

USING TEXTURE IN IMAGE SIMILARITY AND RETRIEVAL

SELIM AKSOY AND ROBERT M. HARALICK

*Intelligent Systems Laboratory, Department of Electrical Engineering,
University of Washington, Seattle, WA 98195-2500, U.S.A.*

E-mail: {aksoy,haralick}@isl.ee.washington.edu

Texture has been one of the most popular representations in image retrieval. Our image database retrieval system uses two sets of textural features, first one being the line-angle-ratio statistics which is a macro texture measure that uses a texture histogram computed from the spatial relationships of intersecting lines as well as the properties of their surroundings, second one being the variances of gray level spatial dependencies computed from co-occurrence matrices as micro texture measures. This paper also discusses a line selection algorithm to eliminate insignificant lines and statistical feature selection methods to select the best performing subset of features to adjust the parameters of the feature extraction algorithms. Average precision is used to evaluate the retrieval performance in comparative tests with three other texture analysis algorithms. Experiments on a database of approximately 10,000 images show that low-level textural features can help in grouping images into semantically meaningful categories and our method is fast and effective with an average precision of 0.73 when 12 images are retrieved.

1 Introduction

Texture has been one of the most important characteristics which have been used to classify and recognize objects and scenes. Haralick and Shapiro¹⁶ defined texture as the uniformity, density, coarseness, roughness, regularity, intensity and directionality of discrete tonal features and their spatial relationships. Haralick¹⁴ gave a review of two main approaches to characterize and measure texture: *statistical* approaches and *structural* approaches. A recent texture survey was done by Tuceryan and Jain,³⁰ where texture models were classified into statistical methods, geometrical methods, model-based methods and signal processing methods.

Image databases are becoming increasingly popular due to the large amount of images that are generated by various applications and the advances in computer technology. Initial work on content-based retrieval focused on using low-level features like color and texture. One of the first systems that used texture in finding similarities between images is the IBM's QBIC Project, where Flickner *et al.*¹² used features based on coarseness, contrast, and directionality that were proposed by Tamura *et al.*²⁹ In the MIT Photobook Project, Pentland *et al.*²⁷ used 2-D Wold-based decompositions²² as texture descriptions. In the Los Alamos National Lab.'s CAN-

DID Project, Kelly *et al.*¹⁹ used Laws' texture energy maps and a sum of weighted Gaussians to model the texture. Manjunath and Ma²⁴ used Gabor filter-based multi-resolution representations to extract texture information. Li and Castelli²⁰ used 21 different spatial features like gray level differences, co-occurrence matrices, moments, autocorrelation functions and fractals on remote sensing images. Smith²⁸ used energies of the quadrature mirror filter wavelet filter bank outputs at different resolutions as textural features. The main disadvantages of these approaches are that they are either ineffective when a large database of images with non-homogeneous textures are used, or require a lot of computation.

More recent approaches by Carson *et al.*⁷ and Ma and Manjunath²³ developed region-based query systems which involve image segmentation based on color and texture but the region segmentation algorithms are still too slow to be used in an image retrieval application. Even though there have been further approaches that use post-processing methods like relevance feedback to improve retrieval accuracy, these methods still depend heavily on the feature representations and therefore it is crucial to improve the low-level feature extraction algorithms to improve overall retrieval.

In this paper we attempt to improve retrieval efficiency using easy-to-compute low-level features that combine macro and micro aspects of the texture in the image. The first set of texture features are the line-angle-ratio statistics. These are macro texture measures that use spatial relationships between lines as well as the properties of their surroundings. A statistical line selection algorithm to eliminate insignificant lines is part of the method. The second set of texture features consist of the variances of gray level spatial dependencies. These are micro texture measures that use second-order (co-occurrence) statistics of gray levels of pixels in particular spatial relationships. Both sets of features are integrated for a multi-scale texture analysis which is crucial for a compact representation, especially for large databases containing different types of complex images.

We use a two-class pattern classification approach to find statistical measures of how well some of the features perform better than others to avoid having less significant or even redundant features that increase computation but contribute very little in the decision process. Retrieval performance is evaluated using average precision computed for the manually groundtruthed data set.

The rest of the paper is organized as follows. First, the textural features are presented in Sections 2, 3 and 4. Then, feature selection methods are described in Section 5. Experiments and results are discussed in Section 6. Finally, conclusions are given in Section 7.

2 Line-Angle-Ratio Statistics

Experiments on various types of images showed us that one of the strongest spatial features of an image is its line segments. Edge and line information have been extensively used in both very early and recent approaches to texture. Our algorithm is composed of two stages; pre-processing and texture histogram generation.

2.1 Pre-processing

Each image is processed by Canny's edge detector, ⁶ Etemadi's edge linker, ¹¹ a line selection operator and a line grouping operator to detect line pairs to associate with it a set of feature records. Edge detection followed by line detection often results in many false alarms. It is especially hard to select proper parameters for these operators if one does not have groundtruth information as training data. After line detection, we use hypothesis testing to eliminate lines that do not have significant difference between the gray level distributions in the regions on their right and left.

The algorithm we developed for line selection is given as follows. Let the set of N gray levels x_1, x_2, \dots, x_N be considered as iid $N(\mu_x, \sigma_x^2)$ samples from the region to the right of a line and the set of M gray levels y_1, y_2, \dots, y_M be considered as iid $N(\mu_y, \sigma_y^2)$ samples from the region to the left of that line. Define

$$\bar{x} = \frac{1}{N} \sum_{n=1}^N x_n \quad \sim N\left(\mu_x, \frac{\sigma_x^2}{N}\right)$$

and

$$\bar{y} = \frac{1}{M} \sum_{m=1}^M y_m \quad \sim N\left(\mu_y, \frac{\sigma_y^2}{M}\right).$$

Then the random variable $z = \bar{x} - \bar{y}$ has a distribution

$$N(\mu_z, \sigma_z^2) = N\left(\mu_x - \mu_y, \frac{\sigma_x^2}{N} + \frac{\sigma_y^2}{M}\right).$$

Define the null hypothesis as $H_0 : \mu_x = \mu_y$ and $\sigma_x = \sigma_y$ which means both sets of gray levels come from the same distribution, and the alternative hypothesis as $H_1 : \mu_x \neq \mu_y$ and $\sigma_x \neq \sigma_y$. To form the test statistic, define two random variables A and B as

$$A = \left(\frac{z - \mu_z}{\sigma_z}\right)^2 \quad \sim \chi_1^2 \quad (1)$$

and

$$B = \frac{1}{N-1} \sum_{n=1}^N \left(\frac{x - \bar{x}}{\sigma_x} \right)^2 + \frac{1}{M-1} \sum_{m=1}^M \left(\frac{y - \bar{y}}{\sigma_y} \right)^2 \sim \chi_{N+M-2}^2. \quad (2)$$

Then, define the test statistic $F = \frac{A/1}{B/(N+M-2)}$ which, under the null hypothesis, becomes

$$F = \frac{z^2 \frac{(N+M-2)}{\left(\frac{1}{N} + \frac{1}{M}\right)}}{\frac{1}{(N-1)} \sum_{n=1}^N (x - \bar{x})^2 + \frac{1}{(M-1)} \sum_{m=1}^M (y - \bar{y})^2} \sim F_{1, N+M-2}. \quad (3)$$

Given a threshold for the F -value, if the null hypothesis H_0 is true, the line is rejected, if the alternative hypothesis H_1 is true, the line is accepted as a significant one.

After obtaining relatively significant lines, we use a line grouping operator to find intersecting and/or near-intersecting line pairs. We allow near-intersection instead of exact end-point intersection because of the perturbation due to noise.

Given two lines L_1 and L_2 with end-points $(P_1, P_2) = ([r_1 \ c_1]', [r_2 \ c_2]')$ and $(P_3, P_4) = ([r_3 \ c_3]', [r_4 \ c_4]')$ respectively, equations of them can be written as

$$L_1 : P = P_1 + \lambda_1(P_2 - P_1), \quad (4)$$

$$L_2 : P = P_3 + \lambda_2(P_4 - P_3) \quad (5)$$

where λ_1 and λ_2 are real constants between 0 and 1. The following conditions should be satisfied for intersection:

$$r_1 + \lambda_1(r_2 - r_1) = r_3 + \lambda_2(r_4 - r_3), \quad (6)$$

$$c_1 + \lambda_1(c_2 - c_1) = c_3 + \lambda_2(c_4 - c_3). \quad (7)$$

If $(r_4 - r_3)(c_2 - c_1) = (r_2 - r_1)(c_4 - c_3)$, lines L_1 and L_2 are parallel. If also $(r_2 - r_1)(c_3 - c_1) = (r_3 - r_1)(c_2 - c_1)$, end-points P_1, P_2, P_3, P_4 are co-linear. If neither of these cases are true, λ_1 and λ_2 can be derived from Eq. (6) and (7) as

$$\lambda_2 = \frac{(r_2 - r_1)(c_3 - c_1) - (r_3 - r_1)(c_2 - c_1)}{(r_4 - r_3)(c_2 - c_1) - (r_2 - r_1)(c_4 - c_3)} \quad (8)$$

and

$$\begin{aligned} \lambda_1 &= \frac{r_3 - r_1}{r_2 - r_1} + \lambda_2 \frac{r_4 - r_3}{r_2 - r_1} \text{ if } r_1 \neq r_2 \\ \text{or} \\ &= \frac{c_3 - c_1}{c_2 - c_1} + \lambda_2 \frac{c_4 - c_3}{c_2 - c_1} \text{ if } c_1 \neq c_2. \end{aligned} \quad (9)$$

We define *Tol* as the tolerance, in number of pixels, for the end points of the lines to intersect. We need to define this tolerance to allow near-intersection instead of exact end-point intersection. To determine the tolerances for λ_1 and λ_2 , two new tolerances τ_1 and τ_2 can be defined as

$$\tau_1 = \frac{Tol}{\|P_2P_1\|} \quad \text{and} \quad \tau_2 = \frac{Tol}{\|P_4P_3\|}. \quad (10)$$

If $\tau_1 \leq \lambda_1 \leq 1 - \tau_1$ and $\tau_2 \leq \lambda_2 \leq 1 - \tau_2$, two lines cross each other, if ($\tau_1 \leq \lambda_1 \leq 1 - \tau_1$ and ($|\lambda_2| < \tau_2$ or $|\lambda_2 - 1| < \tau_2$)) or ($\tau_2 \leq \lambda_2 \leq 1 - \tau_2$ and ($|\lambda_1| < \tau_1$ or $|\lambda_1 - 1| < \tau_1$)), two lines have a T-like intersection, and if ($|\lambda_1| < \tau_1$ or $|\lambda_1 - 1| < \tau_1$) and ($|\lambda_2| < \tau_2$ or $|\lambda_2 - 1| < \tau_2$), two lines intersect at the end-points within the given tolerance. Then, the intersection point $[r \ c]'$ can be found by substituting λ_1 into the Eq. (4) as

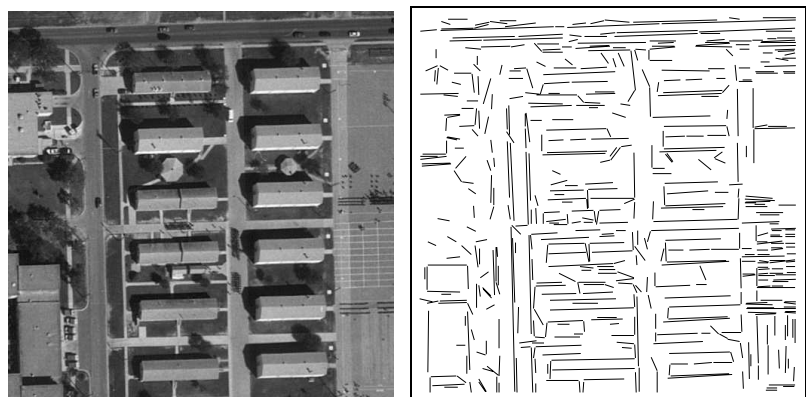
$$\begin{bmatrix} r \\ c \end{bmatrix} = \begin{bmatrix} r_1 \\ c_1 \end{bmatrix} + \lambda_1 \begin{bmatrix} r_2 - r_1 \\ c_2 - c_1 \end{bmatrix}. \quad (11)$$

Examples for the pre-processing steps are given in Figure 1.

2.2 Texture Histogram

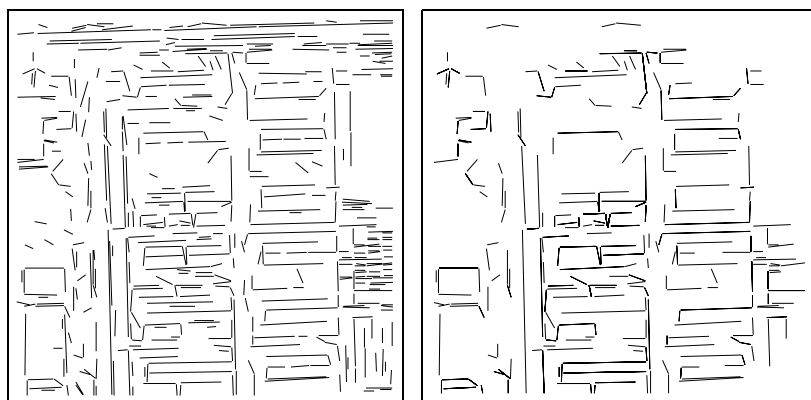
The features for each pair of intersecting line segments consist of the angle between two lines and the ratio of mean gray level inside the region spanned by that angle to the mean gray level outside that region. Angle values are in the range $[0^\circ, 180^\circ]$. An example for region convention is given in Figure 2. Since the possible range of ratio values is infinite, we restrict them to the range $[0, 1)$ by taking the reciprocal if the inner region is brighter than the outer region.

The final features form a two-dimensional space of angles and the corresponding ratios, which is then partitioned into a fixed set of Q cells. The feature vector for each image is designed to be the Q-dimensional vector which has for its q 'th component the number of angle-ratio pairs that fall into that q 'th cell. As can be seen in Figure 3(a), these features do not have a uniform



(a) Grayscale image.

(b) Extracted lines after line detection operator.



(c) Accepted lines after line selection operator.

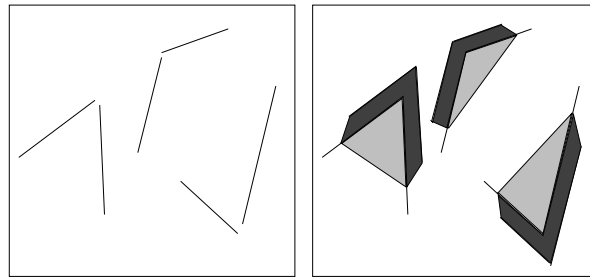
(d) Resulting lines after line grouping operator.

Figure 1. Line selection and grouping pre-processing steps.

distribution so we use vector quantization²¹ to form the Q -cell partition. Resulting partitions and their centroids for an example of 20 cells are given in Figure 3(b).

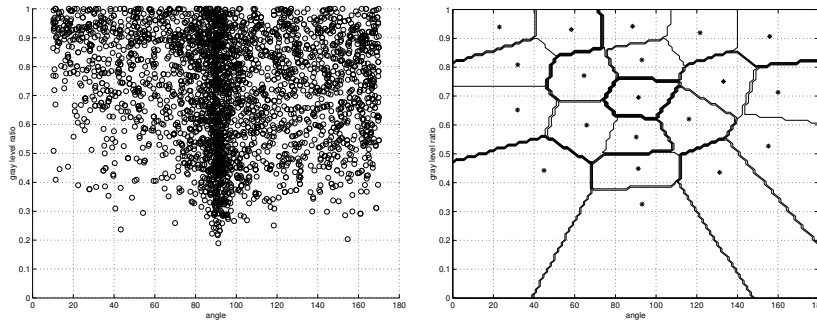
3 Variances of Gray Level Spatial Dependencies

Structural approaches to texture analysis use the idea that texture is composed of primitives with different properties appearing in particular spatial arrangements. On the other hand, statistical approaches try to model texture



(a) Pairs of intersecting lines. (b) Regions used for mean calculation.

Figure 2. Examples of region convention for mean ratio calculation. Light and dark shaded regions show the *in* and *out* regions respectively.



(a) Distribution of the training samples. (b) Resulting partitions for 20 quantization cells.

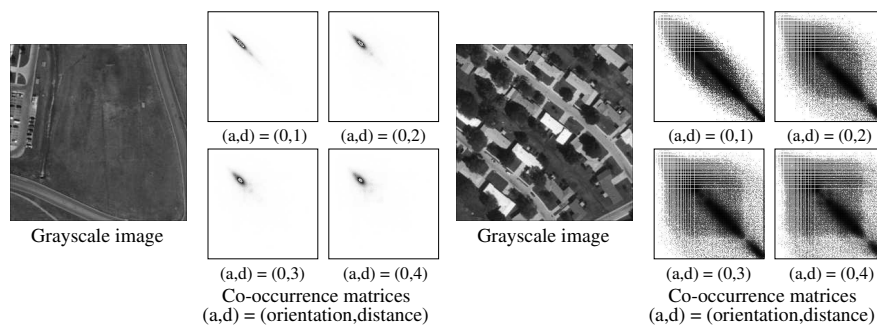
Figure 3. Line-Angle-Ratio feature space distribution and centroids of the resulting partitions.

using statistical distributions either in the spatial domain or in a transform domain. One way to combine these two approaches is to define texture as being specified by the statistical distribution of the properties of different textural primitives occurring at different spatial relationships.

This information can be summarized in two-dimensional co-occurrence matrices that are matrices of relative frequencies $P(i, j; d, \theta)$ with which two pixels separated by distance d at orientation θ occur in the image, one with gray level i and the other with gray level j . These 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.

The initial work on co-occurrence matrices¹⁵ and some comparative studies^{10,26} showed that gray level spatial dependency matrices were very successful in discriminating images with relatively homogeneous textures. *Weszka et*

al.³¹ discussed that if a texture is coarse and the distance d used to compute the co-occurrence matrix is small compared to the sizes of the texture elements, pairs of pixels at separation d should usually have similar gray levels. This means that high values in the matrix $P(i, j; d, \theta)$ should be concentrated on or near its main diagonal. Conversely, for a fine texture, if d is comparable to the texture element size, then the gray levels of points separated by d should often be quite different, so that values in $P(i, j; d, \theta)$ should be spread out relatively uniformly. Similarly, if a texture is directional, i.e. coarser in one direction than another, the degree of spread of the values about the main diagonal in $P(i, j; d, \theta)$ should vary with the orientation θ . Thus texture directionality can be analyzed by comparing spread measures of $P(i, j; d, \theta)$ for various orientations. Example co-occurrence matrices are given in Figure 4.



(a) Co-occurrence matrices for an image with a small amount of local spatial variations. (b) Co-occurrence matrices for an image with a large amount of local spatial variations.

Figure 4. Example co-occurrence matrices.

3.1 Pre-processing

Before computing co-occurrence matrices we use equal probability quantization¹⁵ to make the features invariant to distortions resulting in monotonic gray level transformations. We use 64 quantization levels (N_g) which performed the best among 16, 32 and 64 levels in terms of “total cost” that will be defined in Section 5. In the literature, usually a small number of levels were used because the images under consideration usually contained homogeneous textures, but our images are much more complex than those images and a small number of levels causes significant information loss.

3.2 Co-occurrence Variance

In order to use the information contained in the gray level co-occurrence matrices, Haralick *et al.*¹⁵ defined 14 statistical measures. Since many distances and orientations result in a large amount of computation, we decided to use only the variance³

$$v(d, \theta) = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} (i-j)^2 P(i, j; d, \theta) \quad (12)$$

which is a difference moment of P and measures the contrast in the image. It will have a large value for images which have a large amount of local spatial variation in gray levels and a smaller value for images with spatially uniform gray level distributions. Gotlieb and Kreyszig¹³ used heuristic selection methods to select the best subset of features for co-occurrence matrices and found out that the variance feature performed the best.

4 Multi-Scale Texture Analysis

Line-angle-ratio features capture the global spatial organization in an image by using relative orientations of lines extracted from it; therefore, they can be regarded as a macro-texture measure but are not effective if the image does not have any line content. On the other hand, co-occurrence variances capture local spatial variations of gray levels in the image; therefore, they are effective if the image is dominated by a fine, coarse, directional, or repetitive texture and can be regarded as a micro-texture measure. Another important difference is that line-angle-ratio features are invariant to rotation because they use relative orientations. On the contrary, co-occurrence variances are not rotation invariant because they are angularly dependent. If one also wants rotation invariance for these features, the feature vector can be modified by averaging the feature values for each distance over all orientations.¹⁵

In order to approximately equalize ranges of the features and make them have approximately the same effect in the computation of similarity, each component x in the feature vector is normalized as $y = F_x(x)$, where $F_x(\cdot)$ is the cumulative distribution function of that component. This makes y a random variable uniformly distributed in the $[0, 1]$ interval. Then, both feature vectors are appended to form the final vector. In the rest of the paper size of a feature vector will be denoted by Q .

5 Feature Selection

In a content-based retrieval system, features that are used to represent images should have close values for similar images and significantly different values for dissimilar ones. In many complex feature extraction algorithms, there are many parameters that, when varied, result in a large number of possible feature measurements. These high dimensional feature spaces may cause a problem of having less significant or even redundant features that contribute very little in the decision process.

Most of the times this feature selection process is done heuristically. Formal methods from the statistical pattern recognition literature include algorithms to form a new set of features from a set of available ones either by selecting a subset or by combining them into new features.^{9,32,25,17} Only a few researchers^{27,24,7} presented feature selection algorithms in their papers on database retrieval. Manjunath and Ma²⁴ used the total spectral energy to select among many possible Gabor filters and Carson *et al.*⁷ used the minimum description length principle to select the number of Gaussians that best model the feature space.

We use a two-class pattern classification approach to find statistical measures of how well some of the features perform better than others. In doing so, we define two classes, the relevance class \mathcal{A} and the irrelevance class \mathcal{B} , in order to classify image pairs as similar or dissimilar. Assume that we are given two sets of image pairs for the relevance and irrelevance classes respectively.^{4,2} Differences of feature vectors for each image pair are assumed to have a normal distribution and sample means $\mu_{\mathcal{A}}$ and $\mu_{\mathcal{B}}$ and sample covariance matrices $\Sigma_{\mathcal{A}}$ and $\Sigma_{\mathcal{B}}$ are estimated using the training data. According to our experiments, the line-angle-ratio feature differences follow double-exponential distributions and the co-occurrence feature differences follow normal distributions.⁵ Modeling the joint feature differences using a multivariate normal density worked better than using independently fitted double-exponentials or normals because of the covariance matrix that captures the correlation between features.

5.1 Classification Tests

Given a groundtruth image pair (n,m) with Q -dimensional feature vectors $x^{(n)}$ and $y^{(m)}$ respectively, first the difference $d = x^{(n)} - y^{(m)}$ is computed. From Bayes' law, the probability that these images are relevant is $P(\mathcal{A}|d) = P(d|\mathcal{A})P(\mathcal{A})/P(d)$ and that they are irrelevant is $P(\mathcal{B}|d) = P(d|\mathcal{B})P(\mathcal{B})/P(d)$.

The image pair is assigned to the relevance class if $P(\mathcal{A}|d) > P(\mathcal{B}|d)$, and

to the irrelevance class otherwise. This can be written as

$$\text{Assign to } \mathcal{A} \text{ if } \frac{P(\mathcal{A}|d)}{P(\mathcal{B}|d)} > 1. \quad (13)$$

Assuming that both classes are equally likely, (13) becomes the likelihood ratio

$$\begin{aligned} \frac{P(d|\mathcal{A})}{P(d|\mathcal{B})} &= \frac{P(d|\mu_{\mathcal{A}}, \Sigma_{\mathcal{A}})}{P(d|\mu_{\mathcal{B}}, \Sigma_{\mathcal{B}})} \\ &= \frac{\frac{1}{(2\pi)^{Q/2} |\Sigma_{\mathcal{A}}|^{1/2}} e^{-(d-\mu_{\mathcal{A}})' \Sigma_{\mathcal{A}}^{-1} (d-\mu_{\mathcal{A}})/2}}{\frac{1}{(2\pi)^{Q/2} |\Sigma_{\mathcal{B}}|^{1/2}} e^{-(d-\mu_{\mathcal{B}})' \Sigma_{\mathcal{B}}^{-1} (d-\mu_{\mathcal{B}})/2}} \\ &> 1. \end{aligned} \quad (14)$$

After taking the natural logarithm of (14) and eliminating some constants, we obtain

$$(d - \mu_{\mathcal{A}})' \Sigma_{\mathcal{A}}^{-1} (d - \mu_{\mathcal{A}}) / 2 < (d - \mu_{\mathcal{B}})' \Sigma_{\mathcal{B}}^{-1} (d - \mu_{\mathcal{B}}) / 2 + \ln \frac{|\Sigma_{\mathcal{B}}|^{1/2}}{|\Sigma_{\mathcal{A}}|^{1/2}}. \quad (15)$$

Therefore, if the difference d of the feature vectors of two images satisfy the inequality in (15), this image pair is assigned to the relevance class, otherwise it is assigned to the irrelevance class.

5.2 Experimental Set-up

Suitable measures for the classification performance are misdetection and false alarm. In content-based retrieval we are more concerned with misdetection because we want to retrieve all the images similar to the query image. False alarm rate is also important because the purpose of querying a database is to retrieve similar images only, not all of them. We define total cost as 3 misdetection and 2 false alarm and use it as the criterion for “goodness”, i.e. if a subset of features has a small total cost compared to others, it is called “good”.

If the dimension of the feature space is large, it is computationally too expensive to do classification tests using all possible subsets of the features. In our work, first, we do tests using only one of the features at a time. The second test, which shrinks down feature sets, is done by first computing the total cost using all Q features. The feature with the worst total cost is discarded and the total cost using the remaining $Q-1$ features is computed. Then, the worst feature among the remaining $Q-1$ features is discarded and this procedure continues until one feature is left. A third test, which builds up feature sets,

is done by starting with the total cost for each individual feature and selecting the best one. Given this best one, pairs of features are formed using one of the remaining features and this best feature. Total cost is computed for each pair and the one having the smallest cost is selected. Given the best two features, next, triplets of features are formed using one of the remaining features and these two best features. This procedure continues until all or a preselected number of features are used. These tests do not guarantee the optimal subset of features is found but they allow us to select a suboptimal subset without doing an exhaustive search.

6 Experiments and Results

6.1 Feature Selection

The images in our database come from the Fort Hood Data ¹ of the RADIUS Project and also from the LANDSAT and Defense Meteorological Satellite Program (DMSP) Satellites. The RADIUS images consist of visible light aerial images of the Fort Hood area in Texas, USA. The LANDSAT images are from a remote sensing image collection. The test database for feature selection contains 10,410 256×256 images with a total of 38,240 groundtruth image pairs. Therefore, experiments for each parameter combination tested consist of classifying approximately 38,000 image pairs.

Line-Angle-Ratio Statistics:

The goal of these feature selection tests is to select the quantizer that performs the best. In order to reduce the search space, we consider only 15, 20 and 25 as the possible number of quantization cells. The quantizer with 15 cells resulted in a total cost of 30.20% whereas quantizers with 20 and 25 cells had 30.05% and 30.22% total costs respectively. As a result, we decided to use the quantizer with 20 cells.

Co-occurrence Variances:

The goal of our feature selection tests is to select the set of distances, among distances of 1 to 20 pixels, that perform the best according to the classification criteria. Each distance considered here is a combination of four features which correspond to variances computed at 0, 45, 90 and 135 degree orientations for that distance.

Results of the feature selection tests are given in Figure 5. In the experiments, building up feature sets decreased the total cost faster than shrinking down the set of all features. Another observation was that after using approximately 2 or 3 distances, total cost did not decrease much. As a result, using the distances 1 and 20 together had the minimum total cost of 29.36% among

all the possible combinations of 2 distances. This resulted in an 8-dimensional feature vector.

When we consider the Remote Sensing Dataset and the Fort Hood Dataset separately, smaller distances resulted in smaller total costs for Remote Sensing images, while smaller total costs were obtained using large distances for Fort Hood images. This is consistent with the results of Weszka *et al.*³¹ who stated that co-occurrence matrices computed for small distances performed better for a LANDSAT dataset. We believe that the reason for this is the strong micro-scale texture information in Remote Sensing images in contrast to larger structures in the Fort Hood images.

Although these feature selection tests do not guarantee to have an optimal solution, they resulted in a suboptimal one in 1,560 classification tests without using exhaustive search which would then require $2^{20} - 1$ classification tests.

6.2 Retrieval Performance

Two traditional measures for retrieval performance in the information retrieval literature are precision and recall. Precision is defined as the percentage of retrieved images that are actually relevant and recall is defined as the percentage of relevant images that are retrieved. For these tests, we randomly selected 340 images from the total of 10,410 and formed a groundtruth of 7 categories; parking lots, roads, residential areas, landscapes, LANDSAT USA, DMSP North Pole and LANDSAT Chernobyl. Likelihood values² which were derived from Eq. (15) were used to rank the database images. Average precision was used to evaluate the retrieval performance.

For comparison, UCSB's Gabor texture features,²⁴ IBM's QBIC texture features¹² and TUT's moments texture features⁸ were also tested with Euclidean distance as the distance measure. Gabor features result in a 60-dimensional vector of means and variances of the image values filtered by a Gabor filter bank of 5 scales and 6 orientations. QBIC features result in a 4-dimensional vector of coarseness, contrast, directionality and orientation. Moments features result in a 36-dimensional vector of the means, variances, medians and absolute median deviations of the image values filtered by moment filters of up to 3rd order (which makes 9 2-dimensional filters).

Precision averaged over all 340 images when 12 images were retrieved was 0.73, 0.83, 0.48, and 0.51 for the ISL features, Gabor features, QBIC features, and moments features respectively. In most of the groundtruth groups, our features performed similarly to the Gabor features and both of them always performed significantly better than both QBIC and moments features. Figure 6 shows the average precision for some of the groundtruth

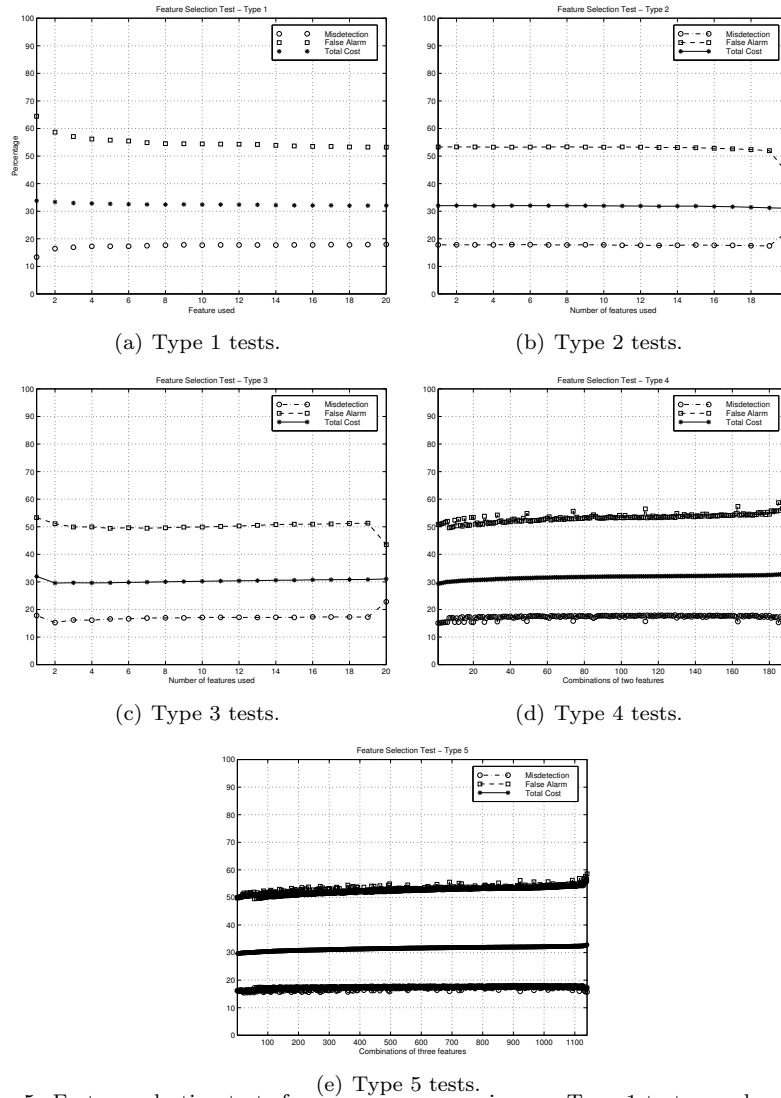
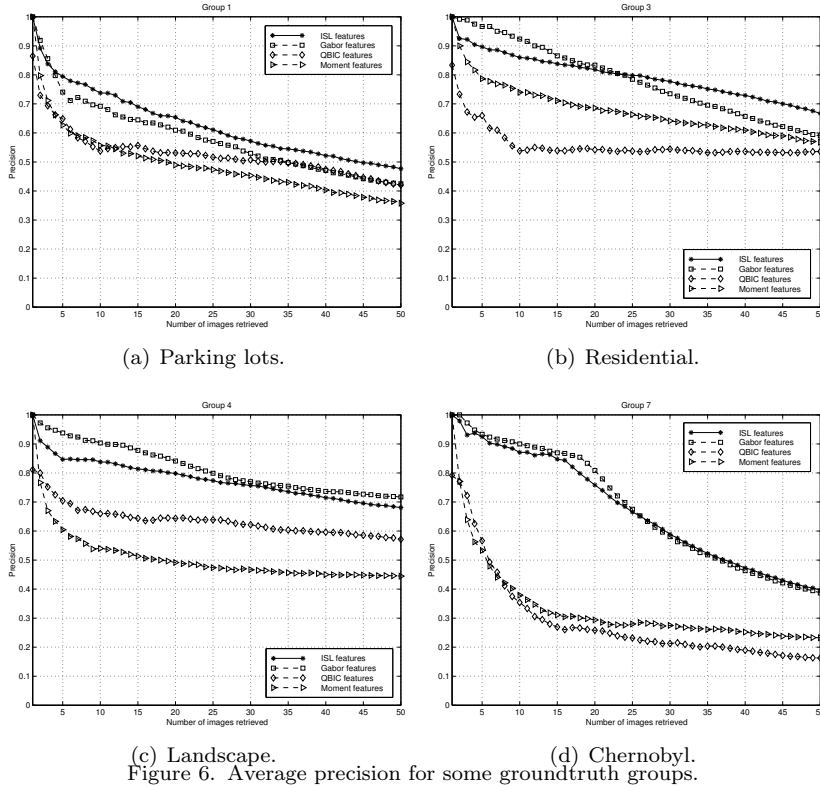
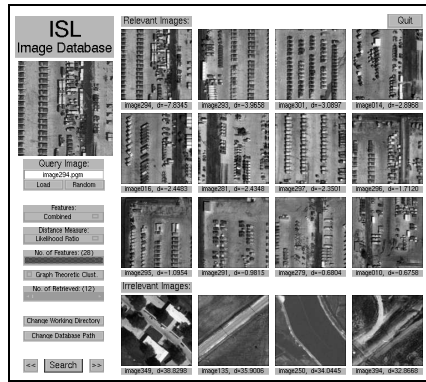


Figure 5. Feature selection tests for co-occurrence variances. Type 1 tests are done using only one feature at a time, type 2 tests are done using all features first and discarding the worst features one by one, type 3 tests are done using the best feature first and adding the next best features one by one, type 4 tests are done using all possible combinations of 2 features, type 5 tests are done using all possible combinations of 3 features. The criterion for “goodness” is the total cost which is plotted as (star,*).

groups. The feature extraction time for our features was approximately 30 times faster than that of the Gabor features. Some example queries are shown in Figure 7. More examples and our groundtruth data set can be found at <http://isl.ee.washington.edu/~aksoy/research/database.shtml>.



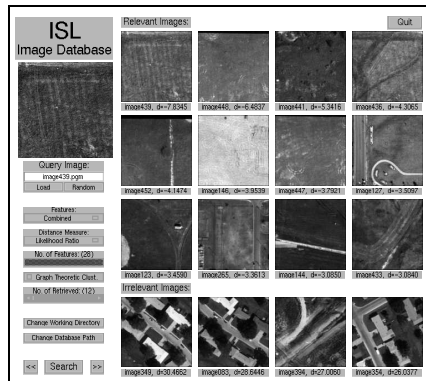
Feature space visualizations for our features and the other features are given in Figure 8. Each feature is first normalized by a transformation to a Uniform[0,1] random variable using its marginal cumulative distribution function. Then, the high-dimensional feature space is projected into the first three principal components to reduce the dimensionality and Sammon's non-linear mapping algorithm¹⁸ is used to refine the projections. These plots show how the feature space is structured compared to the manually generated groundtruth. Although there is no perfect mapping to 3D, it can be seen from Figure 8 that our features and Gabor features create a feature space



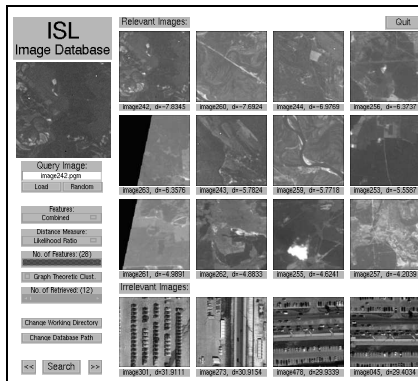
(a) Example query for parking lots.



(b) Example query for residential.



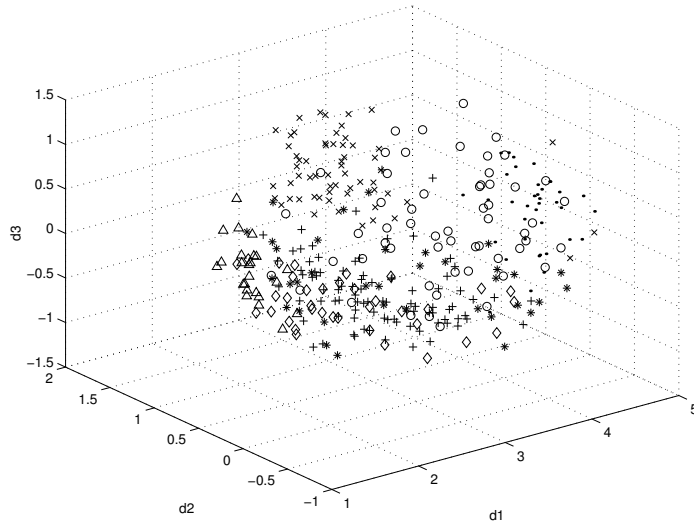
(c) Example query for landscape.



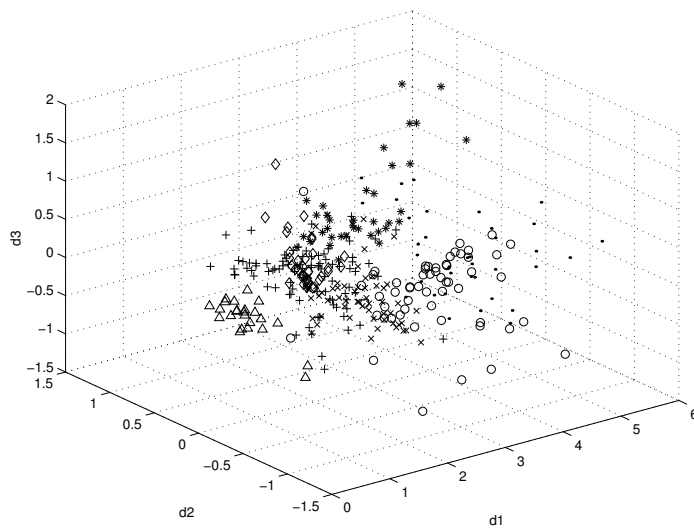
(d) Example query for Chernobyl.

Figure 7. Retrieval examples with the upper left image as the query. Among the retrieved images, first three rows show the 12 most relevant images in descending order of similarity and the last row shows the 4 most irrelevant images in descending order of dissimilarity.

structure that is closer to the manual groundtruth. Some of the individual query results look much better than the average precision results for some groundtruth groups. The reason for this are the difficulties encountered during assigning complex aerial images (e.g. roads and buildings) into single categories. Another observation is that different features can represent different classes of images well so the retrieval performance can be improved significantly if we can find effective ways of combining features from multiple feature extraction algorithms.

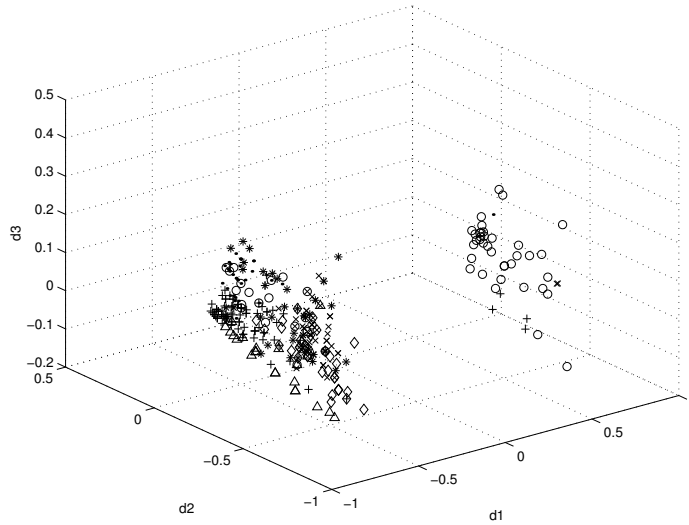


(a) ISL features.

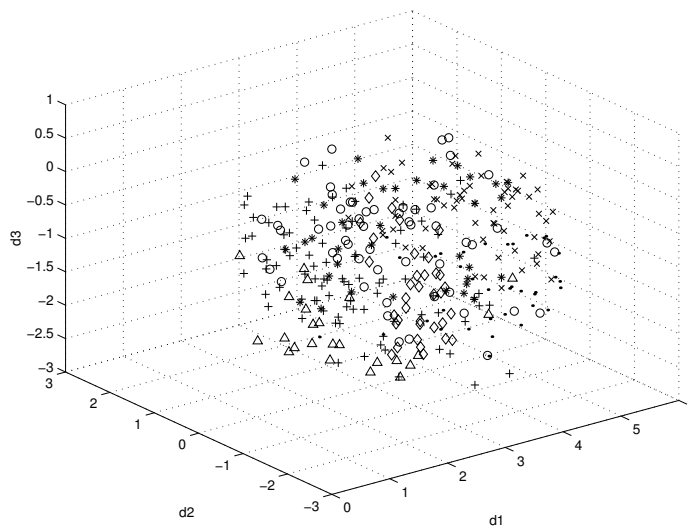


(b) Gabor features.

Figure 8. Feature space projections. The groundtruth groups are: parking lots (point), roads (circle), residential areas (x-mark), landscapes (plus), LANDSAT USA (star), DMSP North Pole (diamond) and LANDSAT Chernobyl (triangle).



(c) QBIC features.



(d) Moments features.

Figure 8. Feature space projections (cont.).

7 Conclusions

We described easy-to-compute but effective low-level textural features. The first set of features capture the global spatial organization in the image using

the edge and line information. The second set of features are effective if the image is dominated by a fine, coarse, directional, or repetitive texture. Some key aspects of this work include a statistical line selection algorithm that uses hypothesis testing to eliminate lines that do not have significant difference between the gray level distributions in the regions on their right and left, and also feature selection tests to determine the parameters of the feature extraction algorithms to avoid having less significant or even redundant features that increase computation but contribute very little in the decision process.

Precision averaged over 340 randomly selected queries on a database of approximately 10,000 images was used to evaluate the retrieval performance in comparative tests with three other texture analysis algorithms. These tests showed that our features had an average precision of 0.73 when 12 images were retrieved and performed significantly better than two of the other features while having similar performance as the third one. They can be combined with other features to further improve the performance and make better inferences about the high-level descriptions of the images.

Acknowledgments

The authors would like to thank Tom Minka from MIT Media Laboratory, USA, for the code for QBIC texture features, Bogdan Cramariuc from Tampere University of Technology, Finland, for the code for moment texture features, and Wei-Ying Ma from the University of California, Santa Barbara, USA, for the code for Gabor texture features.

References

1. *Fort Hood Datasets*, <http://www.mbvlab.wpafb.af.mil/public/sdms/datasets/fthood/index.htm>.
2. S. Aksoy, *Textural features for content-based image database retrieval*, Master's thesis, University of Washington, Seattle, WA, June 1998, Online: <http://isl.ee.washington.edu/~aksoy/thesis.shtml>.
3. S. Aksoy and R. M. Haralick, *Content-based image database retrieval using variances of gray level spatial dependencies*, Proceedings of IAPR International Workshop on Multimedia Information Analysis and Retrieval (Hong Kong), August 13–14 1998, as Lecture Notes in Computer Science, vol. 1464, pp. 3–19.
4. ———, *Textural features for image database retrieval*, Proceedings of IEEE Workshop on Content-Based Access of Image and Video Libraries, CVPR'98 (Santa Barbara, CA), June 1998, pp. 45–49.

5. ———, *Probabilistic vs. geometric similarity measures for image retrieval*, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (Hilton Head Island, South Carolina), June 2000.
6. J. Canny, *A computational approach to edge detection*, IEEE Transactions on Pattern Analysis and Machine Intelligence **8** (1986), 679–698.
7. C. Carson, S. Belongie, H. Greenspan, and J. Malik, *Region-based image querying*, Proceedings of IEEE Workshop on Content-Based Access of Image and Video Libraries, 1997.
8. F. A. Cheikh et al., *Muvis: A system for content-based indexing and retrieval in large image databases*, SPIE Storage and Retrieval of Image and Video Databases VII (San Jose, CA), January 1999, pp. 98–106.
9. Y. T. Chien, *A sequential decision model for selecting feature subsets in pattern recognition*, IEEE Transactions on Computers **C-20** (1971), no. 3, 282–290.
10. R. W. Connors and C. A. Harlow, *A theoretical comparison of texture algorithms*, IEEE Transactions on Pattern Analysis and Machine Intelligence **2** (1980), no. 3, 25–39.
11. A. Etemadi, *Robust segmentation of edge data*, IEE Image Processing Conference, 1992.
12. M. Flickner et al., *The QBIC project: Querying images by content using color, texture and shape*, SPIE Storage and Retrieval of Image and Video Databases, 1993, pp. 173–181.
13. C. C. Gotlieb and H. E. Kreyszig, *Texture descriptors based on co-occurrence matrices*, Computer Vision, Graphics, and Image Processing **51** (1990), no. 1, 70–86.
14. R. M. Haralick, *Statistical and structural approaches to texture*, Proceedings of the IEEE **67** (1979), no. 5, 786–804.
15. R. M. Haralick, K. Shanmugam, and I. Dinstein, *Textural features for image classification*, IEEE Transactions on Systems, Man, and Cybernetics **SMC-3** (1973), no. 6, 610–621.
16. R. M. Haralick and L. G. Shapiro, *Glossary of computer vision terms*, Pattern Recognition **24** (1991), no. 1, 69–93.
17. A. K. Jain and R. Dubes, *Feature definition in pattern recognition with small sample size*, Pattern Recognition **10** (1978), no. 2, 85–97.
18. J. W. Sammon Jr., *A nonlinear mapping for data structure analysis*, IEEE Transactions on Computers **C-18** (1969), no. 5, 401–409.
19. P. M. Kelly and T. M. Cannon, *CANDID: Comparison algorithm for navigating digital image databases*, Proceedings of the Seventh International Working Conference on Scientific and Statistical Database Management, September 1994, pp. 252–258.

20. C. S. Li and V. Castelli, *Deriving texture set for content based retrieval of satellite image database*, IEEE International Conference on Image Processing, 1997, pp. 576–579.
21. Y. Linde, A. Buzo, and R. M. Gray, *An algorithm for vector quantizer design*, IEEE Transactions on Communications **COM-28** (1980), 84–95.
22. F. Liu and R. W. Picard, *Periodicity, directionality, and randomness: World features for image modeling and retrieval*, IEEE Transactions on Pattern Analysis and Machine Intelligence **18** (1996), no. 7, 722–733.
23. W. Y. Ma and B. S. Manjunath, *NETRA: A toolbox for navigating large image databases*, Proceedings of IEEE International Conference on Image Processing, 1997.
24. B. S. Manjunath and W. Y. Ma, *Texture features for browsing and retrieval of image data*, IEEE Transactions on Pattern Analysis and Machine Intelligence **18** (1996), no. 8, 837–842.
25. P. M. Narendra and K. Fukunage, *A branch and bound algorithm for feature subset selection*, IEEE Transactions on Computers **C-26** (1977), no. 9, 917–922.
26. P. P. Ohanian and R. C. Dubes, *Performance evaluation for four classes of textural features*, Pattern Recognition **25** (1992), no. 8, 819–833.
27. A. Pentland, R. W. Picard, and S. Sclaroff, *Photobook: Content-based manipulation of image databases*, SPIE Storage and Retrieval of Image and Video Databases II, February 1994, pp. 34–47.
28. J. R. Smith, *Integrated spatial and feature image systems: Retrieval, analysis and compression*, Ph.D. thesis, Columbia University, 1997.
29. H. Tamura, S. Mori, and T. Yamawaki, *Textural features corresponding to visual perception*, IEEE Transactions on Systems, Man, and Cybernetics **SMC-8** (1978), no. 6, 460–473.
30. M. Tuceryan and A. K. Jain, Handbook of Pattern Recognition and Computer Vision, ch. Texture Analysis, pp. 235–276, World Scientific Publishing Company, River Edge, NJ, 1993, pp. 235–276.
31. J. S. Weszka, C. R. Dyer, and A. Rosenfeld, *A comparative study of texture measures for terrain classification*, IEEE Transactions on Systems, Man, and Cybernetics **SMC-6** (1976), no. 4, 269–285.
32. A. Whitney, *A direct method of nonparametric measurement selection*, IEEE Transactions on Computers **20** (1971), no. 9, 1100–1103.