A Color Identification System Based on Class-oriented Adaptive Color Space Quantization

by

Yuedong Zhao

Thesis submitted to the faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

Electrical Engineering

APPROVED:

A. Lynn Abbott

A. Lynn Abbott, Chairman

Roger W. Ehrich

Roger W. Ehrich

John S Bay

John S. Bay

September, 1996

Blacksburg, Virginia

Keywords: color, quantization, classification, image, texture

A Color Identification System Based on Class-oriented Adaptive Color Space Quantization

by

Yuedong Zhao

Committee Chairman: A. Lynn Abbott

Electrical Engineering

(ABSTRACT)

This thesis describes an automatic computer vision system for color identification. The system deals with color objects, such as wooden parts, that exhibit large texture variations and subtle color differences. In recent years, color has been used more and more as an important cue for recognizing textured objects. Many proposed methods use color histograms as representations of color images. Most of these methods depend on proper quantization of the color space. In our system, a novel adaptive color space quantization scheme has been employed. The method is class-oriented and is integrated with a supervised training algorithm. From a set of training samples, a partition of the original RGB color space is determined, based on the intersection of meaningful parameteric descriptions of the classes. Color histograms are constructed relative to the resulting partition of the color space, and serve as the representations of both the test images and the models in the database. Relative entropy, an information-theoretic similarity measure, has been used to perform the recognition. The system described in this thesis has been extensively tested in the laboratory and has shown a high recognition accuracy.

Acknowledgments

I would like to take this opportunity to express my gratitude to my major advisor, Dr. A. Lynn Abbott, for his invaluable guidance and encouragement throughout this work. My experience of working with him has been very helpful both intellectually and personally. I would also like to thank Dr. Roger W. Ehrich and Dr. John S. Bay for their very helpful advice and for serving on my committee.

The financial support of the project by American Woodmark Corporation (AWC) is sincerely appreciated. I wish to thank the graduate students in the AWC project for their assistance.

Finally, special gratitude goes to my beloved wife, Hong, my mother, and my sisters and brother for their continuous love, understanding and encouragement during my study at Virginia Tech. I also thank my son, Wilson, for making life so beautiful.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1 Introduction

1.1 Motivation

For many years, color has been a great help in identifying objects. Color classification involves extracting useful information concerning the spectral properties of object surfaces, and comparing that with a set of known objects to find the best match. The goal of many computer vision systems is to automatically analyze multispectural images of object surfaces, and then find the best match from a set of known descriptions or class models in order to implement the recognition task.

Color classification is a difficult problem, however, for many types of objects. Natural objects such as wood, for example, exhibit random textures that make it difficult to distinguish different types based on color alone. This thesis describes the development of a computer vision system that uses color as the cue to identify the color and species of painted or stained wood components. The thesis describes an innovative approach of color quantization, in which a partition of the original color space is formed by the intersections of meaningful parametric representations obtained in a supervised learning mode.

1.2 Background

American Woodmark Corporation (AWC) is a manufacturer and distributor of kitchen cabinets and vanities. One of its factories produces finished wooden doors, frames and other components that are shipped to other AWC locations for final assembly. Since there are a great number of components types, and many of them are very similar in appearance, it is tedious, labor-intensive and error-prone to identify these components manually. Virginia Tech is developing several automatic vision-based classification systems for AWC to identify the finished components[1][2][3][4].

The finished components are made of several species of wood and are painted or stained in a number of colors. Some of these colors are very close, with only a slight difference between them. Variations in the color of natural wood cause a lot of confusion between these colors. The variations of wood grain cause some trouble between the different species. It is difficult even for humans to identify all of these components accurately.

In order to improve manufacturing operation, a machine-vision system is needed that can identify the color and species of components automatically. Such a system will simplify production problems and increase productivity. In early work on this problem[4], a laboratory system was developed that considered color and species classification separately. There were two cameras. One was a Cohu 4810 camera that generated black-and-white images for species analysis. Another was a JVC 1070U color video camera that produced separate red, green, and blue images for color analysis. A statistical texture-based classification scheme was used for species identification while minimum centroid distance was employed in the color classification. The system yielded good results in the laboratory using still components, yielding 100% accuracy in one test of 200 components.

However, when the components were moving on the conveyor, the test results were unsatisfactory because of image blurring. In addition to the image blurring problem, the approach needed extensive laboratory experimentation and mathematical computation to determine the thresholds. It became necessary to develop a more sophisticated and robust approach to simplify the system hardware and build a flexible system for the industrial setting.

1.3 Objective and contributions

The goal of this research is to develop a robust and flexible automatic machine vision system that can identify the species and color of painted or stained wooden components in a real time industrial setting. The system should have high accuracy, and should be preferably classify something as "unknown" rather than assign an incorrect ID code. The system should be expandable to permit the addition of new color and species, and it should be appropriate for use in other industry applications in addition to the one described here. Also, the system should be easy to set up and to maintain.

The major contributions of this research include the following:

a) An improved color and species classification strategy has been developed, in which color distribution is used as a pattern recognition cue to deal with both color and texture together. The complexity of the system hardware has been reduced because only one color camera is used.

b) A novel approach to color space quantization has been developed, in which the intersection of meaningful parametric representations results in a partition of the color space.

c) The original imaging station has been improved. This has involved the careful choice of illumination and image sensing geometry and the development of software to provide user interface for the calibration of the camera and light sources.

d) Different color classification methods, color spaces, histogram distance measures, and parametric spatial representations have been analyzed and tested.

e) A user-friendly system has been developed. The system allows the user to train the system in order to change the colors or species that are classified. It is also easy to change system parameters to control the ratio of “unknown” to “misclassification” cases. Software to perform system calibration, system training, and classification functions under menu control has also been developed.

f) The system has been extensively tested in the laboratory. The laboratory tests have demonstrated a high level of accuracy for the system.

1.4 Hardware overview

The color and species identification system is illustrated in Figure 1-1. A JVC TK-1070U color digital camera generates color red-green-blue (RGB) images of a moving component on the conveyor. These are captured in a frame grabber, which resides in a 486-based PC. The PC processes the image, and then compares it to the pre-stored models in a database. The output is a color and species identification code.

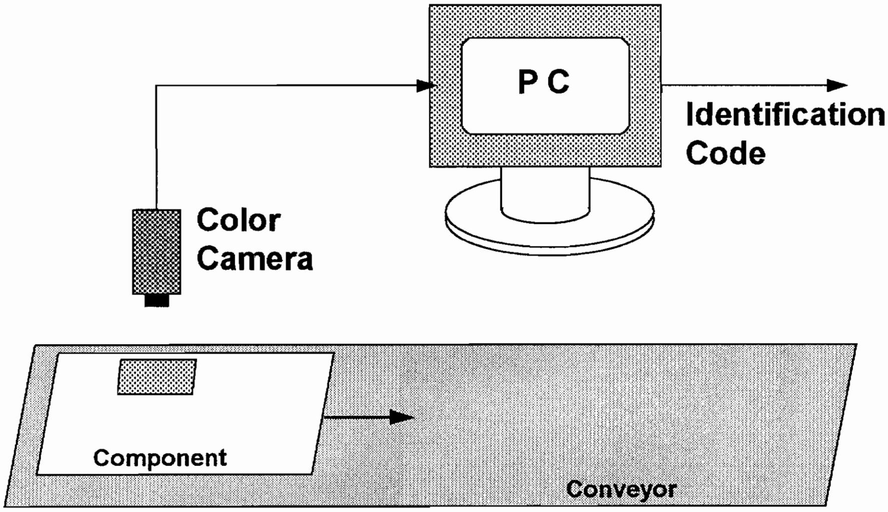


Figure 1-1. Layout of color and species classification system. The PC processes the images produced by the color camera and compares them with pre-stored models in a database, and outputs an identification code for each component.

1.5 Outline of the thesis

The remaining chapters of this thesis describe the problem and the identification system that has been developed. Chapter 2 presents a review of related work and describes the problem in more detail. Chapter 3 describes a novel approach to color space quantization. Chapter 4 gives a system overview in detail. Chapter 5 discusses the experimental results of the laboratory system. Chapter 6 presents a conclusion to the thesis.

Chapter 2 Literature Review

2.1 Overview

The pattern recognition problem can be described as follows: Given an object S, and a set of class labels $\Omega=\{\omega_i \mid i = 0 \ldots N\}$, assign the best label ω_i to S. Assume that ω_1 to ω_N correspond to N known object classes, and ω_0 represents the "unknown" class. If the object S fits the criteria for class i better than the criteria for any other class, then label ω_i should be assigned to the object S.

In a color recognition system, the following issues must be considered:

- Selection of a suitable color space.
- Quantization of the color space.
- Formulation of an appropriate matching method.
- Acquisition of a model database.

The following sections will discuss these topics, in addition to a consideration of texture and inductive learning.

2.2 Color spaces

Color is a perceptual phenomenon related to human response to different wavelengths in the visible electromagnetic spectrum[5]. Although the number of basis functions required to completely describe the full spectrum is essentially infinite, a small number of basis functions can provide good spectral approximations of most perceivable colors. Usually, a color is presented as a weighted combination of stimuli of three primary colors that form a natural basis.

Many different color spaces are used today. Three color spaces are discussed here: RGB, normalized RGB (Nrgb) and HSI.

2.2.1 RGB space

One of the most common color spaces is red-green-blue (RGB) space. Most color display systems use separate red, green, and blue as light sources so that other colors can be obtained by a weighted combination of these three primaries. Although any three colors can be mixed in varying proportions to generate many different colors, the set of red, green and blue can produce the greatest number of colors[6].

In RGB space, all colors that can be displayed are specified by the red, green and blue components. One color corresponds to one point in a three-dimensional space whose axes are the three primaries. Therefore, all colors can be represented within a cube.

Figure 2-1 shows the RGB space and its associated color cube. The origin is black and the opposite vertex of the cube is white. The points on the axes are red, green and blue with varying brightness, respectively. The diagonal from black to white represents different levels of gray whose three components have the same magnitudes.

In computer applications, the RGB space is discrete. Commonly, each dimension has 256 levels, numbered 0 to 255. A total of 256^3 different colors can be represented a (R, G, B), where R, G, and B are the magnitudes of the three components, respectively. Normally, black is represented as (0, 0, 0) while white is represented as (255, 255, 255).

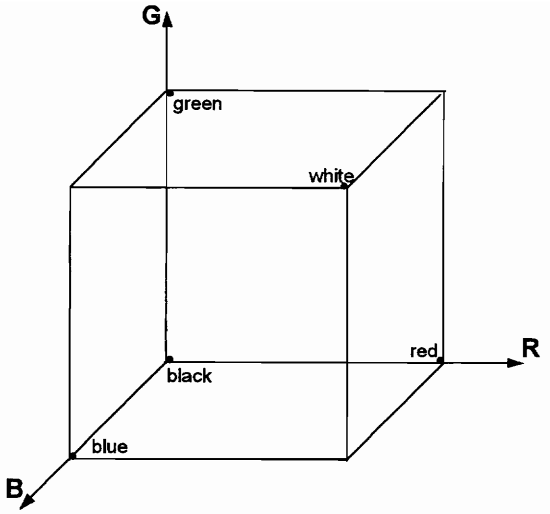


Figure 2-1. RGB color space, and the color cube. The three axes are the three primary colors red, green, and blue. The origin (0, 0, 0) represents the black while the opposite vertex (255, 255, 255) is white. Any point in this cube represents a color and can be presented as (*R*, *G*, *B*). For example, red is (255, 0, 0), green is (0, 255, 0), and blue is (0, 0, 255).

2.2.2 Normalized RGB space

Normalized RGB space is independent of uniformly varying lighting levels. The three primaries of normalized RGB space can be calculated from the primaries of RGB space using the following function:

$$N_c = \frac{C}{(R+G+B)} \qquad \text{for } C \in \{R, G, B\} \text{ and } R + G + B \neq 0.$$

However, the primaries of the normalized RGB space are redundant because $N_R + N_G + N_B =$.

Typically, a normalized RGB space is given as

$$Y = c_1R + c_2G + c_3B$$

$$T_1 = \frac{R}{(R+G+B)}$$

$$T_2 = \frac{G}{(R+G+B)}$$

where c_1, c_2, and c_3 are constants chosen such that $c_1+c_2+c_3 = 1$ [7]. Y is considered as the luminance of the image pixel while T_1 and T_2 are chromatic and are independent of illumination to some extent [8].

2.2.3 HSI space

HSI (hue-saturation-intensity) space is another popular color space because it is based on human color perception. Visible light is electromagnetic radiation in the range of wavelengths to which the normal human visual system is sensitive, about 400 to 700 nanometers [6]. In general, hue is related to the wavelength of a light while intensity is an interpretation of the amplitude of a light. Saturation is a parameter that measures the "colorfulness".

One transformation from RGB to HSI is given as below, even though there are many alternate definitions [5][7] [9].

Intensity: $I = R + G + B$

Saturation: $S = 1 - \dfrac{3 \times \min(R,G,B)}{I}$

Hue:

$$H = \begin{cases} 2\pi - \cos^{-1}\left\{ \dfrac{[(R-G)+(R-G)]}{2\sqrt{(R-G)^2+(R-B)(G-B)}} \right\}, & B > G \\ \cos^{-1}\left\{ \dfrac{[(R-G)+(R-G)]}{2\sqrt{(R-G)^2+(R-B)(G-B)}} \right\}, & otherwise \end{cases}$$

Conceptually, HSI space can be thought of as a cylinder as shown in Figure 2-2, where the coordinates r, θ, and z correspond to saturation, hue, and intensity, respectively.

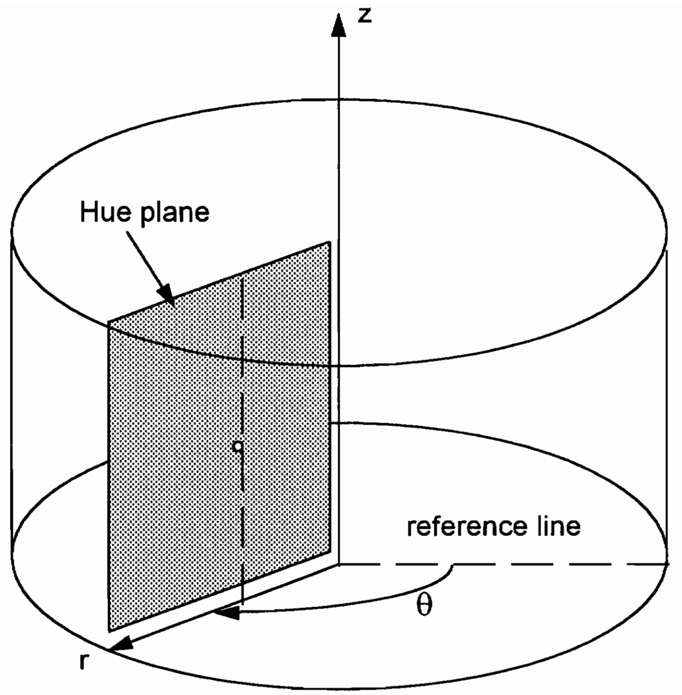


Figure 2-2. HSI color space, and the cylinder. The coordinates *r*, θ, and *z* correspond to saturation, hue, and intensity, respectively. θ is the angle between the reference line and the hue plane.

2.3 Color quantization

Color spaces are often quite large, typically having 256^3 discrete colors if a standard RGB camera is used, digitized using 8 bits per color dimension. It is usually not practical to use this large color space directly in computing histogram distances. The effort of reducing the size of the color space, by partitioning the original color space into larger cells, is known as color quantization. Color quantization has been studied extensively because the set of colors that result from the quantization process has a significant impact on the performance of a color classification system.

The simplest method of color quantization is uniform quantization as illustrated in Figure 2-3. In this case, each color cell is a cube whose edges are aligned with the color axes and all cells have same size. The scheme can be formulated as

$$C = \{(R_i, G_j, B) | it \leq R < (i+1)t, jt \leq G < (j+1)t, kt \leq B < (k+1)t; i, j, k = 0, 1, \ldots, n-1\},$$

where (R, G, B) is the original color; C is the color space after quantization; (R_i, G_j, B_k) is a chosen color that represents the corresponding (i, j, k)th cell, n is the number of quantization cells in each dimension and t is the size of quantization cells so that nt=256. The total number of colors is n^3.

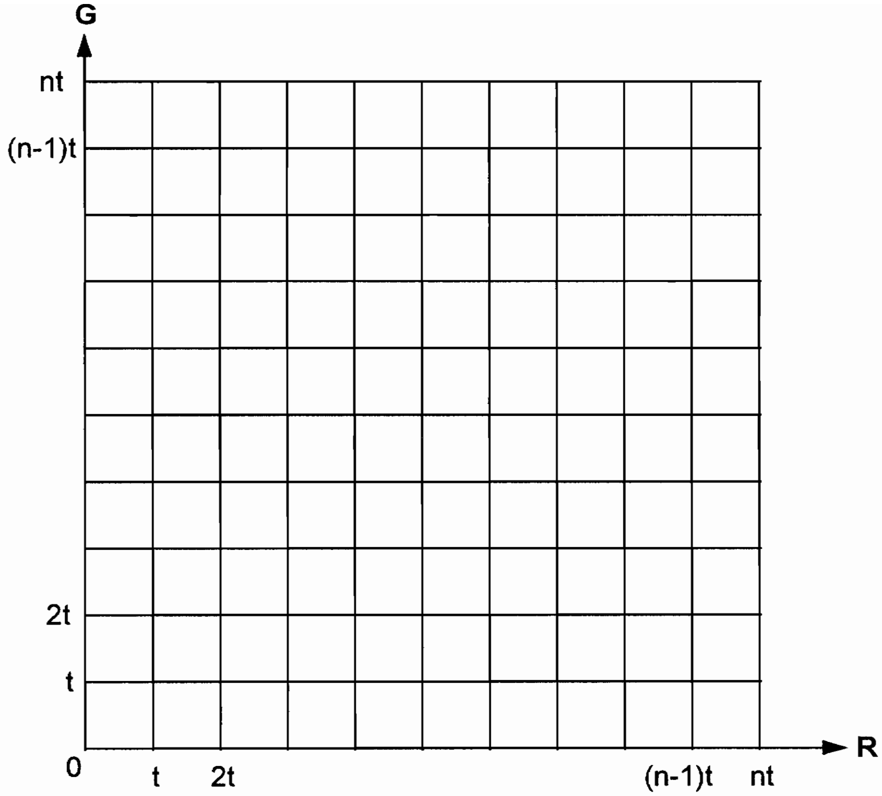


Figure 2-3. Illustration of uniform quantization in two dimensions. Color cells are squares and are of the same size. *n* is the number of quantization cells in each dimension so that the total number of cells is n^2. The *t* is determined by nt=256.

In many systems [10] [11] [12] [13], nonuniform quantization is used to offer improved performance. The method divides the color space using multiple thresholds on each color axis separately. The thresholds are chosen based on an analysis of the original distribution in the color space. The result is a set of rectangular boxes whose edges are aligned with the color axes. Figure 2-4 illustrates the multiple thresholds quantization in two dimensions case. The formula of multiple thresholds quantization may be given as

$$C = \{(R_i, G_j, B_k) | Rt_i \le R < Rt_{i+1}, Gt_j \le G < Gt_{j+1}, Bt_k \le B < Bt_{k+1};$$
$$i = 0,1,\ldots,r-1, j = 0,1,\ldots,g-1, k = 0,1,\ldots,b-1\}$$

where (R, G, B) is the original color; C is the color space after quantization; Rt_i, Gt_j and Bt_k are chosen thresholds, in which Rt_0, Gt_0, and Bt_0 are equal to 0 while Rt_r, Gt_g, and Bt_b equal 256. The r, g, and b are the numbers of quantization cells in the three dimensions and are not necessary to be the same. (R_i, G_j, B_k) is a chosen color that represents the corresponding (i, j, k)th cell. In this case, the cells may not be of the same size. However, the edges of color cells are aligned with the axes so that colors cells are rectangles. The total number of cells is $r \times g \times b$

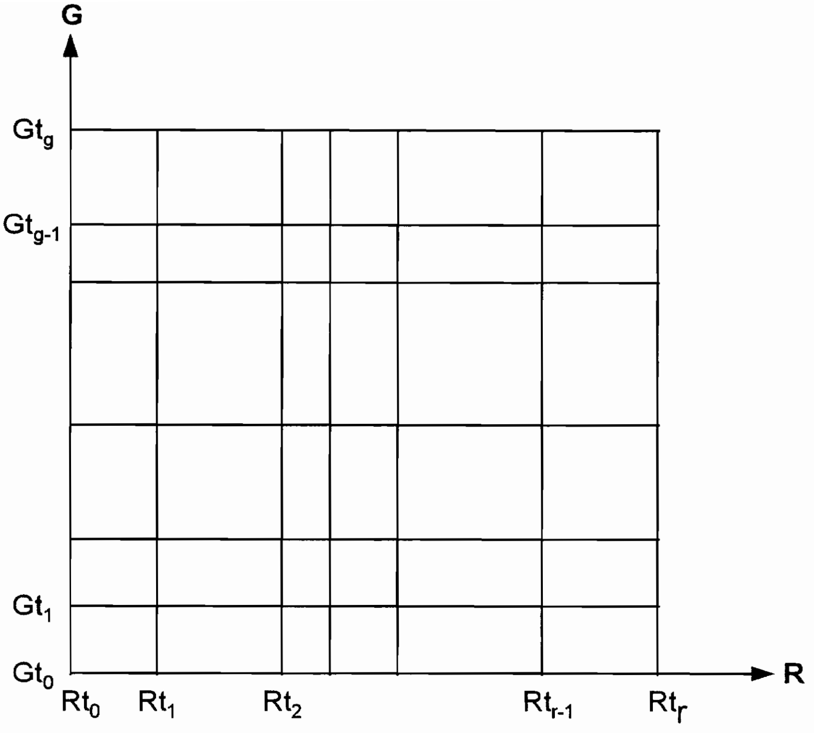


Figure 2-4. Illustration of nonuniform quantization in two dimensions. The r and g are the numbers of quantization cells and they may not be same. The Rt_i and Gt_j are chosen thresholds. The Rt_0 and Gt_0 are equal to 0 while Rt_r and Gt_g equal 256. The color cells may not be of same size. However, the edges of color cells are aligned with the axes so that colors cells are rectangles.

A significant improvement in system performance can be obtained in some cases if all color dimensions are considered simultaneously during quantization. Many proposed algorithms [14][15][16] use clustering or vector quantization methods to quantify the mulitspectral space and the cells no longer are constrained to be rectangular. The resulting quantization is equivalent to the Voronoi tessellation of the color space relative to class centers. Figure 2-5 illustrates the case in a two dimensional color space.

Generally, clustering attempts to find a partition of the input vectors. It is often used in unsupervised systems to find the "best" locations for class representative in multidimensional space. Vector quantization means mapping input vectors to a finite number of weight vectors and requests to find a set of weighted vectors that can represent the original multidimensional data with the least distortion.

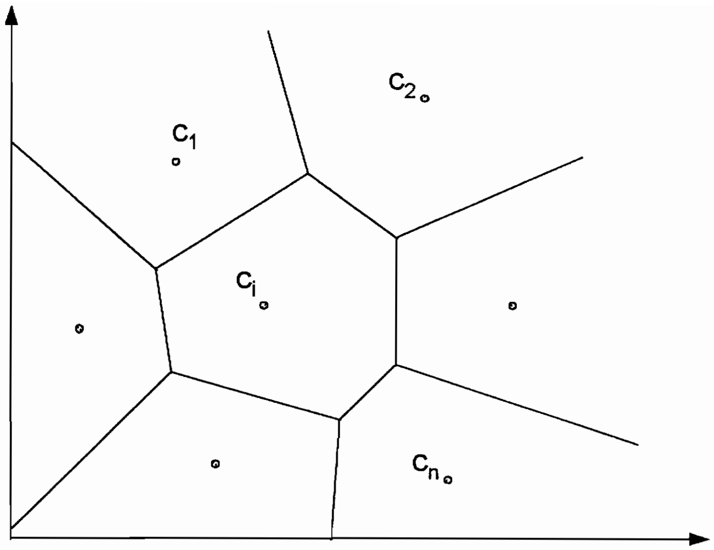


Figure 2-5. Illustration of clustering or vector quantization in two dimensions. Each color cell represents one color.

2.4 Color classification methods

For a long time, color has been considered as an important cue for object recognition tasks. These involve comparing a color image with a set of known color images to find a best match. Different descriptions and distance measurements can yield different results for a given recognition task.

2.4.1 Extraction of color feature

An intuitive description of a color image is the centroid of the distribution of colors in a 3-dimensional color space. In the RGB color space, the centroid of a color image is given as:

$$\begin{bmatrix} R_c \\ G_c \\ B_c \end{bmatrix} = \frac{1}{N}\sum_{i=1}^{N}\begin{bmatrix} R_i \\ G_i \\ B_i \end{bmatrix}$$

where N is the number of pixels of an image. This description is very useful for images with uniform texture because of its simplicity of computation. However, it is insufficient to describe an image with sharply varying texture.

Most color classification methods involve using histograms to provide the pixel distribution information of an image over the color space. Generally, a histogram has K bins, which relate to a set of K non-overlapping subspaces, called cells. Typically, the set of cells forms a partition of the color space. The value contained in each bin in the histogram is the number of pixels that have the $[R, G, B]^T$ values associated with this cell.

The histogram can be considered a K-dimensional feature of a color image, where K is the number of histogram bins.

2.4.2 Classification methods

The Minimum Distance Classifier is one of the most commonly used classifiers because of its simplicity and efficiency. It can be used to classify an observed image for n known classes. Assume that a set of k-dimensional features $\{\mathbf{x}_i \mid i = 1,2,\ldots,n\}$ has been computed from a set of training samples and stored as class models. The k-dimensional feature $\mathbf{y}$ of an observed sample is matched with the models, and class label ω_i is assigned if $d(\mathbf{x}_i,\ \mathbf{y}) < d(\mathbf{x}_j,\ \mathbf{y}),\ \forall\ i \neq j$, where d is the distance measure that is chosen.

Here, we assume that each class can be represented by one model. However, in most cases, one model may not represent a class sufficiently. There may be several subsets for each class and each subset needs its own model. Under the condition that all subsets of a class have to be assigned a unique class label, the formula of the classification problem can be expressed as follows:

$$S = \{\ \omega_i \mid \ \min(\ d(\mathbf{x}_{ik},\mathbf{y})\ ) < \min(\ d(\mathbf{x}_{jk},\mathbf{y})\ )\ \forall\ i \neq j,\ k = 1,\ \ldots,\ n_i\},$$

where $\mathbf{x}_{ik}$ is the kth model of the ith class, and n_i is the number of models of the ith class, $\mathbf{y}$ is the test sample.

Euclidean distance is the most common distance measure. Let $\mathbf{x}=[x_1, x_2, \ldots, x_k]^T$ be a k-dimensional feature of model and $\mathbf{y}=[y_1, y_2, \ldots, y_k]^t$ be a test sample. The Euclidean distance d_1 is given as:

$$d_1^2(\mathbf{x},\mathbf{y}) = (\mathbf{x}-\mathbf{y})^T(\mathbf{x}-\mathbf{y}) = \sum_{i=1}^{k}(x_i - y_i)^2, \text{ or}$$

$$d_1(\mathbf{x},\mathbf{y}) = [(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots + (x_k - y_k)^2]^{1/2},$$

where x_i is the ith component of $\mathbf{x}$, and similarly for $\mathbf{y}$. A simpler distance measure is the city-block metric,

$$d_2(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{k}|x_i - y_i|.$$

The k-dimensional feature is sometimes presented as a normalized form. In this case, a histogram approximates the probability distribution of the pixel values. It is, then, possible to use relative entropy (Kullback Leibler distance) as a distance measure. The definition of relative entropy is

$$d_3(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{k} y_i \log\frac{y_i}{x_i}$$

$$= \sum_{i=1}^{k} y_i \log y_i - \sum_{i=1}^{k} y_i \log x_i \;,$$

where

$$\sum_{i=1}^{k} x_i = \sum_{i=1}^{k} y_i \; = \; 1, \text{ and } x_i \geq 0 \text{ and } y_i \geq 0.$$

This is not a true metric because it is not symmetric and does not satisfy the triangle inequality. Relative entropy, however, is always non-negative and is zero if and only if $\mathbf{x} = \mathbf{y}$. It is often useful to think of relative entropy as a "distance" between distributions[17].

A metric variation of relative entropy is given [18] by

$$d_4(x, y) = \sum_{i=1}^{k} y_i \log \frac{y_i}{x_i} + \sum_{i=1}^{k} x_i \log \frac{x_i}{y_i}$$

$$= \sum_{i=1}^{k} (x_i - y_i)(\log x_i - \log y_i),$$

This is a symmetric distance measure because $d_4(\mathbf{x}, \mathbf{y}) = d_4(\mathbf{y}, \mathbf{x})$.

2.5 Color texture

Texture is an important characteristic for the analysis of many types of images. Despite its importance and ubiquity in image data, however, a formal approach or precise definition of texture does not exist [19]. One of the problems confronted in this work is to accept a texture region, and determine to which of a finite number of classes the region belongs [20].

In general, texture analysis can be categorized into statistical methods and structural methods. Statistical methods use statistical results of distribution of pixels and their relationship as the features of a texture image. This kind of methods works well for many natural texture analyses. However, most of the statistical methods are based on gray level images and are inappropriate to deal with multidimensional data simultaneously. Structural methods consider that a texture image is made up of primitives that appear in repetitive spatial arrangements. Before describing the texture images, the primitives and the placement rules have to be described [21].

Real visual textures are in fact colored textures. They may present different colors, different structural patterns, or different colors and different structural patterns [22]. In the field of computer vision, research has confirmed that color is one of the important properties which humans use in object recognition, in particular, pattern recognition in image texture [23]. Instead of treating color and texture separately, more and more effort has been made to use the combination of the color and texture features in object recognition.

Caelli and Reye [24] proposed an approach to encode color, texture, and shape in a single spatio-chromatic feature space and use multi-scaled filtering and correlation methods to extract the features for object recognition. They considered the color, texture, and shape conjointly in a systematic and interpretable fashion.

Swain and Ballard [25] suggested using color histograms to represent multicolored objects because the histograms are invariant to translation and rotation about the viewing axis and change only slowly under change of angle of view, change in scale, and occlusion. They introduced a Histogram Backprojection algorithm that uses color information instead of spatial information to solve the location estimation problem.

There are also many proposed approaches using color distribution as the cue for color objects classification [26][27][28]. All of these proposed methods are based on color histograms constructed on a quantized color space and involve histogram similarity measures between test sample and model database.

In this work, we deal with the color problem and the texture problem together and use color distribution as the cue for both color and species classification. The color histograms used to represent test samples and models are constructed on a quantized color space obtained in a training procedure using a novel color quantization method.

2.6 Inductive learning

Inductive learning is the process of finding general descriptions from example objects. These descriptions can be temporal or spatial, but are generally abstract in nature. In general, inductive learning system has to consider several issues:

- Find subsets or clusters of the input objects.
- Categorize the subsets into classes.
- Find the class descriptions that cover the classes.

A system that involves the last two parts listed above is called a supervised learning system (learning from examples) while unsupervised learning (learning from observation) includes all three parts [20].

Unsupervised learning uses the training examples without preassigned class labels. When processing begins, the system does not know to what class each object belongs, and may even not know the number of classes, until a partition of the given objects has been found. Unsupervised learning can be considered as "data-driven" approach which is based on the observation of the input data.

Clustering is the technique that is often used in unsupervised learning to generalize a subset of the objects. It can be divided into "free clustering" without the knowledge of the number of clusters in advanced and "constraint clustering" where the number of clusters is predetermined [29]. Minimization of within-cluster variance is a common criterion used in clustering, though there are many proposed different cohesion methods [30].

In supervised learning, all training examples have been preassigned a class label from a set of known classes. The problem of supervised learning is to extract the underlying characteristics of the objects, to find adequate descriptions of the objects, and to set up a set of rules categorizing each object into a class including an unknown class that contains the objects beyond all known classes. In a sense, the system or the algorithm knows what it is looking for. It can be considered as a "model-driven" approach which imposes one of several preconceived structures onto the data.

The goal of this work is to develop a color/species identification system. All test components have to be assigned one of eight known classes unless the component is unrecognizable. Therefore, supervised learning has been employed in a system training procedure in which the training samples are used to extract the features of the classes and set up discrimination rules. The system training procedure will be discussed in section 4.3. In this work, the feature used to distinguish images between different classes is color histograms over a quantified color space and all four distance measures discussed in section 2.4.2 are tested. Chapter 5 will show the results. The next chapter will describe a novel approach of color quantization developed in this work, which is different from the common approaches of color quantization discussed in this chapter.

Chapter 3 Adaptive Color Quantization by Intersection of Parametric Fits

3.1 Motivation

Typically, an RGB color space has 256^3 discrete colors. This number is too large to be practically used for histogram distance measurement when processing time is a critical issue. It is necessary to find some ways to reduce the size of color space and still keep, and maybe improve, the accuracy of the classification. The problem of finding an appropriate subset of the color space is known as color quantization, which may be viewed as the partitioning of the color space into larger cells, each representing a single color. Previous research indicates that the result of color quantization has a significant impact on the performance of a color classification system. In some situations, a coarser quantization may result in better classification performance than is possible with a larger number of quantization cells.

In this thesis, we provide a novel approach for color quantization in which color cells are determined by the intersections of descriptive parametric representations of color classes. The primary motivation in development of this method is to develop a color identification system with high recognition accuracy, for objects that show a large texture variation.

American Woodmark Corporation (AWC) currently uses three different species of wood (maple, oak, and cherry) for their furniture products. These wood types exhibit great variability and irregularity of grain. Figure 3-1 shows images of unpainted maple, oak, and cherry. Because of the variability and irregularity of grain, it is difficult to use texture characteristics alone to distinguish the different species of wood. Color is the only cue we can use as a discrimination between classes that have the same species but different colors. Meanwhile, different species of wood may be stained to the same shade. These finished components show only a little difference in color between different species of wood.

The slight differences in color may seem to require a large number of color cells when histogram-based similarity measures are used. However, histogram distance measures with a large number of bins tend to be computationally prohibitive in a time restricted system. So, color quantization plays an important role in successful recognition. The rest of this chapter describes the color quantization method developed in this work.

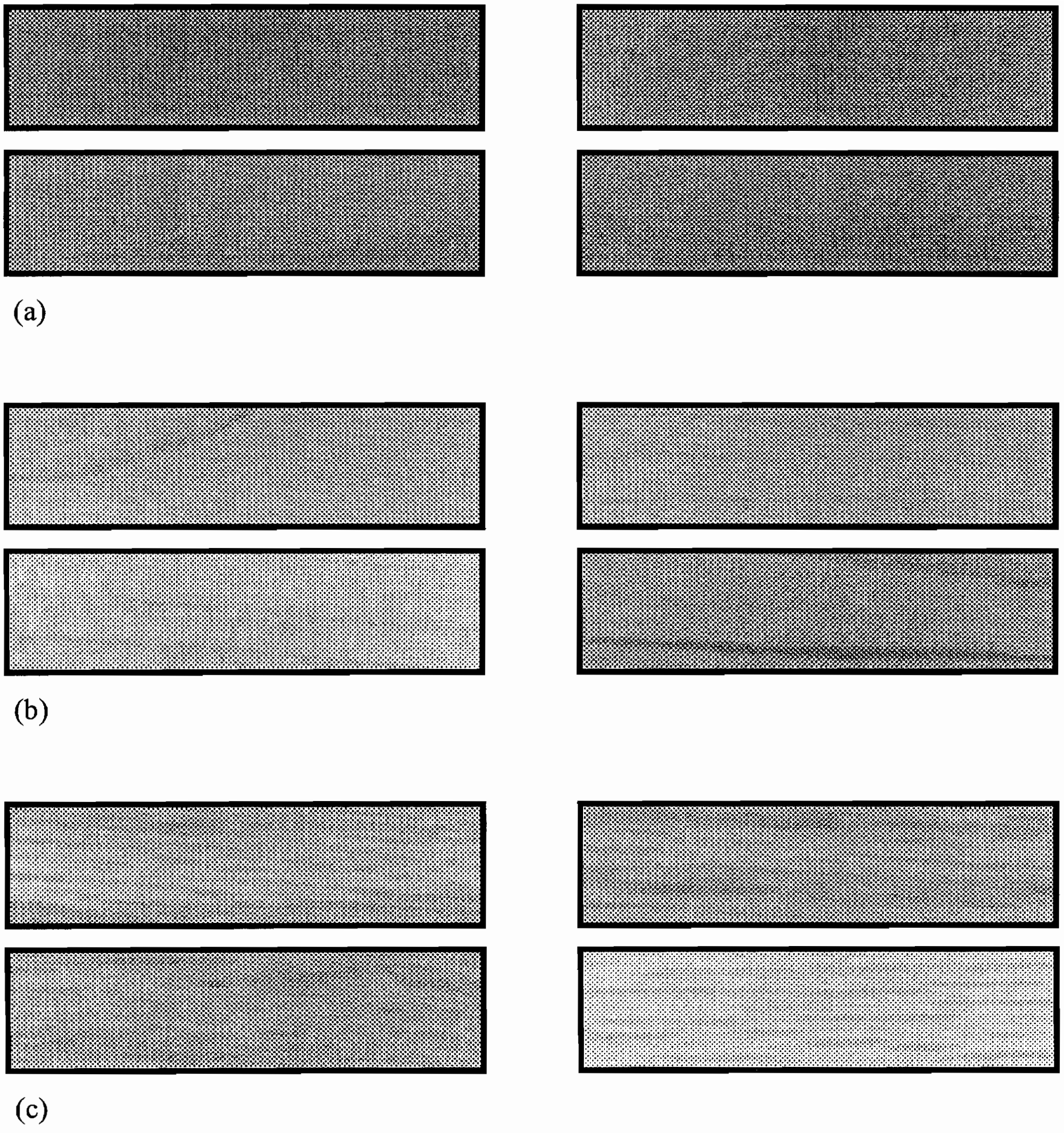


Figure 3-1. Examples of natural wood (a) cherry, (b) maple, and (c) oak. Notice the variations in both color and texture.

3.2 Color quantization based on class intersection

In general, color quantization can be considered as finding a set of colors that each represents a cluster of colors in the original color space in order to reduce the total number of colors that have to be represented. For the image segmentation problem, the goal of color quantization is to achieve a minimum distortion between the new image and the original image. In a color classification system, the purpose of color quantization is to obtain a set of subdivisions of the original space, in which the number of color bins is reduced but sufficient information is still kept for color classification. A partition $\mathcal{P}$ of the color space may be formulated as follows:

$$\mathcal{P}=\left\{P_j \mid j=1\cdots N_b;\ P_j\cap P_k=\varnothing\, for\, j\neq k;\ \bigcup_j P_j=C\right\},$$

where N_b is the final number of color bins and C is the original (large) color space. Each subset P_j represents one color. The goal is to determine a partition so that histograms defined over the bins $\{ P_j \}$ yield good system performance at low computational cost.

Section 2.3 has described several common color quantization schemes. Here we describe a novel approach to color quantization developed in this work.

Assume there are N different object types that are to be recognized using color. The color pixels of each type of object appear in a certain subspace of the original color space. Ideally, these subspaces are isolated from each other. If this is the case, each class of objects can be depicted by one color bin and there are N+1 color bins in the final histogram, where the additional bin is for pixels that do not belong to any one of the N kinds of objects. Figure 3-2 shows this case in two dimensions.

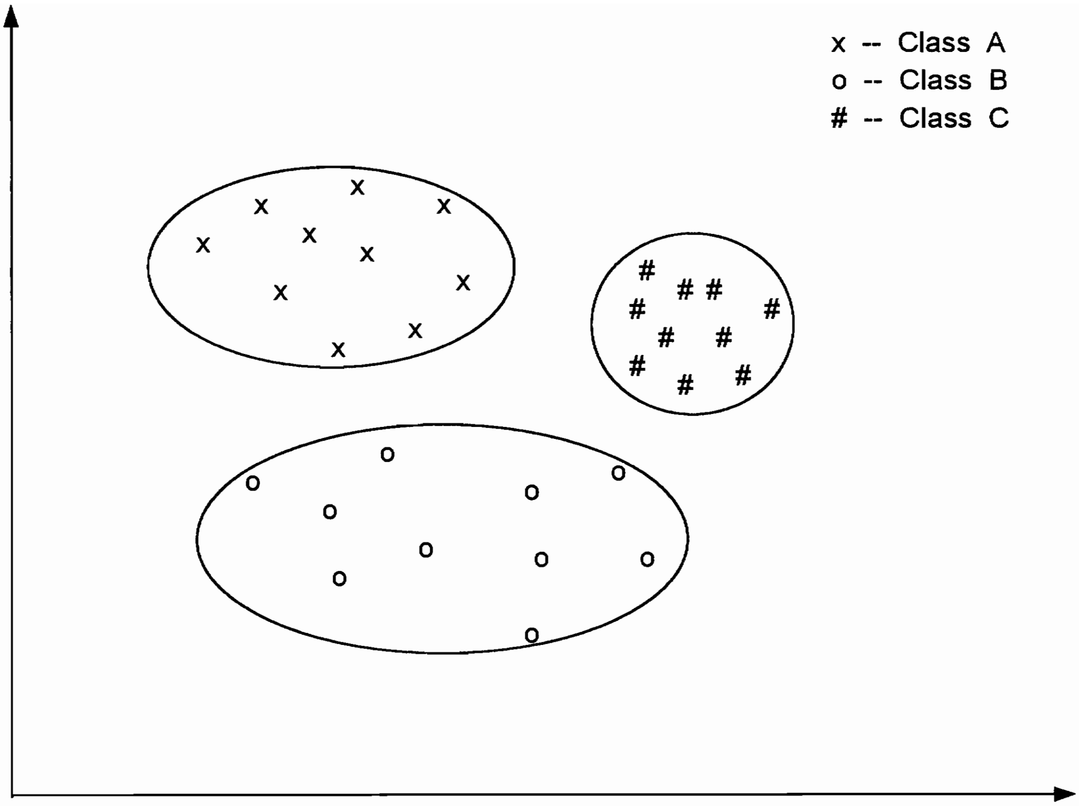


Figure 3-2. Example of nonintersecting color cells in two dimensions. The x's represent the colors that may be found only for the objects of class A, while the o's and #'s are the colors found only for the objects of class B and class C, respectively. There are four color cells after color quantization. The three ellipses correspond to three new colors for the three classes, and the fourth color cell is the complement of these ellipses in the original color space.

In most cases, however, the subspaces are not isolated from each other; instead, there are some intersections among them. Figure 3-3 illustrates a possible situation in two dimensions. When there are some intersections among the classes, one color or description for each class is not sufficient to distinguish the objects, especially for objects that possess large texture variations. In our approach, we use the intersections of classes in the original color space to determine the new color cells.

In Figure 3-3, class A and class B intersect to each other and class B contains class C. In the framework developed here, 6 color bins will be defined. Table 3-1 shows the relationships between bins of the resulting histogram, and the color points of the original color space. Bin #0 contains the color points not contained in either of the three ellipses. The color points appearing only in class A or class B are contained in bin #1 or bin #2, respectively. Bins #3, #4 and #5 represent the color points that are in the intersections of classes A and B, classes B and C, and all three classes, respectively.

In this case, objects of class A, class B and class C are represented by three, four, and two colors, respectively.

Table 3-1. The relationship between new color bins and color bins of the original color space.

bin #	color points	intersection of classes
0	outside classes A, B, and C	$\overline{A}\bullet\overline{B}\bullet\overline{C}$
1	in class A only	$A\bullet\overline{B}\bullet\overline{C}$
2	in class B only	$\overline{A}\bullet B\bullet\overline{C}$
3	in both class A and B	$A\bullet B\bullet\overline{C}$
4	in both class B and C	$\overline{A}\bullet B\bullet C$
5	in all three classes	$A\bullet B\bullet C$

For an N-class system, there are $\sum_{i=0}^{N} C_i^N = 2^N$ color bins at most. Fortunately, in most practical problems, the final number of quantization bins is much less than 2^N because many possible intersections do not exist. For example, in Table 3-1, there are no bins corresponding to the intersections $\overline{\mathrm{A}} \bullet \overline{\mathrm{B}} \bullet \mathrm{C}$ or $\mathrm{A} \bullet \overline{\mathrm{B}} \bullet \mathrm{C}$. The case illustrated in Figure 3-2 is the simplest case of Figure 3-3.

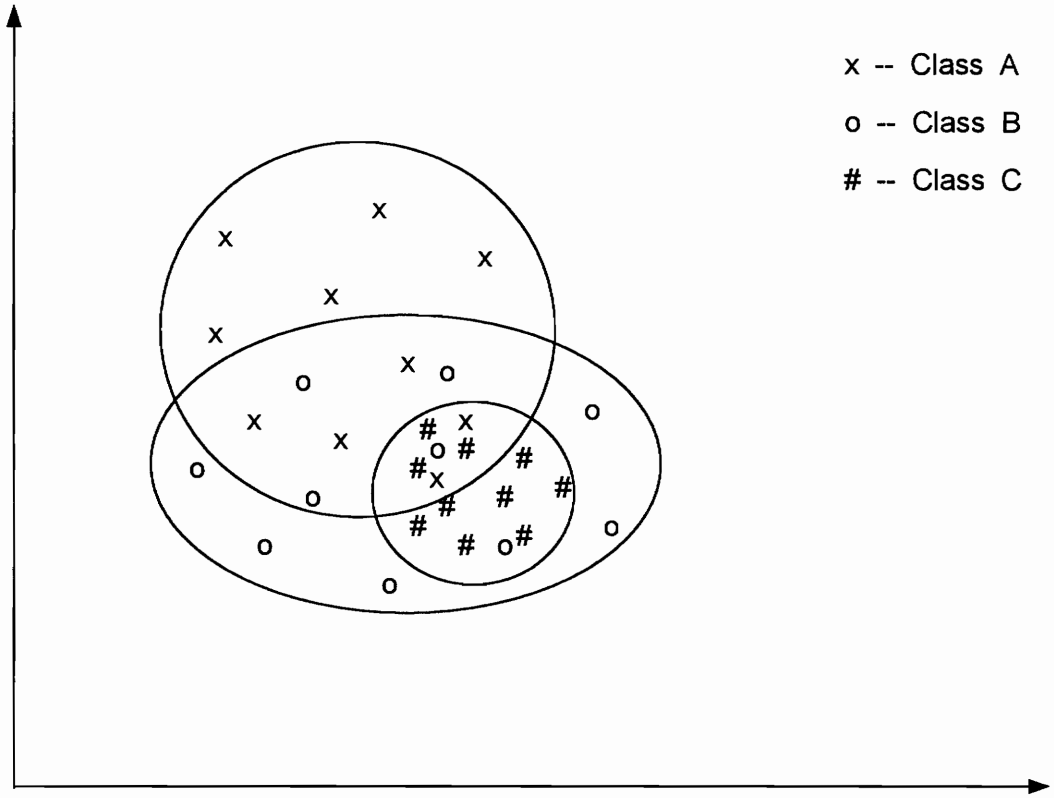


Figure 3-3. Example of color cells for intersecting classes in two dimensions. Each ellipse contains the colors found in the objects of the related class. Six color cells can be selected based on the intersections of these ellipses after quantization. One is the intersection of the three ellipses, which contains 'x', 'o', and '#'. Colors in this cell may be found in objects of all three classes. The colors in the intersections containing two symbols, which are 'x' and 'o' or 'o' and '#', may be found in objects of class A and B, or class B and C, respectively. The colors in the areas contain only one symbol, which is 'x' or 'o', may be found in objects of class A or class B. Another color cell contains the colors that may not be found in either of the three classes. In this case, objects of class A, class B, and class C may be represented by three, four, and two quantified colors, respectively.

In this thesis, we use parametric boundaries to describe the quantized colors, with one enclosing boundary for each class. These boundaries are determined by a set of training samples in a supervised learning mode. Each boundary separates the original color space into two parts: points that appear in the training objects of an associated class, and those that do not. The intersections of the boundaries and their complements determine the color quantization cells. This color quantization scheme considers both similarities and differences between classes. It provides a simple way to successfully distinguish objects of the classes such as class B and class C in Figure 3-3.

The boundaries can be described using ellipsoids, rectangular boxes, or other appropriate shapes. The next section discusses the boundary shapes used in this work.

3.3 The shapes of the boundaries

In order to describe a set of multidimensional data, the ellipsoid is often employed because of its compact shape and simple parametric description. The parameters of an ellipsoid can be easily determined to provide a statistically good fit to the original data by the method of Principal Component Analysis (PCA). Points inside the ellipsoid may be described as

$$(\mathbf{x} - \mathbf{x_c})^{\mathrm{T}} \mathbf{A} (\mathbf{x} - \mathbf{x_c}) \leq 1$$

where $\mathbf{x}$ is a data point vector; $\mathbf{x}_c$ is the mean vector of the data; and $\mathbf{A}$ is a matrix determined by the covariance matrix of the data.

From a set of points $\{\mathbf{x}_c \mid i=1, 2, ..., N\}$, we can obtain the mean vector $\mathbf{x}_c$ and covariance matrix Σ as follows:

$$\mathbf{x_c} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x_i}$$

$$\Sigma = \frac{1}{N}\sum_{i=1}^{N} (\mathbf{x_i} - \mathbf{x_c})(\mathbf{x_i} - \mathbf{x_c})^{T}$$

For a 3-dimensional feature space, the covariance matrix Σ is a symmetric 3×3 matrix so that there are 3 real eigenvalues λ_1, λ_2, and λ_3, and 3 real eigenvectors ϕ_1, ϕ_2, and ϕ_3, that are orthogonal to each other. All these parameters determine an ellipsoid. Figure 3-4 shows the case in two dimensions.

The ellipsoid matrix **A** can be computed from $k\Phi^T\Lambda\Phi$, where

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix}$$

$$\Phi = [\phi_1 \;\; \phi_2 \;\; \phi_3]$$

and k is a scale factor.

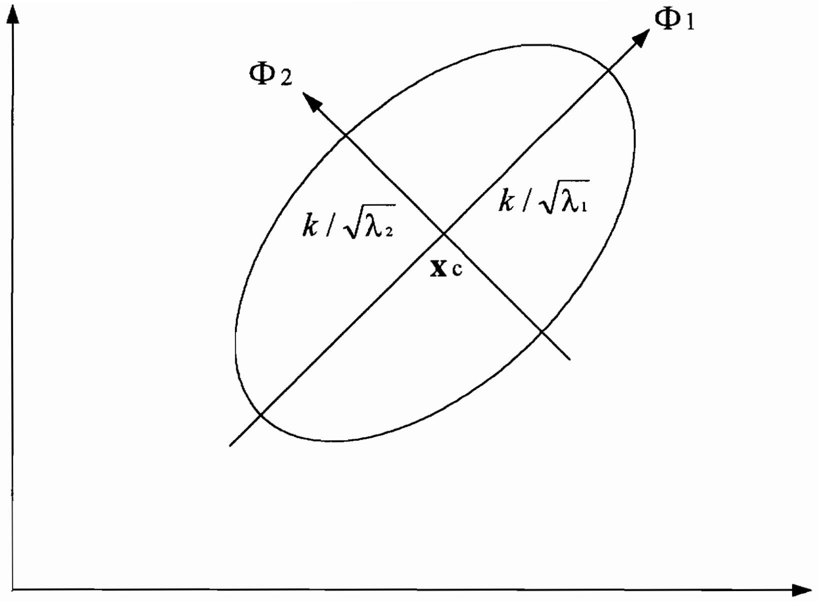


Figure 3-4. Example of an ellipse determined by eigenvalues, eigenvectors and mean values of a data distribution.

The rectangular box (or hyperprism) is another commonly used shape for representing a distribution of data. Generally, a rectangular box can be determined by the principal components approach also. Figure 3-5 shows the case in two dimensions.

For a three dimensional feature space, a rectangular box can be represented as follows:

$$\sum_{i=1}^{6} |A_i x + B_i y + C_i z + D_i| = 1$$

Here 6 planes consist of three pairs of parallel planes; and different pairs of the planes are orthogonal to each other. The orientations of the planes, that are presented by the parameters A_i, B_i, and C_i are determined by the ϕ_1, ϕ_2, and ϕ_3. The parameter D_i is determined by the mean vector $\mathbf{x}_c$ and eigenvalues λ_1, λ_2, and λ_3.

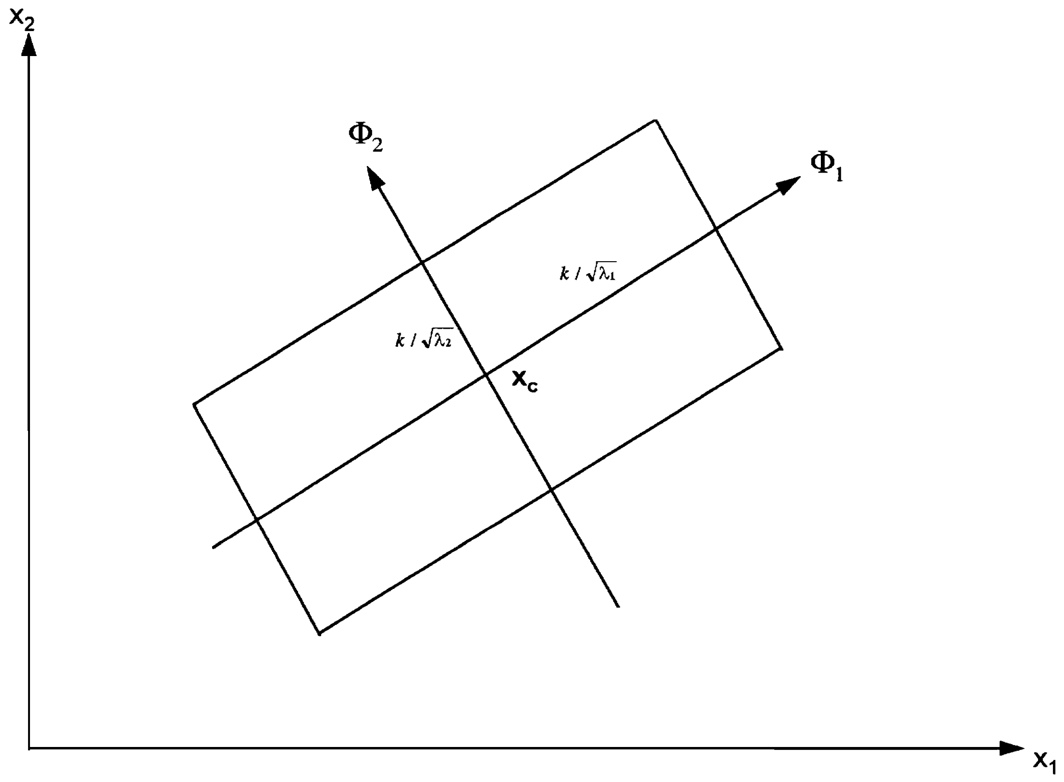


Figure 3-5. Example of rectangle determined by the principal components of the data distribution.

We also consider the rectangular box that is constrained to have edges that are aligned with the axes of the coordinates of the original feature space. The boundaries of such a rectangular box are chosen to provide a good fit to the histogram of training samples.

The calculation of the histogram for an image is very simple and quick when the rectangular boxes that have edges aligned with axes of coordinates are used. However, in most cases, such rectangular boxes can not provide a good fit to the data distribution. Figure 3-6 gives an example to illustrate the problem. In Figure 3-6, the data is distributed in a long narrow space and the principal axes of the data distribution are unaligned with the axes of the coordinates. The rectangular box is too large to provide a good fit to the data distribution.

In such a case, it is possible to subdivide the large box into several small boxes and remove the empty boxes. Finally, the description of a class is several boxes, which are not necessarily connected to each other. The description may be presented as

$$S_i = \{S_{i1} \cup S_{i2} \cup \ldots \cup S_{im} \mid S_{ij} \cap S_{ik} = \varnothing,\ \forall\, j \neq k\}$$

Figure 3-7 displays the split boxes corresponding to the distribution in Figure 3-6.

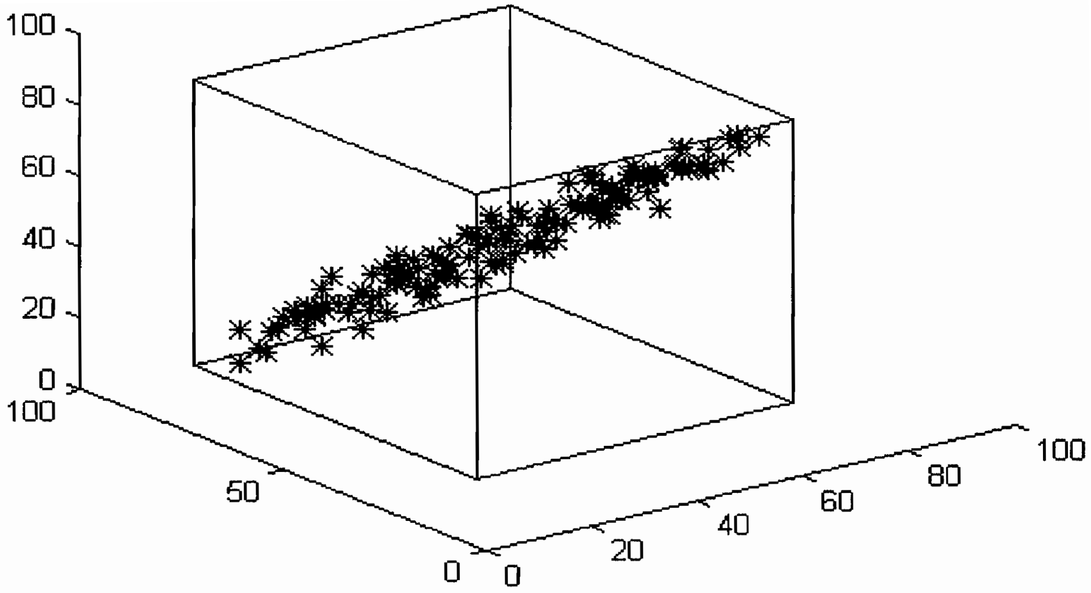


Figure 3-6. Example of rectangular box that has edges aligned with the axes of the coordinates. Because the principal axes of the data distribution are unaligned with the axes of the coordinates and the data is distributed in a long narrow space, the rectangular box is too large to provide a good fit to the data.

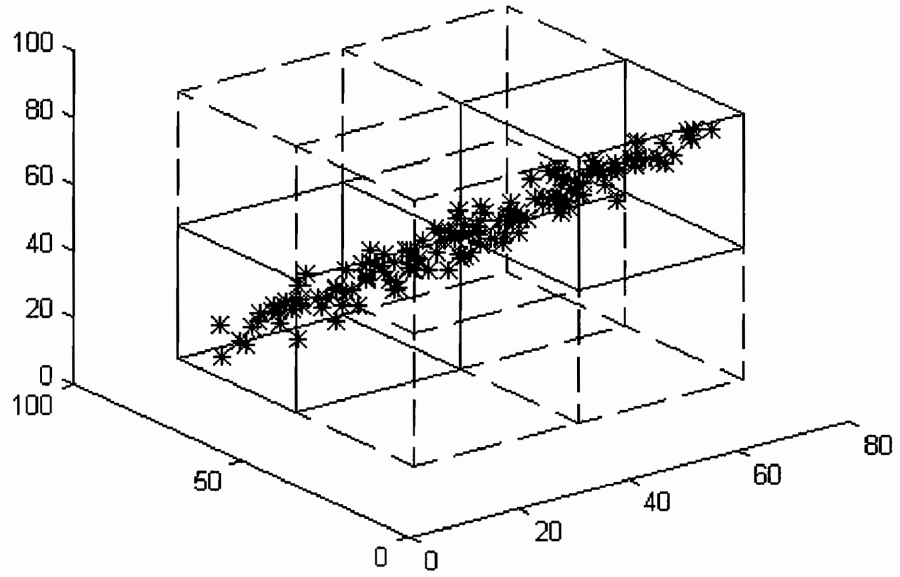


Figure 3-7. Split boxes corresponding to the large box in Figure 3-6. Only the boxes with solid lines are retained in the description of the data distribution. The other boxes that have dashed lines are removed from the description of the data distribution.

Chapter 4 System Overview

This chapter discusses system hardware, calibration, the system training procedure, and the classification mechanism.

4.1 General introduction

The imaging station consists of a host PC, a color video camera, a video digitizer, a set of fiber optic light sources and a photoelectric sensor. When components are moving on the conveyor, the photoelectric sensor detects the presence of a component and triggers image acquisition. The PC then directs the video digitizer to grab a digitized color image (256×64 pixels) from the color video camera. The PC analyzes the image and outputs an ID code of species and color for the component. The fiber optic light sources provide a stable and even illumination for the area sensed by the color video camera.

The software developed in this work consists mainly of three parts: system calibration, system training, and component identification. The system calibration software aids in setting up the system. The system training software contains the functions that perform color space quantization from a set of training samples, and set up a database of class models. The identification software analyzes an image and compares it to the models in the database to find a best match.

4.2 Imaging station and system calibration

Figure 4-1 shows the imaging station, and Table 4-1 lists all the equipment used in this system.

Table 4-1. Equipment list.

Equipment	*Quantity*	*Function*	*Manufacture & model*
Host computer	1	This PC controls image acquisition, processes the images, identifies the components, and provides user interface for system calibration and system training.	Dell 486DX33
Color video camera	1	This color video camera produces RGB color images.	JVC TK-1070U
Video digitizer	1	This PC plug-in board digitizes the outputs of the color camera as 24-bit RGB digital image. Each image can be up to 512×512 pixels.	Imaging Technology, Inc. CFG (Color Frame Grabber)
Photoelectric sensor (transmitter, receiver)	1	This sensor detects the presence of a component and triggers image acquisition.	Honeywell CP18RDND2 CP18EDX2
fiber optic light sources	2	These light sources provide even and stable illumination in the camera sensing area.	Dolan-Jenner A-240

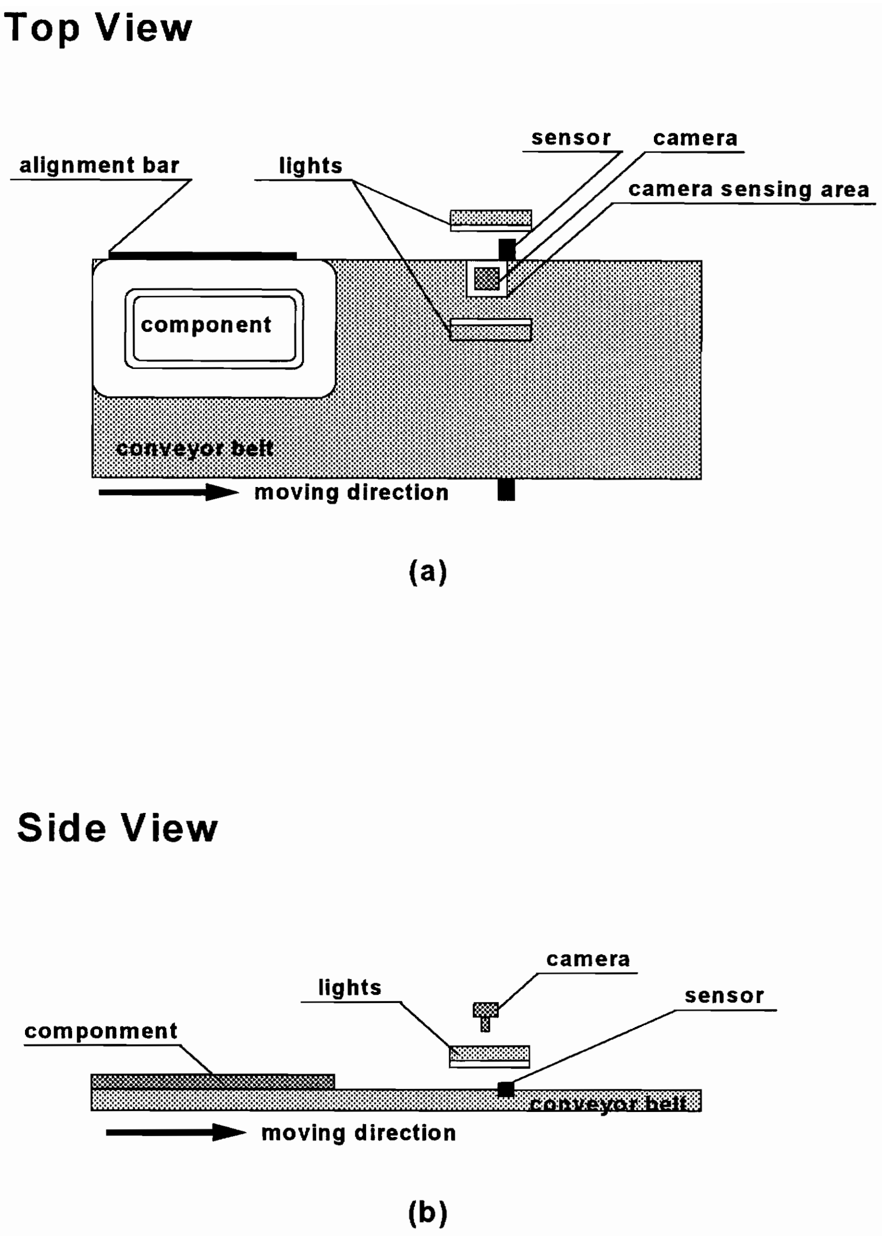


Figure 4-1. Imaging Station. (a) Top view. (b) Side view. Fiber optic lights are on both sides of the camera, causing even illumination in the camera sensing area.

The color camera is mounted vertically and its position was chosen to provide an imaging area as large as possible. In general, the image should be taken from a large area on the component. The images taken from a very small area may have extensive grain or have no grain, which may cause some trouble to identify the species, for example, oak that usually has more grain and maple that has less grain. However, the size of the imaging area is constrained by several issues. First of all, image of large size requires more processing time, which is a problem for a time-critical system. It is a wise to use a small image that still can provide both grain and non-grain information. Reducing the digital resolution may be a good choice. Secondly, it is difficult to provide an even and stable illumination on a large area, which not only increases the cost of the light system but also raises the problem for system maintenance. Finally, there are various sizes of components. All images must be taken from a common area without any profiles to avoid the illumination distortion caused by the reflection and shadow. Figure 4-2 illustrates this consideration.

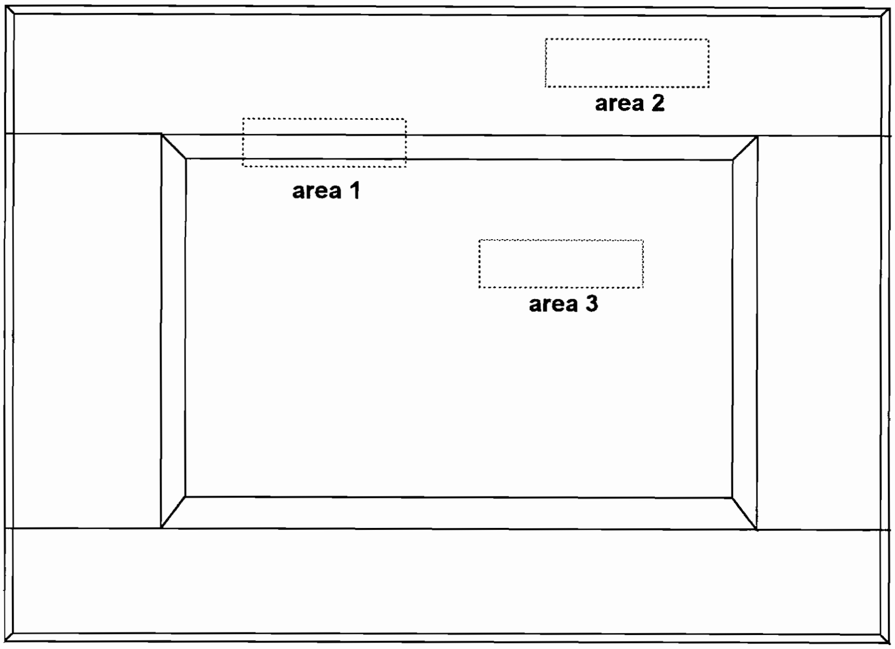


Figure 4-2. Example of the imaging area on a door. Area 1 cannot provide good image because of reflections and shadows. Area 2 and area 3 are good for imaging.

In this work, we chose area 2 in Figure 4-2 as a common imaging area, and the size is approximately 0.8”×3”. The width of the imaging area is determined to ensure that various components can avoid the impact of profiles. The length of the imaging area depends on the area on which light sources may provide an even illumination. The image size is 64×256 pixels, so that the resolution is about 90 pixels per inch.

Since illumination quality and stability are critical to the performance of the system, considerable effort has been devoted to selecting and positioning the lighting components. The fiber optic lights chosen in this system consist of DC power supply with a remote lamp housing and fiber optic illuminator. The DC power supply can be manually or remotely controlled and the output is rated to ±1%. With quartz halogen lamps, the light system is stable for a long period. The fiber optic illuminator provides even illumination in one dimension, and roughly even illumination in another dimension. Figure 4-3 demonstrates the distribution of the illumination intensity. For one source, we have observed that the illumination level in the imaging area varies by about 15~20 %. In order to improve this, two sets of fiber optic lights were mounted on both sides of the camera at an angle of 45°. Figure 4-4 illustrates the positions of the lights and the distribution of illumination. The two set lights overlap in the imaging area so that the illumination intensity in the area varies by less than 3%.

Since strong light causes reflection on the lighter components, the fiber optic light sources can not be set to a high illumination level. Experiments indicate that brighter lights may cause trouble in distinguishing the components of lighter shades, i.e., oak/frost and maple/frost, while lower illumination levels make it difficult to classify dark shades of wood. Practically, we set up the illumination to be as bright as possible, but without bright spots of reflection on white components. Meanwhile, we change the offset of the analog-to-digital converter of the CFG so that the darkest shade, cherry/toffee in this system, is near the origin of the RGB space.

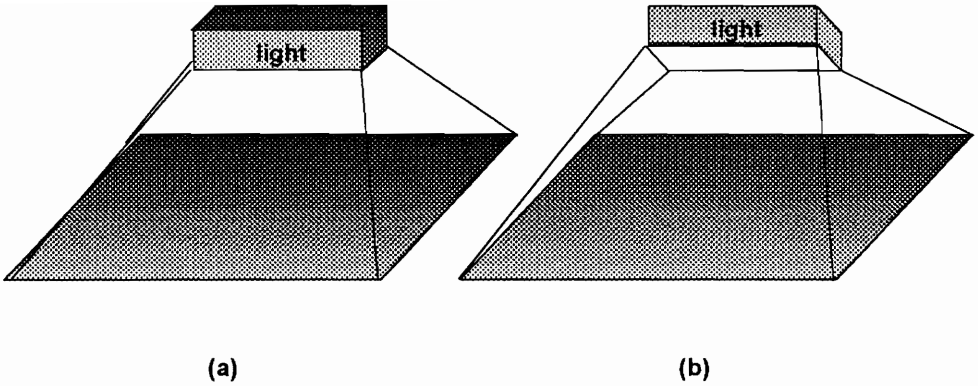


Figure 4-3. Illustration of the distribution of illumination intensity in the imaging area using one illuminator. (a) The illuminator is mounted vertically. (b) The illuminator is mounted at an angle.

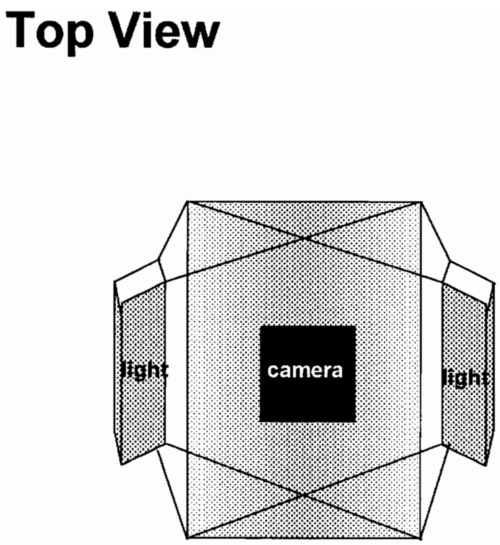


Figure 4-4. Illustration of the positions of the lights and the distribution of the illumination.

4.3 System training

During a supervised learning mode, the system analyzes training images of all of the given classes, and obtains a proper partition of the color space. As described in Chapter 3, color bins are formed by the intersections of descriptive parametric representations. These representations are statistically fit to the color histogram in the original color space. Then, the system sets up a database of models represented as a histogram based on the color partitions $\mathcal{P}$. Figure 4-5 shows the flow chart of the system training procedure. The first step of the training procedure is to collect representative images of components. Under control of the PC, the camera captures images of components when they are moving on the conveyor and these images are saved as binary files to hard disk. A proper class ID code is assigned to each image interactively.

After all images of training components have been collected, the system analyzes the images and determines the boundaries for all of classes. Each boundary separates the original color space into two parts. Inside the boundary are the color points that appear in the training components of the class while the outside points are not supposed to appear on the components of the class. A partition of the original color space is formed by the intersections of these boundaries. This partition serves as a color quantization scheme to reduce the number of the color histogram. So, each image corresponds to a color histogram based on the color partition. These histograms serve as the representations of the images.

Then, database models are chosen based on these representations. Two criteria are used to determine whether an image, $\mathbf{x}$, should be chosen as a model or not. One criterion is the similarity between this image and the database models of same class. Another is the difference between this image and the images of different classes. These two criteria can be calculated as follows:

$$D_1 = \min(d(\mathbf{x}, \mathbf{x}_i)),$$

$$D_2 = \min(d(\mathbf{x}, \mathbf{y}_j)),$$

where $d(\mathbf{x},\mathbf{x}_i)$ is one of the distance measures described in Chapter 2; $\mathbf{x}_i$ is all of the previously chosen database models that are of the same class with $\mathbf{x}$ while $\mathbf{y}_j$ is all of the training images of different classes with $\mathbf{x}$. If image $\mathbf{x}$ satisfies either of the following conditions,

$$D_1 > t_1 \qquad (1)$$

$$\frac{D_1}{D_2} > t_2 \qquad (2)$$

the image $\mathbf{x}$ should be chosen as a database model. The first condition indicates that image $\mathbf{x}$ is not similar to the models of the same class so that there must be a new subset of the class. The second condition indicates that some images similar to $\mathbf{x}$ may be confused with another class. It is necessary to add a new model to represent those images more precisely.

Therefore, one class may have several models that represent several subsets in the class. After sufficiently training, there are enough models in the database to represent all of the subsets for each class.

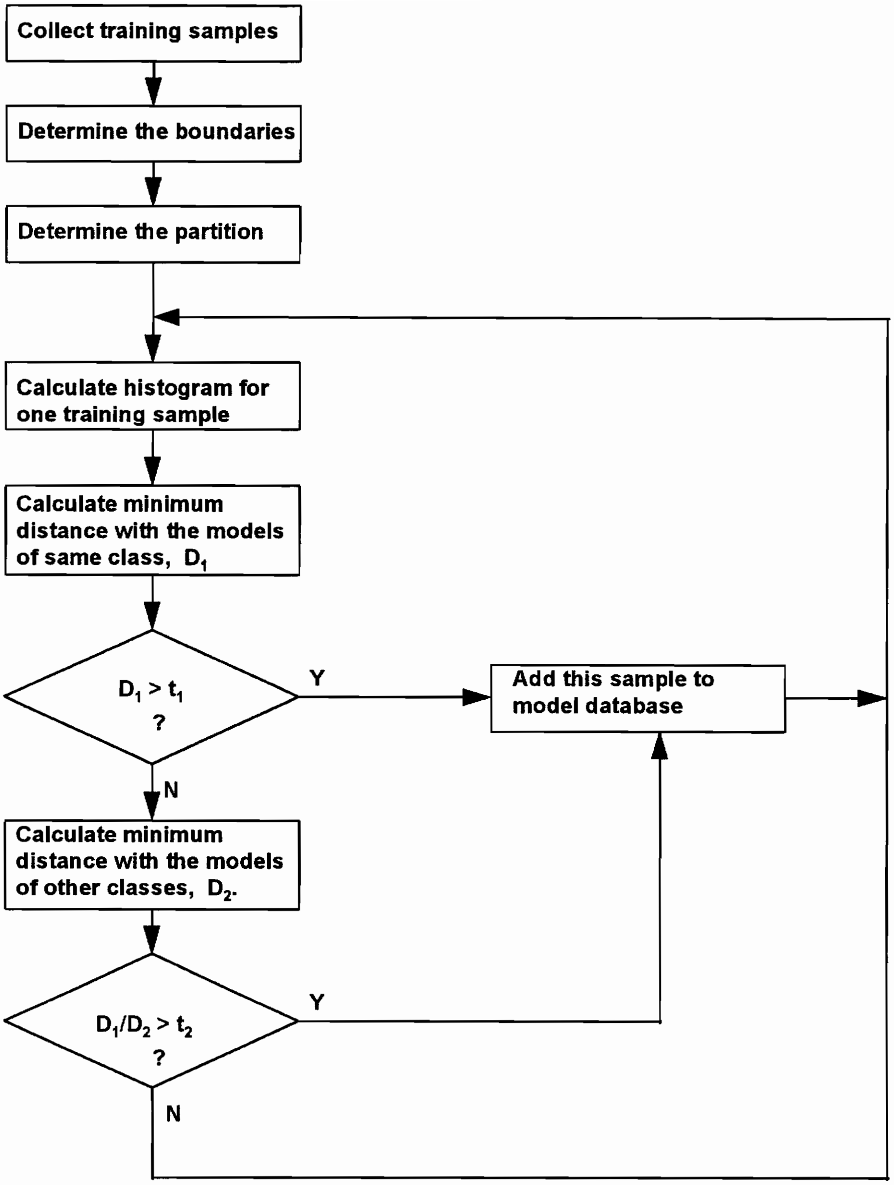

Collect training samples
Determine the boundaries
Determine the partition
Calculate histogram for one training sample
Calculate minimum distance with the models of same class, D1
D1 > t1 ?
Y
Add this sample to model database
N
Calculate minimum distance with the models of other classes, D2.
D1/D2 > t2 ?
Y
N

Figure 4-5. The flowchart of system training procedure.

4.4 Identification

Figure 4-6 shows the flowchart of the identification procedure. After training, the system can be used to identify the components in a real-time mode. All components move on the conveyor. When the of photoelectric sensor detects the presence of a component, it sends a signal to the computer. The computer then acquires an image and computes a color histogram based on the color quantization obtained during the system training procedure. Then, N minimum distances $\{ D_i \}$ are calculated for the classes using the distance measurement described in Chapter 2.

If $D_i > t_1$, for $i = 1, ..., N$, which means the component is not similar to any model of the classes, an "unknown" label is assigned to the component.

If $D_i / D_j < t_2$, $\forall\ i \neq j$, which means the component is similar enough to a model of class i comparing to the models of other classes, then class label ω_i is assigned to the component.

Otherwise, "unknown" is assigned to the component because the component is similar to the models of two or more classes so that it cannot be identified.

Here t_1 and t_2 are the chosen thresholds. Changing t_1 or t_2 may control the ratio of the "unknown" and "misclassified" cases.

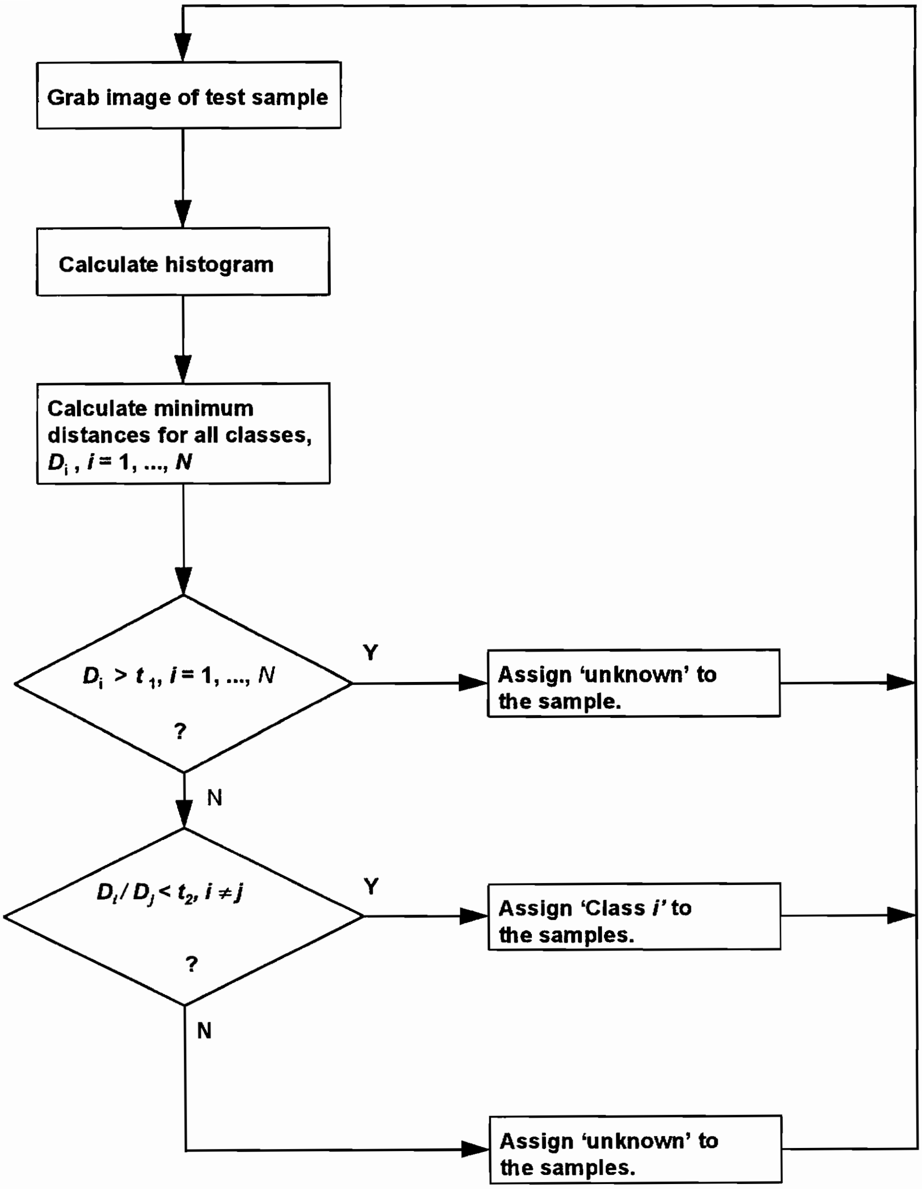

Grab image of test sample
Calculate histogram
Calculate minimum distances for all classes, D_i, $i = 1, ..., N$
$D_i > t_1$, $i = 1, ..., N$?
Y
Assign 'unknown' to the sample.
N
$D_i / D_j < t_2$, $i \neq j$?
Y
Assign 'Class i' to the samples.
N
Assign 'unknown' to the samples.

Figure 4-6. Flowchart of component identification procedure.

Chapter 5 Experimental Results

This chapter presents the experimental results obtained from the laboratory system. For comparison, the new approach developed here is compared with the more traditional approach of minimum centroid distance. The results obtained using different color space quantization approaches, different color subspace shapes and different histogram distance measurements are compared here.

Eight different classes of species and/or colors were used to test the laboratory system. They are the combinations of three species of wood (maple, oak, and cherry), and five colors (known as white, frost, natural, honey, and toffee). Table 5-1 shows the eight colors and species of the components that were used to test the system. Figure 5-1 contains example color images of each case from Table 5-1. These images were obtained using a JVC TK-1070U color video camera in the laboratory system. All images comprise 64 × 256 pixels, captured from an area of approximately 0.8"×3", and each pixel is represented by 24 bits, with 8 bits each for red, green and blue. Due to limitations of the color printer, all images showed here are only an approximation to the digitized images.

Table 5-1. Colors and species combinations used to test the system. The colors are listed in light-to-dark order.

	maple	oak	cherry
white	✓		
frost	✓	✓	
natural	✓	✓	
honey		✓	
toffee		✓	✓

A total of 480 images were obtained in the laboratory, similar to those shown in Figure 5-1. 95 images among them were manually selected as training samples to determine the 8 class representations, and the remaining 385 images were used to test the system. Table 5-2 and Table 5-3 show the test results that were obtained using rotated rectangular boxes to represent the color subspace for each class, and using the formula

$$d(\mathbf{x},\mathbf{y}) = \sum_{k} (x_k - y_k) \log(x_k / y_k)$$

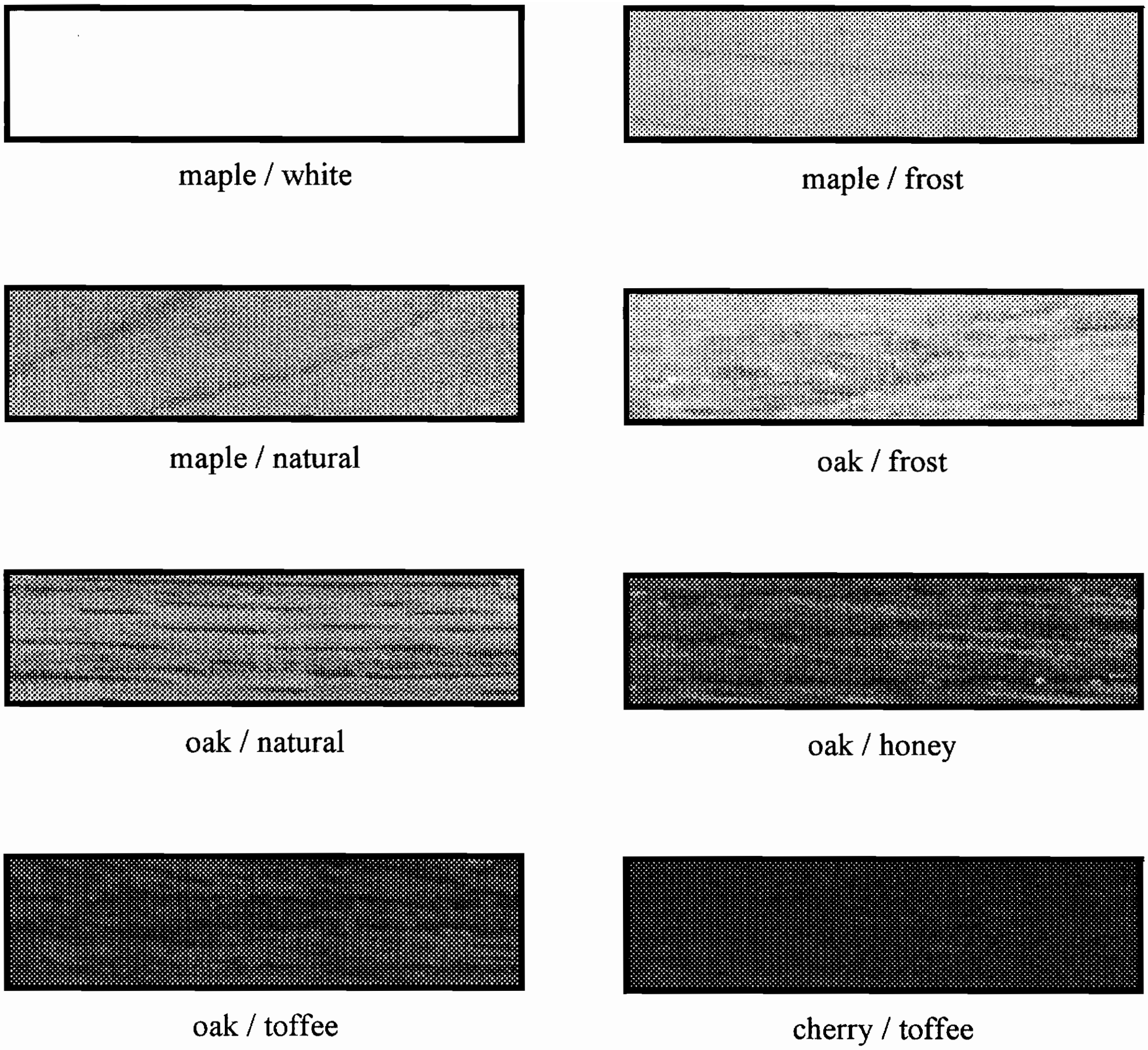
to measure the distance between two image histograms. The thresholds t_1 =0.5 and t_2 = 0.7 were picked to keep the number of misclassified cases smaller than or equal to 1% while making the unknowns small. In this case, over 96% of the test images were recognized correctly while only 1% were misclassified. Figure 5-2 shows some of the misclassified and unknown cases.

Table 5-2. Results using rotated rectangular boxes and $d(\mathbf{x},\mathbf{y}) = \sum_k (x_k - y_k)\log(x_k / y_k)$. The thresholds t_1 and t_2 are chosen to make misclassified cases equal to or less than 1%. The first column contains the names of the eight classes; the 2nd, 3rd, and 4th columns contain the number of training samples, number of models obtained by the training procedure, and number of test samples for each class, respectively. The remaining columns show the test results of identified cases, unknown cases, and misclassified cases for each class. Both the number and percentage of test samples are shown in this table.

	training samples	models	testing samples	identified		unknown		misclassified	
				#	%	#	%	#	%
maple / white	2	1	6	6	100	0	0	0	0
maple/frost	14	1	54	53	98.1	0	0	1	1.9
maple/natural	7	7	30	28	93.3	2	6.7	0	0
oak/frost	10	4	39	38	97.4	1	2.6	0	0
oak/natural	14	14	66	61	92.4	4	6.1	1	1.5
oak/honey	24	24	98	96	98.0	1	1.0	1	1.0
oak/toffee	20	20	78	74	94.9	3	3.8	1	1.3
cherry/toffee	4	4	14	14	100	0	0	0	0
overall	95	75	385	370	96.1	11	2.9	4	1.0

Table 5-3. Confusion matrix for the results given in Table 5-2. The top row contains the true class designations, and the left column represents the class labels that were assigned during the test procedure. So, the numbers in the diagonal are the number of correctly identified cases. The other entries except the last row are the number of misclassified cases. The last row contains the numbers of "unknown" cases.

	MW	MF	MN	OF	ON	OH	OT	CT
MW	6	0	0	0	0	0	0	0
MF	0	53	0	0	0	0	0	0
MN	0	0	28	0	0	0	0	0
OF	0	1	0	38	0	0	0	0
ON	0	0	0	0	61	0	0	0
OH	0	0	0	0	1	96	1	0
OT	0	0	0	0	0	1	74	0
CT	0	0	0	0	0	0	0	14
unknown	0	0	2	1	4	1	3	0



maple / white

maple / frost

maple / natural

oak / frost

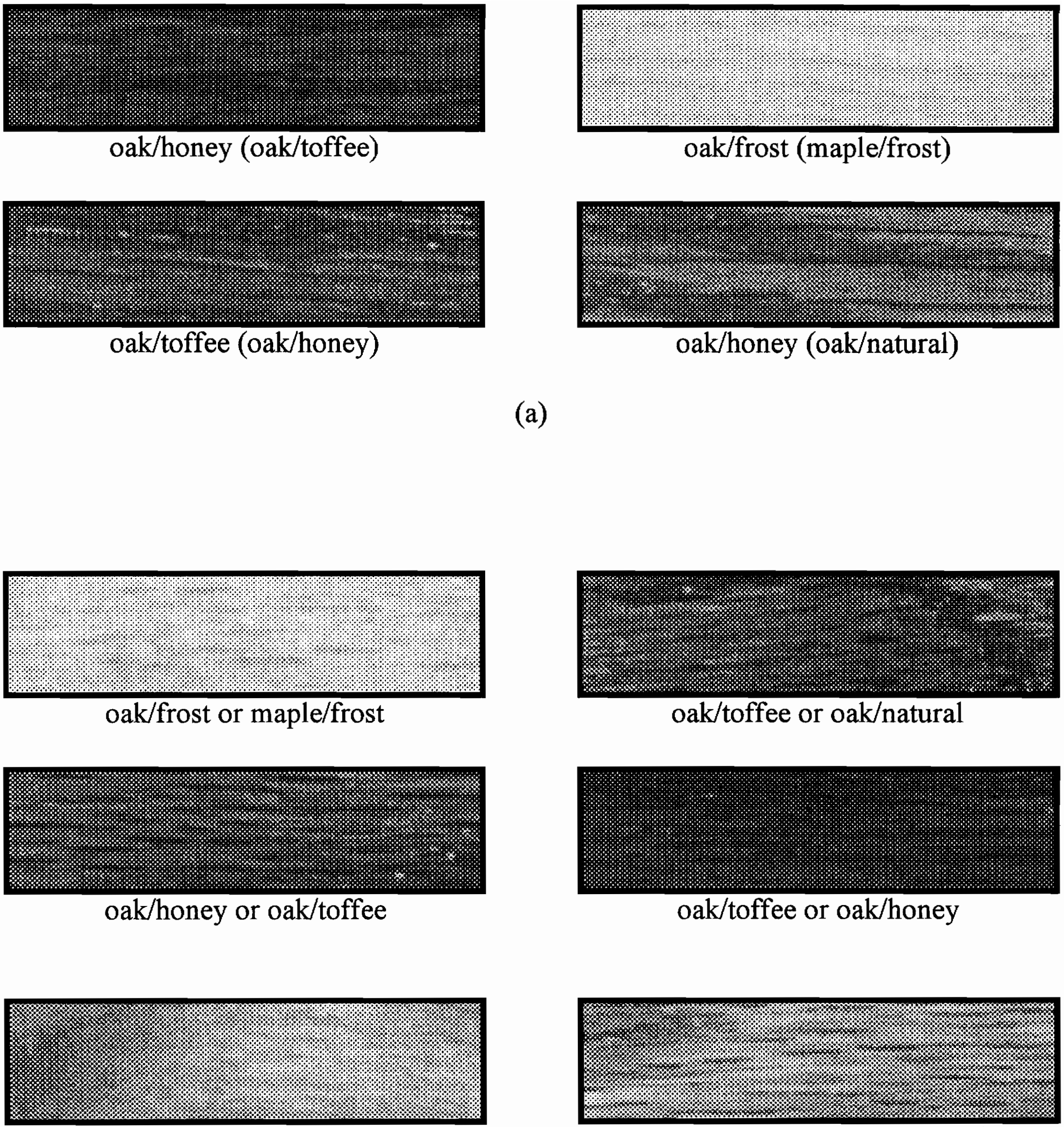
oak / natural

oak / honey

oak / toffee

cherry / toffee

Figure 5-1. Example images of all eight classes of color and species combinations used to test the system.



oak/honey (oak/toffee)

oak/frost (maple/frost)

oak/toffee (oak/honey)

oak/honey (oak/natural)

(a)

oak/frost or maple/frost

oak/toffee or oak/natural

oak/honey or oak/toffee

oak/toffee or oak/honey

maple/natural or oak/natural

oak/natural or maple/natural

(b)

Figure 5-2. Examples of (a) misclassified images and (b) unknown cases. In (a) the correct class is given in parentheses. In (b) each case was classified as unknown because it could not be distinguished from the class labels shown.

Table 5-4 shows the results using different representations for color subspaces and different distance measurements, but with the same set of training images and testing images as was used for Table 5-1. In order to compare the results between the methods of different parameteric shapes and/or different distance measurements in a same basis, which is the allowed misclassification percentage in Table 5-4, different thresholds are used. The thresholds that are described in Section 4.5 may control the results of classification. A smaller value of threshold t_1 may cause more "unknown" cases while a smaller value of threshold t_2 may reduce the number of unknowns but cause more errors. Table 5-5 contains the results using different values for t_2 and using rotated boxes and $d(\mathbf{x},\mathbf{y}) = \sum_k (x_k - y_k)\log(x_k / y_k)$.

In Table 5-4, the distance measurement $d(\mathbf{x},\mathbf{y}) = \sum_k (x_k - y_k)\log(x_k / y_k)$ yields better results than the other three distance measurements. This distance measurement is symmetric so that **x** and **y** play the same roles in the distance measure. The value of $d(\mathbf{x},\mathbf{y}) = \sum_k (x_k - y_k)\log(x_k / y_k)$ is strongly influenced by the bins that have large value of $|x_k - y_k|$. The distance measurement $d(\mathbf{x},\mathbf{y}) = \sum_k x_k \log(x_k / y_k)$ mainly depends on the bins that have large x_k, and this means that the test image plays a more important role than the models of the database. For the distance measurement $d(\mathbf{x},\mathbf{y}) = \sum_k |x_k - y_k|$, all bins make same contributions to the distance measure so it is detail-oriented.

Using different parameteric shapes of color cells also yields different results. Rotated boxes and ellipsoids may provide more accurate boundaries for the color subspaces, but they depend on the training samples to a large extent. Different training samples cause different orientations for the rotated boxes and ellipsoids. Figure 5-3, Figure 5-4 and Figure 5-5 show the rectangular boxes, ellipsoids, and rotated boxes, which are used as the parameteric descriptions of the eight classes.

Table 5-4. Results of different color space representations and histogram distance measurements. Different thresholds t_1 and t_2 are chosen to make misclassified cases equal to or less than 1%. There is a total of 385 test images.

$$d(\mathbf{x},\mathbf{y}) = \sum_k (x_k - y_k)\log(x_k / y_k)$$

	identified		unknown		misclassified	
	#	%	#	%	#	%
box	218	56.6	164	42.6	3	0.8
ellipsoid	258	67.0	123	32.0	4	1.0
rotated box	370	96.1	11	2.9	4	1.0

$$d(\mathbf{x},\mathbf{y}) = \sum_k x_k \log(x_k / y_k)$$

	identified		unknown		misclassified	
	#	%	#	%	#	%
box	169	43.9	213	55.3	3	0.8
ellipsoid	243	63.1	139	36.1	3	0.8
rotated box	361	93.8	20	5.2	4	1.0

$$d(\mathbf{x},\mathbf{y}) = \sum_k (x_k - y_k)^2$$

	identified		unknown		misclassified	
	#	%	#	%	#	%
box	44	11.4	337	87.5	4	1.0
ellipsoid	103	26.8	280	72.7	2	0.5
rotated box	288	74.8	94	24.4	3	0.8

$$d(\mathbf{x},\mathbf{y}) = \sum_k |x_k - y_k|$$

	identified		unknown		misclassified	
	#	%	#	%	#	%
box	77	20.0	304	79.0	4	1.0
ellipsoid	148	38.4	236	61.3	1	0.3
rotated box	334	86.8	47	12.2	4	1.0

Table 5-5. Results using $t_1 = 0.5$ and different values of threshold t_2. The color cells are defined using rotated boxes and the distance measurement is $d(\mathbf{x},\mathbf{y}) = \sum_k (x_k - y_k)\log(x_k / y_k)$. There is a total of 385 test images.

threshold	identified		unknown		misclassified	
t_2	#	%	#	%	#	%
0.20	228	59.2	157	40.8	0	0
0.30	283	73.5	101	26.2	1	0.3
0.40	327	84.9	56	14.5	2	0.6
0.50	355	92.2	27	7.0	3	0.8
0.60	365	94.8	17	4.4	3	0.8
0.70	370	96.1	11	2.9	4	1.0
0.80	377	97.9	3	0.8	5	1.3

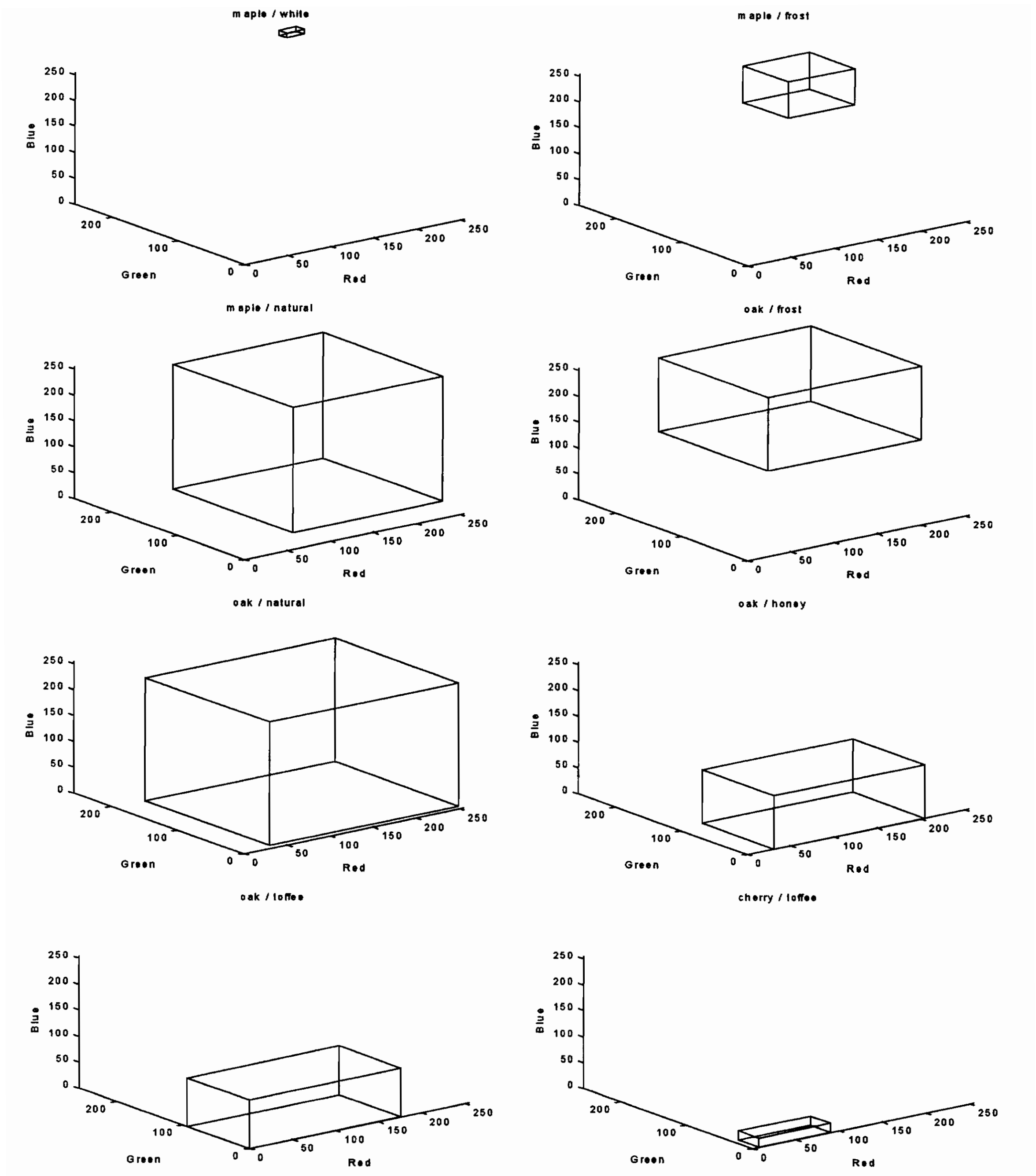


Figure 5-3. Rectangular box representations of all eight classes. Each rectangular box is fit to the histogram of color points that are found in the training images of corresponding class. Intersections and the complement of these rectangular boxes determine a partition of the original color space.

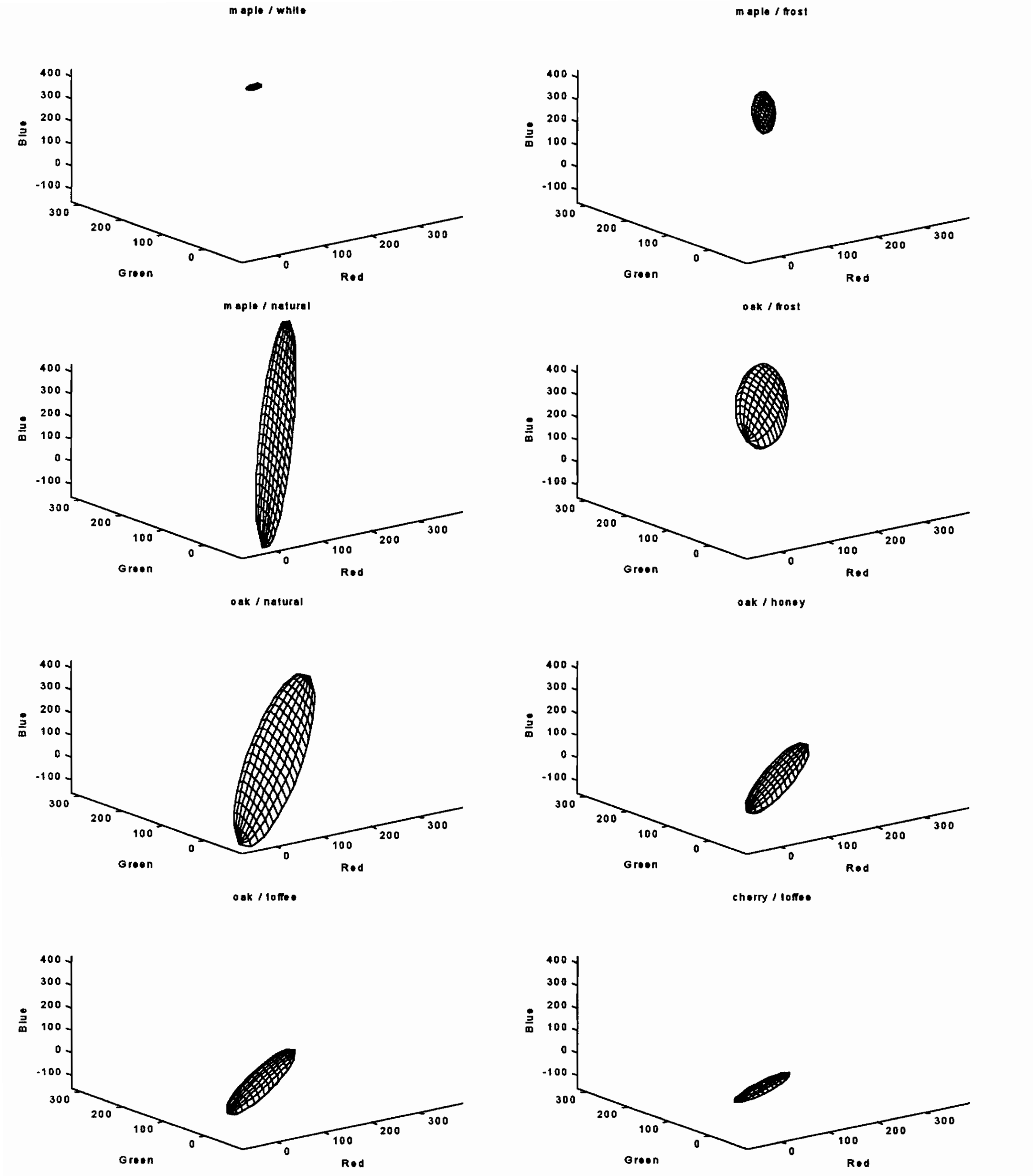


Figure 5-4. Ellipsoid representations of all eight classes. All of the ellipsoids are determined by PCA to fit color points that are found in the training images of corresponding class. Intersections and the complement of these ellipsoids determine a partition of the original color space. The some parts of the ellipsoids are out of the the color cube.

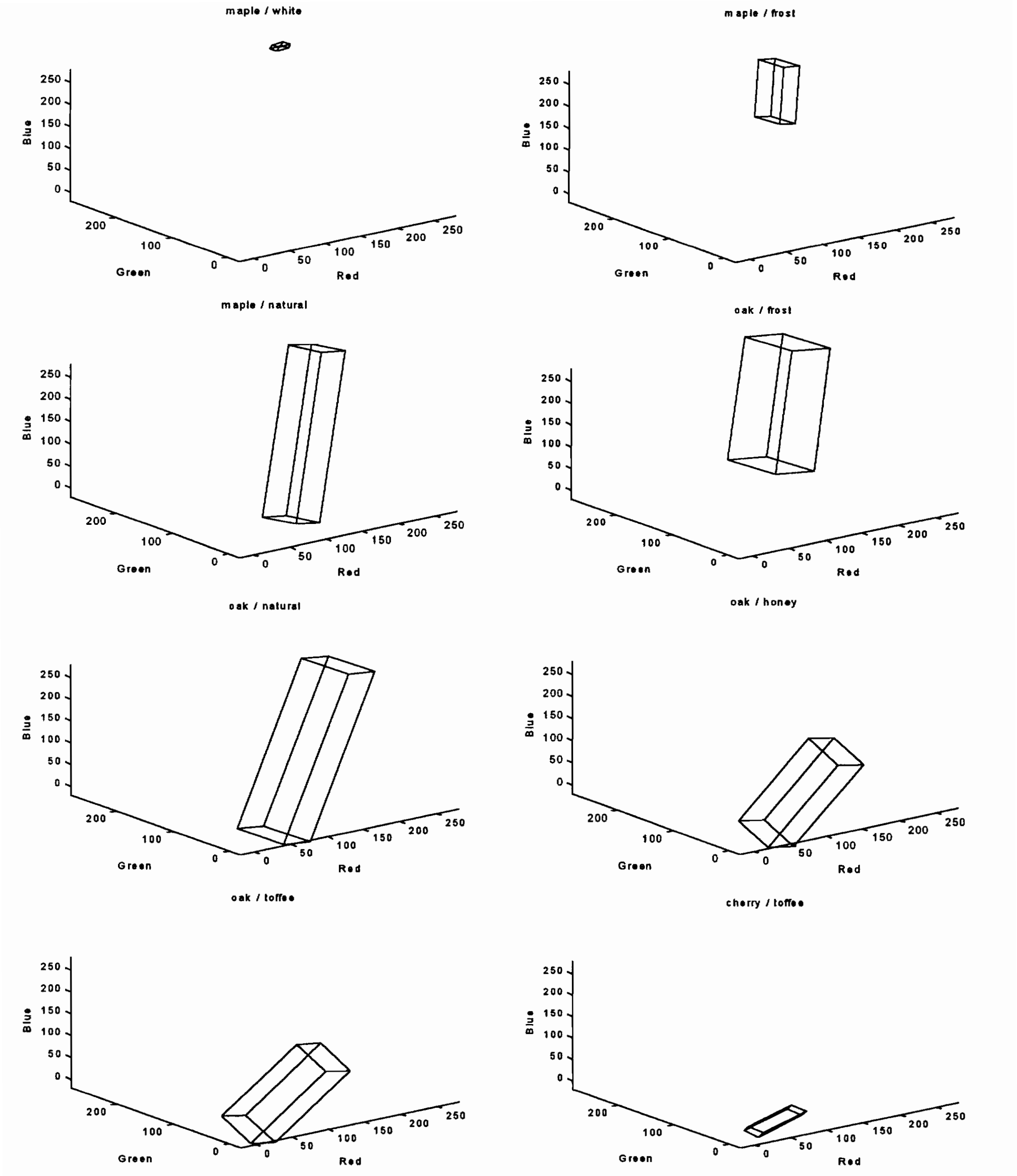


Figure 5-5. Rotated box representations of all eight classes. All of the rotated boxes are determined by PCA to fit color points that are found in the training images of corresponding class. Intersections and the complement of these rotated boxes determine a partition of the original color space.

Table 5-6 contains the test results using the more common uniform quantization with different numbers of color bins. The sets of training images and test images are shown in Table 5-2, which are same as the sets used for the results in Table 5-4.

Table 5-6 indicates that using uniform quantization with different numbers of color bins gets almost same results except the cases that have 4^3 and 256^3 color bins. In this work, because most images do not have sharp edges between grain parts and non-grain parts, the thresholds for the uniform quantization may not cause serious trouble. However, it is difficult to improve the results of uniform quantization by selecting the training images manually. There are 95 models for most cases of uniform quantization. The results become worse if the number of models is reduced by changing the thresholds t_1 and t_2 described in Section 4.4.

Uniform quantization method needs to process a lot of data when a test sample and a model are compared. This is a problem for a time-critical system. For the case of $8^3 = 512$ bins in the histogram, using uniform quantization, 75.3% correct identification and 23.9% unknowns result. The color quantization approach developed in this work requires fewer bins in a histogram for comparable accuracy. Table 5-7 shows the bins generated in the color quantization methods developed here. There are fewer than 20 bins for aligned boxes and ellipsoids, while there are 24 bins for rotated boxes. These color bins are obtained from the intersections of the aligned boxes, ellipsoids, and rotated boxes shown in Figure 5-3, Figure 5-4, and Figure 5-5. The aligned boxes, ellipsoids, and rotated boxes are generated based on the training images during the system training procedure. In Table 5-7, the term “mw” means that only the colors appeared on the images of maple / white class. The term “MW” should be considered as the simple form of “$\mathrm{mw}\cap\overline{\mathrm{mf}}\cap\overline{\mathrm{mn}}\cap\overline{\mathrm{of}}\cap\overline{\mathrm{on}}\cap\overline{\mathrm{oh}}\cap\overline{\mathrm{ot}}\cap\overline{\mathrm{ct}}$”. Similarly, “$\mathrm{mn}\cap\mathrm{of}\cap\mathrm{on}$” should be viewed as “$\mathrm{mn}\cap\mathrm{of}\cap\mathrm{on}\cap\overline{\mathrm{mw}}\cap\overline{\mathrm{mf}}\cap\overline{\mathrm{oh}}\cap\overline{\mathrm{ot}}\cap\overline{\mathrm{ct}}$” while the term “unknown” is equivalent to “$\overline{\mathrm{mw}}\cap\overline{\mathrm{mf}}\cap\overline{\mathrm{mn}}\cap\overline{\mathrm{of}}\cap\overline{\mathrm{on}}\cap\overline{\mathrm{oh}}\cap\overline{\mathrm{ot}}\cap\overline{\mathrm{ct}}$”.

Table 5-6. Results of uniform quantization. There is a total of 385 test images. The distance measurement is $d(\mathbf{x},\mathbf{y}) = \sum_k (x_k - y_k)\log(x_k / y_k)$. Different thresholds t_1 and t_2 are chosen to make misclassified cases equal to or less than 1%.

# of bins	# of models	identified		unknown		misclassified	
		#	%	#	%	#	%
256^3	95	233	60.5	148	38.4	4	1.0
128^3	95	292	75.8	90	23.4	3	0.8
64^3	95	295	76.6	87	22.6	3	0.8
32^3	95	304	79.0	77	20.0	4	1.0
16^3	94	292	75.8	91	23.6	2	0.5
8^3	94	290	75.3	92	23.9	3	0.8
4^3	93	135	35.1	246	63.9	4	1.0

Table 5-7. Histogram bins of adaptive color space quantization methods. The "mw", "mf", "mn", "of", "on", "oh", "ot", and "ct" represent the subspace contained by the parameteric boundaries of Maple/White, Maple/Frost, Maple/Natural, Oak/Frost, Oak/Natural, Oak/Honey, Oak/Toffee, and Cherry/Toffee, respectively. These boundaries are shown in Figure 5-3, Figure 5-4, and Figure 5-5. The "MW" and "OF" are equivalents of "$\mathrm{mw}\cap\overline{\mathrm{mf}}\cap\overline{\mathrm{mn}}\cap\overline{\mathrm{of}}\cap\overline{\mathrm{on}}\cap\overline{\mathrm{oh}}\cap\overline{\mathrm{ot}}\cap\overline{\mathrm{ct}}$" and "$\mathrm{of}\cap\overline{\mathrm{mw}}\cap\overline{\mathrm{mf}}\cap\overline{\mathrm{mn}}\cap\overline{\mathrm{on}}\cap\overline{\mathrm{oh}}\cap\overline{\mathrm{ot}}\cap\overline{\mathrm{ct}}$". Similarly, the term "$\mathrm{mf}\cap\mathrm{mn}\cap\mathrm{of}$" should be viewed as "$\mathrm{mf}\cap\mathrm{mn}\cap\mathrm{of}\cap\overline{\mathrm{mw}}\cap\overline{\mathrm{on}}\cap\overline{\mathrm{oh}}\cap\overline{\mathrm{ot}}\cap\overline{\mathrm{ct}}$". The term "unknown" represents "$\overline{\mathrm{mw}}\cap\overline{\mathrm{mf}}\cap\overline{\mathrm{mn}}\cap\overline{\mathrm{of}}\cap\overline{\mathrm{on}}\cap\overline{\mathrm{oh}}\cap\overline{\mathrm{ot}}\cap\overline{\mathrm{ct}}$".

rectangular boxes		ellipsoid		rotated boxes	
bin	intersections	bin	intersections	bin	intersections
1	MW	1	MW	1	MW
2	OF	2	MN	2	MF
3	OT	3	OF	3	MN
4	mf∩of	4	ON	4	OF
5	mn∩of	5	mf∩of	5	ON
6	mn∩on	6	mn∩of	6	OH
7	ot∩ct	7	mn∩on	7	OT
8	mf∩mn∩of	8	on∩oh	8	mf∩of
9	mn∩of∩on	9	on∩ot	9	mn∩of
10	mn∩on∩oh	10	mn∩of∩on	10	mn∩on
11	on∩oh∩ot	11	mn∩on∩oh	11	mn∩of
12	oh∩ot∩ct	12	on∩oh∩ot	12	on∩oh
13	mf∩mn∩of∩on	13	on∩ot∩ct	13	oh∩ot
14	mn∩on∩oh∩ot	14	mn∩on∩oh∩ot	14	ot∩ct
15	on∩oh∩ot∩ct	15	on∩oh∩ot∩ct	15	mn∩of∩on
16	unknown	16	mn∩on∩oh∩ot∩ct	16	mn∩on∩oh
		17	unknown	17	mn∩oh∩ot
				18	on∩oh∩ot
				19	on∩oh∩ct
				20	oh∩ot∩ct
				21	mn∩on∩oh∩ot
				22	on∩oh∩ot∩ct
				23	mn∩on∩oh∩ot∩ct
				24	unknown

For comparison, Table 5-8 shows the results when the more traditional minimum centroid distance method is used. In this method, each image is represented using its average color. A test image is assigned label ω_i iff $D_i / D_j <$ threshold, $\forall\ i \neq j$, where D_i is the distance between the centroid of the test image and ith reference centroid. There are a lot of error and unknown cases here, especially for maple/frost and oak/frost. The color distributions of these classes overlapped with each other to such a large extent that the centroids are insufficient to distinguish them.

Table 5-8. Results of minimum centroid distance method. The threshold is 0.5.

	reference centroids			identified		unknown		misclassified	
	red	green	blue	#	%	#	%	#	%
maple/white	250.58	253.97	252.46	6	100	0	0	0	0
maple/frost	180.68	159.94	237.56	6	11.1	30	55.5	18	33.3
maple/natural	186.76	124.91	135.37	24	80.0	4	13.3	2	6.7
oak/frost	176.73	159.19	242.84	18	48.6	17	46.0	4	10.8
oak/natural	135.43	69.95	58.39	57	86.4	4	6.1	5	7.6
oak/honey	106.10	40.19	12.46	80	81.6	9	9.2	9	9.2
oak/toffee	65.18	21.36	6.93	59	75.6	6	7.7	13	16.7
cherry/toffee	45.00	6.06	0.75	8	57.1	5	35.7	1	7.1
overall				258	67.0	75	19.5	52	13.5

Chapter 6 Conclusion

This work has developed a machine vision system that can identify the colors of textured objects with satisfactory accuracy. It has been motivated by the need for an industrial color classification system that can handle the cases of large texture variations.

The major contribution of this work is a novel approach to quantization of the color space. The approach developed here makes use of meaningful color representations that focus on the similarities and differences of different class representatives. Compared with the more traditional uniform quantization approach, it dramatically reduces the data required when two histograms are compared. It also provides higher recognition accuracy. In addition, the approach is adaptive because the color space quantization is based on the training samples.

The system has shown a successful performance in laboratory experiments. A total of 385 samples has been used to test the system while only 95 samples have been used to training the system. There were 96.7% cases classified correctly. Only 1% of the cases were misclassified. The remaining 2.3% were unknowns. Since the illumination level fluctuated when these samples were taken, the results may be better if the illumination was more stable.

Future research may involve the following issues:

- Improve the light sources to make the illumination more stable.
- Improve the algorithm of histogram distance measurement to reduce the image processing time.
- Take several images from one component to reduce the impact of noise caused by defects of the component.

References

1. B. Yuan, *Automatic Classification of Wooden Cabinet Doors Using Computer Vision*, M.S. thesis, 1994.

2. F. Bari, *A Machine Vision System for Classifying Rectangular Cabinet Frames*, M.S. thesis, 1994.

3. C. J. DelGigante, *A Recognition System for Rectangular Compontents Based on Structural decomposition*, M.S. thesis, 1995.

4. Z. Huang, *Classification of Species and Color of Finished Wooden Components*, M.S. thesis, 1996.

5. D. H. Ballard and C. M. Brown, *Computer Vision*, Prentice-Hall Inc., 1982.

6. H. J. George and G. Donald, "Colour Spaces for Computer Graphics," *Computer Graphics*, vol. 12, no. 3, pp. 20-25, 1978.

7. F. Perez and C. Koch, "Toward Color Image Segmentation in Analog VLSI: Algorithm and Hardware," *International Journal of Computer Vision*, vol. 12, no. 1, pp. 17-42, 1994.

8. R. Nevatia, "A Color Edge Detector and Its Use in Scene Segmentation," *IEEE Trans. Syst. Man, Cyb*. 7(11), pp. 820-826.

9. M. Adel, D. Wolf, R. Vogrig and R. Husson, "Evaluation of Colour Spaces in Computer Vision Application of Wood Defects Dectection," *Proceedings, IEEE International Conference on Systems, Man, and Cybernetics*, pp. 499-504, 1993.

10. Y. Otha, T. Kanada and T. Sakai, "Color Information for Region Segmentation," *Computer Vision, Graphics, and Image Processing*, vol. 13, pp. 222-241, 1980.

11. P. Heckbert, "Color Image Quantization for Frame Buffer Display," *Computer Graphics*, vol. 16, no. 3, pp. 297-307, July 1982.

12. S. Tominaga, "Color Classification of Natural Color Image," *Color Research and Applications*, vol. 17, pp. 230-239, 1992.

13. R. Massen, "Method and Arrangement for Automatic Optical Classification of Plants," *U.S. Patent 5,253,302*, Oct. 1993.

14. T. Uchiyama and M. A. Arbib, "Color Image Segmaentation Using Competitive Learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 12, pp. 1197-1206, Dec. 1994.

15. J. Bryant, "On the Clustering of Multidimensional Pictorial Data," *Pattern Recognition*, vol. 11, pp. 115-125, 1979.

16. Y. Linde, A. Buzo and R. M. Gray, "An Algorithm for Vector Quantizer Design," *IEEE Trans. Commun.*, vol. COM-28, pp. 84-95, 1980.

17. T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons, 1991.

18. A. D. Brink and N. E. Pendock, "Minimum Cross-entropy Threshold Selection", *Pattern Recognition*, vol. 29, no. 1, pp. 179 - 188, 1996.

19. R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, vol. I (1992) and vol. II (1993), Addison-Wesley.

20. R. W. Ehrich and J. P. Forth, "A View of Texture Topology and Texture Description," *Computer Graphics and Image Processing*, vol. 8, pp. 174-202, 1978.

21. A. Rosenfeld and B. S. Lipkin, Texture Synthesis in Processing and Psychopictorics, New York, Academic Press, 1970.

22. J. Scharcanski, J. K. Hovis, and H. C. Shen, “Representing the color aspect of texture images,” *Pattern Recognition Letters*, vol. 15, Iss. 2, pp. 191-197, 1994.

23. K. Y. Song, J. Kittler, M. Petrou, and I. Ng, “Chromato-Structural Approach towards Surface Defect Detection in Random Textured Images,” *Proceedings of SPIE*, vol. 2183, pp. 193-203, 1994.

24. T. Caelli and D. Reye, “On the Classification of Image Regions by Color, Texture and Shape,” *Pattern Recognittion*, vol. 26, no. 4, pp. 461-470, 1993.

25. M. Swain and D. Ballard, “Color Indexing,” *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11-32, 1991.

26. F. Ennesser and G. Medioni, “Finding Waldo, or Focus of Attention Using Local Color Information,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp.805-809, 1995.

27. Y. Wang and B. J. Griffiths, “Colored Object Inspection with N-tuple System,” *Journal of Electronic Imaging*, 3(2), pp.127-132, 1994.

28. J. Hafner, H. S. Sawhney, W. Equitz, M. Flickner, and W. Niblack, “Efficient Color Histogram Indexing for Quadratic Form Distance Functions,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 17, no. 7, pp.729-736, 1995.

29. R. S. Mickalski, “Knowledge Acquisition Through Conseptural Clustering: a Theoretical Framework and an Algorithm for Partitioning Data into Conjunctive Concepts,” *International Journal of Policy Analysis and Information System*, 4(3), pp. 219-244, 1980.

30. L. Abbott, “Cohesion Methods in Inductive Learning,” *Computational Intelligence*, vol. 3, no. 4, pp. 267-282, 1987.

VITA

Yuedong Zhao was born on June 24, 1959, in Shanghai, China. He received a Bachelor of Science degree in Electrical Engineering in Shanghai Jiao Tong University in 1987. Then he worked for Computer Center in East China University of Science and Technology for five years. He received a Master of Science degree in Electrical Engineering from Virginia Polytechnic Institute and State University in 1996.

Yuedong Zhao