Report Number: WUCS-87-24

1987-10-01

An Evaluation of Two Texture Classification Methods

Will D. Gillett

AN EVALUATION OF TWO TEXTURE CLASSIFICATION METHODS

Will D. Gillett

WUCS-87-24

October 1986

Department of Computer Science
Washington University
Campus Box 1045
One Brookings Drive
Saint Louis, MO 63130-4899

This research was supported in part by the Defense Mapping Agency under contract number DMA800-85-C-0010.

TABLE OF CONTENTS

1. Overview 1
   1.1. Approach 2
   1.2. Training Set Paradigm 3

2. Co-Occurrence Matrices 7
   2.1. Definition and Metrics 7
   2.2. Implementation 10
   2.3. Results and Analysis 11

3. Laws' Texture Energy Measures 15
   3.1. Laws' General Approach 15
   3.2. Implementation 17
   3.3. Results and Analysis 20

4. Comparison and Conclusions 21

TABLE OF FIGURES

Figure 1: Original Image 1
Figure 2: Training/Test Set `tr` 4
Figure 3: Training/Test Set `te` 4
Figure 4: Training/Test Set `ed` 5
Figure 5: Convolution with E5E5 Mask 18
Figure 6: Average Over 15x15 Macrowindow 18

TABLE OF TABLES

Table 1: COM Percentage of Correct Classification - Center Distances 12
Table 2: COM Percentage of Correct Classification - Individual Distances 14
Table 3: Laws' TEM Percentage of Correct Classification - Center Distances 20

1. Overview

This working paper presents the results of a comparative study between two texture classification methods: co-occurrence matrix (COM), as define by Haralick, Shanmugam, and Dinstein[1] and extended by many others[2,3,4,5], and texture energy measures (TEM), as defined by Laws[6]. The co-occurrence matrix approach has not been very effective on the specific images with which we are working. However, the Laws' approach is giving us quite good results.

The comparison analysis in this report is performed using a single image, shown in Figure 1. A training set paradigm is used; three training/test sets were developed and used as a common basis for the comparative analysis. All three training sets contain 84 points, 12 points in each of 7 categories. Two of these training/test sets are composed of points residing in the center or middle of texture fields; they are designated as *central training sets.* The third training/test set is composed of points residing near the edge (2 or 3 pixels from the edge) of

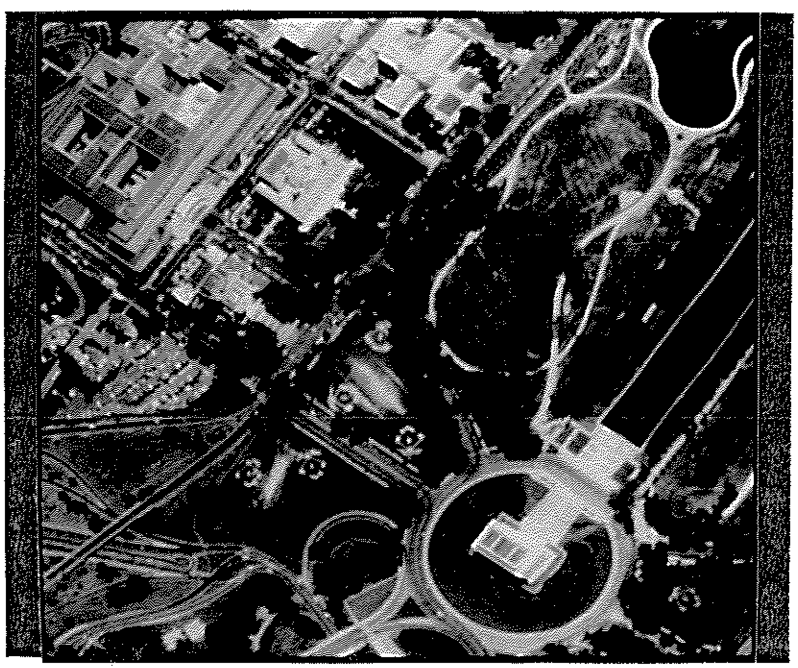


Figure 1: Original Image

texture fields; it will be designated as an *edge training set.*

Here we summarize the results obtained using the two central training sets applied using different context window sizes, ranging from 5×5 to 15×15. Four metrics were used in the COM method. The percentage of correct classification ranges obtained using these metrics were: 2nd order statistics — 17-40% correct classification; L1 norm — 32-70% correct classification; L2 norm — 35-70% correct classification; vote (between the 3 previous metrics) — 33-69% correct classification. The results for Laws' texture energy measures were analyzed in two different spaces: the original texture space, and discriminant space. The percentage of correct classification ranges obtained were: texture space — 58-86% correct classification; discriminant space — 67-98% correct classification.

Our conclusion from this comparative analysis is that Laws' texture energy measures are significantly more effective than COM methods for texture analysis of the kind of image we are interested in analyzing, which are aerial photographs containing mostly adirectional textures.

1.1. Approach

Our general approach to texture analysis is somewhat different than that applied by others[2,3,4,5]. We are attempting to determine texture on a pixel-by-pixel basis. (Of course, this requires knowledge of the surrounding pixels.) This is in contrast to other methodologies in which determining the texture of larger regions (either predefined or undefined) is the objective. We feel that our approach has the advantage that the underlying texture analysis can be applied in more flexible ways. For instance, a variety of segmentation or region growing algorithms can be developed if a pixel-by-pixel texture classification analysis is used underneath. There are also various efficiency advantages that occur because the texture at a specific pixel can be calculated immediately without having to analyze an entire region to find the texture at just one specific point.

The general approach is to consider context windows of size n×n (where n ranges from 5 to 15 in increments of 2). For each point in a training set certain information is extracted about the pixels present in this window; this information is assumed to represent a summary of the texture at the original point. This information is collected for each point in the training set along with the category into which each point has been placed (a priori). Then when an arbitrary point in an image is to be classified into a category, the same information about that point is collected and compared with that extracted for the points in the training set. The point is classified as being in whichever category the summary data is "closest" to. There are many metrics that might measure this concept of "closest", and these can be applied on (a) a point-by-point basis for each point in the training set or (b) a category basis, in which the information has been summarized (usually by a simple average) for all points in each category.

1.2. Training Set Paradigm

Our approach to texture classification is the classical training set paradigm.

(a) Identify certain predefined categories of texture within the image (this is done ad hoc by eye, depending on the specific textures present in the original image).

For the image shown in Figure 1, we have selected 7 categories of interest: tree, grass, water, roof, earth, road, and shadow.

(b) Develop a specific training set based on these predefined categories.

Based on these categories, we have developed 3 training sets (which will also be used as test sets): `tr`, `te`, and `ed`. Each of these training sets contain 84 points, 12 points in each of the 7 categories. Figures 2 and 3 display the `tr` and `te` training sets, for which points have been selected in the centers of texture fields. Figure 4 displays the `ed` training set, in which points have been selected near the edge of texture fields.

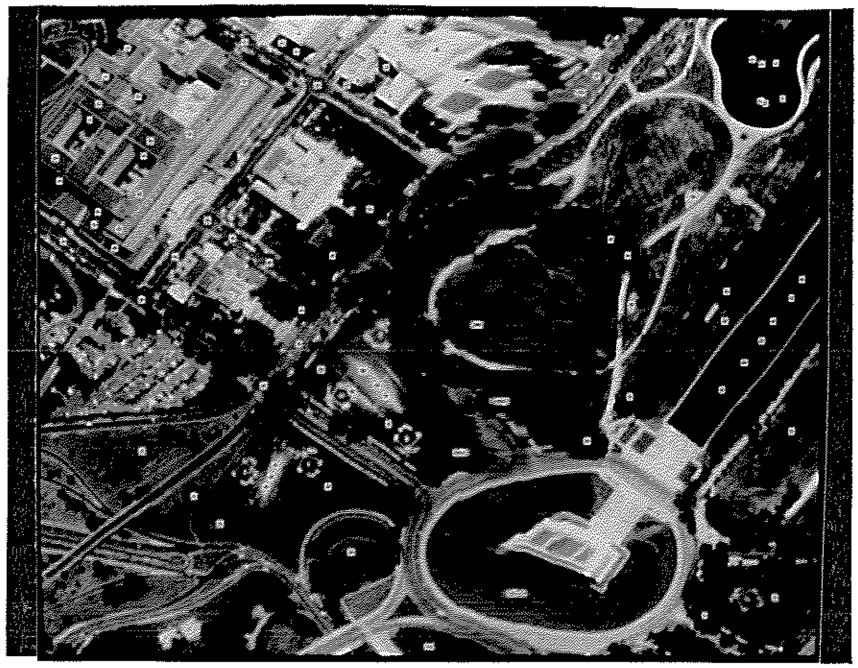


Figure 2: Training/Test Set tr

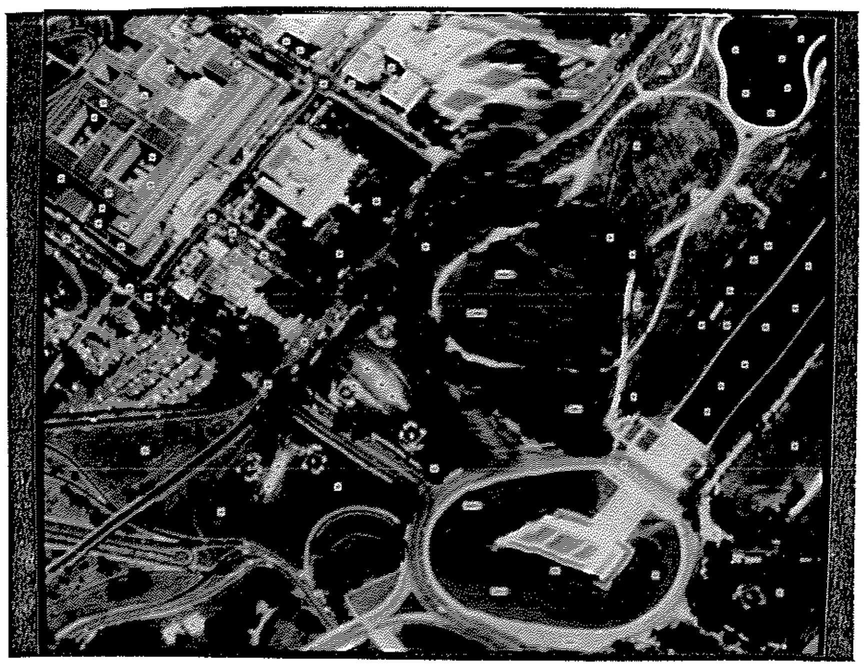


Figure 3: Training/Test Set te

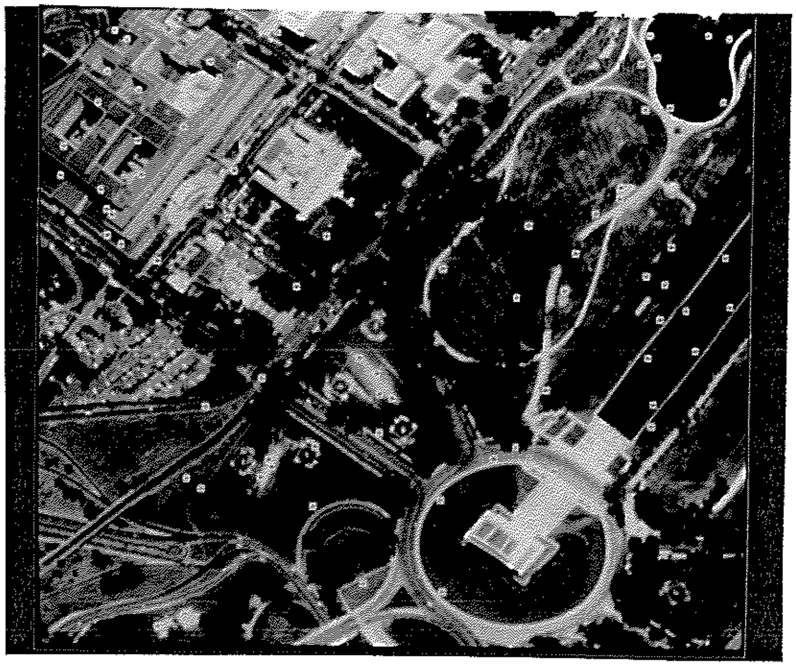


Figure 4: Training/Test Set ed

Multiple training sets were developed for several reasons. First, we wanted to determine how sensitive these methods are to different training sets, e.g., training sets whose points reside in the centers of or at the edges of texture fields. Second, we wanted multiple test sets to be able to evaluate the accuracy of the classification on test sets other than the original training set. Specifically, we were interested in how accuracy dropped off near the edges of texture fields.

(c) Extract certain information about the points in the training set.

Different information is extracted for the two methods analyzed. In the case of co-occurrence matrices, the major information extracted is the COM itself. However, three different metrics will be used to measure the concept of "closest". First, seven 2^{nd} order statistics will be extracted for this matrix; this will be considered to be a vector in 7-dimensional space. Second, an L1 norm will be applied to the matrix (when viewed as a vector). Third, an L2 norm will be applied to the matrix (when viewed as a vector).

Our implementation of Laws' TEM produces a vector in 15-dimensional space. This is the information extracted in this case.

(d) Perform any transformation analysis (such as a discriminant analysis[7]) that may help to "separate" the points in the different categories.

In the case of co-occurrence matrices, no transformation analysis is performed. In the case of Laws' TEM, a discriminant analysis is performed to attempt to separate the categories.

It is possible to attempt to apply a discriminant analysis to the data extracted form the co-occurrence matrices; however, this has not been done because it is relatively expensive to apply for the L1 and L2 norms. However, in order to show the affect of the discriminant analysis, all results shown for Laws' TEM will be presented in both the original space (called texture space) before any transformation is performed and the final transformed space (called discriminant space).

(e) Calculate the "centers" of the categories in the transformed space (this is usually done by simple averaging).

In the case of the co-occurrence matrices, all centers are determined by simple averaging over all the points in a specific category. Specifically, for the 2^{nd} order statistics, the vectors in 7-dimensional space are simply averaged, component by component. For the L1 and L2 norms, the matrices are averaged, component by component.

In the case of Laws' TEM, centers are calculated in a similar manner. The vectors in 15-dimensional space are simply averaged, component by component.

(f) Extract certain information about the points in the test set (to be classified) and transform it into the transformed space (assuming that the transform analysis of (d) is not vacuous).

In the case of co-occurrence matrices, no transformation is performed, since there was no transformation analysis done.

In the case of Laws' TEM, the discriminant analysis produces a linear transformation, a 15×15 matrix. The transformation into discriminant space is achieved by a simple matrix multiplication.

(g) For each point of the test set (after transformation), determine which center it is closest to and classify it as being in the corresponding category.

In the case of co-occurrence matrices, four classifications are actually performed: simple Euclidean distance is used on the 2^{nd} order statistics vectors in 7-dimensional space; an L1 norm is applied to the COM; an L2 norm is applied to the COM; a vote is taken between the previous 3 methods.

In the case of Laws' TEM, simple Euclidean distance is used in 15-dimensional space.

2. Co-Occurrence Matrices

Co-occurrence matrices and certain metrics applied to them have been found to be very useful in texture analysis. In this section we define co-occurrence matrices and some of these metrics. We also describe our implementation and present our results.

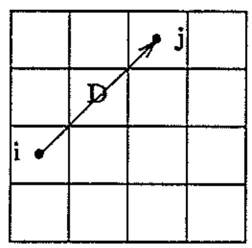
2.1. Definition and Metrics

A COM is basically a probability distribution specifying the probability of transiting from one intensity level i to another intensity level j along a specific displacement vector, D (having a given direction and length), as the displacement vector is allowed to range over a given region of interest, **R**. The basic idea is that if **R** is allowed to range over different portions of a field of uniform texture, these probabilities will not change very much. Within our context, the region of interest, **R**, will be an **n**×**n** window.

DEFINITION

Let **R** be a region of interest; **D** be a displacement vector; **i** and **j** be the *intensity* at the tail and head of **D**, respectively:

$$S_{i,j} = \text{Prob}(\text{tail}(D) = i \wedge \text{head}(D) = j \text{ for } D \in R)$$



Note that the size of the COM is a function of the intensity levels present in the image, eg., if the intensity levels range over 0-255, then the COM is a 256×256 matrix. Also note that S may be very sparse.

EXAMPLE

If R is given by the following image

1	1	4	2
3	2	5	2
5	3	4	1
1	1	4	3

and D is a unit vector in the positive x direction, then the COM is given by:

$$S = \frac{1}{12}\begin{bmatrix} 2 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Given that the assumption that S will not change very much as **R** ranges over a field of uniform texture is true, the pertinent question is how do we determine when two different co-occurrence matrices are "close" to one another? Many metrics have been proposed. Haralick, Shanmugam, and Dinstein[1] have defined a large number of 2^{nd} order statistics that might be used as components of such a metric, and Conners, Trivedi and Harlow[2] and Conners and Harlow[3] have found the following 7 to be of particular interest. Here, S is an L×L COM whose indices range from 0 to L-1.

Inertia

$$I = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}(i-j)^2 S_{i,j}$$

Cluster Shade

$$A = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}(i+j-\mu_x-\mu_y)^3 S_{i,j}$$

Cluster Prominence

$$B = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}(i+j-\mu_x-\mu_y)^4 S_{i,j}$$

Correlation

$$C = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}(i-\mu_x)(j-\mu_y)\frac{S_{i,j}}{\sigma_x\sigma_y}$$

Local Homogeneity

$$L = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\frac{1}{1+(i-j)^2}S_{i,j}$$

Energy

$$E = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} S_{i,j}^2$$

Entropy

$$H = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} S_{i,j}\log(S_{i,j})$$

where

$$\mu_x = \sum_{i=0}^{L-1} i \sum_{j=0}^{L-1} S_{i,j}$$

$$\mu_y = \sum_{j=0}^{L-1} j \sum_{i=0}^{L-1} S_{i,j}$$

$$\sigma_x^2 = \sum_{i=0}^{L-1} (i-\mu_x)^2 \sum_{j=0}^{L-1} S_{i,j}$$

$$\sigma_y^2 = \sum_{j=0}^{L-1} (j-\mu_y)^2 \sum_{i=0}^{L-1} S_{i,j}$$

We will use these, concatenated together as a vector in 7-dimensional space (along with Euclidean distance), as one metric for "closeness".

A second metric is the L1 norm, when S is considered to be a linear vector:

$$L1({}_1S - {}_2S) = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} \left| {}_1S_{i,j} - {}_2S_{i,j} \right|$$

A third metric is the L2 norm, when S is considered to be a linear vector:

$$L2({}_1S - {}_2S) = \sqrt{\sum_{i=0}^{L-1}\sum_{j=0}^{L-1} \left({}_1S_{i,j} - {}_2S_{i,j} \right)^2}$$

2.2. Implementation

Co-occurrence matrices have been found to be very useful in textures that have a specific "grain" to them, i.e., there is a specific directionality to the texture and a regular interval over which the texture repeats itself. The COM is very sensitive to the displacement vector, D, which should be in the direction of the "grain" and have the appropriate length. In our case, all the textures in the image we are analyzing are adirectional. Thus, it is appropriate to average co-occurrence matrices over many different vectors to reduce or eliminate this sensitivity to D. We have chosen to average co-occurrence matrices over Ds with four primary directions (north, northeast, east and southeast) and all lengths that remain within the **n**$\times$**n** window.

Our images are composed of grey scales that range from 0-255. Thus, a conceptual S matrix is 256$\times$256, i.e., it contain 2^{16} entries. If such matrices are not compressed in some manner, memory constraints become a significant consideration (1 MEG = 16 points). There are several ways of compressing these matrices. We have chosen to compress co-occurrence matrices into matrices of size 10$\times$10 by linearly mapping intensity levels between the minimum and maximum intensity levels within a window onto a scale ranging from 0-9, as done by Chen and Pavlidis[5]. This also tends to suppress intensity level variation over a uniform texture field.

2.3. Results and Analysis

We have determined the percentage of correct classification (using the four metrics discussed prior) over a variety of window sizes and utilizing (in all combinations) each training/test set as a training set and a test set. The results of the analysis, using centers of categories to determine the closest category, are presented in Table 1. First, note that when the edge training set is compared to the central test sets (or visa versa), accuracy drops off markedly, as might be expected. Only when the edge training set is compared to itself as a test set does the accuracy remain relatively high, although even in this case, results drop off slightly.

Second, note that the 2^{nd} order statistics give relatively poor results. In the central training/test sets over all window sizes, the range percentage is 17-40% correct classification; not very impressive. This is somewhat surprising, since other researchers have found these to be very powerful in texture analysis, at least in textures that have a specific "grain". We suspect that the reason we are not obtaining better results may be a combination of the following

Table 1
COM Percentage of Correct Classification - Center Distances

Test Set	Window Size	Training Set (2^{nd} order statistics, L1, L2, Vote) tr	te	ed
tr	5	25,56,55,56	32,32,35,33	21,30,31,30
	7	32,56,52,54	24,45,40,45	19,33,29,32
	9	31,59,58,59	35,56,50,54	23,38,37,38
	11	36,68,65,67	36,67,60,67	18,50,52,49
	13	32,63,58,63	29,61,56,63	20,60,58,58
	15	39,68,64,68	38,64,60,64	20,52,56,52
te	5	17,36,35,37	25,57,54,56	14,29,26,26
	7	19,50,50,49	26,57,58,58	23,40,43,42
	9	39,56,58,56	44,61,61,61	20,48,46,48
	11	40,64,65,64	39,68,63,68	17,51,57,51
	13	39,64,64,64	37,65,67,65	19,56,55,58
	15	36,69,70,69	40,70,67,69	20,63,60,63
ed	5	21,35,42,38	20,21,29,21	27,44,50,45
	7	24,36,36,36	18,35,33,33	31,48,51,49
	9	19,40,39,40	21,45,44,45	26,68,62,65
	11	20,44,44,44	21,49,49,50	26,69,63,68
	13	21,48,46,48	19,49,46,49	25,67,64,67
	15	24,45,50,46	19,45,52,45	23,69,65,68

properties.

- small window sizes

  The extent of many of the texture fields present in the image we are analyzing is already less that our maximum window size of 15×15. It did not seem reasonable to attempt larger window sizes. Also, note that as window size increase, the accuracy *tends* to increase, but this is not uniformly true. It is probable that the accuracy here has reached a plateau at about the 15×15 window size.

- compressed intensity ranges

  This seemed appropriate for efficiency in both time and space and tends to suppress intensity level changes over uniform texture fields.

- averaging over multiple displacement vectors

  This was done to reduce the sensitivity of the COM to the direction and magnitude of the displacement vector, since our textures are adirectional.

- unscaled vectors

  A typical vector in 7-dimensional space looks something like

  $$[0.0286, 3.8327, 0.0304, 9.5000, 0.3150, -14.3791, 308.6492]$$

  where the magnitudes of the values shown here are representative of the values normally present in these components. We did not scale (or weight) the separate components of the vector so that a relative change in one component was comparable to a corresponding relative change in another component. One mechanism for achieving this goal would be to perform a mean/standard deviation analysis on the vectors in the training set and transform vectors in the test set assuming that they will have approximately the same distribution.

The first three reasons above may have to be discounted somewhat since we obtain much better results for the L1 and L2 norms, to which the same negative arguments could be applied.

The results for the L1 and L2 norms are much better, ranging (for the central training/test sets) from 32-70% correct classification; this is significantly better than the 2^{nd} order statistics. Note that the increased accuracy as a function of window size is much more uniform than that for the 2^{nd} order statistics. It may be that these types of norms are much better for analysing adirectional textures than the 2^{nd} order statistics, which have given good results in textures having a specific "grain".

The statistics for the vote metric are driven by the L1 and L2 norms and are essentially a reflection of their increased accuracy over the 2^{nd} order statistics.

We were interested in determining how the classification accuracy would change if instead of using the centers of categories, we compared each test point to *each* training point. The results of this analysis is shown in Table 2. Looking down the main diagonal of the table, we note accuracies of 100%. This is expected, but not significant, since the test set was the training set. Looking at the off diagonal entries for the central training/test sets, we obtain a range of 17-46% correct classification for the 2^{nd} order statistics and 31-82% correct classification for the L1 and L2 norms. In general, this seems to represent an increase of about 5-10% on the average. However, note that this increase in accuracy is not uniform over the entire table.

This is not a significant increase given the increased computational complexity of having to compare *all* points in the training set of a category instead of just 1 summary point, the

Table 2
COM Percentage of Correct Classification - Individual Distances

		Training Set (2^{nd} order statistics, L1, L2, Vote)		
Test Set	Window Size	tr	te	ed
tr	5	100,100,100,100	20,36,35,32	30,23,27,26
	7	100,100,100,100	29,50,45,49	23,35,27,30
	9	100,100,100,100	37,58,51,58	23,40,39,39
	11	100,100,100,100	36,75,67,75	21,51,50,49
	13	100,100,100,100	43,73,71,73	25,52,51,52
	15	100,100,100,100	39,77,75,76	26,54,56,52
te	5	17,33,31,33	100,100,100,100	23,27,20,26
	7	32,49,45,49	100,100,100,100	25,38,33,39
	9	44,63,61,64	100,100,100,100	20,40,35,40
	11	38,75,77,75	100,100,100,100	20,57,52,55
	13	39,77,76,80	100,100,100,100	37,56,55,56
	15	46,82,81,82	100,100,100,100	31,58,62,57
ed	5	17,36,38,33	15,26,25,26	100,100,100,100
	7	35,32,31,32	33,43,38,43	100,100,100,100
	9	24,43,38,39	23,50,46,48	100,100,100,100
	11	29,52,51,52	26,51,48,51	100,100,100,100
	13	24,51,51,51	23,51,51,50	100,100,100,100
	15	25,52,50,52	29,57,51,57	100,100,100,100

center of the category. Comparison with Laws' TEM, in which only centers of categories were used, will be performed only for Table 1. It was assumed that a similar increase in accuracy would result if individual comparisons were used for Laws' TEM.

The results obtained here are surprisingly poor for the 2^{nd} order statistics and surprisingly good for the L1 and L2 norms. We have not seen these norms used as metrics for co-occurrence matrices in the literature. More work should probably be done to verify these results on more extensive data and determine if this degree of accuracy is retained or increased when applied to textures with "grain".

3. Laws' Texture Energy Measures

In this section, we discuss Laws' general approach, our specific implementation of it, and and our results using the same image and training/test sets as those used for the co-occurrence matrices.

3.1. Laws' General Approach

Laws found, after analyzing many texture analysis methods, that certain 1-dimensional convolution masks identified or extracted certain important properties of an image that are useful in texture analysis. Specifically, the four most important that he found were:

$$L5 = [1 \quad 4 \quad 6 \quad 4 \quad 1]$$

$$E5 = [-1 \quad -2 \quad 0 \quad 2 \quad 1]$$

$$S5 = [-1 \quad 0 \quad 2 \quad 0 \quad 1]$$

$$R5 = [1 \quad -4 \quad 6 \quad -4 \quad 1]$$

These convolution masks enhance intensity level, edges, spots, and ripples, respectively, in any specific direction that the mask is applied. The cross product of these four masks in the horizontal and vertical directions produce 16 2-dimensional convolution masks, which he found extracted information useful in texture analysis. As an example, the E5E5 2-dimensional convolution mask is given by:

$$\begin{bmatrix} 1 & 2 & 0 & -2 & -1 \\ 2 & 4 & 0 & -4 & -2 \\ 0 & 0 & 0 & 0 & 0 \\ -2 & -4 & 0 & 4 & 2 \\ -1 & -2 & 0 & 2 & 1 \end{bmatrix}$$

The L5L5 mask does not produce useful information for direct texture analysis, but was found useful for normalizing the other 15 convolutions as a function of intensity level. Laws found the other 15 masks to be useful in texture analysis, and these are the ones we have used in our implementation.

The images produced by applying the 15 convolutions masks to the original image are not directly useful in texture analysis because the information at each pixel is very local to that pixel (no more than two pixels away). However, it does extract local artifacts of the original image (eg., edges, spots, and ripples). In essence, a blurring affect is needed to collect information from more distant pixels. He found that taking a standard deviation over a macrowindow of size $\mathbf{n}\times\mathbf{n}$ was appropriate for producing this blurring affect by pulling in information from adjacent pixels. Taking the standard deviation is a computationally intensive process, and he found that a very close approximation could be obtained by instead taking an average (over the same $\mathbf{n}\times\mathbf{n}$ macrowindow) of the absolute value of the convoluted image. This is much less computationally intensive, and is the method that we have chosen.

Using these 15 images (i.e., the averages over the absolute value of the convolution for the 15 different convolution masks), each pixel can be considered to be a vector in 15-dimensional space. Using these vectors as his definition of texture space, he performed a texture analysis by specifying predefined categories, selecting a training set, performing a discriminant analysis, and classifying an entire image based on closeness to centers in discriminant space, as previously discussed.

3.2. Implementation

In this subsection, we describe our implementation of Laws' TEM, and present a pictorial representation of convolution images and averaged images. The important components of the implementation will be presented by giving a scenario of the computational activities, in the sequence in which they occur.

Step 1: Convolution Masks

Fifteen different 2-dimensional convolution masks are applied to the original image. Each convolution is performed by a standalone preprogrammed convolution process. We have convolution software that runs directly on the VAX/750 and separate software that runs on the DVP of the DeAnza. We have created the convolution images using the VAX/750 software. Each of these convolutions extracts or enhances certain artifacts of the original image. As an example of what these convolution masks extract, the results of applying convolution mask E5E5 to the image in Figure 1 is shown in Figure 5; the E5E5 mask extracts edges in both the horizontal and vertical directions.

Step 2: Averaging over an $\mathbf{n}\times\mathbf{n}$ Window

The absolute value of each of the 15 convoluted images is then averaged over an $\mathbf{n}\times\mathbf{n}$ macrowindow (where $\mathbf{n}$ ranges from 5 to 15 in increments of 2). Again, this is performed by a standalone preprogrammed averaging software; in this case we have used a version that runs on the DVP of the DeAnza. This averaging has the affect of blurring the image and pulling in information from distant pixels (a maximum of 7 pixels away). The averaging over a 15$\times$15 pixel macrowindow for the convoluted image shown is Figure 5 is presented in Figure 6. Note the blurring affect.

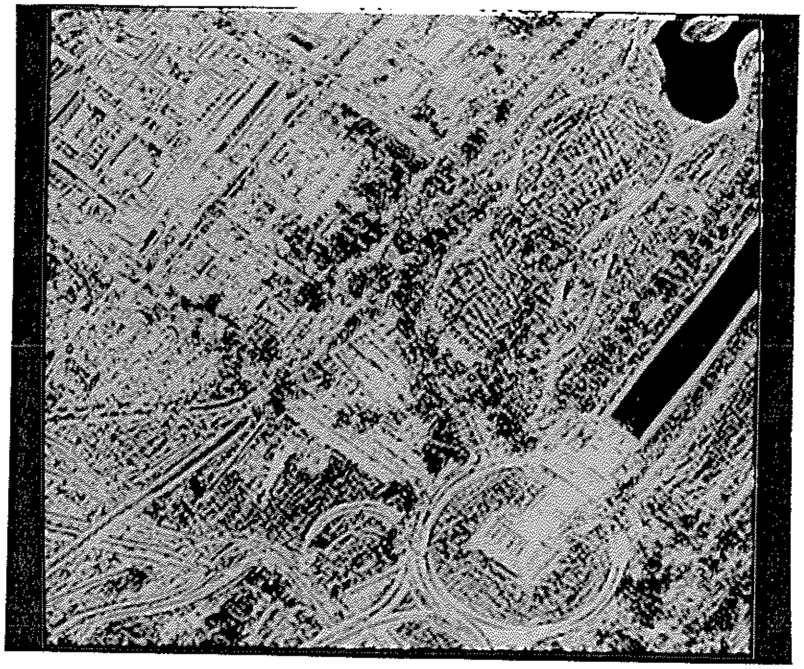


Figure 5: Convolution with E5E5 Mask

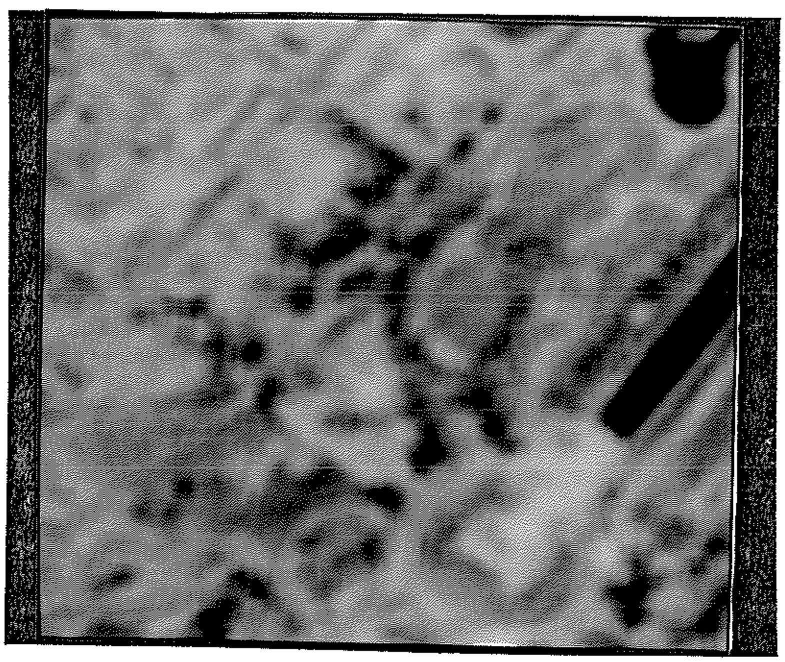


Figure 6: Average Over 15x15 Macrowindow

The results of Step 2 supply the preprocessed information needed by the remaining processing, including category and training set selection, discriminant analysis, and classification. As subsequent processing changes (eg., using different training/test sets), these two steps need not be repeated.

Step 3: Category and Training Set Selection

The seven categories and 3 training sets (`tr`, `te`, and `ed`) are used.

Step 4: Discriminant Analysis

A discriminant analysis based on the texture space of a specific training set is performed. We are currently using an interactive statistical package called S[8] to perform this analysis; however, we suspect that the discriminant analysis of any given statistical package would be sufficient for this computational component. S has been very useful to us in understanding the nature of our training sets because it is very easy to interactively compute and display (in both tabular and graphical form) the results of our analyses. The insights that can be obtained from such an interactive statistical tool have been invaluable. The discriminant analysis in S produces a transformation (in the form of a 15×15 matrix) of texture space into discriminant space.

Step 5: Classification

A classification analysis is performed both in the original texture space and in the discriminant space. Centers of the categories are calculated in the original texture space. Then each point of the given training set is transformed into discriminant space and the centers of the categories are calculated there.

Each point in a given test set is compared against the centers of the categories in the original texture space, and classified into the corresponding category there. Then each point is transformed into discriminant space and compared against the centers of the categories in discriminant space.

Calculations of the centers, transformation into discriminant space, and comparisons to categories are all done in S.

3.3. Results and Analysis

Analyses similar to those done for co-occurrence matrices were performed for Laws' TEM using all combination of training/test sets. The results are shown in Table 3. Entries for the central training/test sets over all window sizes range from 58-86% correct classification in texture space and 67-98% correct classification in discriminant space. Even the results in texture space are a significant improvement over any of the metrics used for the COM method.

Note that, in general, accuracy tends to increase (in both texture space and discriminant space) as window size increases. However, this increase is not uniform. We suspect that the

Table 3
Laws' TEM Percentage of Correct Classification - Center Distances

		Training Set (texture, discriminant)		
Test Set	Window Size	tr	te	ed
tr	5	69,87	63,69	45,64
	7	76,88	67,77	48,65
	9	77,90	68,81	49,73
	11	79,92	69,87	48,74
	13	80,93	74,89	54,77
	15	77,92	75,87	56,75
te	5	58,67	74,81	42,50
	7	68,79	75,82	46,62
	9	79,85	80,88	48,76
	11	79,81	77,98	48,77
	13	83,83	86,96	55,82
	15	83,86	85,96	58,80
ed	5	43,38	35,38	64,82
	7	44,48	42,44	65,81
	9	45,50	44,50	63,86
	11	49,52	43,49	62,85
	13	48,55	45,48	63,86
	15	46,52	49,52	63,86

accuracy in both spaces has reached a plateau at window sizes around 15×15.

Note that again when the edge training/test set is used in conjunction with the central training/test sets, accuracy is degraded, as expected. Texture space accuracy seems to be enhanced slightly when ed is used as a training set as opposed to a test set. Also, when ed is used as a test set, the discriminant analysis is not very effective in improving classification accuracy; however, when it is used as a training set, a more significant increase is observed. We have no explanation for these artifacts.

4. Comparison and Conclusions

The most meaningful comparison of these data is between the L1 and L2 norms of Table 1 and the texture space entries of Table 3. These data have the common assumptions of (a) no discriminant analysis, and (b) "closeness" is determined on the basis of centers of categories.

Restricting our attention to the central training/test sets, Laws' texture space percentages range from 58-86% correct classification and COM percentages range from 32-70% correct classification. Clearly Laws' TEM perform much better here. However, notice that when the edge training/test set is used in conjunction with the central training/test sets it is no longer clear that Laws' TEM are superior, Laws' TEM percentages ranging from 35-58% correct classification and COM percentages ranging from 21-63% correct classification.

Turning our attention to the discriminant space of Laws' TEM, percentages range from 67-98% correct classification. These are very good results and clearly superior to any of the COM metrics. However, of course, a powerful statistical back end (the discriminant analysis) has been put in place.

Such a statistical back end might also be applied to the L1 and L2 norms of the COM method. However, both the discriminant analysis and subsequent linear transformation would be very expensive here (for large images). Our implementation of co-occurrence matrices produces matrices of size 10×10 (i.e., 100 entries). When viewed as a linear vector, this requires a discriminant analysis in 100-dimensional space, and a subsequent matrix multiplication (transformation) by a 100×100 array. Of course, these complexities might be reduced by a principle component analysis or a similar data compression method. However, since Laws' TEM are clearly superior to the COM method prior to discriminant analysis, there is no reason to believe that there would be an inversion after discriminant analysis.

We conclude from this comparative analysis that Laws' TEM is significantly superior to the COM method when applied to images composed of adirectional texture fields. However, COM methods may be comparable or superior when applied to textures with a specific "grain".

---

1. Robert M. Haralick, K. Shanmugam, and Its'hak Dinstein, "Textural Features for Image Classification," *IEEE-SMC* **3**(6) pp. 610-621 (November 1973).

2. Richard W. Conners, Mohan M. Trivedi, and Charles A. Harlow, "Segmentation of a High-Resolution Urban Scene Using Texture Operators," *CGIP* **25** pp. 273-310 (1984).

3. Richard W. Conners and Charles A. Harlow, "Toward a Structural Textural Analyzer Based on Statistical Methods," *CGIP* **12** pp. 224-256 (1980).

4. Steven W. Zucker and Demetri Terzopoulos, "Finding Structure in Co-Occurrence Matrices for Texture Analysis," *CGIP* **12** pp. 286-308 (1980).

5. Patrick C. Chen and Theodosios Pavlidis, "Segmentation by Texture Using a Co-Occurrence Matrix and a Split-and-Merge Algorithm," *CGIP* **10** pp. 172-182 (1979).

6. Kenneth Ivan Laws, *Textured Image Segmentation,* University of Southern California (January 1980). Ph. D. Dissertation.

7. Sing-Tze Bow, *Pattern Recognition,* Marcel Dekker, Inc., New York (1984).

8. Richard A. Becker and John M. Chambers, *S: An Interactive Environment for Data Analysis and Graphics,* Wadsworth, Belmont, CA (1984).