

Figure 2. The shaded area shows the 9/16 overlapping between two 128×128 sub-images. Sub-images relevant to the shifted sub-image at (32,32) are at (0,0), (0,64), (64,0) and (64,64). An example for an irrelevant sub-image is the one at (192,256).

In order to estimate the distribution of the relevance class, we first compute the differences d, $d = x^{(n)} - y^{(m)}$, $(n,m) \in \mathcal{A}$, $x^{(n)}, y^{(m)} \in \Re^Q$ where Q is the dimension of the feature vector and $x^{(n)}$ and $y^{(m)}$ are the feature vectors of sub-images n and m respectively. We assume that these differences have a normal distribution, and compute the sample mean, $\mu_{\mathcal{A}}$, and sample covariance, $\Sigma_{\mathcal{A}}$. Similarly, we compute the differences d, $d = x^{(n)} - y^{(m)}$, $(n,m) \in \mathcal{B}$, then the sample mean, $\mu_{\mathcal{B}}$, and sample covariance, $\Sigma_{\mathcal{B}}$, for the irrelevance class.

Making the decision: Suppose for the moment that the user query is a K×K image. First, its feature vector x is determined. Then, the search goes through all the feature vectors y in the database. For each feature vector pair $(x,$

$y)$, the difference $d = x - y$ is computed. The probability that the input image with feature vector x, and a sub-image in the database with feature vector y are relevant is $P(\mathcal{A}|d) = P(d|\mathcal{A})P(\mathcal{A})/P(d)$. Similarly, the probability that they are irrelevant is $P(\mathcal{B}|d) = P(d|\mathcal{B})P(\mathcal{B})/P(d)$. Then, assuming the prior probabilities are equal, the likelihood ratio can be defined as

$$r(d) = \frac{P(d|\mathcal{A})}{P(d|\mathcal{B})} = \frac{\frac{1}{(2\pi)^{Q/2}|\Sigma_{\mathcal{A}}|^{1/2}}e^{-(d-\mu_{\mathcal{A}})'\Sigma_{\mathcal{A}}^{-1}(d-\mu_{\mathcal{A}})/2}}{\frac{1}{(2\pi)^{Q/2}|\Sigma_{\mathcal{B}}|^{1/2}}e^{-(d-\mu_{\mathcal{B}})'\Sigma_{\mathcal{B}}^{-1}(d-\mu_{\mathcal{B}})/2}}. \quad (2)$$

If this ratio is greater than 1, the sub-image is considered to be relevant to the input query image. After taking the natural logarithm and eliminating constants, a new measure r' can be defined as

$$r'(d) = (d-\mu_{\mathcal{B}})'\Sigma_{\mathcal{B}}^{-1}(d-\mu_{\mathcal{B}}) - (d-\mu_{\mathcal{A}})'\Sigma_{\mathcal{A}}^{-1}(d-\mu_{\mathcal{A}}). \quad (3)$$

To find the sub-images that are relevant to an input query image, sub-images are ranked by their likelihood values in (3). Among them, k sub-images having the highest r'-values are retrieved as the most relevant ones.

3.2. Nearest neighbor rule

In the nearest neighbor approach we assume each sub-image in the database is represented by its feature vector y in the Q-dimensional feature space. Given the feature vector x for the input query, goal is to find the y's which are the closest neighbors of x. Then, the k-nearest neighbors of x will be retrieved as the most relevant ones. For the distance metric ρ we use Euclidean distance, $\rho(x,y) = \|x - y\|$, or infinity norm, $\rho(x,y) = \max_{i=1,...,Q} |x_i - y_i|$.

4. Experimental set-up

To populate the database, we used 300 512×512 images from the Fort Hood Data, supplied for the RADIUS program. These images are visible light images of the Fort Hood area at Texas. To construct the database, we partitioned all images into 256×256 sub-images. Co-occurrence variances and line-angle-ratio statistics each resulted in 20-dimensional feature vectors. We also combined two feature vectors into a 40-dimensional vector by appending.

To test the classification effectiveness of the features, the likelihood ratio in (2) is used to classify each groundtruth pair. Since we know which non-shifted sub-images and shifted sub-images overlap, we also know which sub-image pairs should be assigned to class $\mathcal{A}$ and which to class $\mathcal{B}$. So, to test the algorithm, we check whether each pair that should be classified into class $\mathcal{A}$ or $\mathcal{B}$ is classified into class $\mathcal{A}$ or $\mathcal{B}$ correctly.

Two traditional measures for retrieval performance are precision and recall. Precision is the percentage of retrieved images that are correct and recall is the percentage of correct images that are retrieved. Note that computation of these measures requires prior goundtruthing of the database. Since our automatically generated groundtruths are not the ones required for precision and recall, these measures can not be used directly. We use modified versions of them to evaluate the performance of our algorithm. After manually grouping a smaller set of sub-images in our database, we will evaluate the performance using precision and recall too.

To test the retrieval performance, we use the following procedure. Given an input query image of size K×K, images are retrieved in descending order of likelihood ratio (3) or ascending order of distance for nearest neighbor rule. If the correct image is retrieved as one of the k best matches, it is considered a success. Average rank of the correct image is also computed. This can also be stated as a nearest neighbor classification problem where the relevance class is defined to be the best k matches and the irrelevance class is the rest of the images. For this experiment, we use the non-shifted sub-images to compute the best case performance and the shifted sub-images to compute the worst case performance. We call this the worst case because the shifted sub-images overlap a sub-image in the database by only half the area. All other possible sub-images have a sub-image in the database which they overlap by more than half the area. This experimental procedure is appropriate to our problem of retrieving images which have some section in them that is like the user input image.

5. Results

Classification effectiveness: In this experiment non-shifted and shifted sub-image databases consist of 2,700 and 1,200 256×256 sub-images respectively. There are 4 relevant and 4 irrelevant non-shifted sub-images for each of the 1,200 shifted sub-images, which make a total of 9,600 groundtruth sub-image pairs. As can be seen in Tables 1-3, approximately 80% of the groundtruth $\mathcal{A}$ pairs were assigned to $\mathcal{A}$ with an overall success around 63%.

We can say that most of the groundtruth $\mathcal{A}$ pairs were assigned to $\mathcal{A}$ but the groundtruth $\mathcal{B}$ pairs seem to be split between being assigned to $\mathcal{A}$ or $\mathcal{B}$ because although the assumption that overlapping sub-images are relevant almost always holds, we can not always guarantee that non-overlapping sub-images are irrelevant. Hence, some of the assignments which we count as incorrect are not in fact incorrect [2]. Thus the 80% relevance class identification rate is a lower bound.

Retrieval performance: Results for the retrieval performance experiments are presented in Tables 4-6. In all of these experiments a success means the correct image is re-

Table 1. Classification effectiveness test for Co-occurrence Variances.

	Assigned to $\mathcal{A}$	Assigned to $\mathcal{B}$	Success (%)
G.truth $\mathcal{A}$	3,828	972	79.75
G.truth $\mathcal{B}$	2,584	2,216	46.17
Overall	6,412	3,188	62.56

Table 2. Classification effectiveness test for Line-Angle-Ratio Statistics.

	Assigned to $\mathcal{A}$	Assigned to $\mathcal{B}$	Success (%)
G.truth $\mathcal{A}$	3810	877	81.29
G.truth $\mathcal{B}$	2699	1974	42.24
Overall	6509	2851	61.79

Table 3. Classification effectiveness test for combined features.

	Assigned to $\mathcal{A}$	Assigned to $\mathcal{B}$	Success (%)
G.truth $\mathcal{A}$	3,985	815	83.02
G.truth $\mathcal{B}$	2,589	2,211	46.06
Overall	6,574	3,026	64.54

trieved as one of the best 20 matches among a total of 2,700. In the best case performance tests with 2,700 queries, all features had a success between 93-100%. Nearest neighbor approach performed better than likelihood ratio. In the worst case performance tests with 1,200 queries, line-angle-ratio statistics had a success around 70% which was better than co-occurrence variances which had 57%. All decision methods performed almost equally. To illustrate the bounds found in these experiments, the database was queried with 500 randomly extracted 256×256 sections from images in the database.

In general, line-angle-ratio statistics performed better than co-occurrence variances. Experimenting on sub-image size showed that smaller sub-images give better results for co-occurrence features, which is a measure of micro texture, because texture tends to be more homogeneous. Larger sub-images favor line-angle-ratio features because they include more line information. Among the decision methods, Euclidean distance performed slightly better than the rest. Also, it retrieved the correct image at a higher rank which was around 5.5 on the average at the worst case. For the combined features case, it seems worse performing features dominated the distance computation and performance although Table 3 shows that adding more features increases classification effectiveness. The solution we propose to this problem is to sort individual components of the 40-dimensional feature vector according to their distances and to use only the n-best components to measure "similarity according to at least n features" instead of "similarity according to all features".

Table 4. Retrieval performance tests for Co-occurrence Variances.

	Original sub-images (2,700 tests)			Shifted sub-images (1,200 tests)			Random sub-images (500 tests)		
	Likelihood Ratio	Euclidean Distance	Infinity Norm	Likelihood Ratio	Euclidean Distance	Infinity Norm	Likelihood Ratio	Euclidean Distance	Infinity Norm
# successes	2,536	2,683	2,684	683	701	681	326	330	325
% success	93.93	99.37	99.41	56.92	58.42	56.75	65.20	66.00	65.00
avg. rank	4.0430	2.0078	2.0138	6.1830	5.5706	5.5727	5.7301	4.2576	4.2523

Table 5. Retrieval performance tests for Line-Angle-Ratio Statistics.

	Original sub-images (2,700 tests)			Shifted sub-images (1,200 tests)			Random sub-images (500 tests)		
	Likelihood Ratio	Euclidean Distance	Infinity Norm	Likelihood Ratio	Euclidean Distance	Infinity Norm	Likelihood Ratio	Euclidean Distance	Infinity Norm
# successes	2,665	2,696	2,696	829	866	667	398	422	359
% success	98.70	99.85	99.85	69.08	72.17	55.58	79.60	84.40	71.80
avg. rank	3.3989	1.0308	1.0308	6.2979	4.2517	5.1049	5.3166	2.8436	4.0334

Table 6. Retrieval performance tests for combined features.

	Original sub-images (2,700 tests)			Shifted sub-images (1,200 tests)			Random sub-images (500 tests)		
	Likelihood Ratio	Euclidean Distance	Infinity Norm	Likelihood Ratio	Euclidean Distance	Infinity Norm	Likelihood Ratio	Euclidean Distance	Infinity Norm
# successes	2,540	2,685	2,685	683	702	682	325	330	325
% success	94.07	99.44	99.44	56.92	58.50	56.83	65.00	66.00	65.00
avg. rank	4.0449	1.9966	1.9966	6.1654	5.5769	5.5865	5.7354	4.2909	4.2831

6. Conclusions

In this paper, we presented two feature extraction methods and two decision methods that allow a user to input an image or a section of an image and to retrieve all images from a database having some section in them that is like the user input image.

Feature extraction methods used are the variances of gray level spatial dependencies computed from co-occurrence matrices, and the line-angle-ratio statistics constituted by a 2-D histogram of the angles between two lines and the ratio of mean gray levels inside and outside the regions spanned by those angles.

A likelihood ratio is defined to measure the relevancy of two images one being the query image and one being a database image so that image pairs which had a high likelihood ratio were classified as relevant and the ones which had a lower likelihood ratio were classified as irrelevant. Also k-nearest neighbor rule is used to retrieve k images which have the closest feature vector to the feature vector of the query image in the high dimensional feature space.

To evaluate the performance, we discussed a protocol that translates a frame throughout every image to automatically construct groundtruth image pairs for the relevance and irrelevance classes. Results of the classification effectiveness tests showed that the algorithm assigned 80% of the sub-image pairs we were sure were relevant, to the relevance class correctly. A total of 13,200 queries showed that all of our methods retrieved correct images successfully as one of the best 20 matches, which is less than 1 percent of the total 2,700, in more than 94% and 57% of the experiments for the best and worst case analysis of the nearest neighbor classifier respectively. A method for performance characterization will be to plot classification success as a function of the number of images retrieved.

References

[1] S. Aksoy and R. M. Haralick. Content-based image database retrieval using variances of gray level spatial dependencies. In *Proc. of IAPR Intl. Workshop on Multimedia Information Analysis and Retrieval*, Aug. 1998.

[2] S. Aksoy, M. L. Schauf, and R. M. Haralick. Content-based image database retrieval based on line-angle-ratio statistics. Technical Report, Intelligent Systems Lab., University of Washington, Seattle, WA, Nov. 1997.

[3] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Region-based image querying. In *Proc. of IEEE Workshop on Content-Based Access of Image and Video Libraries*, 1997.

[4] R. W. Conners and C. A. Harlow. Some theoretical considerations concerning texture analysis of radiographic images. In *Proc. of IEEE Conference on Decision and Control*, pages 162–167, 1976.

[5] M. Flickner *et al.* The QBIC project: Querying images by content using color, texture and shape. In *SPIE Storage and Retrieval of Image and Video Databases*, 1993.

[6] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE Trans. on SMC*, SMC-3(6):610–621, Nov. 1973.

[7] W. Y. Ma and B. S. Manjunath. NETRA: A toolbox for navigating large image databases. In *Proc. of ICIP*, 1997.

[8] A. Pentland *et al.* Photobook: Content-based manipulation of image databases. In *SPIE Storage and Retrieval of Image and Video Databases II*, pages 34–47, Feb. 1994.

[9] A. Rosenfeld and E. B. Troy. Visual texture analysis. In *Conf. Record for Symposium on Feature Extraction and Selection in Pattern Recognition*, pages 115–124, Oct. 1970.

[10] J. S. Weszka, C. R. Dyer, and A. Rosenfeld. A comparative study of texture measures for terrain classification. *IEEE Trans. on SMC*, SMC-6(4):269–285, April 1976.