The Pennsylvania State University

The Graduate School

Department of Computer Science and Engineering

THREE-DIMENSIONAL TEXTURE CLASSIFICATION USING THE DISCRETE COSINE TRANSFORM

A Thesis in

Computer Science and Engineering

by

Rodney Lee Summerscales

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

August, 2005

The thesis of Rodney Lee Summerscales was reviewed and approved* by the following:

Dennis F. Dunn
Associate Professor of Computer Science and Engineering
Thesis Adviser

Octavia Camps
Associate Professor of Electrical Engineering

Raj Acharya
Professor of Computer Science and Engineering
Head of the Department of Computer Science and Engineering

*Signatures are on file in the Graduate School.

Abstract

This thesis proposes a novel approach to classifying texture in three dimensional (volumetric) images using a bank of 3-D filters derived from the discrete cosine transform (DCT). This method is tested on collections of synthetic and natural 3-D textures. Classification accuracy is compared to that obtained with 2-D DCT and 3-D Gaussian Markov random field (GMRF) texture features [20].

The experiments in this thesis show that 3-D DCT texture features generally provide better classification accuracy than 2-D DCT and 3-D GMRF features. This result holds even when the image resolution in the third dimension is lower than the image resolution in the first two dimensions. This result is dramatically illustrated in the case where all methods are asked to discriminate between 3-D image samples of normal and emphysemic lung tissue. These image samples are extracted from low resolution CT scans of normal and emphysemic human lungs. The 3-D DCT texture features provide better discrimination between normal and emphysemic lung tissue than 2-D DCT and 3-D GMRF texture features.

Table of Contents

List of Tables

List of Figures

Acknowledgments

The completion of this degree was made possible through the support and cooperation of many people and organizations.

First of all, I thank my thesis adviser, Dr. Dennis Dunn. I am grateful for his guidance. Above all, I appreciate his time and patience as he reviewed the numerous drafts of this thesis. I am also grateful to Dr. Octavia Camps for serving on my committee. I appreciate her comments and her willingness to sacrifice her time and energy to help me. I thank Dr. Frank Ritter for providing me with space in his Applied Cognitive Science (ACS) Lab. I appreciate the moral support that I received from Dr. David Mudgett and the members of the ACS lab. I am also grateful to Dr. William Higgins for his advice concerning the use of human image data in this thesis. The CT image data used in this thesis was provided by Dr. Joseph M. Reinhardt, Department of Biomedical Engineering at the University of Iowa. I am deeply grateful to Dr. Reinhardt for his assistance.

I am also grateful to my employers who have provided me with financial support throughout my graduate studies. These include the Penn State Department of Computer Science and Engineering, Penn State School of Information Science and Technology, and the Centroid Corporation of Howard, PA. I thank Joe McCulloch and John Roe of Centroid for allowing me to work part time this year as I completed this thesis.

I would also like to thank the administrative staff in the Department of Computer Science and Engineering for their help. I am particularly grateful to Vicki Keller and Karen Corl for their assistance.

Finally, I thank my family for their love and support over the last few years. In particular, I thank my loving wife, Tiffany. Without her unfailing love and emotional support, I doubt that I would ever have finished. Furthermore, she helped by reviewing numerous drafts of this thesis. Above all, I thank God for granting me the opportunity and ability to finish this degree.

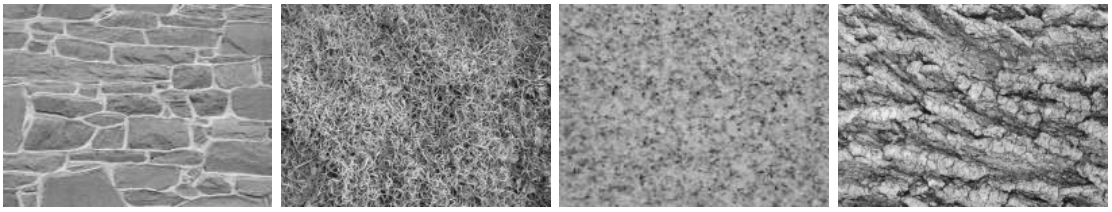
Obtaining a graduate degree can be a difficult and emotionally draining process. Feelings of frustration and discouragement constantly haunted me. As I write, I realize just how many people have supported me and contributed to this degree in one form or another. I am deeply grateful to everyone that has helped and supported me over the years.

Chapter 1

Introduction

Our senses experience texture in many ways. We use texture to characterize various sources of information such as the surfaces of objects we touch, the overall sound of songs we hear and the appearance of objects we see.

The form of texture our eyes observe is especially useful for identifying objects in two-dimensional (2-D) images which only provide visual information. Although there is no accepted mathematical definition of image texture, this thesis will define it loosely as repeated patterns of pixel values. With images containing natural texture, the patterns and their repetitions have been modified by various forms of noise. Examples of natural texture in 2-D images can be seen in Figure 1.1.



(a) (b) (c) (d)

Fig. 1.1. Samples of 2-D texture. (a) Stone wall. (b) Curly grass. (c) Granite. (d) Bark.

Due to its usefulness in interpreting images, texture has been a source of great interest in the field of computer vision. Computers can use texture to recognize objects, determine their shape, judge their condition [24] and identify flaws. Texture can be used to detect diseases such as emphysema in lung images [28]. The analysis of image texture generally falls into three categories: classification, segmentation and shape recovery. Texture classification is concerned with identifying the correct texture class to which a given pixel or region belongs. Texture segmentation partitions an image into separate regions of homogeneous texture. Shape recovery uses apparent distortions in surface texture to determine the shape of objects. This thesis is concerned with texture classification.

Texture is not a property that can be defined given a single pixel value. As illustrated in Figure 1.2, identifying the texture class to which a pixel belongs involves examining its relationship with the surrounding pixels. Texture features are computed to characterize this relationship. Once texture features are calculated, a classifier assigns the pixel or region to the texture class mostly likely to have these features. Depending on the application, the texture classes may be healthy lung/emphysemic lung or grass/brick/stone/tree bark.

Although 2-D images are useful for capturing the surfaces of objects, they contain little information regarding their internal structure. While this situation is satisfactory for many applications, fields such as medical imaging and seismic analysis are more concerned with the interior of objects than their surface. A third spacial dimension is needed to sufficiently capture internal structure. The introduction of three-dimensional sensor technology has led to the production of 3-D (volumetric) images. Whereas a 2-D

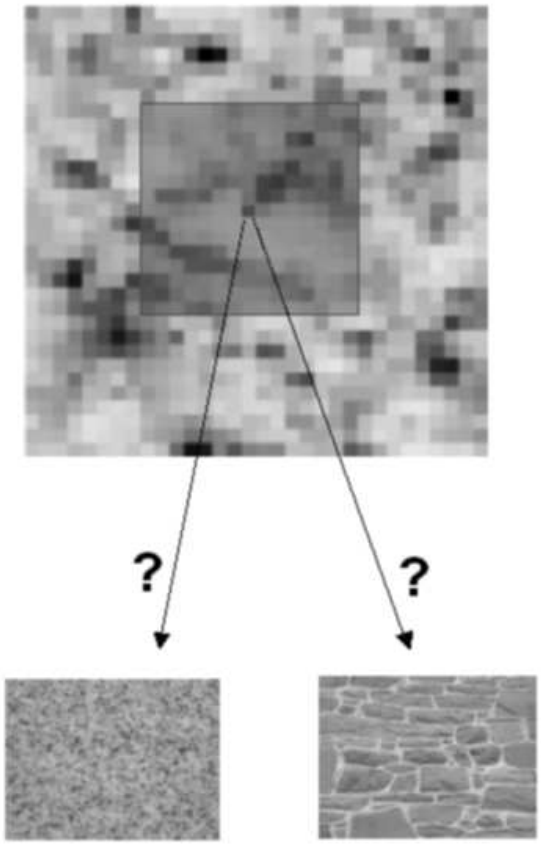


Fig. 1.2. Texture classification is the task of identifying the correct texture class to which a pixel belongs. This identification requires examining the pixel relationships in the area around the pixel to be classified.

image is a 2-D lattice of picture elements (pixels), a volumetric image (volume) is a 3-D lattice of volumetric pixels (voxels). Figure 1.3 shows the difference between 2-D and 3-D images. With 3-D images, texture can be used to identify objects by their internal structure.

Volumes are often processed as a series of 2-D images (or slices). 2-D texture features are computed for pixels in each slice [28, 8]. Unfortunately, by processing volumes as a series of separate 2-D slices, texture information across slices is ignored. Various methods for computing 3-D texture features have been developed to include this extra texture information. These methods include the use of Laws filters [13], co-occurrence matrices [11, 12], run-length matrices [31], wavelets [10], sub-band filtering [22], Gaussian Markov random fields(GMRF) [20, 19] and a combination of co-occurence matrices and Gabor filters [15].

Since texture analysis was first applied to 2-D images, there are significantly more methods to choose from for computing 2-D texture features than 3-D texture features. While many of these 2-D methods can be extended to 3-D, only a few have been. One particularly successful 2-D method involves using 2-D discrete cosine transform (DCT) filters. Unser found DCT filters to perform better at 2-D texture classification than Laws filters [27]. Ng et al [16] found a 2-D DCT filter bank to be superior to a bank of Gabor filters for 2-D texture segmentation. Tan and Kittler used 2-D DCT filter bank for analyzing texture in color images [26]. Furthermore, Randen and Husøy [18] have found a bank of 2-D DCT filters to be superior to most of the popular 2-D texture classification methods including optimized Gabor filters and co-occurrence matrices.

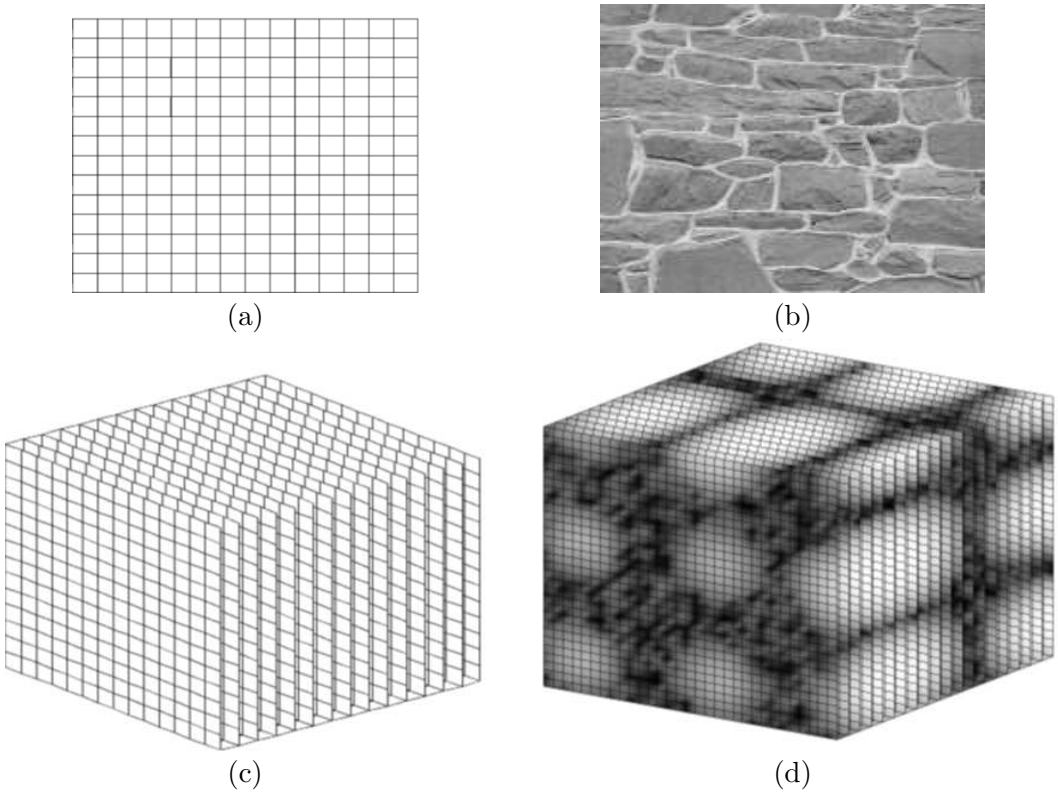


Fig. 1.3. A comparison of 2-D and 3-D images. (a) A 2-D lattice of pixels. (b) Sample 2-D image. (c) 3-D lattice of voxels. (d) Sample 3-D (volumetric) image.

Given the effectiveness of 2-D DCT texture features, this thesis explores the value of using 3-D DCT texture features for classifying texture in 3-D images. The effectiveness of 3-D DCT texture features is compared with 2-D DCT texture features. The 3-D DCT features are also compared with features from an existing 3-D method that fits texture to a 3-D GMRF model [20].

Although it may seem intuitive that 3-D DCT features are better than 2-D DCT features for classifying texture in 3-D images, the addition of a third dimension does raise some concerns. First of all, calculating 3-D DCT features involves extra computations across slices, making the calculation of 3-D DCT features more expensive than 2-D DCT features. Also, the 3-D DCT method generates more features than the equivalent 2-D DCT method. More features can improve classification, however it often requires more training data to adequately train the classifier. Furthermore, there is the concern that 3-D texture features may not provide any discriminatory advantage for 3-D images with low slice resolution. If the distance between consecutive slices is relatively large compared with the distance between pixels within a slice, the 3-D image may contain little or no useful information across slices. To address this concern, 3-D DCT features are compared with 2-D DCT features on various synthetic textures, with and without discriminatory information across slices, as well as samples of real 3-D data with a low slice resolution. The samples of real 3-D data are extracted from low resolution CT scans of healthy and emphysemic human lungs. With these samples the distance between slices is 3mm whereas the distance between pixels within slices is about 0.7mm. Figure 1.4 shows sample slices from CT scans of a healthy lung and an emphysemic lung and enlarged 2-D samples of normal and emphysemic lung tissue.

Based on the experiments presented in this thesis, 3-D DCT features generally provide better classification accuracy than 2-D DCT and 3-D GMRF features. Despite the low slice resolution, the 3-D DCT features can more easily distinguish between normal and emphysemic lung tissue than either 2-D DCT or 3-D GMRF features. Based on these results, this thesis suggests using the 3-D DCT features for classifying texture in 3-D images even when the resolution between slices is low.

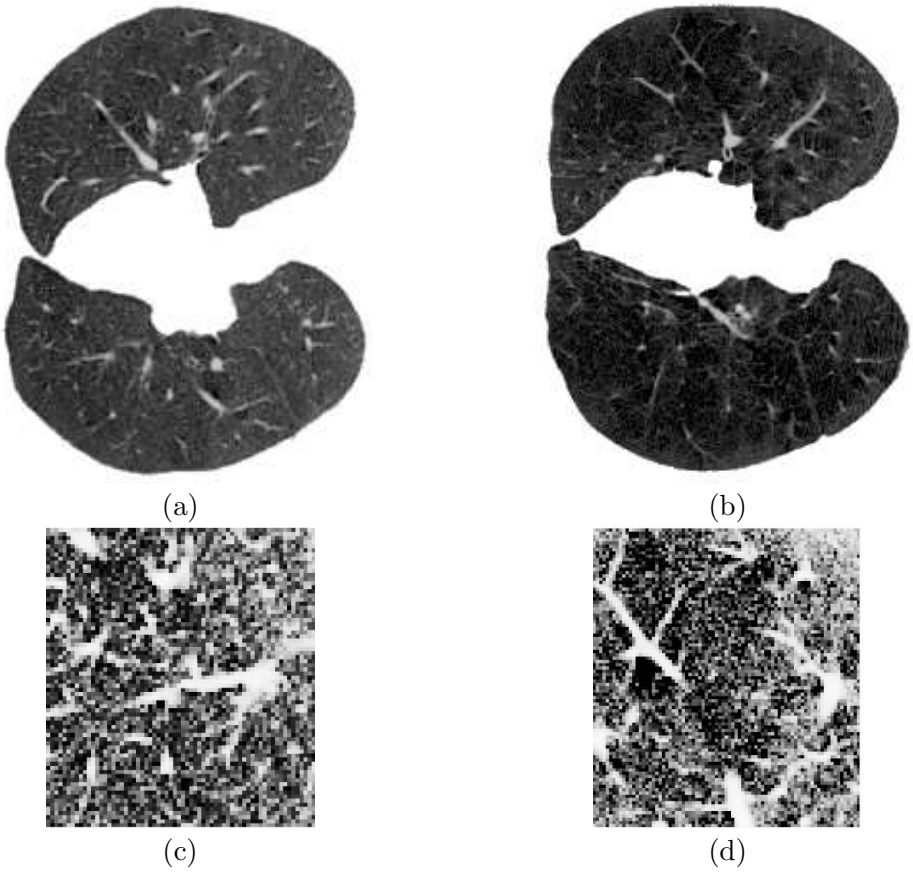


Fig. 1.4. Sample slices from CT scans of a healthy lung and an emphysemic lung. (a) healthy lung (b) emphysemic lung. (c) Enlarged sample of normal lung tissue. (d) Enlarged sample of emphysemic lung tissue. The white structures in both samples are blood vessels and airways. The texture of the emphysemic lung tissue appears to be finer, containing more high frequency detail than the sample of normal lung tissue.

Chapter 2

3-D Texture Classification

This chapter presents a novel approach to classifying texture in 3-D images using 3-D discrete cosine transform (DCT) features. This method is a 3-D extension of a popular 2-D method [18] which relies on multi-channel filtering. Using 2-D DCT filters in the 2-D multi-channel approach has already proved to be effective for 2-D texture analysis [18]. Since the 2-D DCT has been shown to be effective with 2-D texture and is less computationally complex than the 3-D DCT, this thesis compares the classification accuracy of 3-D DCT features with that of 2-D DCT features.

This chapter also introduces an existing 3-D texture approach which uses 3-D Gaussian Markov random field (GMRF) features. Like the 2-D DCT, 2-D GMRF features have proved to be useful for analyzing texture in 2-D images [3, 4]. Ranguelova and Quinn first extended the GMRF to 3-D for segmenting texture in 3-D images [20]. They demonstrated that 3-D GMRF features provide better segmentations of 3-D textured images than 2-D GMRF features.

The 3-D GMRF provides an different approach to characterizing texture than does the 3-D DCT. The 3-D GMRF is a model based approach where as the 3-D DCT is a frequency based approach. The 3-D DCT approach assumes texture to be characterized by the energy in predefined frequency channels. Texture classes are identified by their energy values in these channels. The 3-D GMRF fits textures to a 3-D GMRF model.

Texture classes are characterized by their 3-D GMRF model parameter values. This thesis compares classification results for 3-D DCT features with results using 3-D GMRF features.

2.1 3-D Texture Classification

This thesis is concerned with the task of classifying voxels in 3-D images based on the texture characteristics of the volumetric regions surrounding them. The key to accurate classification is for different textures to have significantly different texture feature values. Samples from the same texture class should have similar texture features values. The process of classifying a voxel, based on texture, consists of the following steps:

1. **Compute features** - Compute a vector of 3-D texture features for the voxel. The purpose of the feature vector is to characterize the texture in the area around the voxel.

2. **Classify features** - Find the texture class with the greatest likelihood of containing this voxel's vector of texture features. Assign the voxel to this texture class.

2.2 Filter-Based Texture Features

Since this thesis defines texture as repeated pixel or voxel patterns, it views texture as being essentially periodic. Texture classes are characterized by the pattern and pattern repetition frequencies. Significant random spacial and pixel noise causes some textures

to appear non-periodic. In order to capture the periodic nature of image texture, this thesis uses a frequency-based method for computing texture features. The effectiveness of these features is compared with features from a non frequency-based method which assumes image texture can be approximated by a Gaussian Markov random field.

A popular frequency-based method for extracting texture features from 2-D images uses 2-D filters. Figure 2.1 illustrates the process of classifying pixels in an image using this approach. First, a set of filters is applied to an image. Applying filters amplifies a band or channel of spacial frequencies while suppressing the rest of the spacial frequencies. Each filter amplifies a different channel of spacial frequencies. A local energy function is applied to each of the pixels in the filtered images. This function estimates the energy in a window around the pixel. The texture features are the local energies of the amplified channels. This approach to texture classification is easily extended to 3-D

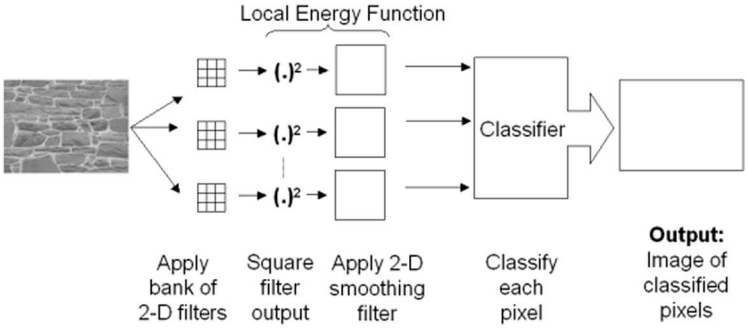


Fig. 2.1. 2-D filter-based approach to texture classification

by using 3-D filters and extending the local energy function to 3-D.

2.2.1 Local Energy Function

The purpose of a 3-D texture feature is to characterize the texture in a 3-D window around a voxel with a single value. A way to characterize a signal with a single value is by measuring its "size". Since 3-D images are 3-D signals that have finite energy, size may be measured by computing the *energy* of the image. Energy in a discrete 1-D signal can be defined as

$$E_f = \sum_{\imath=0}^{N-1} f(\imath)^2. \tag{2.1}$$

The magnitude of the signal is squared in order to rectify the signal so that the energy is greater than zero. This definition for energy is extended to 3-D by squaring and summing all voxels in the volume.

Since a volume may contain multiple textures, computing the energy over the entire volume is not generally useful for identifying the texture at a particular location in the volume. For this reason, energy is computed within a window around the voxel to be classified. This value is usually referred to as the *local energy*. In this thesis, the local energy at sample t in a 1-D signal is defined as

$$E_f(t) = \frac{1}{W} \sum_{\imath=-\lfloor W/2 \rfloor}^{\lfloor W/2 \rfloor} f(t+\imath)^2 \tag{2.2}$$

where W is odd and is the length of the window. Scaling the energy by $\frac{1}{W}$ computes the average energy per sample within the window. Equation 2.2 can be extended to 3-D by computing the average energy within a 3-D window instead of a 1-D window.

The local energy of a voxel is computed by the local energy function. In 3-D, the local energy function is implemented by squaring each voxel in the volume, then applying a 3-D smoothing filter to average the squared values within the windows around each voxel. The size of the 3-D smoothing filter defines the size of the 3-D window over which the local energy for a voxel is estimated. As Figure 2.2 illustrates in 1-D, if the length of the window matches the period of the signal, then the local energy for the signal will be the same when estimated at any sample in the signal. This situation is ideal for texture classification since every pixel in a texture region will have the same local energy value. Unfortunately, textures are often much more complicated than the signal in Figure 2.2(a). Defining a smoothing filter that matches the period of a particular texture class is a complex process and may not be possible or practical in many cases. Furthermore, the filter would need to match the periods of all the possible texture classes in order to be fully effective.

In the absence of a 3-D filter that matches the period of the texture, it is best to use as large a smoothing filter as is possible. As the size of the smoothing filter increases, more periods of the texture are covered, improving the estimation of the local energy. Unfortunately, while a large smoothing filter may be useful for texture classification, it can be a problem for texture segmentation. The smoothing filter must be smaller than the smallest, valid texture regions. If it is larger, small regions may not be detected.

Unfortunately, the size of the smallest, valid texture regions is not always known in practice.

For the textures used in this thesis, a $15 \times 15 \times 15$ mean filter was found to generally provide the best classification accuracy compared to other sizes of mean filters. For this reason, this 3-D filter will be the smoothing filter used in the local energy function.

2.2.2 3-D DCT Filter Bank

While energy is a way of describing a signal with a single value, it is possible for signals with different frequency content to have the same energy. This statement means that visually different textures can have the same local energy. Thus, local energy alone is not a good texture feature. Using this thesis's definition of texture, different texture will have characteristically different frequency spectrum. Either the frequencies in the repeated patterns will be different or the frequency of the repetition will be different. So, the effectiveness of the local energy can be improved by applying a bank of channel filters to the texture before computing the energy. If the textures are different, the local energy in at least some of these channels should be different.

The choice of filter bank determines which channels will be analyzed. Since various filter banks perform differently on the same textures [18], it appears some channels are more useful for classifying texture, or certain types of textures, than others. Without an accepted mathematical definition of texture, however, it is difficult to design a filter bank that will perform well with all types of texture. This situation has led to the use of many different types of filter banks for analyzing texture in 2-D images. Laws filters

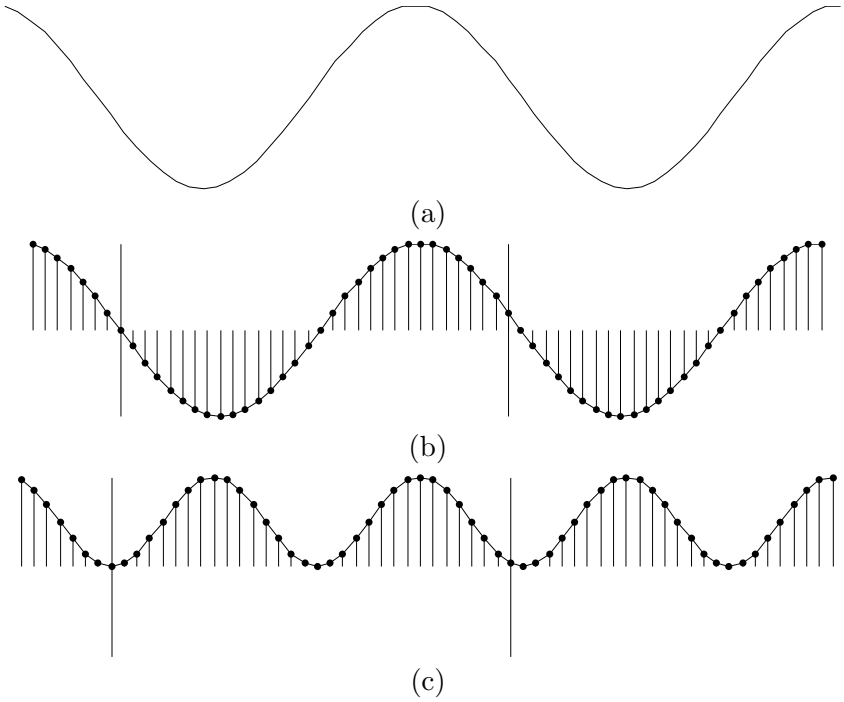


Fig. 2.2. (a) A sample 1-D signal. (b) Discrete signal with window matching period of signal. Sum of signal values in window is zero. (c) Signal has been squared. Sum of squared signal values in the window is positive. The energy within the window remains constant as the window shifts along the squared signal.

[14], Gabor filters [2, 7], optimized Gabor filters [6], quadrature mirror filters [17] and linear transform filters [27] are just some of the types of filters applied to 2-D texture.

Filters derived from the 2-D discrete cosine transform (DCT) have been shown to be effective for analyzing texture in 2-D images [27, 16, 26, 18]. Furthermore, Randen and Husøy [18] have found a bank of 2-D DCT filters to be superior to most of the popular 2-D texture classification methods including optimized Gabor filters and co-occurrence matrices. It was outperformed by only the most complex wavelet and quadrature mirror filter methods. For this reason, this thesis uses a bank of 3-D DCT filters for computing 3-D texture features. Although this thesis presents the first use of the 3-D DCT for 3-D texture classification, the 3-D DCT has already been proposed for compressing 3-D medical images [25] and video sequences [23, 29] where time is the third dimension. The 3-D DCT has also been used to watermark 3-D data [30].

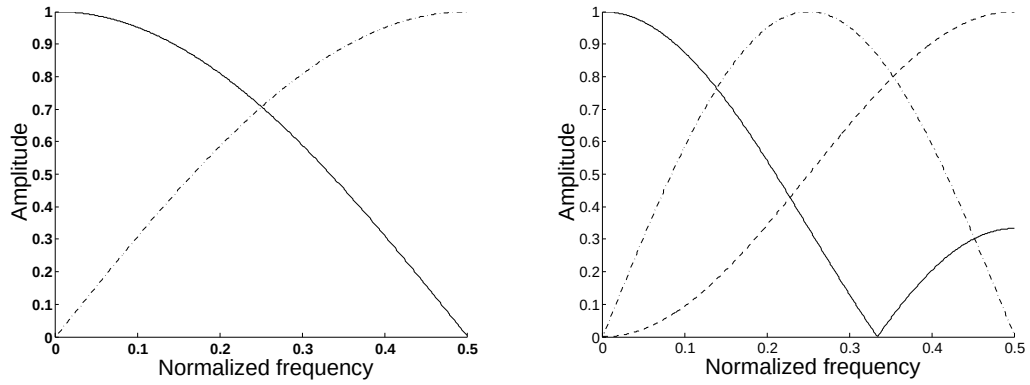
The DCT is a linear transform similar to the discrete Fourier transform. Unser [27] first proposed using linear transforms, including the DCT, for analysis of texture in 2-D images. Since the DCT is a separable transform, 3-D DCT filters can be constructed from three 1-D DCT filters and can be applied by convolving these 1-D filters over each dimension. The DCT was first reported by Ahmed et al [1] and defined for a 1-D signal of length N as

$$c(u) = \alpha(u) \sum_{\imath=0}^{N-1} f(\imath) \cos\left[\frac{(2\imath+1)u\pi}{2N}\right], \quad u = 0, 1, \ldots, N-1, \tag{2.3}$$

where

$$\alpha(u) = \begin{cases} \frac{1}{\sqrt{N}} & u = 0 \\ \sqrt{\frac{2}{N}} & u = 1, 2, \ldots, N-1 \end{cases} . \tag{2.4}$$

Figure 2.3 contains amplitude responses for 1-D DCT filters of length $N = 2$ and $N = 3$. This figure illustrates that DCT filters collectively cover the entire frequency spectrum.



(a) (b)

Fig. 2.3. Amplitude responses for 1-D DCT filters of length (a) $N = 2$ and (b) $N = 3$.

A 3-D filter bank is constructed from the 1-D DCT basis vectors. $N \times 1$ DCT basis vectors are created using

$$\Phi_u(\imath) = \begin{cases} \frac{1}{\sqrt{N}} & u = 0 \\ \sqrt{\frac{2}{N}} \cos\left[\frac{(2\imath+1)u\pi}{2N}\right] & u = 1, 2, ..., N-1 \end{cases} . \tag{2.5}$$

Each filter h_m in the filter bank is constructed using

$$h_m(\imath, j, k) = \Phi_u(\imath) \cdot \Phi_v(j) \cdot \Phi_w(k), \quad \imath, j, k = 0, 1, ..., N-1, \tag{2.6}$$

where $m = 9u + 3v + w$. Convolving the volume with all $N \times N \times N$ filters in the filter bank is equivalent to applying the 3-D $N \times N \times N$ DCT

$$c(u, v, w) = \alpha(u)\alpha(v)\alpha(w) \cdot \sum_{\imath=0}^{N-1} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} f(\imath, j, k) \cos\left[\frac{(2\imath+1)u\pi}{2N}\right] \cos\left[\frac{(2j+1)v\pi}{2N}\right] \cos\left[\frac{(2k+1)w\pi}{2N}\right] \tag{2.7}$$

around each voxel in the volume. Each filter output corresponds to a DCT transform coefficient.

As Figure 2.3 shows, increasing the length of the DCT filter increases the number of filters, but reduces the bandwidth of the filters. Analyzing images in narrower frequency bands could improve the effectiveness of the local energy function to characterize texture classes. However, increasing the size 3-D transform window, greatly increases the number of filters in the 3-D filter bank. Since each filter corresponds to a texture feature, increasing the number of filters increases the number of features. As a result, more training data may be needed to effectively train the classifier. Since 2-D DCT texture methods successfully employ 3×3 DCT filters [27, 16, 26, 18], the 3-D DCT filter bank in this thesis consists of 27 filters derived from the $3 \times 3 \times 3$, 3-D DCT.

2.3 Classifier

The classifier assigns labels to voxels based on their texture features. This thesis uses a Bayes classifier [5] to classify the texture features. The classifier is implemented using the following discriminant functions

$$g_\imath(\mathbf{x}) = \ln p(\mathbf{x}|\omega_\imath) + \ln P(\omega_\imath), \tag{2.8}$$

where $\mathbf{x}$ is the feature vector to be classified. The feature vector $\mathbf{x}$ is assigned to texture class $\omega_\imath$ if $g_\imath(\mathbf{x}) > g_j(\mathbf{x}), \forall \imath, j$ such that $\imath \neq j$.

For simplicity, texture features are assumed to obey a Gaussian distribution. This assumption appears to be reasonable after observing histograms of 2-D DCT, 3-D DCT and 3-D GMRF features computed for textures used in this thesis. The feature histograms are unimodal and approximately Gaussian in shape. Using a multi-variate Gaussian distribution leads to the following definition for the discriminant

$$g_\imath(\mathbf{x}) = \mathbf{x}^t \mathbf{W}_\imath \mathbf{x} + \mathbf{w}_\imath^t \mathbf{x} + b_\imath \tag{2.9}$$

where

$$\mathbf{W}_\imath = -\frac{1}{2}\mathbf{\Sigma}_\imath^{-1}, \tag{2.10}$$

$$\mathbf{w}_\imath = \mathbf{\Sigma}_\imath^{-1}\mu_\imath, \tag{2.11}$$

and

$$b_\imath = -\frac{1}{2}\mu_\imath^t \mathbf{\Sigma}_\imath^{-1}\mu_\imath - \ln|\mathbf{\Sigma}_\imath| + \ln P(\omega_\imath). \tag{2.12}$$

μ_i and $\mathbf{\Sigma}_i$ are the mean feature vector and feature covariance matrix for texture class ω_i. The mean vectors and covariance matrices are estimated from a separate set of training data. Texture features are computed for each voxel in the training volumes. From these features, average feature vectors and feature covariance matrices are computed.

2.4 3-D DCT Algorithm

Figure 2.4 illustrates the algorithm proposed by this thesis for classifying texture in volumetric images. This algorithm classifies each voxel in a given volume. The first step in the algorithm is to apply the $3 \times 3 \times 3$, 3-D DCT filters. Since the DCT is a separable transform, the 3-D filters could be applied by applying 1-D DCT filters over each dimension. The filtering stage generates 27 filtered volumes. The local energy function is applied to each of these volumes.

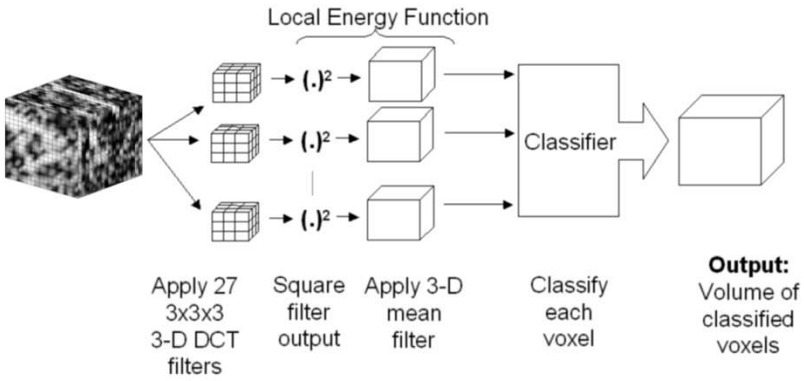


Fig. 2.4. 3-D DCT filter-based approach to texture classification

The local energy is computed by squaring each voxel, then applying a $15 \times 15 \times 15$ mean filter to each of the volumes. At this point we have 27 volumes of texture features which corresponds to a vector of 27 texture features for each voxel in the original volume.

The original voxels are now classified by their feature vectors using the classifier from Section 2.3. The final result is a volume whose voxel values are the texture class labels for the voxels in the original volume.

2.5 2-D DCT vs 3-D DCT

The 2-D DCT has already proved to be effective for analyzing texture in 2-D images [27, 16, 26, 18]. If 3-D images are processed as a series of unrelated images, a 2-D DCT classification scheme can be applied to each slice. Since an $N \times N \times N$, 3-D DCT has an order of magnitude more features than a $N \times N$, 2-D DCT, it is worth questioning whether 3-D DCT features are better than 2-D DCT features for classifying 3-D texture.

The texture features in both methods are local energy values computed on filtered version of the original volume. From Section 2.2.1, we see that this value is the average squared voxel value estimated over a window of voxels around the voxel to be classified. The 2-D DCT method uses a $15 \times 15 \times 1$ window and the 3-D DCT method uses a $15 \times 15 \times 15$ window. As explained in Section 2.2.2, using a larger window should result in better, more consistent estimates of the local energy. Since a texture class is characterized by its average local energy values, improving estimates of the local energy should improve classification. Thus, using a $15 \times 15 \times 15$ window for computing the local energies instead of the smaller $15 \times 15 \times 1$ window should improve classification accuracy.

However, the larger window could be a problem in the context of texture segmentation, if the texture regions are relatively small and only span a few slices. In this case, it might be better to use a window that only includes a few slices or simply use 2-D DCT features.

Since the DCT is a separable transform, an $N \times N \times K$, 3-D DCT can be computed by first applying a $N \times N$, 2-D DCT to each slice in a volume and then applying a $1 \times K$, 1-D DCT across the slices. Thus, an $N \times N$, 2-D DCT and an $N \times N \times N$, 3-D DCT use the same 2-D frequency channels. The only difference is the number of channels used to analyze the spacial frequencies across slices. The 3-D DCT uses N channels, whereas the 2-D DCT only uses one channel that includes all frequencies. Since 3-D DCT divides the frequencies across slices in N channels that collectively cover the frequency spectrum, 3-D can better isolate discriminatory frequencies, resulting in local energy values that better characterize the texture classes.

The fact that $N \times N \times N$, 3-D DCT has more texture features than $N \times N$, 2-D DCT allows it to more effectively distinguish between texture classes than 2-D DCT. Unfortunately, while adding more features can improve classification accuracy, the addition of new features can increase the need for more training data to adequately train the classifier. This problem is referred to as the "curse of dimensionality." Given sufficient training, 3-D DCT features should provide better classification of 3-D images than 2-D DCT features. However, 2-D DCT features may provide better classification accuracy when training data is insufficient for the extra 3-D DCT features.

2.6 3-D GMRF Texture features

This thesis not only compares classification results of 3-D DCT texture features to 2-D DCT texture features, it also compares results with features from a current 3-D method which uses a 3-D GMRF model. The features in this method are model parameters obtained by fitting a 3-D image to a 3-D GMRF model. These parameters define the amount of correlation between voxel pairs within the neighborhood of voxels defined by the GMRF order. The neighborhood of voxels is the collection of voxels surrounding the voxel to be classified. This collection of voxels only includes those who satisfy the 3-D extension of the Maximum Allowable Square (MAS) [21] rule. The size of the neighborhood is determined by the *order* of the GMRF. Neighborhood size increases as the order of the GMRF increases.

A more detailed description of the 3-D GMRF is found in Appendix A. The algorithm used for estimating GMRF features is presented in Section A.3. For the sake of comparison, 3-D GMRF features are classified using the same Bayes classifier used with 3-D DCT features as described in Section 2.3.

Chapter 3

Experimental Results

The 3-D DCT texture classification scheme presented in this thesis was tested on 3-D synthetic and natural textures. Details describing these textures and how they were made are given in Appendix B. Classification results for 3-D DCT features are compared with results using 2-D DCT features and 3-D GMRF features.

3.1 Methodology

All texture methods used a Bayesian classifier, which was implemented using discriminant functions as described in Section 2.3. The 2-D DCT method was applied to each slice in the sample volumes, treating them as sets of unrelated 2-D images. All voxels were classified except those along the boundaries of the volumes where the smoothing and estimation windows did not fit. For the sake of consistent comparison, voxels in the first and last slices that were not classified by 3-D methods were ignored by the 2-D DCT method. Results for all methods are based on the voxel classification error for the same sets of voxels.

The histograms of all sample volumes were equalized. This step was done to ensure that the textures could not be distinguished by first order gray level statistics. The volumes were then zero meaned by computing the mean voxel value for each volume and subtracting it from the value of each voxel. For all texture classes except the lung set,

there are 60, $20 \times 20 \times 20$ sample volumes. The lung texture classes have 60, $18 \times 18 \times 18$ sample volumes each. The lung samples had to be smaller in order to find homogeneous 3-D texture regions in the low resolution CT image data. Classification results from [28] were used as ground truth for the image data. Samples were selected from 3-D regions where at least 99% of the voxels in the region belonged to the same texture class. In all cases, half the number of sample volumes were used for training and the other half for testing.

The effectiveness of 3-D DCT features was compared with features from multiple 3-D GMRF models. A second order 3-D GMRF was used, since results in [20] were obtained using a second order 3-D GMRF. Because a third order 3-D GMRF has a neighborhood with the same dimensions as a $3 \times 3 \times 3$ filter, this thesis also uses a third order 3-D GMRF model. A third order 3-D GMRF model has 13 features. Since the 3-D DCT method used in this thesis has 27 features, classification accuracy is also compared with that obtained with a fifth order 3-D GMRF which has 28 features. Since Ranguelova and Quinn [20] use a $15 \times 15 \times 3$ window for computing features and 3-D DCT features are computed over a $15 \times 15 \times 15$ window, both the second, third and fifth order 3-D GMRF models use a $15 \times 15 \times 15$ window. Not only does this selection lead to an estimation window equal in size to that used by 3-D DCT, increasing the depth of the window from 3 to 15 improves classification accuracy [19].

3.2 Results

Classification error rates for each volume are presented in Table 3.1. From these results we can see that 3-D DCT features provide much better classification accuracy

than 2-D DCT. The 3-D DCT has the lowest classification error for the lung texture set and generally provides better classification accuracy than 3-D GMRF. Results from the lung texture set show that even with relatively low slice resolution, the 3-D DCT provides better discrimination between normal lung tissue and emphysemic lung tissue than 2-D DCT.

Table 3.1. Classification Error rates

Texture set	3D GMRF			2D DCT	3D DCT
	order=2	order=3	order=5		
GMRF-2D	0.0%	0.0%	0.0%	0.1%	0.0%
GMRF-3	0.0%	0.0%	0.0%	8.1%	0.0%
GMRF-5	41.8%	44.8%	8.3%	38.0%	1.3%
cox-xy	1.7%	2.7%	4.5%	0.1%	0.1%
cos-xyz	0.0%	0.3%	0.0%	0.2%	0.0%
cox-z-xy	0.4%	1.5%	0.9%	42.4%	0.0%
lung	16.9%	20.8%	27.6%	24.9%	7.9%

3.2.1 2-D DCT vs 3-D DCT

From the results in Table 3.1, 3-D DCT clearly provides better classification accuracy than 2-D DCT. Not surprisingly, 2-D DCT performs best with texture sets containing little or no information across slices. These sets include GMRF-2D and cos-xy. These texture sets essentially consist of volumes that are a series 2-D textured images. However, even with these texture sets 3-D DCT has equal or better classification

accuracy. It is worth noting that 2-D DCT features did perform rather well on the cos-xyz texture set. Unlike cos-xy, the texture classes in this set contain discriminating frequencies in z (across slices) as well as in the xy plane (within slice). Although 2-D DCT features are blind to the frequencies across slices, it appears that there is enough of a difference in the frequencies in the xy plane for 2-D DCT features to still be useful.

The advantage of using a 3-D method can be seen with texture sets containing distinctive information across slices. With these texture sets, 3-D DCT significantly outperforms 2-D DCT. The most dramatic example of 3-D's superiority is cos-z-xy. This texture set contains two texture classes that have identical frequencies in the xy plane, but different frequencies in z. These texture classes are almost identical as far as 2-D DCT is concerned, but 3-D DCT easily discriminates between the two texture classes.

From Table 3.1 we see that 3-D DCT has significantly lower classification error than 2-D DCT on the lung texture set. This result is interesting since these samples are extracted from low resolution CT scan where the between slice resolution is about four times lower than the pixel resolution within slice. It appears that there is some distinguishing information across slices, despite the relatively low resolution between slices.

3.2.2 GMRF estimation order

The experiments in this thesis used different orders for 3-D GMRF. No particular choice for GMRF order yielded the best classification accuracy in all cases. In most cases, the second order 3-D GMRF was as good as or better than the third or fifth order GMRFs. This result is somewhat surprising since second order features are a subset of

higher order GMRFs. It demonstrates the pitfall of using more features. More features can lead to better classification accuracy as can be seen with the GMRF-5 texture set. However, increasing the number of features complicates classifier training and can lead to worse results when the training set is small. This situation is illustrated with the lung textures. With this texture set, classification error increases as the 3-D GMRF order increases.

3.2.3 3-D DCT vs. 3-D GMRF

The 3-D DCT generally matched or exceeded the performance of the 3-D GMRF. The GMRF-5 texture set is a dramatic example of the 3-D DCT features outperforming 3-D GMRF features. The GMRF-5 textures are fifth order, 3-D GMRFs with identical third order parameters. Because of this parameter selection, the poor performance of the second and third order 3-D GMRFs is not surprising. However, it is surprising that 3-D DCT features were more effective than fifth order 3-D GMRF features. The relative poor performance for fifth order 3-D GMRF features on this texture set is probably the result of not having enough training data. With this texture set about half of the fifth order 3-D GMRF features are useless since both texture classes have the same third order parameters. It is possible that with more training data, the classifier could be trained to ignore the useless features. However, it is still worth noting that given the same training data, 3-D DCT features lead to better classification accuracy. This result implies that 3-D DCT features seem to capture texture information more efficiently than 3-D GMRF features, requiring less training data. This observation is further illustrated with the lung texture set which had less training data available than the synthetic texture sets. With

this texture set, 3-D DCT features significantly outperformed the 3-D GMRF features. This result implies that 3-D DCT features are a better choice for discriminating between normal and emphysemic lung tissue.

Chapter 4

Conclusions and Future Work

This thesis presents a new method for classifying texture in 3-D images using filters derived from 3-D DCT basis vectors. This method usually provides better classification accuracy than 2-D DCT and 3-D GMRF. Furthermore, it is more effective at discriminating between normal and emphysemic lung tissue in low resolution CT scans. These results demonstrate the usefulness of using the 3-D DCT for classification of texture in 3-D images.

While this thesis introduces the use of 3-D DCT filters for 3-D texture classification and demonstrates their effectiveness, there is still room for further investigation. For example, it would be interesting to use 3-D DCT features for segmenting texture in 3-D images as opposed to classifying it.

Although 3-D DCT features generally resulted in low classification error for the textures in this thesis, there are a some changes that can be made to improve their effectiveness in texture classification. The 3-D DCT method in this thesis uses 27 features. Reducing the number of features, reduces the complexity of the classifier which can reduce the amount of training data needed. The number of features could be reduced by mapping the features to a lower dimensional feature space. Another option would be to find and eliminate useless features. Classification accuracy may be improved further

by replacing the $15 \times 15 \times 15$ smoothing filter in the local energy function with a more sophisticated smoothing filter that is optimized for the given textures.

Another option for future work would be to apply the 3-D method presented in this thesis to different types of 3-D textured images. For instance, it would be interesting to examine the usefulness of 3-D DCT features on active textures. These textures are essentially 2-D textures that change over time. In this case, the 3-D images are video sequences and time is the third dimension instead of space.

Appendix A

3-D Gaussian Markov Random Fields

The 3-D texture classification method presented in this thesis was tested on a series of 3-D texture samples generated using the 3-D Gaussian Markov Random Field (GMRF) model [20]. Markov random fields (MRF) provide a natural definition of texture. Ranguelova and Quinn [20] proposed analyzing and synthesizing 3-D texture using a 3-D GMRF model.

A.1 3-D Markov Random Field

A 3-D random field is a family of random variables $X = \{X_1, X_2, ..., X_N\}$ arranged in a 3-D lattice of sites $\Omega = \{t = (\imath, j, k) | 1 \leq \imath \leq N_1, 1 \leq j \leq N_2, 1 \leq k \leq N_3\}$ where $N = N_1 \times N_2 \times N_3$. A realization of the random field is the event when $X = \{X_t, t \in \Omega\}$ takes the values $x = \{x_t, t \in \Omega\}$. Each possible realization x is called a configuration of X.

Markov random fields require a neighborhood of surrounding lattice sites η_t, to be defined for each site $t \in \Omega$. The size of η_t is determined by the order of MRF. A qth order MRF neighborhood is the set of sites $\eta_t = \{s = t + r, r \in \eta(q)\}$ where $\eta(q)$ is a set of displacement triples constructed using the 3-D extension [20] of the Maximum Allowable Square(MAS) [21] rule:

$$\eta_q = \{(\imath, j, k) | (\imath^2 + j^2 + k^2 \leq \text{MAS}(q)), \imath \in \mathbb{Z}, j \in \mathbb{Z}, k \in \mathbb{Z}\}. \tag{A.1}$$

MAS(q) is the qth element in the ordered set $\{(q_1^2 + q_2^2 + q_3^2)|q_1, q_2, q_3 \in \mathbb{Z}\}$. The neighborhood η_t is symmetric and can be partitioned such that $\eta_t = \eta_t^+ \cup \eta_t^-$ where $\eta_t^- = \{(-i, -j, -k)|(i, j, k) \in \eta_t^+\}$. Toroidal boundary conditions are assumed for sites toward the edges of the 3-D lattice [3].

A random field is a Markov random field if it satisfies the following two conditions:

$$P(x) > 0, \qquad \forall x \in \mathbb{X} \qquad (positivity) \tag{A.2}$$

$$P(x_t|x_{\Omega-\{t\}}) = P(x_t|x_{\eta_t}), \qquad \forall t \in \Omega \qquad (Markovianity) \tag{A.3}$$

where $\mathbb{X}$ is the set of all configurations, $\Omega - \{t\}$ is the set of all lattice sites other than point t, $x_{\Omega-\{t\}}$ is the configuration x excluding the value at site t and $x_{\eta_t} = \{x_r|r \in \eta_t\}$ is the 3-D lattice of values in the neighborhood surrounding lattice site t.

The positivity condition requires all configurations to have some chance of occurring. The Markovianity condition characterizes the interactions between lattice sites. A lattice site t does not interact with lattice sites outside of its neighborhood η_t. Due to the Markovianity property, almost any texture can be viewed as a MRF given a large enough neighborhood.

A.2 3-D Gaussian Markov Random Field

If the random variables in the MRF obey a Gaussian distribution, the random process X can be modeled using simultaneous auto-regression (SAR). With this method,

each random variable X_t is modeled using

$$X_t = \sum_{r \in \eta_t^+} \theta_{t,r}(X_{t+r} + X_{t-r}) + E_t, \tag{A.4}$$

where $\theta_{t,r}$ are the auto-regression parameters and $E_t = N(0, \sigma^2)$ is Gaussian noise. The parameters are symmetric $\theta_{t,r} = \theta_{r,t}$ and describe the amount of correlation between the random variables X_{t+r} and X_{t-r}. Equation A.4 assumes X_t is a zero-mean random process.

A.3 3-D GMRF Parameter Estimation

The 3-D GMRF texture features for a voxel t are estimations of the SAR parameters $\theta_{t,r}$ for a window surrounding t. Ranguelova and Quinn use the method of Least Squares to estimate these parameters [20]. This estimation is accomplished by solving equation A.4 for each site x_r in a sub-lattice ω_t around site t. These equations are first written in vector-matrix form as:

$$x_r = y_r^T \theta_r + e_r, \tag{A.5}$$

where $y_r = [x_s + x_{-s}, s \in \eta_r^+]^T$ and $\theta_r = [\theta_{r,s}, s \in \eta_r^+]^T$. The parameters $\theta_{r,s}$ are assumed to be constant over ω_t so $\theta_t = \theta_r$. Writing equation A.5 for all $r \in \omega_t$ leads to

$$x_t = Y_t \theta_t + e_t, \tag{A.6}$$

where matrix $Y_t = [y_r^T, r \in \omega_t]$ and vectors $x_t = [x_r, r \in \omega_t]^T$, $e_t = [e_r, r \in \omega_t]^T$ are created by raster scanning ω_t. The goal when estimating θ_t is to minimize the sum of the squared errors $e_t^T e_t$. The Least Squares solution that minimizes $e_t^T e_t$ is

$$\theta_t^* = (Y_t^T Y_t)^{-1} Y_t^T x_t. \tag{A.7}$$

Appendix B

3-D Texture samples

This chapter contains information describing the creation of the synthetic and natural 3-D texture samples used in this thesis.

B.1 3-D GMRF textures

Several 3-D texture sets used in this thesis were generated using 3-D GMRF model. Table B.1 contains the parameters used to generate the texture classes. Figure B.1 shows consecutive slices from sample volumes in each texture class.

B.1.1 3-D GMRF Synthesis

The 3-D GMRF textures used in this thesis were created with the following algorithm developed by Ranguelova and Quinn [20].

1. $Y := rand(N_1, N_2, N_3)$ *Generate* $N_1 \times N_2 \times N_3$ *array* $Y \sim N(0, \sigma^2)$
2. $Y := \mathcal{F}[Y]$ *Apply 3-D FFT*
3. **for** $\imath := 0$ **to** $N_1 - 1$
4. **for** $j := 0$ **to** $N_2 - 1$
5. **for** $k := 0$ **to** $N_3 - 1$
6. $A(\imath, j, k) := B(1, (j + (\imath - 1)N_2) + (k - 1)N_1N_2)$

end for

end for

end for

7. $A := \mathcal{F}[A]$ *Apply 3-D FFT*

8. **for** $\imath := 0$ **to** $N_1 - 1$

9. **for** $j := 0$ **to** $N_2 - 1$

10. **for** $k := 0$ **to** $N_3 - 1$

11. $Y := Y(\imath, j, k)/\sqrt{A(\imath, j, k)}$

end for

end for

end for

12. $X := \mathcal{F}^{-1}[Y]$ *Apply 3-D inverse FFT*

This algorithm generates a zero-mean 3-D GMRF. The matrix B is constructed by:

$$B_{s,t} = \begin{cases} 1 & \text{if } s = t \\ -\theta_{s,t} & \text{if } s \in \eta_t \\ 0 & \text{otherwise} \end{cases} . \tag{B.1}$$

For X to be a valid 3-D GMRF, the parameters $\theta_{s,t}$ must satisfy the following criterion:

$$\sum_{s \in \eta_t^+} |\theta_{s,t}| < \frac{1}{2}. \tag{B.2}$$

B.1.2 3-D GMRF-2D textures

This set consists of two classes of textures generated using a third order 3-D GMRF. All the parameters specifying correlation between voxels within the same slice are non-zero. Parameters corresponding to voxel pairs from different slices are set to zero. Using these parameters, 3-D GMRF textures are generated with out any correlation between slices. These textures are essentially collections of 2-D GMRF images.

B.1.3 3-D GMRF-3 textures

This set consists of two classes of textures generated using a third order 3-D GMRF model. Parameters were randomly generated for each class.

B.1.4 3-D GMRF-5 textures

The textures in this set were generated using an fifth order 3-D GMRF model. This set contains two classes. The third order parameters of both classes are identical. Only the parameters added by fourth and fifth order GMRF differ between the two classes.

B.2 Cos textures

A convenient method of generating 3-D textures is a finite Fourier cosine series model. Gluckman used finite Fourier cosine series to generate texture in 2-D images [9]. With this method a 3-D volume f is constructed from a linear combination of cosines

Table B.1. Parameters used to generate 3-D GMRF textures

GMRF parameter	GMRF-2D		GMRF-3		GMRF-5	
	Class 1	Class 2	Class 1	Class 2	Class 1	Class 2
$\theta_{1,0,0}$	0.3105	0.1609	-0.0071	0.0289	-0.0059	-0.0059
$\theta_{0,1,0}$	-0.0194	-0.2045	-0.0726	0.0963	-0.0265	-0.0265
$\theta_{0,0,1}$	0.0000	0.0000	0.0117	0.0337	-0.0491	-0.0491
$\theta_{-1,1,0}$	0.1517	0.0733	-0.0478	-0.0366	0.0220	0.0220
$\theta_{1,1,0}$	0.0084	-0.0513	0.0641	0.0217	-0.0116	-0.0116
$\theta_{0,-1,1}$	0.0000	0.0000	-0.0365	-0.0574	0.0073	0.0073
$\theta_{-1,0,1}$	0.0000	0.0000	0.0239	-0.0011	0.0052	0.0052
$\theta_{1,0,1}$	0.0000	0.0000	0.0099	-0.0027	0.0005	0.0005
$\theta_{0,1,1}$	0.0000	0.0000	-0.0418	0.0000	-0.0224	-0.0224
$\theta_{-1,-1,1}$	0.0000	0.0000	-0.0983	-0.0181	-0.0211	-0.0211
$\theta_{1,-1,1}$	0.0000	0.0000	-0.0027	0.0623	-0.0083	-0.0083
$\theta_{-1,1,1}$	0.0000	0.0000	-0.0458	-0.1067	-0.0264	-0.0264
$\theta_{1,1,1}$	0.0000	0.0000	0.0278	0.0244	-0.0235	-0.0235
$\theta_{2,0,0}$	—	—	—	—	0.0328	-0.0091
$\theta_{0,2,0}$	—	—	—	—	0.0012	-0.0233
$\theta_{0,0,2}$	—	—	—	—	-0.0271	0.0137
$\theta_{-2,1,0}$	—	—	—	—	-0.0009	0.0411
$\theta_{2,1,0}$	—	—	—	—	-0.0251	-0.0151
$\theta_{-1,2,0}$	—	—	—	—	-0.0301	-0.0234
$\theta_{1,2,0}$	—	—	—	—	-0.0058	-0.0068
$\theta_{0,-2,1}$	—	—	—	—	0.0212	0.0183
$\theta_{-2,0,1}$	—	—	—	—	0.0029	-0.0236
$\theta_{2,0,1}$	—	—	—	—	0.0146	-0.0331
$\theta_{0,2,1}$	—	—	—	—	-0.0260	-0.0033
$\theta_{0,-1,2}$	—	—	—	—	-0.0103	-0.0018
$\theta_{-1,0,2}$	—	—	—	—	-0.0058	0.0134
$\theta_{1,0,2}$	—	—	—	—	-0.0270	-0.0164
$\theta_{0,1,2}$	—	—	—	—	-0.0294	-0.0041

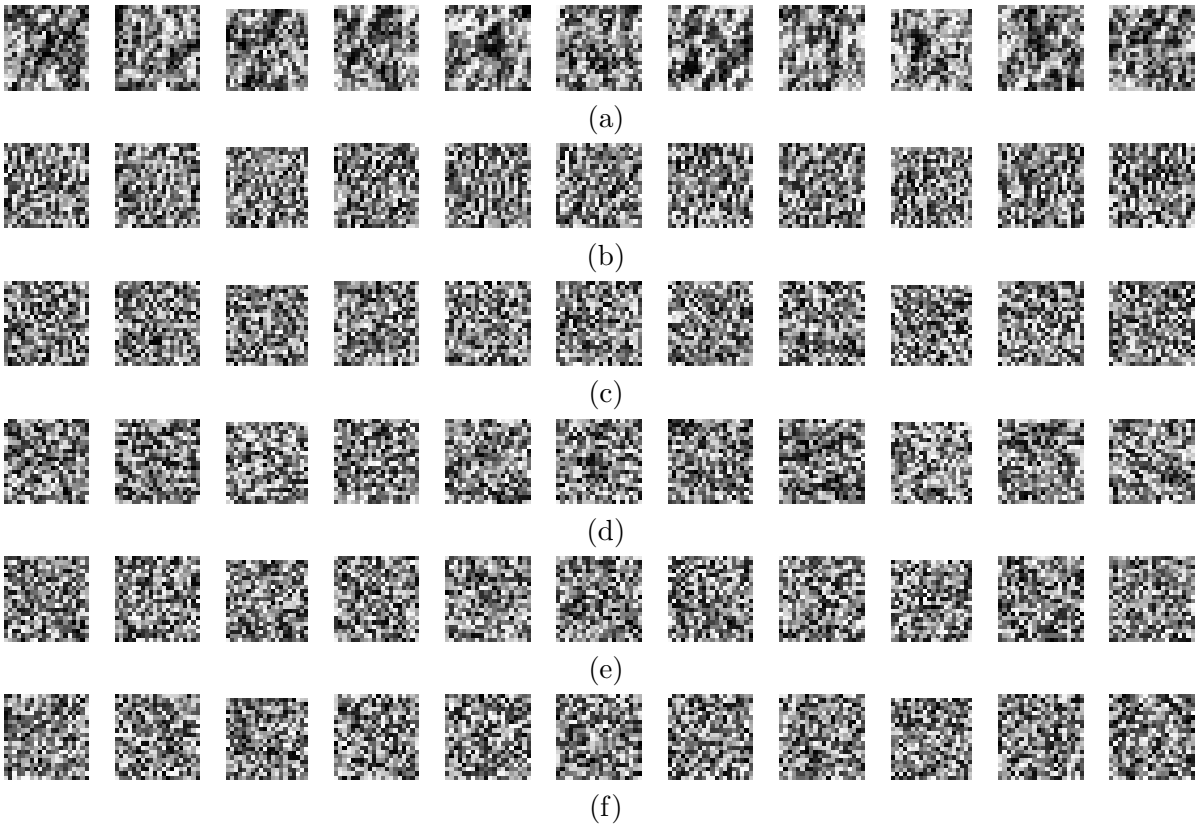


Fig. B.1. Consecutive slices from sample volumes belonging to GMRF texture sets. (a) GMRF-2D class 1 (b) GMRF-2D class 2 (c) GMRF-3 class 1 (d) GMRF-3 class 2 (e) GMRF-5 class 1 (f) GMRF-5 class 2.

rotated in 3-D,

$$f(x,y,z) = \sum_{\imath=1}^{n} c_i \cos(2\pi(u_i x + v_i y + w_i z) + \phi_i) \tag{B.3}$$

where u_i, v_i, w_i specify the frequency along x, y and z axes, c_i is the weight and ϕ_i is the phase angle of the $\imath$th cosine.

Three 3-D texture sets, cos-xy, cos-xyz and cos-z-xy, were created using this model. Cos-xy has different frequencies in the x and y, but no frequencies across slices. This texture is essentially a series of unrelated 2-D images. Cos-xyz has different frequencies in x, y, and z. Cos-z-xy has identical frequencies in the x and y, but different frequencies in the z. Table B.2 contains the frequencies used to define the texture classes in each. These frequencies are defined for a $21 \times 21 \times 21$ volume that is tiled and trimmed to create a final volume of any desired dimensions. Gaussian noise, $N(0, 0.05)$ was added to the frequencies used to generate the texture samples. Noise was only added to non-zero frequencies. This was done mainly to ensure that cos-xy textures did not have any correlation between slices. Gaussian noise was also added to the sample volumes. Voxel values for the sample volumes ranged from -1 to 1 and the additive noise was specified by $N(0, 0.05)$. Figure B.2 shows consecutive slices from sample volumes in each texture class.

B.3 Lung textures

The 3-D DCT was also tested on samples of natural texture. These samples are extracted from low resolution CT scans of human lungs. The CT image data was provided by Dr. Joseph M. Reinhardt from the Department of Biomedical Engineering

Table B.2. (u, v, w) Frequencies used for Cos textures

cos-xy		cox-xyz		cox-z-xy	
Class 1	Class 2	Class 1	Class 2	Class 1	Class 2
(2, 0, 0)	(-3, 0, 0)	(2, 0, 0)	(-3, 0, 0)	(2, 0, 0)	(2, 0, 0)
(0, -4, 0)	(0, 5, 0)	(0, -4, 0)	(0, 5, 0)	(0, -4, 0)	(0, -4, 0)
(0, 0, 0)	(0, 0, 0,)	(0, 0, 2)	(0, 0, 5)	(0, 0, 2)	(0, 0, 5)

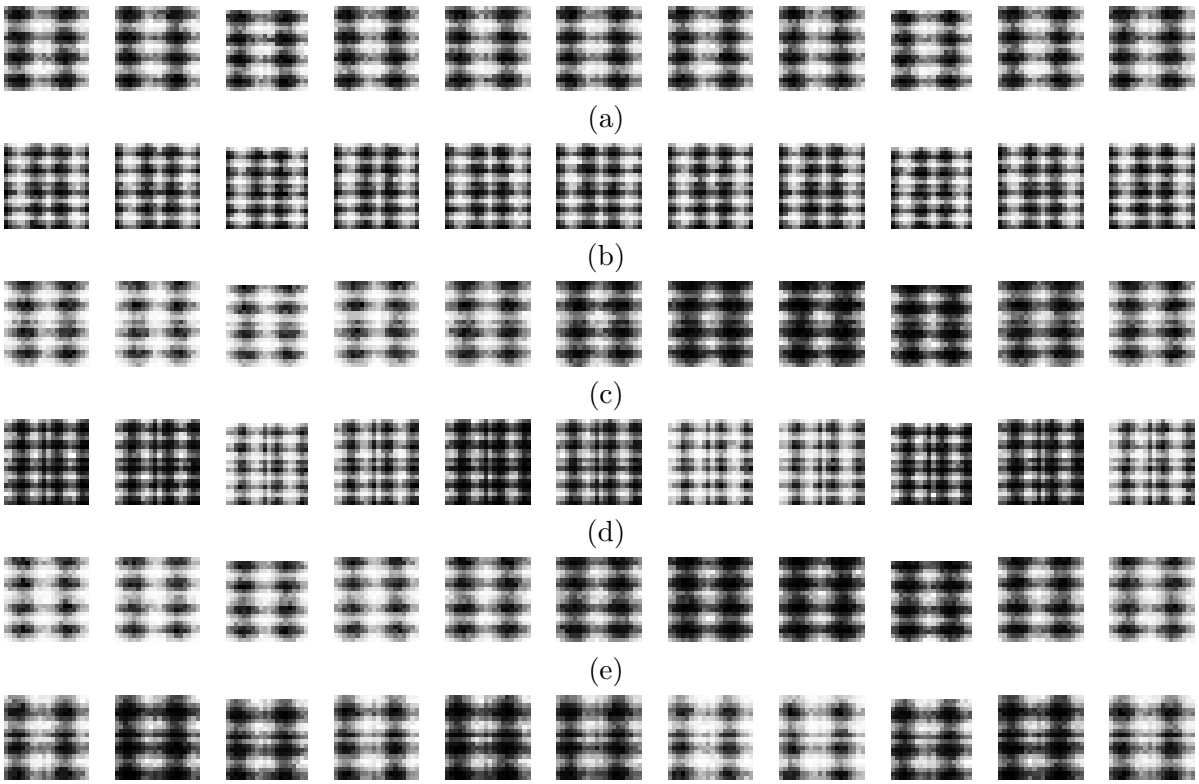


Fig. B.2. Consecutive slices from sample volumes belonging to cos texture sets. (a) cos-xy class 1 (b) cos-xy class 2 (c) cos-xyz class 1 (d) cos-xyz class 2 (e) cos-z-xy class 1 (f) cos-z-xy class 2.

at the University of Iowa. With these images, there is 3mm between slices compared to about 0.7mm between pixels within slices. Figure 1.4 contains sample slices from two of the original CT scans. The texture classes are healthy lung tissue and emphysemic lung tissue. The texture samples were taken from lung images used in [28]. Emphysema samples were extracted from CT scans of four patients diagnosed with emphysema. Normal samples were extracted from CT scans of two patients diagnosed as healthy. Classification results from [28] guided the selection of the samples and were used as ground truth for classification results in this thesis. Figure B.3 show consecutive slices from samples of normal and emphysemic texture.

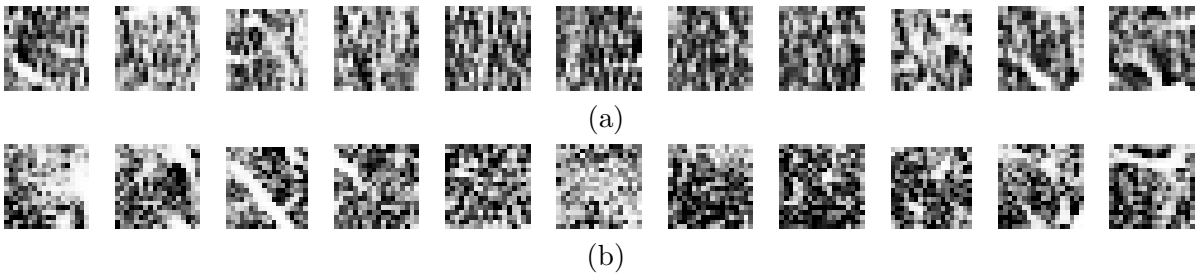


Fig. B.3. Samples slices from lung texture set. (a) normal (b) emphysema.

References

[1] N. Ahmed, T. Natarajan, and K.R. Rao. Discrete cosine transform. *IEEE Trans. on Computers*, C-23:90–93, 1974.

[2] A.C. Bovik, M. Clark, and W.S. Geisler. Multichannel texture analysis using localized spacial filters. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12:55–73, 1990.

[3] R. Chellappa. Two-dimensional discrete gaussian markov random field models for image processing. *Progress in Pattern Recognition 2*, pages 79–112, 1985.

[4] R. Dubes and A. Jain. Random field models in image analysis. *Journal of Applied Statistics*, 16(2):131–164, 1989.

[5] R.O. Duda, P.E. Hart, and Stork D.G. *Pattern Classification*. Wiley, New York, second edition, 2001.

[6] D.F. Dunn and W.E. Higgins. Optimal gabor filters for texture segmentation. *IEEE Trans. Image Processing*, 4:947–964, July 1995.

[7] F. Farrokhnia and A.K. Jain. A multi-channel filtering approach to texture segmentation. *Proc. IEEE Comp. Soc. Computer Vision and Pattern Recognition Conf. (CVPR)*, pages 364–370, 1991.

[8] M. Fernández, A. Mavilio, and M. Tejera. Texture segmentation of a 3d seismic section with wavelet transform and gabor filters. *15th Internation Conf. On Pattern Recognition*, 3:354–357, 2000.

[9] J. Gluckman. On the use of marginal statistics of subband images. *Proc. IEEE Intl. Conf on Computer Vision*, 2003.

[10] K. Jafari-Khouzani, H. Soltanian-Zadeh, K. Elisevich, and S. Patel. Comparison of 2d and 3d wavelet features for tle lateralization. In *Proc. of SPIE Medical Imaging 2004: Physiology, Function, and Structure from Medical Images*, volume 5369, pages 593–601, 2004.

[11] V.A. Kovalev, F. Kruggel, H.-J Gertz, and D.Y. von Cramon. Three-dimensional texture analysis of mri brain datasets. *IEEE Trans. on Medical Imaging*, 20(5):424–433, 2001.

[12] A.S. Kurani, D.-H Xu, J.D. Furst, and D.S. Raicu. Co-occurrence matrices for volumetric data. *7th IASTED Int'l Conf on Computer Graphics and Imaging*, 2004.

[13] Z. Lang, R.E Scarberry, Z. Zhang, W. Shao, and X. Sun. A texture-based direct 3d segmentation system for confocal scanning fluorescence microscopic images. In *Proc. of the 23rd Southeastern Symposium on System Theory,Los Alamitos, CA*, pages 472–476. IEEE Comput. Soc. Press, 1991.

[14] K.I. Laws. Rapid texture identification. *Proc. SPIE Conf. Image Processing for Missile Guidance*, pages 376–380, 1980.

[15] A. Madabhushi, M. Feldman, D. Metaxas, D. Chute, and J. Tomaszewski. A novel stochastic combination of 3d texture features for automated segmentation of prostatic adenocarcinoma from high resolution mri. In *Medical Image Computing and Computer-Assisted Intervention*, volume 2878 of *Lecture Notes in Computer Science*, pages 581–591. Springer-Verlag, 2003.

[16] I. Ng, T.S.C Tan, and J. Kittler. On local linear transform and gabor filter representation of texture. *Proc. Int'l Conf. Pattern Recognition*, pages 627–631, 1992.

[17] T. Randen and J.H. Husøy. Multichannel filtering for image texture segmentation. *Optical Eng.*, 33:2617–2625, Aug 1994.

[18] T. Randen and J.H. Husøy. Filtering for texture classification: A comparative study. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(4):291–310, 1999.

[19] E. Ranguelova. *Segmentation of Textured images on three-dimensional lattices*. PhD thesis, University of Dublin, Trinity College, 2002.

[20] E. Ranguelova and A. Quinn. Analysis and synthesis of three-dimensional gaussian markov random fields. *Proc. of ICIP'99, Kobe, Japan*, 1999.

[21] J. Reichel and A. Quinn. A fast and fully unsupervised scheme for model-based image segmentation. *SPIE Conf. on Bayesian inference for Inverse Problems*, 1998.

[22] C.C. Reyes-Aldasoro and A. Bhalerao. Volumetric texture description and discriminant feature selection for mri. In *Information Processing in Medical Imaging 2003*, volume 2732 of *Lecture Notes in Computer Science*, pages 282–293. Springer-Verlag, 2003.

[23] M. Servais and G. De Jager. Video compression using the three dimensional discrete cosine transform (3d-dct). *Proc. of COMSIG '97*, pages 27–32, 1997.

[24] L.H. Siew, R.M. Hodgson, and E.J. Wood. Texture measures for carpet wear assesssment. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 10(1):92–105, January 1988.

[25] S. Tai, Y. Wu, and C. Lin. An adaptive 3-d discrete cosine transform coder for medical image compression. *IEEE Trans. on Information Technology in Biomedicine*, 4:259–263, September 2000.

[26] T.S.C Tan and J. Kittler. Color texture analysis using colour histogram. *IEE Proc.-Vis. Image Signal Process*, 141(6):403–412, December 1994.

[27] M. Unser. Local linear transforms for texture measurements. *Signal Processing*, 11:61–79, 1986.

[28] R. Uppaluri, T. Mitsa, E.A. Hoffman, G. McLennan, and M. Sonka. Texture analysis of pulmonary parenchyma in normal and emphysematous lung. *SPIE*, 2709:456–467, 1996.

[29] R. Westwater and B. Furht. The xyz algorithm for real-time compression of full-motion video. *Real-Time Imaging*, 2:19–34, 1996.

[30] Y. Wu, X. Guan, M.S. Kankanhalli, and Z. Huang. Robust invisible watermarking of volume data using the 3d dct. *International Conf. on Computer Graphics*, pages 359–362, 2001.

[31] D.-H Xu, A.S. Kurani, J.D. Furst, and D.S. Raicu. Run-length encoding for volumetric texture. *4th IASTED Int'l Conf on Visualization, Imaging and Image Processing*, 2004.