COLOR IMAGE QUANTIZATION & HALFTONING BASED ON HUMAN VISUAL PERCEPTION

Yu Mei Ping, Lily

A dissertation[†] submitted in Partial fulfillment of

the requirements for the degree of

Master of Philosophy

Department of Electronic and Information Engineering

The Hong Kong Polytechnic University

May 2003

[†] This research was supported by the Hong Kong Polytechnic University Grant A/C No. GW-098

ABSTRACT

of the thesis entitled

"Color Image Quantization & Halftoning Based on Human Visual Perception"

submitted by Yu Mei Ping

for the degree of Master of Philosophy

at The Hong Kong Polytechnic University in December 2002

In this dissertation, we propose two algorithms that address two important issues in color quantization. The first one being the incorporation of spatial or contextual information of an input image to the process of color quantization, thus allowing quantization to give priorities to different regions of the image and focus on the regions of the image having important color information.

In considering the way the human visual system deals with color difference, the second algorithm is concerned with color space. The algorithm is a simple but effective color quantization technique, which performs 3D frequency diffusion in RGB color space. By manipulating a diffusion filter, a cross-space operation is achieved, obtaining quantization effect in a color space without transforming the image to that space.

For most color displays, processing of colors is performed on the R, G and B components. The requirement for human vision system, however, would favour a different scheme because the Euclidean distance in the RGB color space is not consistent with the way the human visual system deals with color difference. Theoretically, a luminance–chrominance color space that correlates with human color perception is more suitable to be adopted than the RGB for many applications. The transformation processes between color spaces is, however, bound up with computational errors. As a result, the overall performance of any color quantization scheme that involves conversion between color spaces may not be better than a quantization completely in RGB.

This dissertation begins with a brief history of color quantization and halftone development as well as descriptions of major techniques. Followed by that, the essential color knowledge encountered in the context of color image quantization, as well as the fundamental concepts, models, metrics and their formulation for evaluating the quality of quantization outputs is examined. Experimental results show that our algorithms perform better in some aspects than two popularly used algorithms for color quantization. Finally, description of the algorithms representing the core of the research along with in depth discussions of each algorithm will be given.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to Dr. K.C. Lo for his constant guidance, invaluable advice and sustained interest throughout the preparation of my work. Special thanks is also extended to Dr. Y.H. Chan and those who have assisted in providing and sharing information, insights and their valuable experience in directing my way to complete this project.

STATEMENTS OF ORIGINALITY

We summarize below the major contribution of this thesis.

- This dissertation proposed two novel algorithms, the Contextual algorithm and the 3D FD algorithm, that address two different problems that commonly encountered in performing color quantization.

- The contextual algorithm is a color quantization algorithm based on the contextual information of the input image. In this algorithm, a method for evaluating the contextual information of the input image is provided.

- Experimental results indicate a good performance of the Contextual algorithm with the capability to focus on the regions of an image having important color information.

- The 3D FD algorithm treats color quantization as a problem of vector error diffusion. The process requires only simple arithmetical calculations. The algorithm works well in all color space.

- The dissertation demonstrates that the proposed 3F FD algorithm is able to perform a cross-space operation without transforming the image between color spaces by manipulating the geometric shape and coefficients of the 3D diffusion.

- It is found that the 3D FD algorithm, while operating in the RGB color space, is capable of producing similar effect as it does under the YUV color space with no transformation of the image required.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

Full color displays typically use 24 bits to represent the color of each pixel on the screen. With 8 bits for each of the primary component, Red, Green and Blue, approximately 16.7 million possible colors can be generated in a 24 bit full RGB color image. In many imaging applications, it is often desirable to represent images with as few colors as possible while at the same time retain the optimal perceived quality of the images. This is generally achieved by color quantization, in which the number of colors in an image is reduced to a maximum of 256 simultaneously displayable colors and followed by subsequent halftoning to create the illusion of a continuous tone image. Although 24-bits/pixel full color displays have already become affordable for most users, color quantization retains its practical value in many graphic functions such as animation and transparency. In addition, the reduced image file size resulting from color quantization will facilitate storage and transmission efficiencies, particularly via the Internet.

Color quantization basically involves algorithms for performing two tasks. The first task is called colormap design, which is to select the best possible set of representative colors for an image. These colors will comprise the image colormap.

The second task is color mapping that is to locate the closest representative for each original color and replace the original color by its best representative from the colormap. While this mapping of color to its nearest neighbor is optimal from a minimum-average-error standpoint, it often produces objectionable contours in smooth image regions. To solve this problem, digital halftoning is often carried out as a subsequent processing step to create an illusion of continuous tone image by exploiting the properties of the human visual system (HVS).

In the remainder of this chapter, we present a brief history of color quantization and halftone development as well as descriptions of major techniques. The attempt is to classify these techniques from a technological as well as a historical viewpoint. In addition, the objective of this research and an overview of the dissertation will be given at the end of this chapter.

1.1 Color Quantization

Over the past years, extensive research works have been conducted in the area of color quantization and the algorithms proposed can be image dependent or independent. It is, however, confirmed that the use of a fixed image independent colormap, though is easier to implement and less computational intensive, usually

produces unacceptable results when comparing with the use of an adaptive image dependent color palette.

Proposed by Tom Boyle and Andy Lippman at the AMG in the summer of 1978, the *Popularity algorithm* [16] was the first algorithm proposed to adaptively quantize color images with an image dependent colormap. In this algorithm, colormap is constructed simply by selecting a number of the most frequently occurred colors from the histogram of the true image colors. While the popularity algorithm is extremely fast and produces good results for many images, it performs poorly on images with a wide range of colors because colors in sparse region of the color space is often neglected. Braudaway [6] suggested an improved popularity algorithm that prevents the concentration of too many representatives in the neighborhood of one distribution peak by allocating the palette colors sequentially and modifying the histogram after each allocation. This leads to a more uniform distribution of the representatives in the image histogram. With the attempt to use each palette color to represent an equal number of true colors, Heckbert [13] provided an alternative to determine a colormap by a recursive process of splitting the largest cluster into two equal halves. The algorithm is commonly referred to the *Median cut* algorithm. To improve the initial colormap generated, both Heckbert and Braudaway apply the *LBG* algorithm [30] to iteratively map the colors in the image to their nearest

representatives. Since the LBG algorithm converges only to a local minimum, this step often yields only slight improvement.

More significant improvement is obtained by selecting the colors through a sequential splitting process in which the color space of the original image is split iteratively according to certain preference criteria until the expected number of subspaces is reached. The representative color of each subspace will then become the quantized color. A number of algorithms using different splitting procedures and selection criteria have been proposed, some of the examples includes the V*ariance minimization* algorithm [65]-[66] suggested by Wan et al., in which more clusters are assigned to regions with large variance thus reducing the total quantization errors of the partition. Some of the other examples include the center-cut [21], the *Octree* quantization [12]-[13], the *Orchard-Bouman splitting* algorithm [41], and Wu's [67] algorithm by *Principal analysis*.

Median Cut and Octree are the two color quantization schemes that we used later in this work for comparison. A brief description of the two algorithms is given in the following sections.

1.1.1 Median Cut

The basic idea of Median Cut [13] is to use each of the colors in the colormap to represent an equal number of pixels in the original image. The algorithm repeatedly divides the 3D color space into subspaces, with the median point being the boundary for cutting, such that approximately equal numbers of points will fall at each side of the cutting plane. Subdivision is stopped until K required subspaces are generated, and the color representatives that comprise the colormap are obtained by averaging the colors in each subspace.

Figure 1-1 illustrates the initial three steps of the division process in RGB color space leading to the formation eight subspaces. Suppose the first cut, Figure 1-1(a), is in the R-axis and the second cut, Figure 1-1(b), in the G-axis. Since the median point of each subspace will be different after the first division, the cutting point will be different as well for the two subspaces. After that, the third division, Figure 1-1(c), is completed and finally eight subspaces are formed. The division process continues until 256 subspaces are generated.

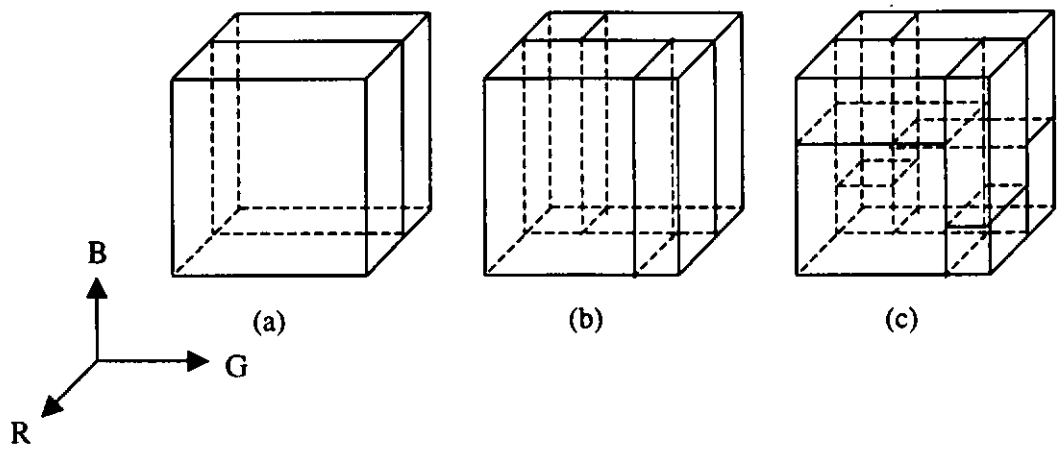


Figure 1-1: Initial three steps of the division process of the Median cut quantization in RGB color space.

1.1.2 Octree

The concept behind Octree [10]-[11] is to build a tree structure containing always a maximum of k different colors, which is the desired number of palette colors. During quantization, the RGB color value is read sequentially from an input image data file and added to the tree structure creating a new leaf node. When the number of leaf nodes n exceeds the number of palette colors k, the tree will be reduced by merging some very closely related colors and both values will be substituted by their mean.

Figure 1-2 shows how the RGB color space can be represented by an octree of depth eight, with the RGB values (0 - 255) being the equivalent of the coordinates in the

3D color space. The bit pattern, corresponding to the same level in the octree as the bit position, is used as the index for the branch into the octree.

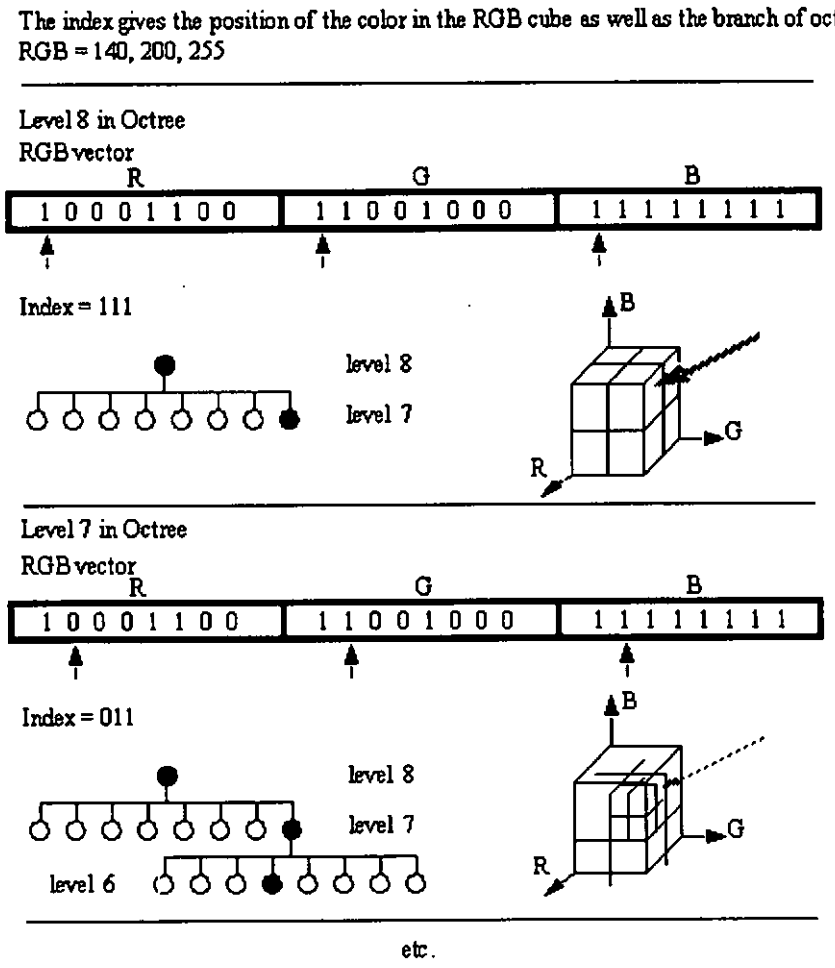


Figure 1-2: Mapping an RGB Value into the RGB Cube/Octree.

1.2 Digital Halftoning

Important developments in digital halftoning include analog screening, noise encoding, ordered dithering, error diffusion and stochastic screening [22]. In addition, numerous model-based halftoning schemes have been developed. Among the various methods, ordered dithering and error diffusion (ED) are the two main classes of conventional digital halftoning techniques.

Ordered dithering provides a fixed pattern of numbers to indicate the order of turning pixels on within a halftone cell. It has two main approaches: clustered dots and dispersed dots. The application of clustered-dot-ordered-dithering is primarily used in the printing industry because clustered dots are insensitive to common printing distortions and reproduce well on printers that have difficulties in producing isolated single pixels [59]. Unlike the clustered-dot approach, dispersed-dot-ordered-dithering turns pixels on individually without grouping them into clusters in the highlight and midtone regions. Being developed to meet the needs of the television and monitor display industries, it has been successfully used for bilevel displays for its greater spatial resolution.

While clustered dots are often preferred for printing, ED is perhaps the most widely used approach for displays [54]. ED is an adaptive algorithm that operates by

spreading or diffusing the quantization error of a current pixel to neighboring pixels. A single pass is made over the input image; each pixel is processed sequentially. At each pixel, error is computed according to the input pixel value and is passed to the neighboring pixels that have not been processed. This method was first proposed by Floyd and Steinberg in 1975 [11], and was originally used and analyzed for gray-scale images. The major part of the analysis is nevertheless applicable to color images as well. For color images, ED is either performed separately in each independent channel (scalar ED) or simultaneously in a 3-D color space (vector ED) [26]. For color displays with an image dependent palette, vector ED is often used.

Based on Anastassiou's analysis of ED and Σ-Δ A/D conversion [3], Knox considered an image resulting from ED as the sum of the original image and a highpass filtered error image [27]. Due to the highpass nature of the "noise" in the reproduction, ED images typically appear closer to the originals than those obtained with ordered dithering. The fact that ED produces high frequency noise, i.e. "blue noise" as named by Ulchney [22], led to the development of various blue noise masks (BNM) [60]-[61], [34], [69]. These BNM methods are collectively known as blue noise halftoning or stochastic screening.

The development of stochastic screening further led to the inclusion of output device model and HVS into halftone algorithms, known as model-based halftoning. For

display of gray-scale images on binary output devices, a number of researchers have suggested iterative schemes that provide good solutions with varying computational requirements [1], [2], [36], [42], [56]. Pappas extended model-based halftoning methods to color [43]-[45] that account for printer distortion as well as quantization effects. Kim and co-workers applied model-based color halftoning to perceptually uniform color spaces such as CIELAB [25]. The algorithms minimize the color distortion between the continuous tone image and it's low pass filtered halftone. For color displays, however, surprisingly little research has been done and the problem remains computationally infeasible at present. With the attempt to minimize a visual model-based error, the optimization of ED for color display applications has been reported in [28] and [57], where optimal ED filter coefficients were determined through a process of autoregressive (AR) modeling of the eye's spatial response.

1.3 Objective

The objective of this research is to develop color quantization algorithms that optimize the overall color image quality on CRT displays. Our algorithms are designed to address two important issues in color quantization. The first one being the incorporation of spatial or contextual information of an input image to the process of color quantization. For most of the color quantization algorithms, the selection of colors for colormap entry is often maneuvered to regions of the color

space with a larger number of pixels. However, colors that amount to a large quantity might not be significant in contributing to the image quality. With this in mind, we propose a heuristic approach (the Contextual algorithm) to perform color quantization based on the contextual information of the input image, thus enabling quantization to focus on the regions of the image having important color information.

The second issue under consideration is related to color space. For most color displays, processing of colors is performed on the R, G and B components. The requirement for human vision system, however, would favour a different scheme. We propose a novel algorithm (the 3D FD algorithm) for color quantization based on frequency diffusion in the histogram of an image. By manipulating a diffusion filter, a cross-space operation is achieved, obtaining quantization effect in a color space without transforming the image to that space.

1.4 Overview of the Dissertation

The rest of this dissertation is organized as follows. A brief review on the basics of color science is given in Chapter 2 (Colorimetry) and Chapter 3 (Digital Color Spaces). Colorimetry is the method of measuring and evaluating colors that lay the foundation for the development of various color spaces. Different color imaging

devices use different color spaces, each of them has a variety of different characteristics that lead to advantages and disadvantages depending on the application. In presenting the basic information according to which colors are specified, related and measured, these two chapters attempt to provide the fundamental color knowledge that is encountered in the context of color image quantization.

Over the past years, the designing of color quantization algorithms has evolved to an integration of various imaging techniques, human vision, color science and device models. In practice, improvements can only be made if image quality can be accessed. Therefore, Chapter 4 (Evaluation of Color Image Quality) examines the fundamental concepts, models, metrics and their formulation for evaluating the quality of quantization outputs.

Description of the algorithms representing the core of the research, along with in depth discussions of each algorithm will be given in Chapter 5 (A Contextual Algorithm for Color Quantization) and Chapter 6 (Color Quantization by Three-Dimensional Frequency Diffusion).

We first deal with the issue of deriving a mechanism to carry out image analysis for allocation of quantization levels in Chapter 5. In this chapter, an algorithm called the

Contextual algorithm is proposed. In our approach, we identify the regions of an image having the greatest need for colors, allocate more quantization levels to them, and then move to another part of the image in a deterministic manner. We achieve this by scanning the elements of the input image in a way determined by their local intensity and select the color representatives that comprise the colormap according to their local popularity. The proposed algorithm is then compared with some of the other well-established color quantization schemes such as Median cut and Octree.

The issue of color space is then tackled in Chapter 6. A novel algorithm for color quantization, three-dimensional frequency diffusion (3D FD), applies to the histogram of an image based on the principle of error diffusion using a 3D error diffusion filter is proposed. With the histogram divided into overlapping cubes, an iterative process is devised to select representative colors from these cubes by a popularity scheme that considers a neighborhood of pixels until a colormap is filled. By transforming the 3D frequency diffusion filter, a cross-color-space operation is achieved with no transformation of the image required. The algorithm is able to produce similar effect as it does under the YUV color space when operating in RGB.

The performance of the two proposed algorithms are then compared and concluded in Chapter 7. Finally, direction for future research is given in Chapter 8.

Chapter 2

COLORIMETRY

Colorimetry is a measurement of color [9]. Color appearance of an object depends on three components, namely the light source and illuminants, the interaction of radiant energy with materials and the human visual response. While the measurement or standardization of light sources and materials provides the necessary physical information for colorimetry, the quantification of human visual response provides the basis of color appearance specification. As a combination of all these areas, colorimetry forms the foundation for the development of various color appearance models or color spaces.

2.1 CIE XYZ (1931)

Based on the data from the measurements of the color-matching abilities of the average human eyes made in 1931, a system of three primaries, XYZ, was developed by the Commission Internationale de l'Eclairage (CIE) capable of representing all visible colors using only positive values of X, Y and Z [68]. The Y tristimulus value is identical to Luminance, X and Z tristimulus values give coloring information. This forms the basis of the CIE 1931 XYZ color space, which is

fundamental to all colorimetry. All color appearance models for practical applications begins with the specification of the stimulus and viewing conditions in terms of CIE XYZ tristimulus values. It is defined such that all other models can be interpreted as different mappings or subsets of this color space.

2.2 Chromaticity Coordinates

Chromaticity diagrams are developed to provide a convenient two-dimensional representation of colors. Mathematically, the projection of the tristimulus space to the two-dimensional X and Y plane can be defined as:

$$x = X / (X + Y + Z),$$

$$y = Y / (X + Y + Z),$$

$$z = Z / (X + Y + Z),$$

$$x + y + z = 1,$$

where x, y, and z are the chromaticity coordinates. They are the normalization of the tristimulus values. Figure 2-1 shows the 1931 CIE chromaticity diagram. The chromaticity coordinates represent the relative amounts of the three stimuli X, Y and Z required to obtain any colors. However, they do not provide information about the color appearance of the stimuli because they lack information about luminance and do not account for chromatic adaptation. Since luminance is incorporated into the Y tristimulus value, the triplet (x, y, Y) should be used to accurately describe a color.

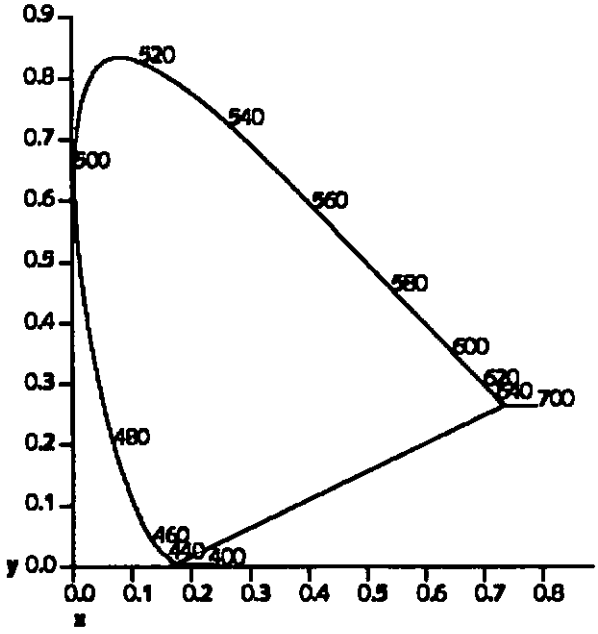


Figure 2-1: The CIE 1931 x – y chromaticity diagram [22].

2.3 CIE Color Spaces

There are perhaps two problems with the specification of colors in terms of tristimulus values and chromaticity coordinates. Firstly, this specification is not easily interpreted in terms of the psychophysical dimensions of color perception such as brightness, chroma and hue. Secondly, the XYZ system and the associated chromaticity diagrams are not perceptually uniform. The second of these points is a problem if we wish to estimate the magnitude of the difference between two color stimuli. Visually uniform color spaces are therefore derived from nonlinear

transforms of the CIEXYZ that describe color using opponent-type axis relative to a given absolute white point reference. Two spaces were developed and recommended by CIE for use in 1976, they are the CIE 1976 (L* a* b*) color space (CIELAB) and the CIE 1976 (L* u* v*) color space (CIELUV).

2.3.1 CIE 1976 (L a* b*)*

The CIELAB color space is defined by the following equations [19]:

$$L^* = 116(Y/Y_n)^{1/3} - 16, \qquad \text{for } Y/Y_n > 0.008856$$

$$L^* = 903.3(Y/Y_n), \qquad \text{for } Y/Y_n \le 0.008856$$

$$a^* = 500[(X/X_n)^{1/3} - (Y/Y_n)^{1/3}],$$

$$b^* = 200[(Y/Y_n)^{1/3} - (Z/Z_n)^{1/3}],$$

$$C^*_{ab} = (a^{*2} + b^{*2})^{1/2}$$

$$h_{ab} = \tan^{-1}(b^* / a^*)$$

where X, Y, Z are the tristimulus values of the stimulus and X_n, Y_n, and Z_n are the tristimulus values of the reference white. Note that when any of the quotients are less than or equal to 0.008856, a slightly different set of equations are used. Illustrated by Figure 2-2 is a Cartesian color space constructed using the L*, a^* and b^* coordinates.

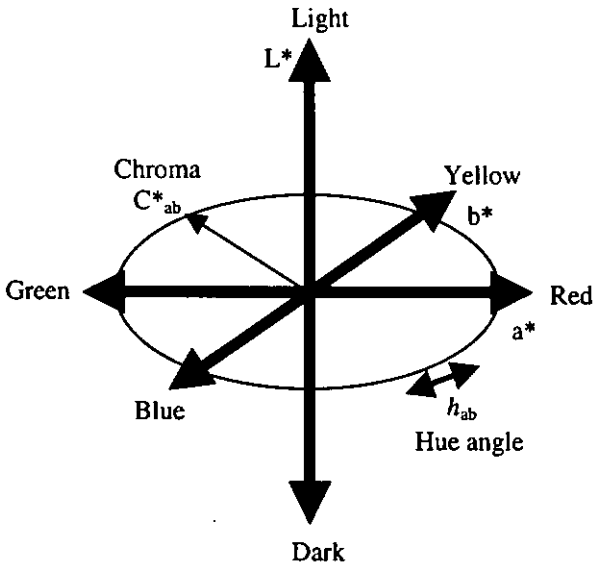


Figure 2-2: Cylindrical representation of the CIELAB color space.

As indicated by the figure, L* represents lightness and extends from 0 (black) to 100 (white). a^* approximate redness-greenness, b^* approximate yellowness-blueness, C^*_{ab} chroma, and h_{ab} hue. The L*, C^*_{ab} and h_{ab} coordinates are the cylindrical representation of the same space.

2.3.2 CIE 1976 (L u* v*)*

Similar to the CIELAB color space, the CIELUV color space is a nonlinear transform of the 1931 CIEXYZ and is defined by the following equations:

$$L^* = 116(Y/Y_n)^{1/3} - 16, \qquad \text{for } Y/Y_n > 0.008856$$

$$L^* = 903.3(Y/Y_n), \qquad \text{for } Y/Y_n \leq 0.008856$$

$$u^* = 13L^* (u' - u'_n)$$

$$v^* = 13L^* (v' - v'_n)$$

$$C^*_{uv} = (u^{*2} + v^{*2})^{1/2}$$

$$h_{uv} = \tan^{-1} (u^* / v^*)$$

where u' and v' are the chromaticity coordinates of the stimulus and u'_n and v'_n are the chromaticity coordinates of the reference white. L* represents lightness, u^* redness-greenness, v^* yellowness-blueness, C^*_{uv} chroma, and h_{uv} hue. Similar to CIELAB, The L*, u^* and v^* coordinates are used to construct a Cartesian color space, while L*, C^*_{uv} and h_{uv} coordinates are the cylindrical representation of the same space.

Although both the CIELAB and CIELUV spaces were recommended in 1976, CIELAB has become almost universally used for color specification and particularly color difference measure.

Chapter 3

DIGITAL COLOR SPACES

A color space is a means by which colors can be specified, related and measured. Since color is tri-variant, most color models are defined in a three-dimensional space and can be considered as a three-dimensional geometric shape with a particular color specified by its vector coordinates within that shape. Different color imaging devices use different color spaces, and therefore colors produced by these spaces are device dependent and do not correlate with the way the HVS perceived colors. The rendering or representation of colors depends solely on the characteristics of the device being used. Digital color spaces have a variety of different characteristics that lead to advantages and disadvantages depending on the application and system for which a particular color space is required. Table 3-1 summarizes the most popular color spaces and some of their applications.

Color Spaces		Applications
Device Independent	XYZ	Colorimetric calculations
	CIELAB	Color difference evaluation, image analysis, color management
Device Dependent	RGB	Computer graphics applications, image processing or analysis
	YIQ, YUV, YCbCr	Television transmission; image compression, coding and storage
	HSV	User interface in computer graphics, human color perception

Table 3-1: The most popular color spaces and some of their applications.

3.1 RGB

RGB is the most frequently used color space in describing colors for computer graphics since no transformation is required. Based on the red, green and blue primaries, the RGB color space can be visualized as a cube with the three axes corresponding to red, green and blue. Using an appropriate scale along each primary axis, the space can be normalized and confined to values between 0 and 1, and all definable colors will lie in the cube as shown in Figure 3-1. The origin of the cube, defined as (0,0,0) corresponds to black and the point with coordinate (1,1,1) corresponds to the system's brightest white.

The major advantage of the RGB color space is that it is very easy to specify display colors. However, this color space is not perceptually uniform, which means a given change in any coordinates with the same Euclidian distance does not correspond to the same perceived color difference in all regions of the color space.

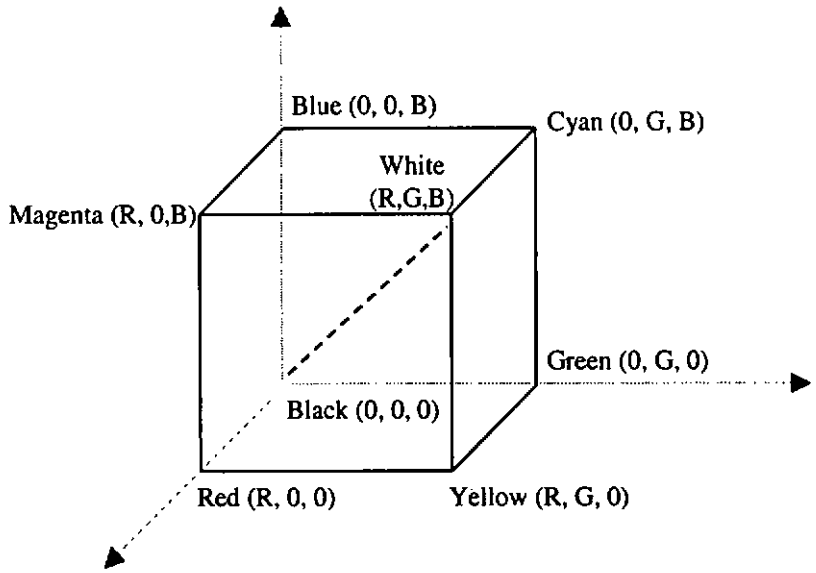


Figure 3-1: The RGB color space [46].

3.2 HSV

HSV (hue, saturation, value) color space is derived for user specification and recognition of a color in an intuitive manner. The HSV coordinate system, proposed originally by Smith [55], is an inverted hexagonal cone as shown in Figure 3-2, with black at the apex and white at the center of the H – S plane.

The HSV color space, though provides better approximations to the perceptual dimensions of color, remains device dependent as it is simply a linear transforms of the RGB color space and hence is perceptually non-uniform. In addition, the three axes are not perceptually independent. As one moves around the H – S color plane,

for instance, the apparent lightness will change on the display, even though the V component remains numerically constant. The biggest discrepancy is between green and blue, which typically differ in luminance by a factor of 5. Conversely, a change in V will generally change the apparent saturation, even though the S component remains numerically constant.

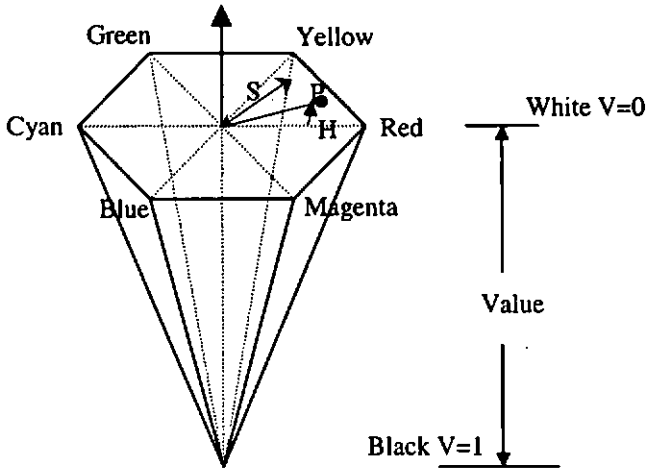


Figure 3-2: The HSV color space [23].

3.3 YIQ, YUV, YCbCr [49]

These are television transmission color spaces adopted in practical video systems such as NTSC and PAL, designed to take the advantage of the greater sensitivity of the HVS to changes in luminance than to changes in hue or saturation. Instead of

transmitting red, green and blue signals, in these color spaces, luminance (Y) and two color difference signals (B-Y and R-Y) are transmitted as a means of conveying the RGB information. Figure 3-3 is a unit RGB cube transformed into luminance Y and color difference components B-Y and R-Y.

The decoupling of luminance (Y) and chrominance components (B-Y and R-Y) provides several advantages in image coding and communication applications. For instance, by allowing more bandwidth to code the luminance and less bandwidth to code chrominance, noise introduced in transmission, processing and storage can be reduced. Another implication is that the luminance (Y) component of an image can be processed without affecting its color content. This color space is widely used in transmission of images, situations where compression is important. It is device dependent, but is intended for use under strictly defined conditions within closed systems.

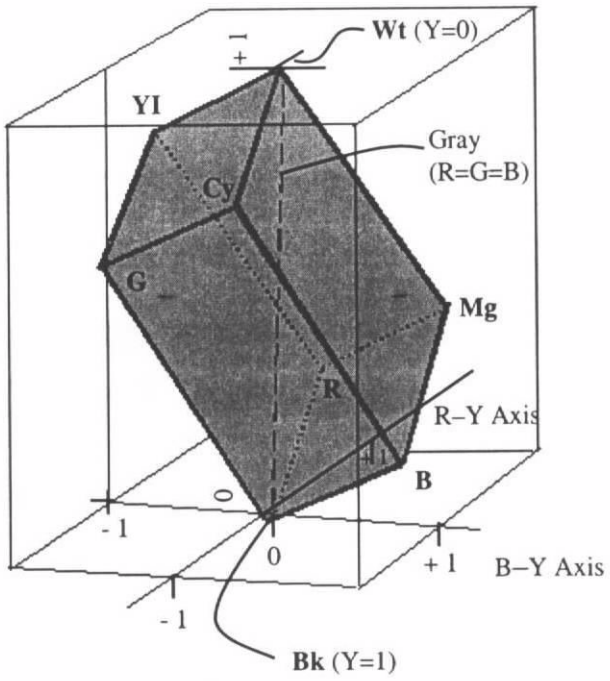


Figure 3-3: RGB and Y, B–Y, R–Y cubes.

Chapter 4

EVALUATION OF COLOR IMAGE QUALITY

As described by Holst [17], image quality is a 'subjective impression ranking of imagery from poor to excellent. It is a perceptual ability, accomplished by brain, affected by and incorporating other sensory systems, emotions learning and memory. The relationships are many and not well understood', visual image quality, as perceived by observer, however, has no single or unique definition.

Although the concept of quality in imaging is complex, especially when human factors are involved, it would be essential if image quality can be quantified and measured so that improvements can be made to optimize the performance of the quantization algorithm to reproduce color images of optimal visual quality.

4.1 PSNR

Peak signal-to-noise (PSNR) [40], [50] is a measure used to estimate the quality of a reconstructed image compared with an original image. The basic idea is to compute a single number that reflects the quality of the reconstructed image. Reconstructed images with higher metrics are judged better. In fact, PSNR measures do not equate with human subjective perception, they are used because of their simplicity in computation.

Given a source image $I(i, j)$ having dimensions of M×N and a reconstructed image $Q(i, j)$, the PSNR in decibels (dB) is computed as follows

$$PSNR = 10\log_{10}\left\{\frac{MN \times 255^2}{\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}[I(i,j)-Q(i,j)]^2}\right\}$$

where $I(i,j)$ and $Q(i,j)$ are the RGB values at location (i, j) of the original image and quantized image respectively.

When $I(i,j)$ and $Q(i,j)$ are vectors, the square of their difference becomes the algebraic sum of the squares of the differences of the respective components. In essence the PSNR is the ratio of 255^2 to the mean square error of the reconstructed picture.

Typical PSNR values range between 20 and 40, and are usually reported to two decimal points. The actual value is not meaningful, but the comparison between two for different reconstructed images gives one measure of quality.

4.2 CIELAB Color Difference

Derived from perceptual measurements of color discrimination of large uniform targets under fixed adaptation condition, the CIELAB color difference metric has been widely used in measuring color reproduction errors for twenty years. Though imperfect, it works

reasonably well in measuring perceptual differences between large uniform color patches viewed under standard illuminants.

Color differences in the CIELAB color space (ΔE^*_{ab}) are measured as the Euclidean distance between the coordinates for two stimuli:

$$\Delta E^*_{ab} = (\Delta L^{*2} + \Delta a^{*2} + \Delta b^{*2})^{1/2}$$

The aim of the CIELAB color space design was to have color differences be perceptually uniform throughout the space, so that equal distances in the color space represent equal perceived differences in appearance. For example, with the just noticeable color difference equal to one ΔE^*_{ab} unit, a ΔE^*_{ab} of 1.0 for a pair of red stimuli should be perceived as equal in magnitude to a ΔE^*_{ab} of 1.0 for a pair of gray stimuli. However, this aim is not achieved precisely, especially for evaluating small color differences.

4.3 Contrast Sensitivity Function

'Quality is not a property of images, but a description of a judge's reaction to images' [10]. Since images are viewed by human, the incorporation of HVS models into image processing algorithms would be essential in maximizing perceived image quality. For digital halftoning, the most important parameter is perhaps spatial frequency and therefore, most of the HVS models for digital halftoning are based on the human contrast sensitivity function (CSF).

Image contrast is the ratio of the local intensity to the average image intensity [64]. For a sinusoidal grating, contrast is typically defined as the difference between the maximum and minimum luminance divided by the sum of the maximum and minimum luminance (Michelson contrast) [32]. Contrast sensitivity is the reciprocal of the contrast threshold, which is the minimum amplitude necessary to just detect a sine wave of a given angular spatial frequency. A CSF describes contrast sensitivity for sinusoidal gratings as a function of spatial frequency expressed in cycle per degree (cpd) of the visual angle. It is the linear spatially invariant approximation of the HVS.

Previous works on luminance CSFs indicates that human contrast sensitivity is affected by background intensity [51], [63]. Under scotopic conditions, human vision has a low pass filtering characteristics; on intense photopic backgrounds, CSF curves are bandpass. Furthermore, chromatic CSF curves behave differently from luminance CSF curves. Figure 4-1 illustrates a chromatic CSF curve and a luminance CSF curve measured by Mullen [35], obtained at high mean luminance level. As indicated by the figure, the luminance CSF is bandpass in nature, whereas the chromatic is of low-pass nature and has a significantly lower cutoff frequency. This implies that the HVS is more sensitive to fine spatial changes in luminance than it is to small changes in chrominance. This explains the reasons why images with spatial degradations in chrominance will usually not be noticed, though similar degradations in luminance might be obvious.

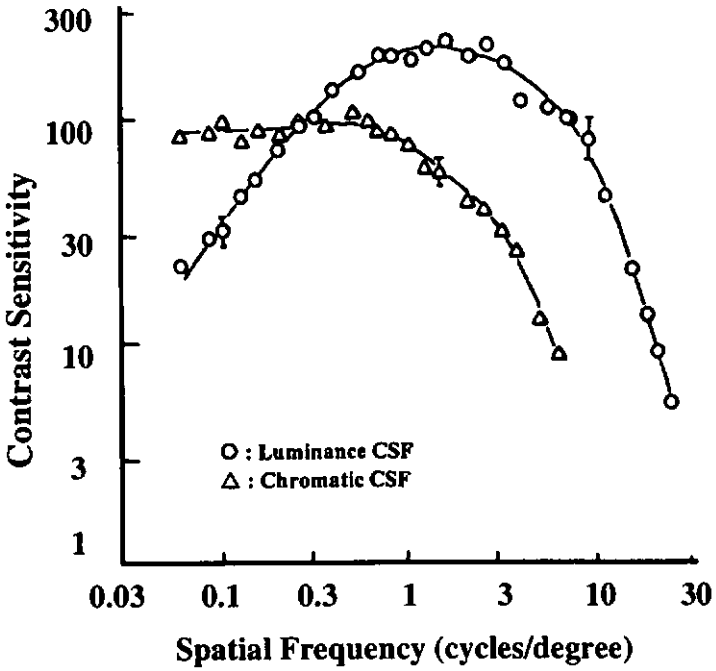


Figure 4-1: Luminance and chrominance (blue-yellow) contrast sensitivity functions [22].

CSFs have been adopted for use in various HVS models and have been exploited in several halftoning algorithms and some of the examples are given in [8], [33], [37]. In addition, Zhang and Wandell [70] proposed a color image fidelity metric that incorporates the spatial-color sensitivity of the human eye into the calculation of CIELAB color difference to account for spatial errors as well as color errors in measuring color reproduction errors of digital images.

4.4 S-CIELAB

The overall quality of a color image is the integrated result of the spatial pattern and color appearance. Due to the fact that real images are not made up of large uniform fields and color perception is dependent on spatial patterns of images [15], [47]-[48], [53], [62], the CIELAB color difference metric is, however, not suitable for measuring color differences between real images. To reflect both spatial and color sensitivity of the HVS, S-CIELAB, a spatial extension of CIELAB, has been developed for color reproduction of digital images.

Since the HVS is not as sensitive to color differences in fine details as compared to large uniform fields, the design goal of S-CIELAB is to apply spatial filtering to the color image in a small-field or fine-pattern area, but reverts to the conventional CIELAB in a large uniform area. The computation of S-CIELAB consists of three processing steps. First, the original and distorted images, specified in terms of XYZ tristimulus values, are converted into three opponent-colors planes representing luminance, red-green and blue-green components for each image. Second, each opponent-colors plane is convolved with a two-dimensional spatial filter selected according to the spatial sensitivity of the human eye for that color component to simulate the spatial blurring by the HVS [39]-[40]. Third, the filtered representations are transformed back to CIEXYZ space and then to CIELAB representations. The difference between S-CIELAB representations of the original and its reproduction (ΔE_s), as an indication to the reproduction error, is then computed precisely as ΔE_{ab} in conventional CIELAB. The ΔE_s values are interpreted in the same way as the

standard CIELAB ΔE_{ab} values; that is distortions of one ΔE_s unit is at threshold visibility at optimum viewing condition, whereas distortions with ΔE_{ab} values around two or below are generally not visible under less controlled viewing conditions.

S-CIELAB has been used to predict texture visibility of printed halftone patterns and the results correlate with perceptual data better than the standard CIELAB [71]. This metric is also used to improve multilevel halftone images [72].

Chapter 5

A CONTEXTUAL ALGORITHM FOR COLOR QUANTIZATION

All the quantization techniques mentioned in Chapter 1 are operated in accordance with the statistical distribution of the colors in the input image and share a common initial condition attempting to allocate more quantization levels to regions of the color space with a larger number of pixels. These algorithms, though image dependent, do not account for the spatial and contextual information of the image. As a result, instead of focusing on the area of high interest in an image containing highly saturated primaries, the grays and low saturated colors are often over represented resulting in the appearance of contouring artifacts, especially in regions with smooth color transition. Aiming at this problem, we developed a color quantization algorithm with contextual information taken into account.

5.1 Contextual Algorithm

Consider a case to color quantize an image I containing millions of colors to an output image Q containing a maximum of 256 simultaneously displayable colors. From I, we generate two pseudo images, PI1 and PI2, as the inputs to our algorithm.

The first pseudo image, PI1, is obtained by summing up the RGB values of I at each pixel as indicated in eq.(1) and is used to locate the areas of I containing important color information.

$$PI1(i, j) = [r(i, j) + g(i, j) + b(i, j)] \qquad (1)$$

where r, g, b are the pixel values at location (i, j) for RGB values of the original image respectively.

The second pseudo image, PI2, is composed of indexing values representing the exact colors of every pixel in the original image I, each of which constitutes the RGB components of a specific color as below,

$$PI2(i, j) = [r(i, j) * 10000000 + g(i, j) * 1000 + b(i, j)] \qquad (2)$$

where r, g, b are the pixel values at location (i, j) for RGB values of the original image respectively.

Therefore, if the RGB values of a pixel I (i, j) is (16, 67, 17) for example, the indexing value for PI2 at the same location will be 16067017, and the popularity of a color can be determined by counting the frequency of occurrence of the corresponding index value. For an image of size 256x256, the total number of indexing values will be 256*256, i.e. 65536, while the total number of different indexing values is dependent on the total number of colors that constitute an image.

Our approach is to identify the regions of the image we want to emphasize from PI1, and then from PI2 at the same regions, we extract the color representatives that comprise the colormap according to their local popularity. Once a color is extracted, we can identify the locations of the image having the same or visually similar colors. We then keep the locations and update the input images by assigning 0 to these locations to prevent the same color being selected in the next iteration of color extraction. Figure 5-1 shows a schematic diagram of the proposed algorithm, which will be elaborated in the following section.

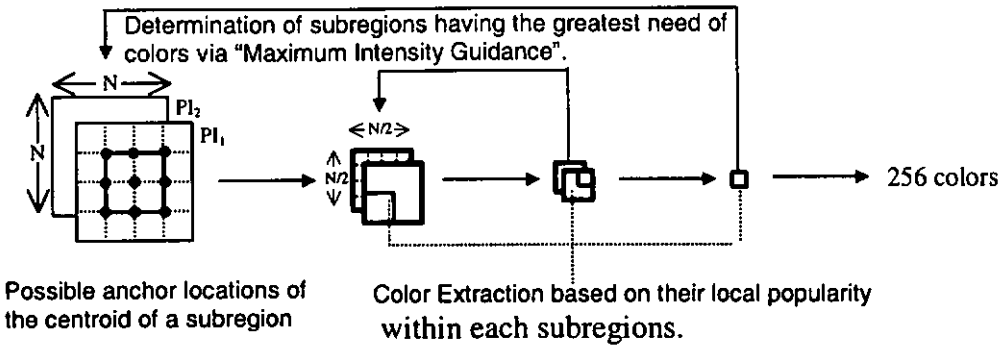


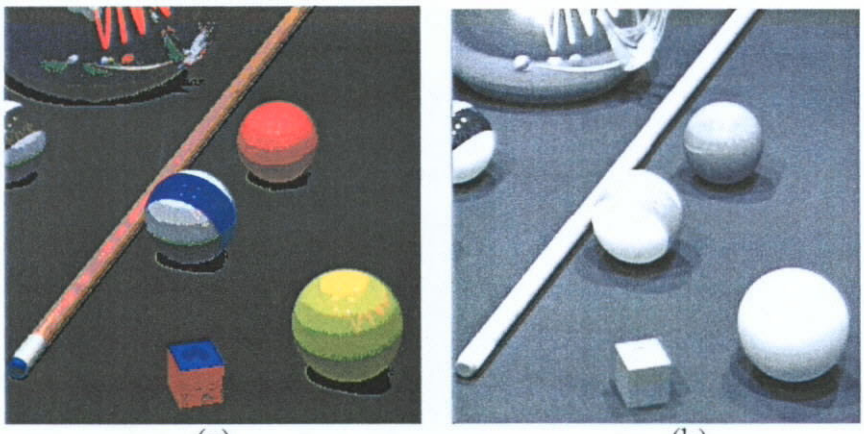
Figure 5-1: Schematic diagram of the Contextual algorithm.

5.2 Determining the Regions of High Interest

Figure 5-2(a) shows a 24 bits/pixel artificial image containing wide range of colors that has been uniformly quantized to contain 256 colors only and Figure 5-2(b) is the corresponding pseudo image (PI1) generated by summing up the RGB values of the original image at each pixel. As indicated by these figures, the regions where contouring is most likely to occur after color quantization are equivalent to the regions in the pseudo image showing excessive brightness, i.e. regions having large sums of RGB values.

Therefore, by locating the areas of PI1 having the largest value, and thus the greatest need for colors, we can easily identify the parts of the image we want to emphasize and assign more colors to them. We achieve this by scanning the elements of PI1 in a deterministic way via "maximum intensity guidance" [7] [24]. The basic idea is to

iteratively search the brightest region of PI1 by partitioning the image into subregions, always choosing the subregion containing the largest sum of elements for assigning colors as well as further division.



(a) (b)

Figure 5-2: (a) An artificial image that is uniformly quantized to contain 256 colors; (b) The corresponding pseudo image generated by summing up the RGB values of the original true color image at each pixel.

5.3 Image Subregion Representation

Let X be the input image array to our algorithm of size K×L. If we consider a square image of a binary size, i.e., $K=L=N=2^r$, the dimensions of which are divisible by 2, when each side of the image is divided by 4, 16 equal subregions are obtained as shown in Figure 5-3(a). Let a side of a subregion be $w = N/4$.

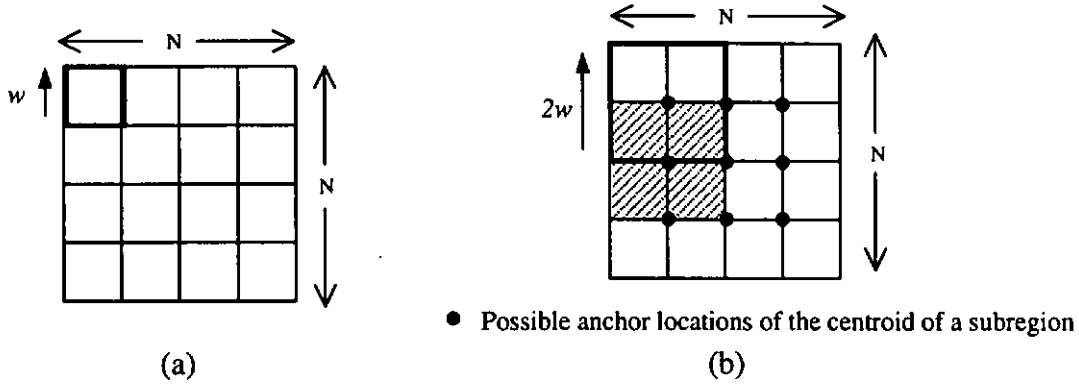


Figure 5-3: (a) A scheme of dividing the image into 16 subregions, with the size of each subregion equals $w \times w$. (b) A scheme of dividing the image into 9 subregions, with the size of each subregion equals $2w \times 2w$.One subregion is shown with bold border and an overlapping subregion is shaded.

With a simple scheme that picks all subregions with the largest sum of all elements for assigning colors, it is likely that the resulting colormap will contain many entries of similar colors, neglecting other less popular ones. To prevent this, we consider a neighborhood or subset of the image:

$$X_k(l_k,m_k) = \{X_{k-1}(l_k+x, m_k+y) \mid x, y \in 0,1,\ldots, 2w-1\} \qquad (3)$$

where $k = 1,2, \ldots, w=2^{r-k-1}$ and l_k, and m_k each takes the values of 0, w and $2w$.

In this way, pixels in an area (or segment) of $2w$ by $2w$ are grouped together as shown in Figure 5-3(b). With this arrangement, 9 overlapping segments are naturally

formed within the whole image with each segment overlapping half the area of an adjacent segment. The elements associated with a segment E_k is defined as

$$E_k(l_k, m_k) = \sum_{x=0}^{(2w-1)} \sum_{y=0}^{(2w-1)} X_k(l_k + x, m_k + y) \tag{4}$$

The highest E_k is selected and the corresponding X_k is subdivided according to eq (3). Before proceeding to the next subsegment of the highest local intensity for further subdivision, colors are extracted as the colormap entries from each of the nine subsegments sequentially based on their local popularity. In eq (3), a higher k refers to a smaller subregion such that X_0 represents the original input, and X_r the finest level containing a single element corresponding to the actual pixel of the X array. It is important to note that the process of subdividing X_k for color extraction within the algorithm will not continue up to k=r-1. This is because the color of a single pixel element is not likely to be an effective representation of any important color information in an image. Instead, the process will terminate at a particular level when X_k has reached a certain size of n × n. We assume to stop extracting colors at a level that is sufficient to include less popular colors that may be essential in contributing to the final image quality, while further color extraction will not contribute much to the final image quality.

5.4 Extracting Colors for the Colormap

From PI1, we identify the regions of the image to emphasize, and then extract the color representatives that comprise the colormap based on their local popularity. The popularity is obtained by counting the frequency of occurrence of each color point by point accordingly from PI2 at the same locations. The colors of the highest frequency counts are selected for construction of colormap. To ensure that the dominant colors will be extracted in case the region contains a large part of background colors, the colors among the first three highest frequency counts are selected, so that more than one color will be extracted at one time.

Referring to Figure 5-4, let ID1 = ($id1_1$, $id1_2$, ..., $id1_n$) be the list of colors extracted from a particular image segment. Starting with the first value ($id1_1$) from the list of ID1, a second list of index values, ID2 = ($id2_1$, $id2_2$, ..., $id2_n$), can be generated consequently to include all possible colors within a distance D from the extracted color in the RGB color space. The colors among the list of ID2 are regarded as visually the same as $id1_1$ and satisfying results are obtained with the distance D equals to 5. After that, we identify any locations in PI2 having the same value as $id1_1$ or as those among the list of ID2, and assign 0 to the input image arrays at these locations to avoid the same color being selected in the next iteration of color

extraction. This is followed by an update of ID1 by neglecting $id1_1$ as well as any subsequent values that are identical to those in the list of ID2.

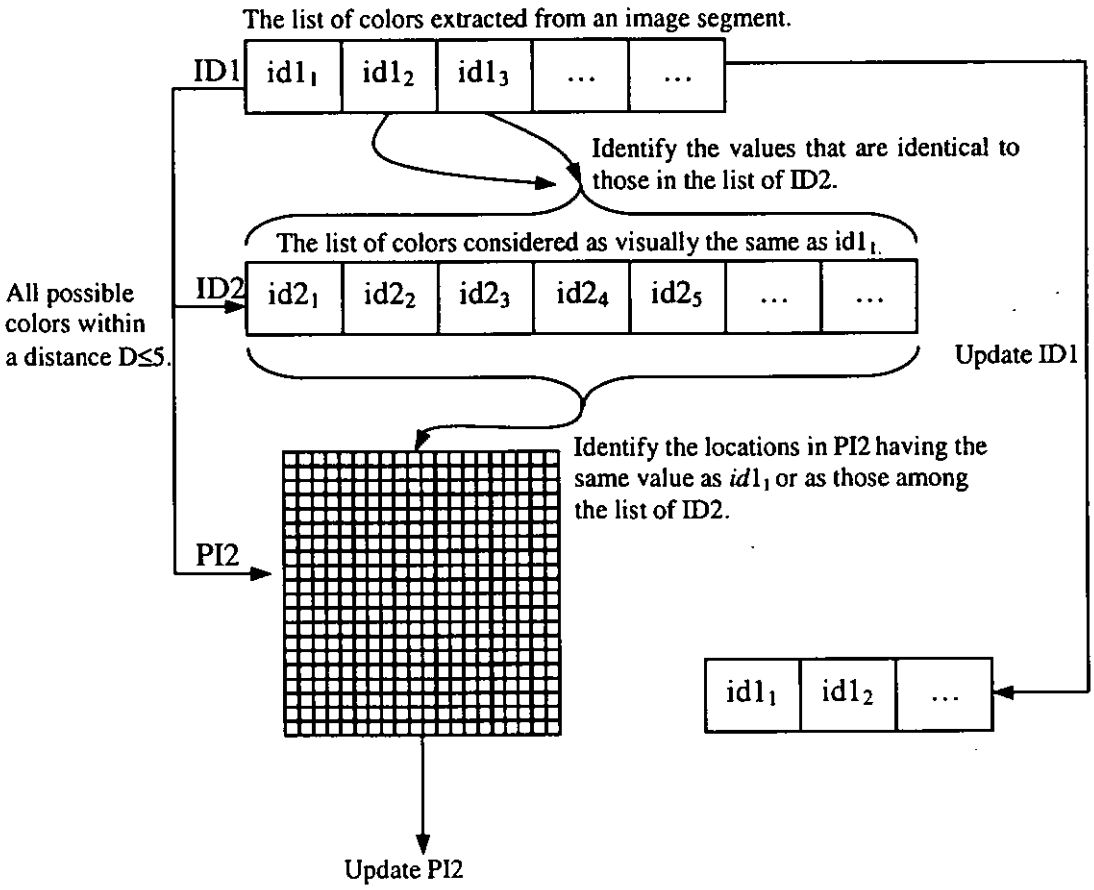


Figure 5-4: Updating of the input image array to avoid the same color being selected in the each iteration of color extraction.

5.5 Algorithm

Step 1: From the original image I of size N×N, generate two pseudo images, PI1 and PI2.

Step 2: Initialize PI1 as the image array X_0 for division. Set $k=1$.

Step 3: Divide the input image X_{k-1} into 9 overlapping segments $X^m{}_k$, m=1, 2, ...9 according to eq (3). Find the sum of all elements E_k associated with each segment $X^m{}_k$ according to eq (4). Select the segment $X^m{}_k$ with the largest E_k to be the new region of interest.

Step 4: Increment k. Set $X^m{}_k$ as X_{k-1}, the input image array, further divide the input segment X_{k-1} into nine subsegments as in Step 3. Label the 9 subsegments as $X^j{}_k$, j=1, 2, ...9. Set j=1.

Step 5: From the subsegment $X^j{}_k$, identify the most frequently occurring colors within the confines of the subsegment according to PI2 at the same location. Select the colors with the three highest frequency counts and store them in a list ID1.

Step 6: Select $id1_1$, the first entry in ID1 as the colormap entry. Generate a list of colors ID2 that is considered as visually the same as $id1_1$. Identify the locations of the image having the same values as those among the list of ID2 from PI2. Update both PI1 and PI2 as well as all the subsequent subsegments

by assigning 0 to the locations to avoid the same color being selected in the next iteration of color extraction.

Step 7: Update ID1 by deleting $id1_1$ as well as any subsequent values that are identical to those among the list of ID2. Set the next color that remains in the list of ID1 as $id1_1$ and repeat Step 6 until ID1 contains no color.

Step 8: Repeat Step 5 to Step 7 for j=2, 3, ...9. Select the subsegment X^j_k with the largest E_k to be the new region of interest as in Step 3.

Step 9: Repeat Step3 to Step 8 until the input image array X_k has reached a size n×n. It is shown later that n=8 is optimal.

Step 10: Repeat Step 2 to Step 9 to extract the remaining colors until the whole colormap is filled by 256 colors.

5.6 Experiment and Discussions

A representative set of images of a size 256×256 have been chosen to include both areas of smooth gradation (low frequency) and fine details (high frequency) for evaluation of the algorithm. The images are shown in Figure 5-5. To start with, we test our algorithm by subdividing X_k until k=5, at which level the process of division and color extraction will terminate as X_k reaches a size 8×8. At this size we shall show later that, if the process of division stops before this level such that colors are extracted from larger areas, some of the less popular colors that may be essential in

contributing to the final image quality will be neglected. On the other hand, further subdivision after this level to extract colors from smaller subregions will be of no significance in contributing to the final image quality.

Figure 5-5: The test images – (a) Pool (b) Woman (c) Shop and (d) Blythe.

To compare our algorithm with other color quantization schemes, we use the peak signal-to-noise ratio (PSNR) as a gauge to measure the effectiveness of the resulting palettes. Given the original image I having dimensions of N×N and a quantized image Q, the PSNR in decibels (dB) is computed as follows

$$PSNR = 10\log_{10}\left\{\frac{N^2 \times 255^2}{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}[I(i,j)-Q(i,j)]^2}\right\} \tag{5}$$

where $I(i,j)$ and $Q(i,j)$ are the RGB values at location (i, j) of the original image and quantized image respectively.

When $I(i,j)$ and $Q(i,j)$ are vectors, the square of their difference becomes the algebraic sum of the squares of the differences of the respective components. In essence the PSNR is the ratio of 255^2 to the mean square error of the reconstructed picture.

The PSNR values of the various test images quantized by our algorithm and by the other two algorithms are given in Table 5-1. According to the table, terminating the process of division and color extraction within the algorithm after the final input image array X_k has reached a size 8×8 gives the highest PSNR, comparing to the other sizes such as 16×16 or 4×4, which is comparable to the other two color

quantization schemes in comparison. Implied by the reduced PSNR, increasing or decreasing the size of the final subsegment in terminating the division process will not made any improvements to the overall performance of the algorithm as we assumed. If colors are extracted from larger areas as the process of subdivision is stopped at the level of 16x16, some of the less popular colors that may be essential in contributing to the final image quality will be neglected. While extracting colors from smaller subsegments of a size 4x4 could result in colors that will not contribute to the final image quality being selected, leaving less possibility for colors that are comparatively more important to be included in the resultant colormap.

		Pool	Woman	Shop	Blythe
Contextual Algorithm	*n=16	30.87	29.54	25.88	25.42
	*n=8	32.87	29.54	27.94	28.53
	*n=4	30.97	29.50	25.84	25.53
Others	Median Cut	32.63	29.92	28.84	28.31
	Octree	33.72	31.27	28.47	29.10

Table 5-1: PSNR (in dB) of pictures reproduced after color quantization by the Contextual algorithm, Median cut and Octree. *The division process within the algorithm terminates when the size of the final input image array has reached the size of n×n.

Apart from altering the size of the final subsegment to terminate the division process, we test our algorithm by changing the number of colors we selected at a time from each subsegment for color extraction. To start with, the number of colors

to be selected is arbitrarily set to include those among the first three highest frequency counts. The reason for this is to ensure the selection of dominant colors, especially when the subsegment contains a large part of analogous background colors. Experimental results indicate that while changing the selection criteria to less than three will degrade the PSNR, increasing the number of colors to be selected will not improve the performance of our algorithm either as shown in Table 5-2. Since colors are selected based on their local popularity, decreasing the number of colors to be extracted will have the possibility of bringing the focus to the background colors, despite the less popular ones are actually more important. Given that our algorithm extracts colors from subsegment to subsegment sequentially after identifying the region of interest. Allocating more quantization levels to each subsegment and hence the whole region will have the possibility of selecting too many colors from a particular region at the beginning, thus giving the subsequent regions insufficient prominence and neglecting some of the important colors in these regions.

		Pool	Woman	Shop	Blythe
Contextual Algorithm	† f=2	30.02	25.54	25.10	28.40
	† f=3	32.87	29.54	27.94	28.53
	† f=4	31.68	28.81	25.43	27.90
Others	Median Cut	32.63	29.92	28.84	28.31
	Octree	33.72	31.27	28.47	29.10

Table 5-2: PSNR (in dB) of pictures reproduced after color quantization by the Contextual algorithm, Median cut and Octree. The division process within the algorithm terminates when the size of the input image array has reached 8×8.
† The colors selected from each subsegment are those among the first two highest frequency counts (f=2), the first three highest frequency counts (f=3) and the first four highest frequency counts (f=4).

The appearance of contouring artefacts is one of the major problems encountered by color quantization, especially in regions exhibiting smooth color transition, such as the background in "Woman" and the billiard balls in "Pool". This problem can be solved with spatial error diffusion by using the Floyd and Steinberg error diffusion filter [9] during pixel mapping. The error, that is the difference between the exact pixel value and the quantized value, is distributed to the neighborhood of four pixels surrounding the central pixel as indicated in Figure 5-6.

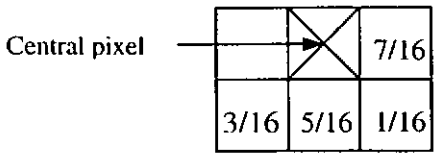


Figure 5-6: Weights of Floyd and Steinberg error diffusion filter surrounding the central pixel.

The resultant images are shown in Figure 5-7 to 5-10. To study the overall performance of our proposed algorithm with spatial error diffusion, we measure the color reproduction errors by using the S-CIELAB color difference metric [70]. The result is an error map with one ΔE value per pixel indicating the difference between the S-CIELAB representation of the original image and the quantized image. We use S-CIELAB because PSNR does not correlate to how the human visual system detects and responds to image inaccuracies, which is the major shortcoming of PSNR measures. Whether the eyes and brain judge an error as significant depends on numerous factors, such as where the error resides in the image, both absolutely and relative to where the eyes' attention is at that point in time; what portions of the color spectrum the error's pixels represent.

Let us consider an example. Table 5-3 (a) and Table 5-3 (b) are the PSNR values of the various test images quantized by the contextual algorithm and by the other two algorithms without performing spatial error diffusion and followed by spatial error respectively after quantization. According to the tables, spatial error diffusion could degrade the PSNR, however, as a means to solve the problem of contouring, the images are expected to have a higher perceived image quality. As illustrated by Figure 5-11, the image "Woman" quantized by Octree followed by spatial error diffusion, though with a lower PSNR, gives a better result than the image without performing spatial error diffusion. Notice especially how the background of the

image showing objectionable contours is improved after spatial error diffusion. Therefore, a higher PSNR does not imply a better perceived image quality, but can only be used to estimate the comparative of the various quantization schemes.

(a)	Pool	Woman	Shop	Blythe
Contextual Algorithm*	32.87	29.54	27.94	28.53
Median Cut	32.63	29.92	28.84	28.31
Octree	33.72	31.27	28.47	29.10

(b)	Pool	Woman	Shop	Blythe
Contextual Algorithm*	32.96	29.74	27.00	28.39
Median Cut	31.42	28.64	27.52	27.09
Octree	32.59	29.98	27.17	27.61

Table 5-3: PSNR (in dB) of pictures reproduced after color quantization by the Contextual algorithm, Median cut and Octree (a) without performing spatial error diffusion and (b) followed by spatial error diffusion. *The division process within the algorithm terminates when the size of the final input image array has reached the size of 8×8.

For this reason, we study the overall performance of our proposed algorithms by measuring the color reproduction errors using the S-CIELAB color difference metric. Table 5-4 (a) and Table 5-4 (b) tabulate some statistics of S-CIELAB ΔE of the test images quantized by the Contetxual algorithm, Median Cut and Octree with and without performing spatial error diffusion after quantization. By definition, ΔE of value 3 or above is noticeable. According to the tables, images after performing spatial error diffusion (Table 5-4 (b)) could greatly reduce the corresponding

percentage of error pixels with $\Delta E > 3$, indicating S-CIELAB is a measure that correlates more with human subjective perception comparing to PSNR. Consequently, it is possible for a quantization algorithm to be significantly pooper than the other algorithms in terms of PSNR and suddenly much better in terms of ΔE.

In addition to the percentage of pixels having $\Delta E > 3$, median and mode are given for evaluating the performance of our algorithm by S-CIELAB. Comparing to mean, which is very sensitive to extremities, median and mode are more informative about the picture quality. Since the median is the middle of a distribution while the mode is the most frequently occurring score in a distribution, the smaller the values the better will be the image quality and the corresponding plot of the distributions of ΔE of the test images are shown in Figure 5-12. At a first glance, our proposed algorithm performs more or less the same as either one of the two algorithms for different images. Obviously, with "Pool" and "Shop", our algorithm performs much better than Octree, while for "Woman", our algorithm resembles Octree and outperforms Median Cut.

(a) Image / Filter	Pool			Woman			Shop			Blythe		
	median	mode	pixels ΔE>3† (%)	median	mode	pixels ΔE>3 (%)	median	mode	pixels ΔE>3 (%)	median	mode	pixels ΔE>3 (%)
Contextual Algorithm	1.1581	3.19	16.01	1.6124	1.13	18.65	1.3563	0.77	18.73	1.9336	0.02	36.54
Median Cut	1.1522	1.09	21.27	1.7912	1.28	27.27	1.8289	1.15	25.76	1.6342	0.06	18.90
Octree	1.4757	3.19	24.27	1.3789	0.85	14.99	2.4282	1.82	36.68	1.4623	0.03	15.18

(b) Image / Filter	Pool			Woman			Shop			Blythe		
	median	mode	pixels ΔE>3† (%)	median	mode	pixels ΔE>3 (%)	median	mode	pixels ΔE>3 (%)	median	mode	pixels ΔE>3 (%)
Contextual Algorithm	0.3602	0.02	2.69	0.9834	0.46	8.99	0.9563	0.570	14.49	1.0178	0.02	15.02
Median Cut	0.7872	1.09	3.63	1.0839	0.61	10.91	1.1337	0.41	13.69	1.0004	0.04	8.86
Octree	0.9753	3.19	18.12	1.0243	0.41	7.82	1.7045	1.35	19.37	1.0463	0.02	7.34

Table 5-4: Statistical parameters of S-CIELAB ΔE of the Contextual algorithm (a) without performing spatial error diffusion and (b) followed by spatial error diffusion after quantization. Median Cut (MC) and Octree (OC) with and without error diffusion are included for comparison. (†A ΔE of value 3 or above is noticeable (Zhang, X. and Wandell, B. A., 1997).

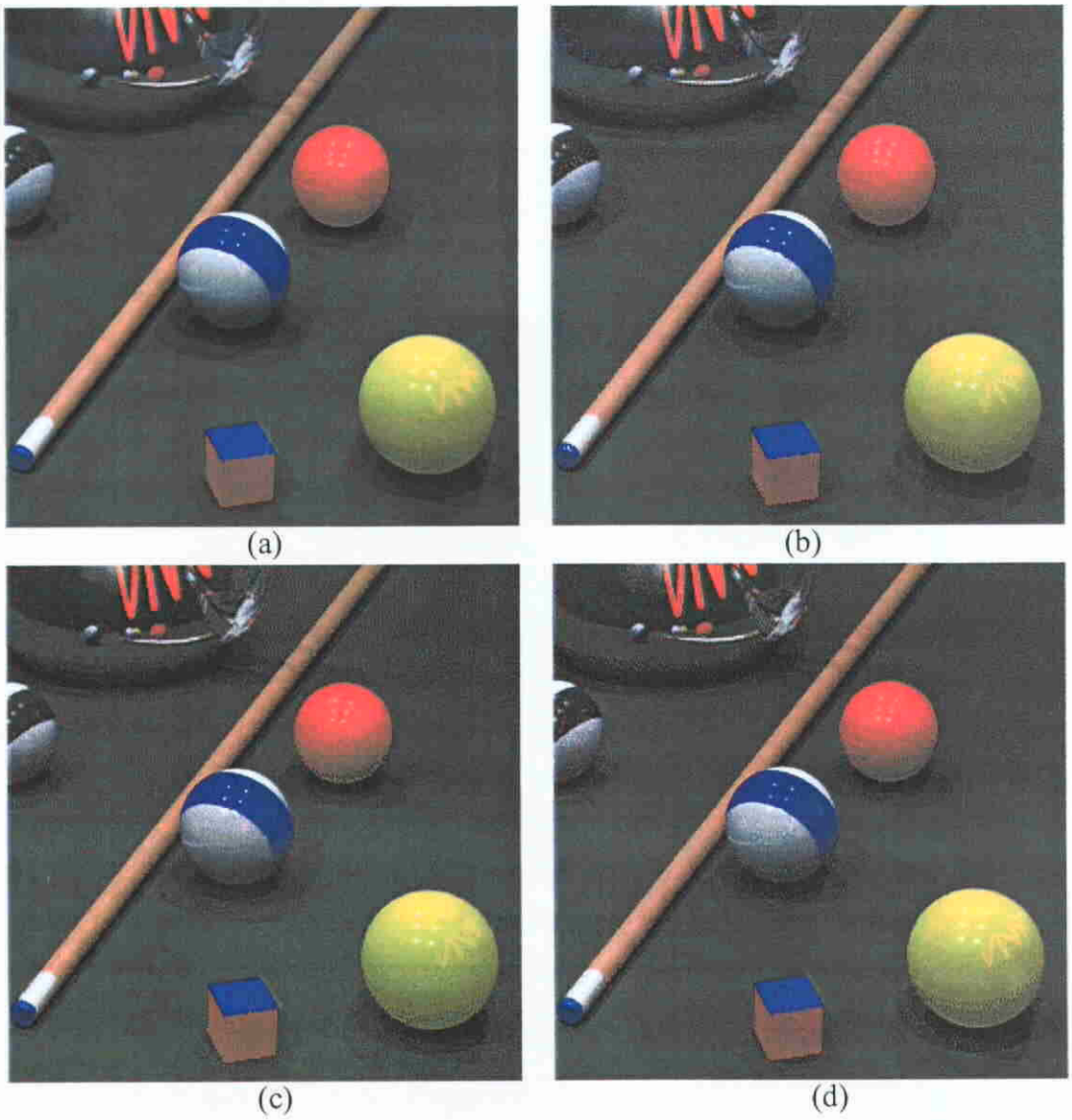


Figure 5-7: (a) Original image "Pool" used for the experiment. (b), (c) and (d) The corresponding images quantized by the Contextual algorithm, Median Cut and Octree respectively followed by spatial error diffusion. Notice especially how the various quantization schemes differ in selecting colors on the billiard balls exhibiting a smooth transition from dark to bright primary colors.

Figure 5-8: (a) Original image "Woman" used for the experiment. (b), (c) and (d) The corresponding images quantized by the Contextual algorithm, Median Cut and Octree respectively followed by spatial error diffusion. Notice especially how the various quantization schemes differ in reproducing the background exhibiting a smooth color transition.

Figure 5-9: (a) Original image "Shop" used for the experiment. (b), (c) and (d) The corresponding images quantized by the Contextual algorithm, Median Cut and Octree respectively followed by spatial error diffusion. Notice especially how the various quantization schemes differ in reproducing colors on the window display, in particular the areas around the lightings that are subjected to contouring artifacts.

Figure 5-10: (a) Original image "Blythe" used for the experiment. (b), (c) and (d) The corresponding images quantized by the Contextual algorithm, Median Cut and Octree respectively followed by spatial error diffusion. Notice especially how the various quantization schemes differ in reproducing colors on the forehead of the doll exhibiting a smooth transition.

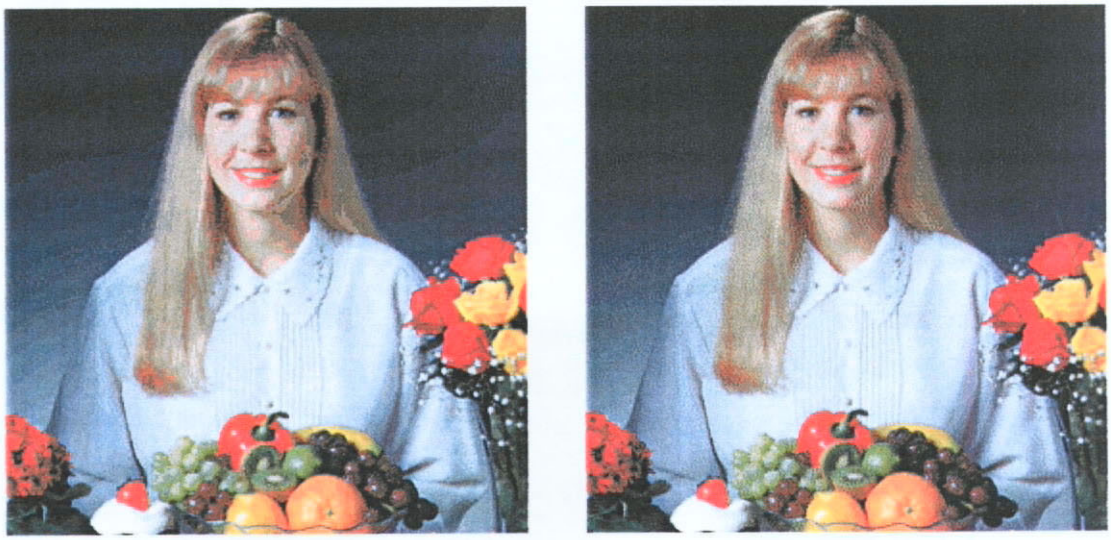


Figure 5-11: Color image "Woman" quantized by Octree (a) without performing spatial error diffusion (b) followed by spatial error diffusion. Notice especially how the problem of contouring is solved after spatial error diffusion in (b).

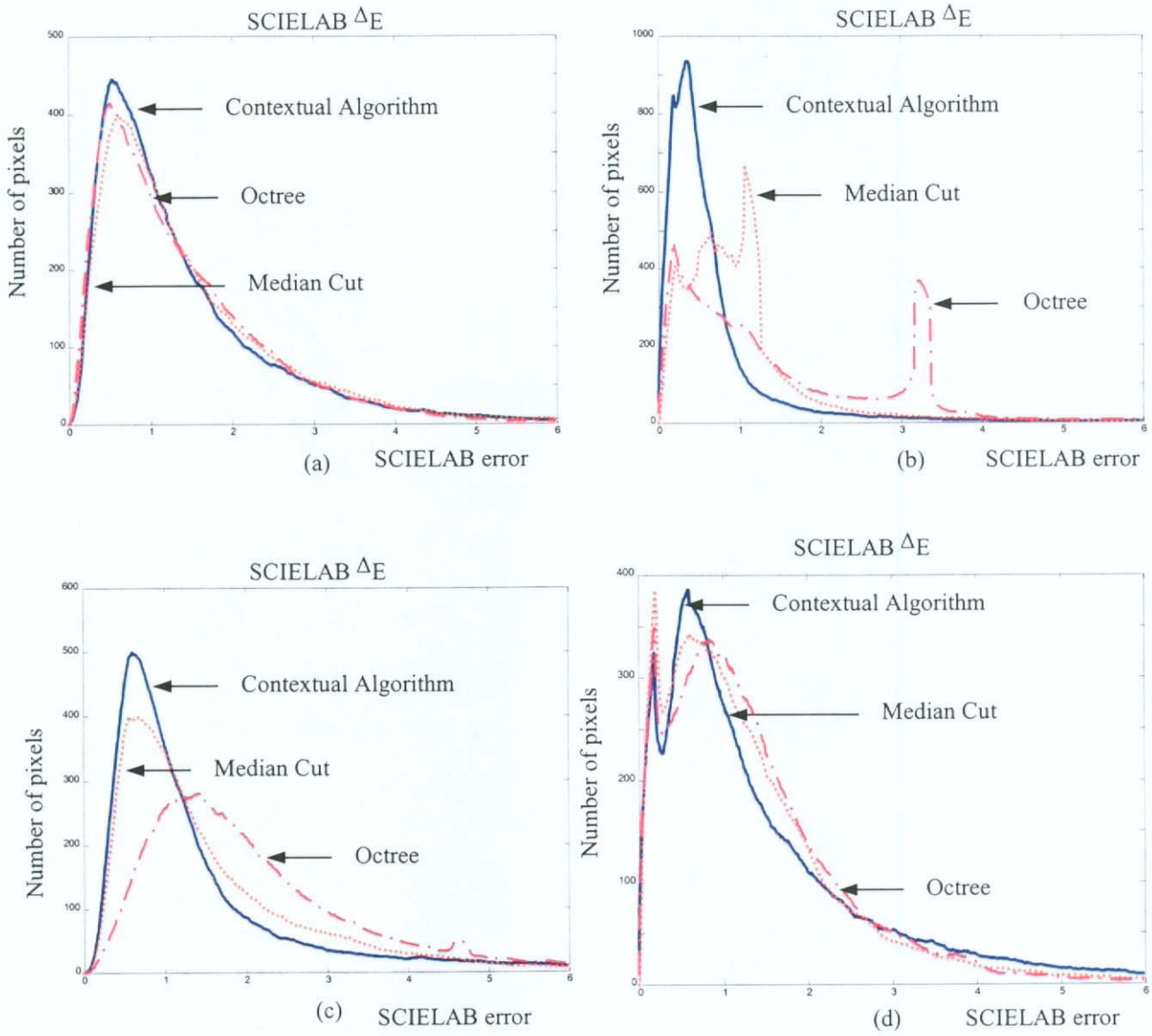


Figure 5-12: Distributions of the S-CIELAB ΔE of (a) "Woman" (b) "Pool" (c) "Shop" after color quantization and error diffusion. According to the distribution plots, the Contextual algorithm is able to produce good results with small median and mode.

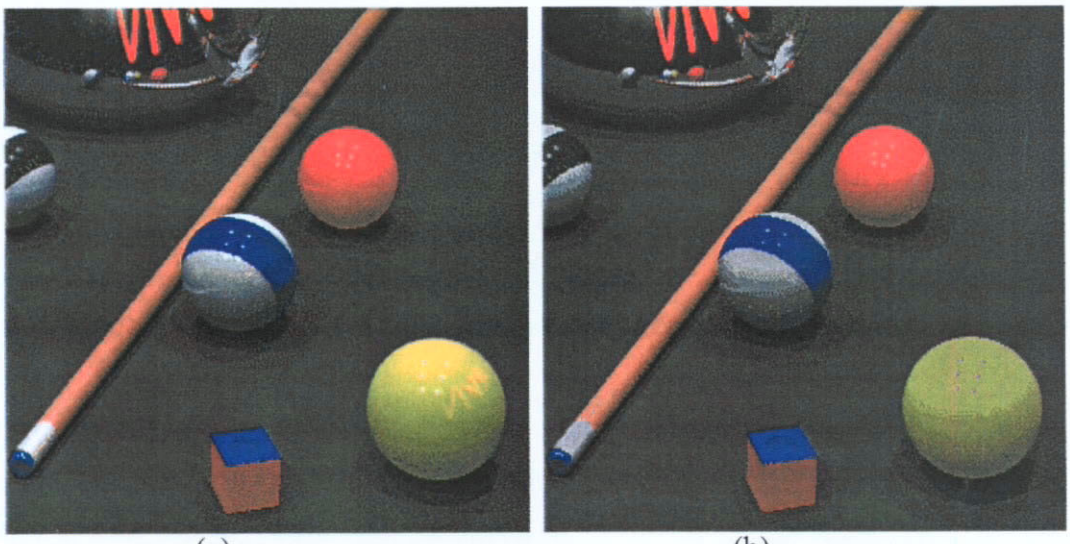
5.7 Quantization by Color Segmentation

We evaluate the contextual information of the input image and identify regions of interest by subdividing a pseudo image into image segments. Another possible way to analyze an image is by color segmentation, which has long been used in many applications to identify regions of interest and objects in the scene. An argument is that instead of using fixed spatial image segments for operation, color segmentation should be conducted first and the operation applied to color segments.

Suppose we conduct color segmentation first to separate the individual object in an image into color segments, and identify the region of interest by locating the color segment having the largest sum of elements for assigning colors. It is expected that more quantization levels will be allocated to the color segment that constitutes a larger area of the image and hence a larger sum of RGB values. Since colors having large coverage might not possibly be the same as colors that are important in contributing to the image quality, our aim to derive a mechanism to account for contextual information of an image might not be achieved.

Figure 5-13(b) is a quantized image "Pool" produced by performing color segmentation first with the technique based on the split and merge algorithm suggested by Horowitz and Pavlidis [18], then applying the quantization operations of our proposed algorithm to the colors segments followed by spatial error diffusion using the Floyd and Steinberg error diffusion filter. Obviously, the image quantized using color segments for identifying

regions of interest, similar to the other statistical quantization algorithms, has problem in allocating enough yellow to the billiard ball at the front due to the presence of large number of green pixels in the image. The result is an increase in color reproduction error, with the percentage of pixels having S-CIELAB $\Delta E > 3$ increasing form about 3% using fixed spatial segments to approximately 9.2% when color segments are used.



(a) (b)

Figure 5-13: Color Images quantized by the Contextual algorithm using (a) fixed spatial segment for operations, and (b) color segments for operations. Notice especially the deficiency in the yellow color in (b), as more quantization levels are allocated to the green pixels that give rise to an image segment of large coverage.

5.8 Summary

We propose a new color quantization algorithm based on the contextual information of the input image. Similar to other color quantization methods, spatial error diffusion or dithering is performed when mapping pixel values. Comparing with two existing popular methods, our algorithm yields comparable PSNR before dithering (Table 5-1) and good distributions of S-CIELAB errors after dithering (Figure 5-12).

By the use of a pseudo image, we provide a method for evaluating the contextual information of the input image, allowing us to easily locate the dominant colors in an image. We achieve this by dividing the image into 9 overlapping segments, which is a consequence of using a binary number 4 as the divisor. Let C be the amount of computations required for checking the pixel values of an N×N image and m be the divisor, $C = (2N/m)^2 \times (m-1)^2$, i.e. $C = 4N^2 (m-1/m)^2$. According to the equation, it is obvious that when m is much greater than 1, C approaches $4N^2$, and hence a smaller value of m will give a smaller value of C. Since 4 is the smallest binary number to be used in this case, a scheme of 9 segments is developed. If a scheme of 16 segments is to be devised and an overlap of half the area of 2 adjacent segments is to be maintained, an image will be divided into 25 squares, i.e., each side being divided by 5. Since each side has a dimension of 2^r, it is indivisible by an odd integer. Thus the resulting squares cannot be all equal. If such a scheme is nevertheless implemented, the amount of computations will be $(2N/5)^2 \times 16 = (64/25)N^2$, while computations for the case with 9

segments is $(2N/4)^2 \times 9 = (9/4)N^2$. The ratio of computations is 1.14 to1, i.e., slightly more computations are required for 16 segments.

Quantization is performed under the RGB color space, and color distance is used for identifying visually similar color. The RGB color space is, however, not perceptually uniform, which means a given change in any coordinates with the same Euclidian distance does not correspond to the same perceived color difference in all regions of the color space. In principle, a perceptually more uniform space such as CIELAB is more appropriate for the process.

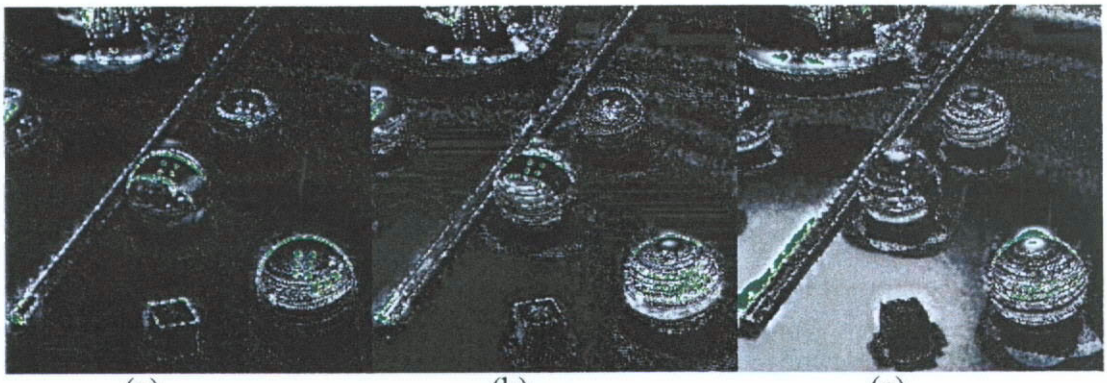
Suppose during the course of color quantization, we transform the color after color extraction from RGB format to CIELAB format, identify any visually similar colors within the CIELAB color space, transform the resulting colors determined in form of L^*, a^* and b^* back to RGB and continue to quantize the image in the RGB space. The PSNR values of the resultant images are given in Table 5-5. The results imply that the conversion could degrade the PSNR by more than 2 dB, which indicates the overall performance with color extraction operating in a uniform color space with a subsequent conversion may not be better than a quantization completely in RGB. Since most common displays operate in the RGB mode, an advantage of performing the quantization wholly in RGB is an elimination of computational errors created by transforming processes between color spaces.

PSNR (dB) / Color Space	Pool	Woman	Shop	Blythe
†RGB	32.87	29.54	27.94	28.53
†CIELAB	30.78	27.70	25.89	26.75

Table 5-5: PSNR (in dB) of pictures reproduced after color quantization by the Contextual Algorithm. The division process within the algorithm terminates when the size of the input image array has reached 8×8. †After color extraction, visually similar colors are identified in RGB and CIELAB color space (Refer to Figure 5-4).

In our experiments, we study the overall performance of our proposed algorithms by measuring the color reproduction errors using the S-CIELAB color difference metric. With contextual information taken into account, the superiority of our algorithm lies in the capability to focus on the regions of an image having important color information, thus allocating more quantization levels to these regions. Let us consider an example. The "Pool" image has been selected for its wide range of colors. Figure 5-14 is the S-CIELAB error images of 3 copies of "Pool" showing the spatial distribution of color reproduction errors resulting from different color quantization schemes. As indicated by the figures, it is apparent that our algorithm outperforms the other algorithms in providing a good selection of colors on the billiard balls exhibiting a smooth transition from dark to bright primary colors. Notice especially the deficiency in the yellow color in the images quantized by other methods. Due to the presence of large number of green pixels in the image, more quantization levels are allocated to the green colors in the background, despite the fact that the billiard balls need more colors such that an image with optimized image quality can be reproduced. This is a fundamental problem of those statistical quantization algorithms that do not account for the spatial or contextual information of the input image.

As more computation is involved in the analysis of the input image, our algorithm requires more execution time as shown in Table 5-6. Operating with a 600-MHz personal computer, the execution times for Median Cut range from 94 to 172 seconds, for Octree 145 to 249 seconds, and for the Contextual algorithm 210 to 372 seconds.



(a) (b) (c)

Figure 5-14: S-CIELAB error images of 3 copies of "Pool" reproduced from (a) the Contextual algorithm (b) Median Cut and (c) Octree. A higher intensity indicates a higher ΔE. Pixels with $\Delta E \geq 5$ are patched in green. In (b) and (c), more errors are found among the billiard balls than in (a).

Quantization Scheme \ Execution time (sec)	Woman	Pool	Shop	Blythe
Contextual Algorithm	210.91	372.84	330.92	289.75
Median Cut	93.7	172.31	99.76	164.8
Octree	181.09	248.65	145.22	227

Table 5-6: Execution times (sec) of different quantization algorithms by a 600-MHz personal computer.

Chapter 6

COLOR QUANTIZATION BY THREE-DIMENSIONAL FREQUENCY DIFFUSION

As mentioned in previous sections, the Euclidean distance in the RGB color space is not consistent with the way the human visual system deals with color difference. Theoretically, a luminance–chrominance color space that correlates with human color perception is more suitable to be adopted than the RGB for many applications. The transformation processes between color spaces is, however, bound up with computational errors. As a result, the overall performance of any color quantization scheme that involves conversion between color spaces may not be better than a quantization completely in RGB. For this reason, we propose a simple but effective color quantization technique. Our approach is to select the colors of the image with the highest frequency of occurrence to be the colormap entries from the corresponding 3D color histogram. We identify the color representatives by iteratively subdivide the 3D color histogram into subspaces in a deterministic manner until the subspace is represented by one representative color. Based on the same principle that once a pixel is quantized, error will be introduced and this error should affect the quantization of the neighboring pixels, we perform error diffusion immediately after a color representative is selected by the use of a 3D diffusion filter. In this way, error is diffused and added to the frequency counts of the

neighboring colors. While operating in the RGB color space, the algorithm is capable of producing similar effect as it does under the YUV color space with no transformation of the image required. We achieve this by using a 3D frequency diffusion filter computed to preserve the shape and hence the color distributing features of the YUV color space.

6.1 Neighbourhood

Consider a population residing in the RGB color space. Every pixel in an image is treated as a 3×1 vector that constitutes a single point, $v_s = (v_s^R, v_s^G, v_s^B)^T$, where $s = 0$, 1, 2, ... $(L_1 \times L_2)$-1 for an image of size $(L_1 \times L_2)$, and T denotes the transpose of a vector. In this color space, pixels of the same color are accumulated, giving rise to a 3D color histogram. Let $X_{(org)}$ be the input 3D color space of size $N \times N \times N$, which contains the histogram, where $N = 2^r$ and r equals to the number of bits per channel. Usually r is smaller than the original bit resolution of the image so as to scale down the histogram to better fit a palette of 256 entries. For example, a 24-bit image having 8 bits/channel would have a histogram of 5 bits/channel or $N = 32$. There are 32×32×32 = 32,768 clusters or cells. With most images, a 5-bit histogram is sufficient to yield good results.

Our algorithm is a two-step iterative algorithm. Each iteration, we first identify the colors of the image having the highest frequency of occurrence according to the input histogram $X_{(org)}$. With a naive scheme that picks all the clusters with the highest frequencies, it is likely that the palette will contain many entries of similar colors, neglecting other less popular colors. To prevent this, we consider a neighbourhood or subset of a color space:

$$X_{(org)\,k}(l_k,m_k,n_k) = \{X_{(org)\,k-1}(l_k+x, m_k+y, n_k+z) \mid x, y, z \in 0,1,\ldots, 2w-1\} \quad (1)$$

where $k = 1,2, \ldots, r-1$, $w = 2^{r-k-1}$, and l_k, m_k and n_k each takes the values of 0, w and $2w$.

In eq (1), a higher k refers to a smaller subspace such that $X_{(org)\,0}$ represents the original input. A space is divided into 27 overlapping subspaces, each occupying 1/8 of that space. The frequency f_k associated with a subspace $X_{(org)\,k}$ is defined as

$$f_k(l_k, m_k, n_k) = \sum_{x=0}^{(2w-1)} \sum_{y=0}^{(2w-1)} \sum_{z=0}^{(2w-1)} X(l_k + x, m_k + y, n_k + z) \qquad (2)$$

The highest f_k is selected and the corresponding $X_{(org)\,k}$ is subdivided according to eq (1) until $X_{(org)\,r-1}$, which is the finest level containing 8 clusters of the original histogram in each subspace. After identifying the particular $X_{(org)\,r-1}$ with the highest

f_{r-1}, the last step is to pick the single cluster having the highest frequency from that subspace, which does not require any summation in eq (2). The color of the final cluster becomes an entry in the palette.

With N=32, r = 5, the index k varies from 1 to 4 and each color space is divided into 27 overlapping subspaces as shown in Figure 6-1. A palette entry is determined after 4 iterations of dividing color spaces and 5 iterations of seeking the maximum frequency.

The process, together with frequency diffusion, allows flexibility by providing more possible paths to reach a particular location without being trapped in a local optimum. Let us consider a simplified example of picking 2 colours from 2 disjoint neighbourhoods N_1 and N_2. Suppose N_1 contains 2 major clusters C_a and C_b, while N_2 contains another 2 major cluster C_c and C_d such that their frequencies $f_a > f_b > f_c > f_d$ but with $f_c + f_d > f_b$. With a simple scheme of taking the highest counts, C_a and C_b, which could be quite close in hue, will be chosen. When considering the frequencies of neighbourhoods, C_a will be first chosen and f_a removed (see eq (5) below). The frequency in N_1 (f_b) is now smaller than that in N_2 ($f_c + f_d$), therefore C_c will be selected instead as the second entry of the palette, which would contain more varieties.

6.2 3D Frequency Diffusion

The principle of frequency diffusion is similar to error diffusion that once a pixel has been quantized, error is introduced, which will affect the next color to be picked. Here frequency diffusion is performed in a 3D color space after choosing a cluster to be an entry in the palette.

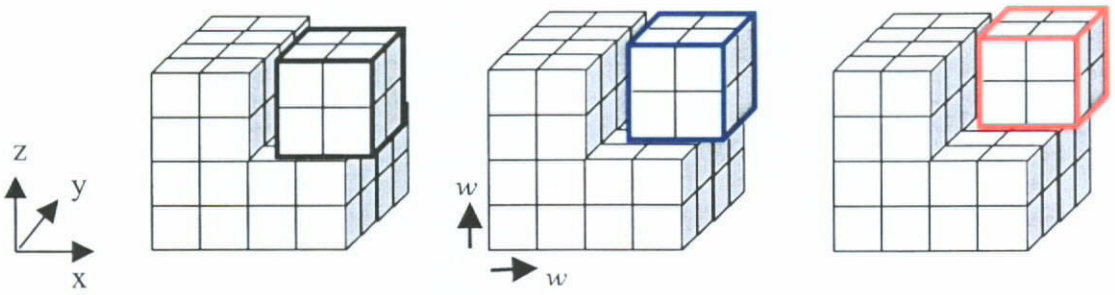


Figure 6-1: A pictorial schematic showing 3 of the 27 overlapping subspaces obtained according to eq (1). From left to right, they are $X_k(2w,0,2w)$, $X_k(2w,w,2w)$ and $X_k(2w,2w,2w)$.

To borrow the concept of quantization, an integer dot value is defined. Let f_t be the total frequency of $X_{(org)}$ and f the frequency of a cluster in $X_{(org)}$. With an 8-bit palette, we can scale f_t to 256. Then the frequency density $\phi = f / f_t$ can be calculated and a fractional dot value $d = 256\times \phi$ can be assigned to a cluster. When a cluster is chosen, a dot value is assigned by a quantization process $Q(\ )$:

$$Q(d) = \text{dot} = \max(1, \text{round}(d)) \tag{3}$$

where round( ) rounds the argument to the nearest integer. This function is required because experimental results show that there may exist in a natural picture a few clusters with a dot value larger than 1. Assuming that values in $X_{(org)}$ are normalized to 256, the error is:

$$e(l, m, n) = X(l, m, n) - \text{dot}(l, m, n) \tag{4}$$

where is X the normalized color space.

The quantization error $e(l,m,n)$ is then diffused and added to the frequency of the neighbouring cluster, giving rise to an error histogram E. A 3D filter with a convenient geometric shape, e.g. a sphere, and coefficients ω is defined for the diffusion process. The error histogram is updated as follows:

$$X(i,j,k) = E(i,j,k) = \begin{cases} 0, & \text{if } (i,j,k) = (l,m,n) \\ X(i,j,k) + \omega_{ijk} e(l,m,n), & \text{otherwise} \end{cases} \tag{5}$$

where the sum of all $\omega_{ijk} = 1$. When a part of the filter extends outside the boundary of the color space, the part is folded back into the space. After the frequency diffusion, the histogram X is updated. The quantization process is repeated until all palette entries are filled.

The frequency diffusion process regulates the selection of the palette entries such that representative colors may be chosen from clusters not too close together. In a normal picture, most dot values d are less than 1, which will generate negative errors e. If Ω is the window of support for the frequency diffusion filter, all clusters within Ω are penalized in being selected in the following iteration. Hence more variation in the palette will be obtained.

6.3 Algorithm

Step 1: Prepare a suitable color histogram X_0 of dimensions $N \times N \times N$ in a color space. For example, with a 24-bit image, we can set up a 5-bit histogram in the RGB space, i.e., there are 32×32×32 clusters. A 5-bit histogram is sufficient for most pictures.

Step 2: Specify the shape and coefficients of a 3D filter for frequency diffusion. Initialise an error histogram $E(x,y,z)$ having the same size as X_0. Set $r = \log_2 N$ and $k = 1$.

Step 3: Divide an input histogram X_{k-1} into 27 overlapping subspace according to eq (1). Find the frequency f_k of each subspace X_k according to eq (2). Seek the subspace X^m_k having the highest f_k.

Increment k. Set X'''_k as X_{k-1}, the input histogram, and repeat Step 3 until $k = r - 1$.

Step 4: Identify the cluster with the highest frequency from the 8 clusters within X_{r-1}, the final histogram found in Step 3. Select the color of that cluster as an entry of the palette. (Referring to Figure 6-1, each subspace of the size 2×2×2 will contain a total of 8 clusters.)

Step 5: Scale the total frequency f_t to 256 dots. Determine the dot value of the cluster found in Step 4 according to eq (3). Compute and diffusion the error as stated in eqs (4) and (5). Update the histogram X_0.

Step 6: Repeat Step 3 to Step 5 until the whole palette is filled.

6.4 Experiment and Discussions

The test images used for evaluation of the algorithm are shown in Figure 6-2. To start with, we test our algorithm in the RGB color space. For the frequency diffusion, spherical filters enclosed in 3×3×3 and 5×5×5 cubes are used, and the filter coefficients are given in Figure 6-3 and Figure 6-4.

The PSNR values of several test images quantized by the 3D frequency diffusion algorithm and by two other algorithms, Median Cut and Octree under RGB color

space are given in Table 6-1. All the images produced by 3D frequency diffusion give high values in PSNR, indicating a good performance of our proposed algorithm compared to the other colour quantization schemes.

To solve the problem of contouring artefacts, we perform spatial error diffusion by using the Floyd and Steinberg error diffusion filter during pixel mapping as in the previous section. The resultant images are shown in Figure 6-5 to 6-9.

Again, we study the overall performance of 3D frequency diffusion with spatial error diffusion by measuring the color reproduction errors using the S-CIELAB color difference metric. Table 6-2 tabulates some statistics of S-CIELAB ΔE of the test images and a typical plot of the distributions of ΔE of the corresponding images is shown in Figure 6-10. The results show that our algorithm behaves equally well as Median Cut and more consistently with different types of images than Octree.

	Image / Filter*	Woman	Pool	Shop	Musicians	Blythe
3D FD	SP3	29.34	30.88	28.82	30.09	29.61
	SP5a	29.23	30.71	28.98	30.01	29.54
	SP5b	29.28	30.76	29.07	30.07	29.52
	SP5c	29.27	30.81	29.01	30.08	29.55
others	Median Cut	29.49	32.63	28.84	29.82	29.90
	Octree	30.50	33.72	28.47	29.76	30.44

Table 6-1: PSNR (in dB) of pictures reproduced after color quantization by 3D frequency diffusion (FD), Median cut and Octree under RGB color space.
* Format of filter in 3D frequency diffusion, SPms :
m – the filter is enclosed in an m×m×m cube, e.g. 3×3×3.
a, b, c – different filter coefficients as shown in Figure 6-4.

Image / Filter	Woman			Pool			Shop			Musician			Blythe		
	median	mode	pixels ΔE>3 (%)	median	mode	pixels ΔE>3 (%)	median	mode	pixels ΔE>3 (%)	median	mode	pixels ΔE>3 (%)	median	mode	pixels ΔE>3 (%)
SP3	1.0526	0.37	11.85	0.7755	1.09	3.36	0.9033	0.41	12.33	0.9963	0.42	10.11	0.7317	0.04	9.82
SP5a	1.0472	0.43	11.59	0.7881	1.09	3.61	0.9268	0.41	11.49	1.0063	0.57	10.44	0.7411	0.03	12.51
SP5b	1.0556	0.39	11.72	0.7814	1.09	3.32	0.9281	0.41	12.19	1.0029	0.57	10.42	0.7434	0.03	12.63
SP5c	1.0418	0.45	12.42	0.7808	1.09	3.29	0.9023	0.41	11.66	1.0080	0.59	10.20	0.7524	0.03	12.51
MC	1.3110	0.65	16.13	0.7872	1.09	3.63	1.1337	0.41	13.69	1.0752	0.85	5.75	0.7856	0.03	9.11
OC	1.1186	0.46	8.20	0.9753	3.19	18.12	1.7045	1.35	19.37	1.3085	1.04	9.56	0.8622	0.04	5.94

Table 6-2: Statistical parameters of S-CIELAB ΔE of 3D frequency diffusion under RGB color space followed by spatial error diffusion. The filters used in 3D frequency diffusion are as stated in Table 6-1. Median Cut (MC) and Octree (OC) under RGB space with error diffusion are included for comparison.

Figure 6-2: The test images – (a) Woman (b) Pool (c) Shop (d) Musicians and (e) Blythe.

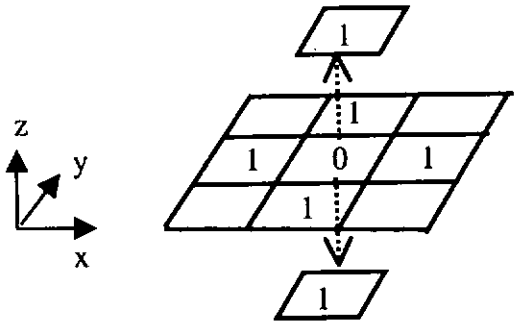


Figure 6-3: Weights of 3D frequency diffusion filter SP3, where the centre carries a weight 0. Radius = 1 in D_4 (4-neighbour) distance. Filter coefficients $\omega_{xyz} = w_{xyz}/6$, where $w_{xyz} = 1$ if distance D = 1 as shown on the diagram; otherwise $w_{xyz} = 0$.

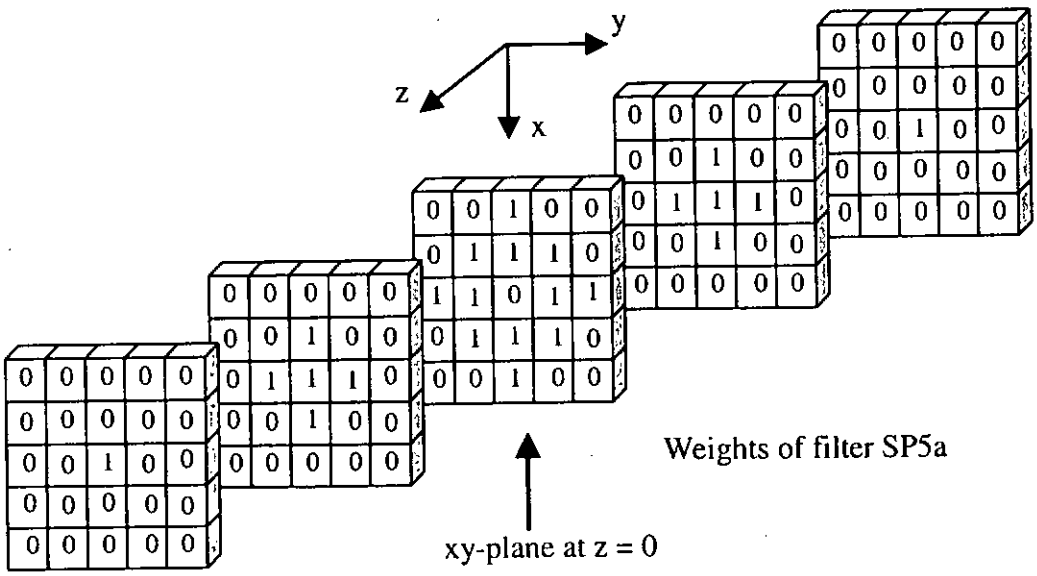


Figure 6-4: Weights of filter SP5a.
SP5a: Radius = 2 in D_4 distance. Weight w = 1 if distance D = 1 and 2; otherwise $w = 0$. Coefficients $\omega_{xyz} = w_{xyz} / \Sigma w_{xyz}$.
SP5b: w = 1 if D = 1 and z = 0; w = 0.5 if D = {1,2} and $z \neq 0$.
SP5c: w = 1 if D = 1; w = 0.5 if D = 2.

Figure 6-5: Test image "Woman" quantized by 3D frequency diffusion under RGB color space followed by spatial error diffusion. The filter used in (a) (b) (c) and (d) are SP3, SP5a, SP5b and SP5c respectively. Notice especially how the algorithm performs in reproducing the background exhibiting a smooth color transition, which is subjected to contouring artifacts. The images show that the various filters work equally well.

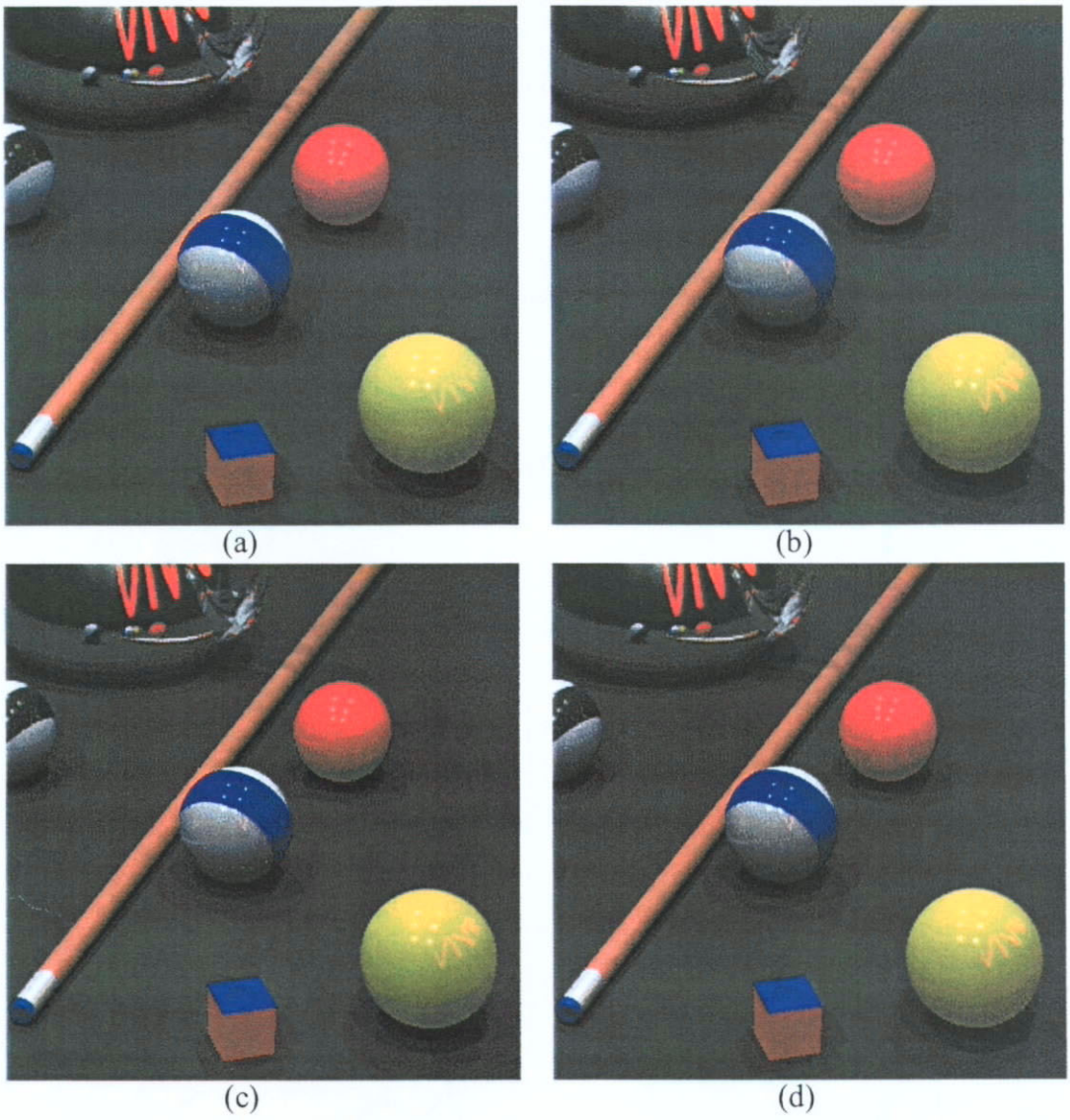


Figure 6-6: Test image "Pool" quantized by 3D frequency diffusion under RGB color space followed by spatial error diffusion. The filter used in (a) (b) (c) and (d) are SP3, SP5a, SP5b and SP5c respectively. Notice especially how the algorithm performs in selecting colors on the billiard balls exhibiting a smooth transition from dark to bright primary colors. The images show that the various filters work equally well.

Figure 6-7: Test image "Shop" quantized by 3D frequency diffusion under RGB color space followed by spatial error diffusion. The filter used in (a) (b) (c) and (d) are SP3, SP5a, SP5b and SP5c respectively. Notice especially how the algorithm performs in reproducing colors on the window display, in particular the areas around the lightings that are subjected to contouring artifacts. The images show that the various filters work equally well.

Figure 6-8: Test image "Musician" quantized by 3D frequency diffusion under RGB color space followed by spatial error diffusion. The filter used in (a) (b) (c) and (d) are SP3, SP5a, SP5b and SP5c respectively. Unlike the other test image, this image does not contain large area exhibiting smooth color transition. Whether the pictures consisting evenly of many different colors or a few dominant hues, the proposed algorithm yields good results by observation. The images show that the various filters work equally well.

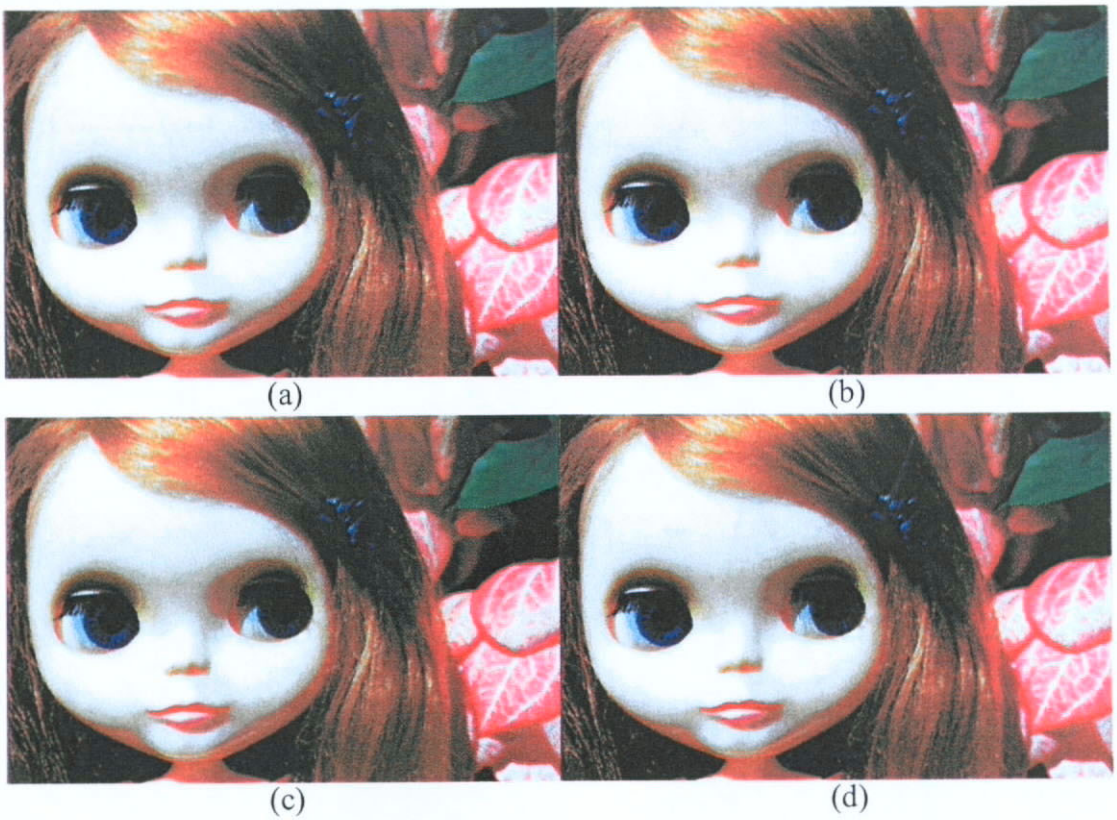


Figure 6-9: Test image "Blythe" quantized by 3D frequency diffusion under RGB color space followed by spatial error diffusion. The filter used in (a) (b) (c) and (d) are SP3, SP5a, SP5b and SP5c respectively. Notice especially how the algorithm performs in reproducing colors on the forehead of the doll exhibiting a smooth transition. The images show that the various filters work equally well.

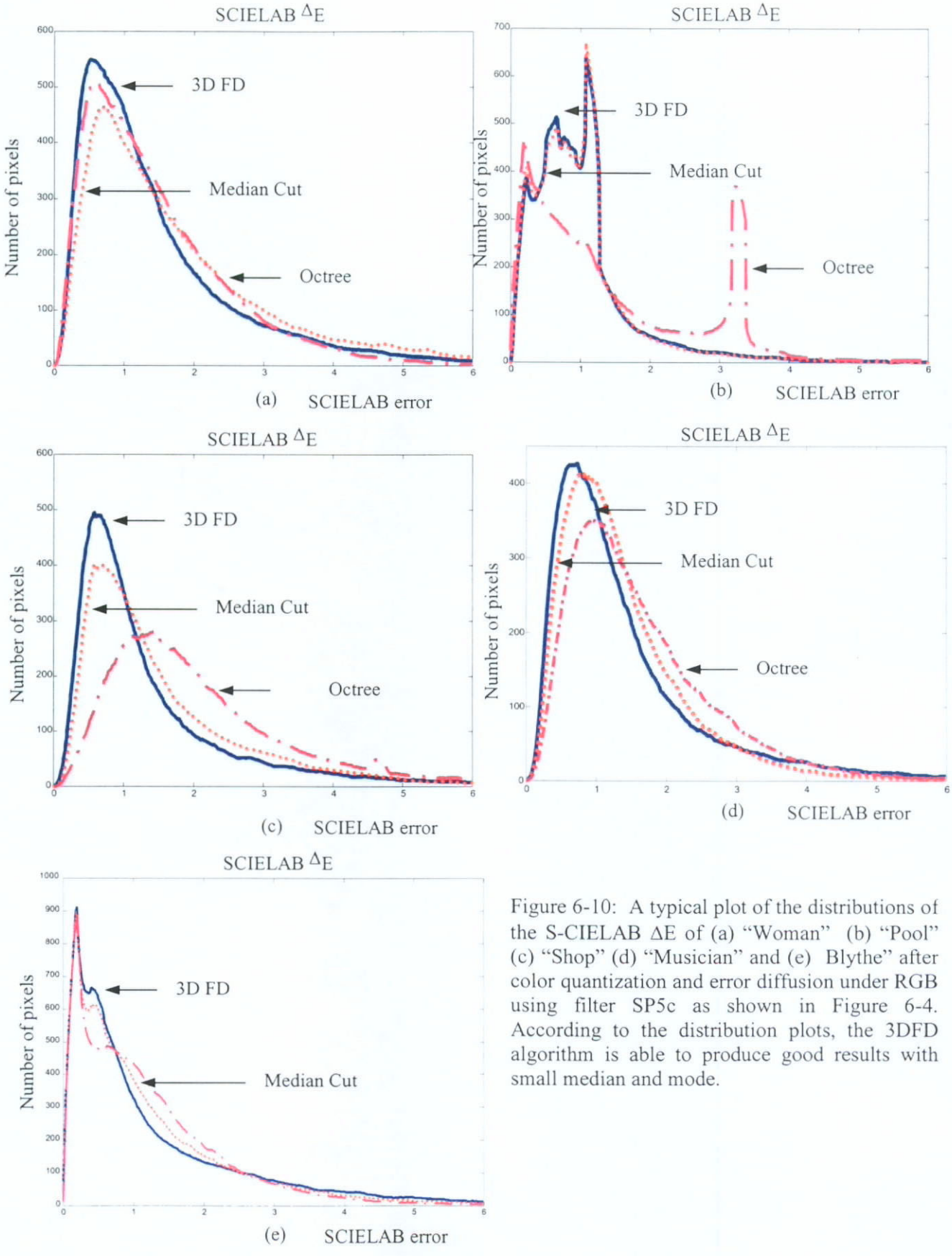


Figure 6-10: A typical plot of the distributions of the S-CIELAB ΔE of (a) "Woman" (b) "Pool" (c) "Shop" (d) "Musician" and (e) Blythe" after color quantization and error diffusion under RGB using filter SP5c as shown in Figure 6-4. According to the distribution plots, the 3DFD algorithm is able to produce good results with small median and mode.

6.5 Cross-Space Operation

Although for most color displays, processing of colors is performed on the R, G and B components, the requirement for human vision system, however, would favour a different scheme. Since human vision is more sensitive to changes in luminance than to changes in chrominance, original image quality can be retained if one is able to maintain optimal details in luminance. This would be difficult to accomplish in the RGB color space because the perception of color differences in RGB is highly non-uniform and the contribution of R, G and B components to brightness sensation is an intricate matter. Given three sources of R, G and B of the same luminance, the green will always appear the brightest and the blue will be the darkest of the three. Although blue has small contribution to the brightness sensation, human vision has extraordinarily good color discrimination capability in blue colors. For this reason, a luminance–chrominance color space, i.e., YC_bC_r or YUV, which provides independent luminance channel, is more suitable to be adopted in some applications than the RGB. In these cases, a cross-space operation is desirable and the 3D diffusion algorithm comes in handy.

Table 6-3 is the PSNR values of the test images quantized by the 3D frequency diffusion algorithm in YUV color space, provided the images are given in YUV format (requiring no conversion from RGB).

	Image / Filter*	Woman	Pool	Shop	Musicians	Blythe
3D FD	SP3	33.26	33.91	32.83	33.52	33.00
	SP5a	33.33	33.88	32.91	33.63	32.91
	SP5b	33.38	33.87	32.88	33.62	33.00
	SP5c	33.41	33.81	32.75	33.61	32.89
others	Median Cut	34.01	35.11	33.48	33.63	33.98
	Octree	35.41	39.39	33.59	34.80	35.71

Table 6-3: PSNR (in dB) of pictures reproduced after color quantization by 3D frequency diffusion (FD), under YUV color space requiring no conversion from RGB. *The filters used in 3D frequency diffusion are as stated in Table 6-1.

Comparing to Table 6-1, it is obvious that a picture quantized in YUV without conversion is able to give a PSNR approximately 3 dB higher than the case in RGB. Nevertheless, most common displays operate in the RGB mode and a conversion of the colormap is required.

Suppose an image is given in the RGB format and it is to be quantized with frequency diffusion in the YUV space. One method is that we transform the image to YUV, process it with a filter in that color space and then transform the resulting colormap back to RGB. The results as shown in Table 6-4 reveal that the conversion could degrade the PSNR by about 4 dB, which indicates that the overall performance in YUV, even if the source image is given in YUV format, may not be better than a quantization in RGB.

	Image / Filter*	Woman	Pool	Shop	Musicians	Blythe
3D FD	SP3	27.43	28.19	23.81	27.94	27.75
	SP5a	27.28	28.46	23.68	27.93	27.65
	SP5b	27.22	28.20	23.77	27.73	27.71
	SP5c	27.31	28.18	23.82	27.90	27.72
others	Median Cut	27.28	26.51	26.68	26.05	27.07
	Octree	28.27	29.46	26.91	27.13	28.57

Table 6-4: PSNR (in dB) of pictures reproduced after color quantization by 3D frequency diffusion (FD) with the image transformed to YUV, processed with a filter in that color space and then transform the resulting colormap back to RGB.

Another approach is that we keep the components as they are, transform the coordinates of the filter from YUV to RGB and perform the frequency diffusion in the RGB space. Obviously, the transforming the filter involves only a small amount of computation as its size is very small. Also, the same transformed filter can be used to process many different images.

The conversion from YUV to RGB with the values of each component kept within a dynamic range of 5 bits (and ignoring the bias) is given by [49]:

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} 1.1644 & 0 & 1.5960 \\ 1.1644 & -0.3918 & -0.8130 \\ 1.1644 & 2.0172 & 0 \end{pmatrix} \begin{pmatrix} Y \\ U \\ V \end{pmatrix} \tag{7}$$

If a filter is regarded as a 3D shape contained within a cube in the YUV space, the above transformation involves a shearing, a translation and a magnification of the cube. The 3 basis vectors in eq (7) are E_1= [1.1644 1.1644 1.1644]^T, E_2= [0 −0.3918 2.0172]^T and E_3= [1.5960 −0.8130 0]^T, where T denotes the transpose. To unify the magnification factor in all dimensions, we normalize the vectors, $e_n = E_n / \|E_n\|$, where $\|E_n\|$ is its norm. Hence we obtain

$$M = \begin{pmatrix} 0.5774 & 0 & 0.8911 \\ 0.5774 & -0.1906 & -0.4539 \\ 0.5774 & 0.9817 & 0 \end{pmatrix} \qquad (8)$$

Taking a 3×3×3 cube as an example, there are altogether 27 cells having coordinates ranging from (1, 1, 1) to (3, 3, 3). Since multiplying by M will result in a shift in origin and decimal parts in the coordinates, modifications are required to handle these situations. The shift can easily be removed by mapping the minimum values of the respective components (R, G and B) to 1. The effect of truncating the decimal part is minimized by increasing the resolution of the coordinates of the cube. We achieve this by enlarging slightly the cube, i.e., the coordinates are multiplied by a constant factor such that, after rounding, the integer parts carry most significance of the numbers. A 3×3×3 cube may be mapped to a 7×7×7 cube such that the centre in the first filter, namely, (2, 2, 2) is mapped to the centre of the second, namely, (4, 4,

4). Having mapped the coordinates between the filters, their associated coefficients are mapped accordingly.

Figure 6-11 shows a part of the quantized image "Blythe" using the SP5c filter (see Figure 6-4) with the coordinates of the filter transformed from YUV to RGB. By using a YUV-transformed frequency diffusion filter in the RGB space, our algorithm is able to reproduce atypical colors as it does under the uniform YUV color space, while at the same time retains similar PSNR (see Table 6-5). As indicated by the Fig. 6-11(c) and 6-11(d), the quantized images produced by the cross-space operation show a marked resemblance to that produced in the original YUV color space, without ignoring the greenish eye shadow around the eyes of the doll. Experimental results shows that the best result is produced using SP5c filter for both high frequency and low frequency images.

Table 6-6 tabulates some statistics of S-CIELAB ΔE of the test images. Obviously, with "Pool" and "Shop", the 3D frequency diffusion algorithm performs more or less the same as Median Cut but better than Octree. With images consisting of a few dominant colors covering relatively large areas, Octree tends to select fewer representative shades of colors from these areas than the other 2 algorithms. Consequently the distribution of error would become bimodal as shown in Figure 6-10(b). In Figure 6-11(b), where the error image of "Pool" from Octree is depicted,

we observe that a large background area near the lower left corner exhibits high errors, whereas with 3D frequency diffusion, errors shown in Figure 6-12(a) are dispersed among the ball and other small items.

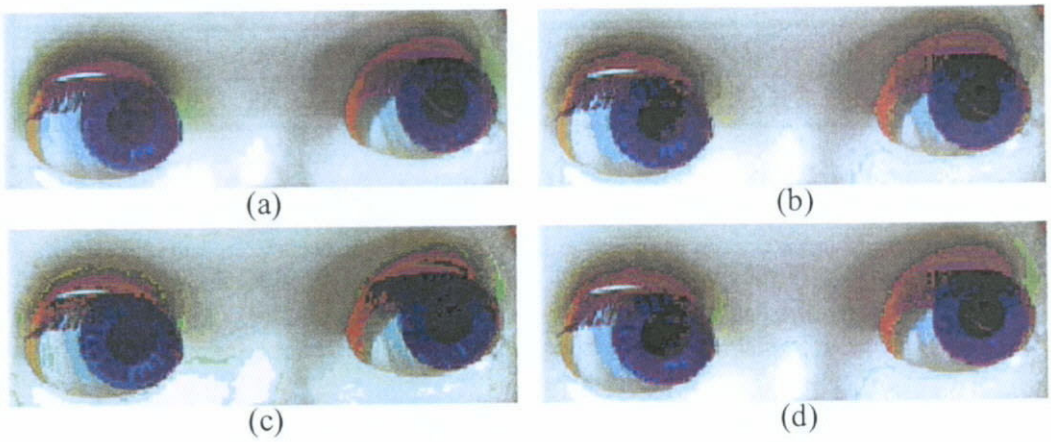


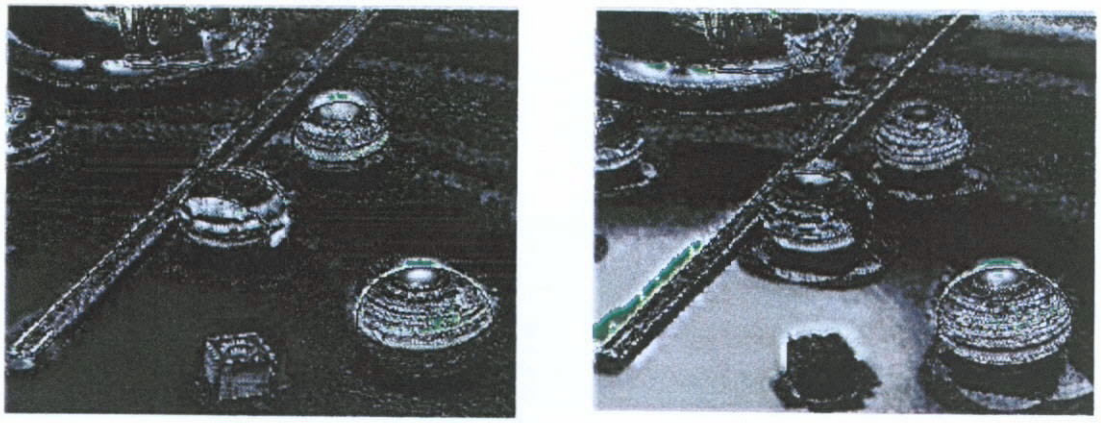
Figure 6-11: Effect of a cross-space operation indicated by the preservation of atypical colors. (a) The eyes of the original "Blythe" (b) Quantized image in the RGB color space (c) Quantized image in the YUV color space (d) Quantized image in the RGB color space with the filter transformed from YUV. Note that in (b), the greenish eye shadow around the eyes is missing. The result in (d) shows a resemblance to that in (c).

	Filter \ Image	Woman	Pool	Shop	Musicians	Blythe
3D FD	SP3	28.92	30.72	28.53	29.70	29.32
	SP5a	28.28	30.57	28.17	29.14	28.56
	SP5b	28.51	30.71	28.21	29.41	28.44
	SP5c	28.70	30.66	28.48	29.33	28.83
others	Median Cut †	27.28	26.51	26.68	26.05	27.07
	Octree †	28.27	29.46	26.90	27.13	28.57

Table 6-5: Performance of cross-space operations in terms of PSNR in dB.
3D FD is performed in RGB with a filter transformed from YUV. Filter coefficients are shown in Figure 6-3 and 6-4, where the z-axis is aligned to the Y (luminance)-axis.
† For Median Cut and Octree, the source images are transformed from RGB to YUV for quantization and the resulting palettes are reconverted to RGB.

Image / Filter	Woman			Pool			Shop			Musician			Blythe		
	median	mode	pixels ΔE†>3 (%)	median	mode	pixels ΔE>3 (%)	median	mode	pixels ΔE>3 (%)	median	mode	pixels ΔE>3 (%)	median	mode	pixels ΔE>3 (%)
SP3	1.0570	0.54	13.00	0.7816	1.09	3.63	0.9467	0.41	13.69	1.0216	0.52	11.56	0.7655	0.03	13.08
SP5a	1.0636	0.54	14.65	0.7792	1.09	3.56	0.9389	0.41	14.99	1.0494	0.65	14.18	0.8128	0.04	15.81
SP5b	1.0563	0.54	14.50	0.7681	1.09	3.29	0.9417	0.41	15.56	1.0249	0.49	13.28	0.7945	0.03	16.71
SP5c	1.0540	0.57	13.64	0.7757	1.09	3.48	0.9362	0.41	14.45	1.0479	0.72	14.27	0.7951	0.04	15.31
MC	1.3110	0.65	16.13	0.7872	1.09	3.63	1.1337	0.41	13.69	1.0753	0.85	5.57	0.7856	0.03	9.11
OC	1.1186	0.46	8.20	0.9753	3.19	18.12	1.7045	1.35	19.37	1.3085	1.04	9.56	0.8622	0.04	5.94

Table 6-6: Statistical parameters of S-CIELAB ΔE of 3D frequency diffusion under RGB color space followed by spatial error diffusion. The filters used in 3D frequency diffusion are transformed from YUV as stated in Table 6-5. Median Cut (MC) and Octree (OC) under RGB space with error diffusion are included for comparison.



(a) (b)

Figure 6-12: S-CIELAB error images of 2 copies of "Pool" reproduced from (a) 3D frequency diffusion and (b) Octree. A higher intensity indicates a higher ΔE. Pixels with $\Delta E \geq 5$ are patched in green. In (b), large errors are found in the lower right corner of the picture, whereas errors are dispersed among the ball and other small items in (a).

6.6 Summary

In this chapter, a new color quantization algorithm, 3D frequency diffusion, based on the principle of frequency diffusion in a color space is proposed. The algorithm is simple yet effective. Although the process requires only simple arithmetical calculations as described in the steps in section 3, it performs consistently well with various types of color images. Similar to other color quantization methods, spatial error diffusion or dithering is performed when mapping pixel values. Comparing

with some existing popular methods, our algorithm yields high PSNR's before dithering (Table 6-1) and comparable or even better distributions of S-CIELAB errors after dithering (Figure 6-12).

Since frequency diffusion is realized with a 3D filter under a color space, the geometric shape and coefficients of the filter can be manipulated to suit a particular application. The most convenient shape of the filter is a cube or a sphere contained within a cube. Quantization results show that cubes with 3 or 5 units per side are good choices. The use of a filter facilitates a cross-space operation mentioned in section 5. Instead of transforming the whole block of data between different color spaces, only the filter coefficients in a desired space are mapped to the existing space of the image. Having transformed a filter once, many images can be processed by the same filter. The advantage of the cross-space operation is a reduction in computational errors, such as rounding errors, because two transformation processes between color spaces, namely, one for the original image and another for the colormap after quantization, can be eliminated. In Table 6-1, we can see that a picture quantized in RGB has a PSNR of about 30 dB. Supposing there is an image given in YUV format (requiring no conversion from RGB) and quantized in YUV, its PSNR would be about 33 dB, i.e., 3 dB higher than the case in RGB. However, most common displays operate in the RGB mode and a conversion of the colormap is required. Empirical results reveal that the conversion could degrade the PSNR by

about 4 dB, which indicates that the overall performance in YUV, even if the source image is given in YUV format, may not be better than a quantization in RGB. With the cross-space operation, the PSNR's are maintained, which is revealed by comparing Tables 6-1 and 6-5. Figure 6-11 gives one example to illustrate the effect of cross-space operation. From Table 6-6, consistency in performance of 3D frequency diffusion in terms of S-CIELAB errors is demonstrated; it is observed that whether with pictures consisting evenly of many different colors or a few dominant hues, the proposed algorithm yields good results.

Chapter 7

CONCLUSIONS

In this work, we have proposed two algorithms, the Contextual algorithm and the 3D FD algorithm. The objective of the Contextual Algorithm is to incorporate spatial or contextual information of an input image to the process of color quantization, thus allowing quantization to give priorities to different regions of the image.

In our approach, we evaluate the contextual information of the input image by the use of a pseudo image generated by summing up the RGB values of the original image at each pixel. To prevent quantization being trapped in a local optimum for assigning colors, a neighborhood or subset of the image is considered. We identify the parts of the image we want to emphasize by iteratively divide the pseudo image into nine overlapping subregions in a deterministic manner. We compare their local intensities and locate the subregion containing the largest value for assigning colors as well as further division. Before proceeding to the next subregion of the highest local intensity, the colors of the highest frequency counts are extracted from each of the nine subsegments sequentially for construction of colormap. Once a color is selected as the colormap entry, the input image arrays will be updated to avoid the same color being selected in the next iteration of color extraction. The process of

division and color extraction is stopped when the final subsegment has reached a particular size and then starts over again until the whole colormap is filled by 256 colors. In this way, priority is given to the regions of an image having the greatest need for colors, allocate more quantization levels to them, and then move to another part of the image.

In considering the way the human visual system deals with color difference, the second algorithm, the 3D FD algorithm, is concerned with color space. The algorithm is a simple but effective color quantization technique, which performs 3D frequency diffusion in RGB color space. Our approach is to select the colors of an image with the highest frequency of occurrence to be the colormap entries from the corresponding 3D color histogram. Similar to the Contextual Algorithm, the frequencies of colors in a neighbourhood are considered to avoid choosing too many entries from clusters nearby. We identify the color representatives by iteratively divide the 3D color histogram into subspaces in a deterministic manner and compare their frequencies until the subspace contains one target color, i.e., a cluster. Each cluster of the histogram carries a fractional dot value d equal to its frequency density times the number of entries of a colormap. When a cluster is chosen, an integer value nearest to d will be removed from the subspace. An error will thus be introduced and diffused to its neighbours according to a 3D frequency diffusion filter. In a normal image, most clusters carry a d less than 1. In this way, a self-

correcting task is performed to penalize the neighbours of a color selected each iteration. Under this arrangement, a cross-color-space operation can be achieved by transforming the 3D frequency diffusion filter. As the geometric shape of a frequency diffusion filter in space A is transformed to space B, the color distributing features of space B is preserved. While operating in the RGB color space, our algorithm is capable of producing similar effect as it does under the YUV color space with no transformation of the image required.

7.1 Comparison of the Contextual algorithm and the 3D FD algorithm

Instead of using those images commonly used such as Lena and Pepper containing large areas in similar tones, in our experiments, we use images with wider range of colors including both areas of smooth gradation (low frequency) and fine details (high frequency) for evaluation of our algorithms. As indicated in Figure 5-5, for example, "Pool" has a large background with nearly uniform color, and billiard balls exhibiting a smooth transition from dark to bright primary colors. "Woman" has a large continuous tone background in grayscale with small details comprised of primary colors in the front. "Shop" has a large coverage of smooth gradation in the window display and fine details at the bottom. "Blythe" is composed of several large areas of colors in different ranges.

In this section, the images we used to compare the performance of the two proposed algorithms are those used in evaluating the Contextual algorithm as shown in Figure 5-5. For the Contextual algorithm, the results we used for comparison are those obtained by terminating the division process within the algorithm when the final input image array has reached the size of 8×8 and the colors selected from each subsegment are those among the first three highest frequency counts. For the 3D FD algorithm, frequency diffusion is performed in RGB using filter SP5c transformed from YUV (see Section 6.5).

In our experiments, we use the peak signal-to-noise ratio (PSNR) as a gauge to measure the effectiveness of the resulting palettes. Similar to other color quantization methods, spatial error diffusion is performed when mapping pixel values to solve the problem of contouring artefacts. The resultant images are depicted in Figure 7-1 to 7-4, and the corresponding color reproduction errors are measured by the use of the S-CIELAB color difference metric. Comparing with two existing popular methods, both algorithms yield high PSNR before dithering (Table 7-1), and comparable or even better distributions of S-CIELAB errors after dithering (Table 7-2 and Figure 7-5).

According to the results, the performance of the Contextual Algorithm and the 3D FD algorithm in solving color quantization problem is more or less the same. With spatial information of the input image incorporated into the process of color quantization, it is obvious that the Contextual Algorithm produce images with better perceived quality than images produced by 3D FD, and better results in reducing color reproduction errors referring to Table 7-2 and Figure 7-5. However, the Contextual Algorithm is very computational intensive comparing to the 3D FD method. As more computation is involved in the analysis of the input image, excessive execution time is required by the Contextual Algorithm as shown in Table 7-3, which is often the trade off between quality and complexity.

Figure 7-1: Test image "Woman" quantized by (a) the Contextual algorithm, (b) 3DFD, (c) Median cut and (d) Octree followed by spatial error diffusion. Notice especially how the various quantization schemes differ in reproducing the background exhibiting a smooth color transition.

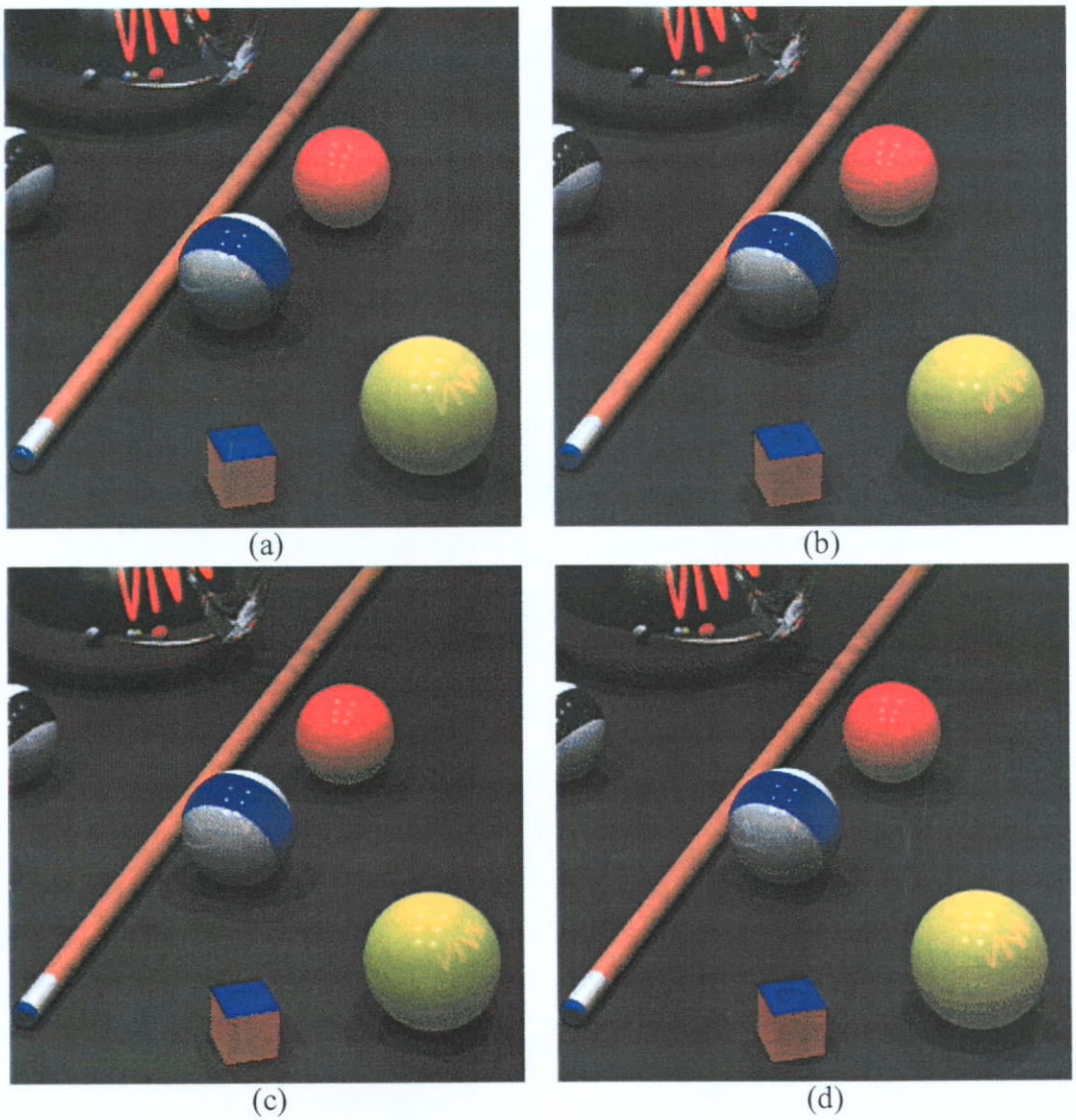


Figure 7-2: Test image "Pool" quantized by (a) the Contextual algorithm, (b) 3DFD, (c) Median cut and (d) Octree followed by spatial error diffusion. Notice especially how the various quantization schemes differ in selecting colors on the billiard balls exhibiting a smooth transition from dark to bright primary colors.

Figure 7-3: Test image "Shop" quantized by (a) the Contextual algorithm, (b) 3DFD, (c) Median cut and (d) Octree followed by spatial error diffusion. Notice especially how the various quantization schemes differ in reproducing colors on the window display, in particular the areas around the lightings that are subjected to contouring artifacts.

Figure 7-4: Test image "Blythe" quantized by (a) the Contextual algorithm, (b) 3DFD, (c) Median cut and (d) Octree followed by spatial error diffusion. Notice especially how the various quantization schemes differ in reproducing colors on the forehead of the doll exhibiting a smooth transition.

Quantization Schemes \ Image	Woman	Pool	Shop	Blythe
Contextual Algorithm	29.54	32.87	27.94	28.53
3D FD	29.40	30.66	28.48	28.20
Median Cut	29.92	32.63	28.84	28.31
Octree	31.27	33.72	28.47	29.10

Table 7-1: PSNR (in dB) of pictures reproduced after color quantization by the Contextual algorithm, 3D FD, Median cut and Octree.

Quant-ization Schemes \ Image	Woman			Pool			Shop			Blythe		
	median	mode	pixels >3ΔE (%)	median	mode	pixels >3ΔE (%)	median	mode	pixels >3ΔE (%)	median	mode	pixels >3ΔE (%)
Contextual	0.9834	0.46	8.99	0.3602	0.02	2.69	0.9563	0.57	14.49	1.0180	0.02	15.02
3DFD	0.9411	0.47	12.41	0.7757	1.09	3.48	0.9362	0.41	14.45	1.3106	0.03	20.37
MC	1.0839	0.61	10.91	0.7872	1.09	3.63	1.1337	0.41	13.69	1.0004	0.04	8.86
OC	1.0243	0.41	7.82	0.9753	3.19	18.12	1.7045	1.35	19.37	1.0463	0.02	7.34

Table 7-2: Statistical parameters of S-CIELAB ΔE of the Contextual algorithm, 3D FD, Median Cut (MC) and Octree (OC) followed by spatial error diffusion.

Quantization Scheme \ Execution time (sec)	Woman	Pool	Shop	Blythe
Contextual Algorithm	210.91	372.84	330.92	289.75
3D FD	19.77	20.93	18.73	17.96
Median Cut	93.7	172.31	99.76	164.8
Octree	181.09	248.65	145.22	227

Table 7-3: Execution times (sec) of different quantization algorithms by a 600-MHz personal computer.

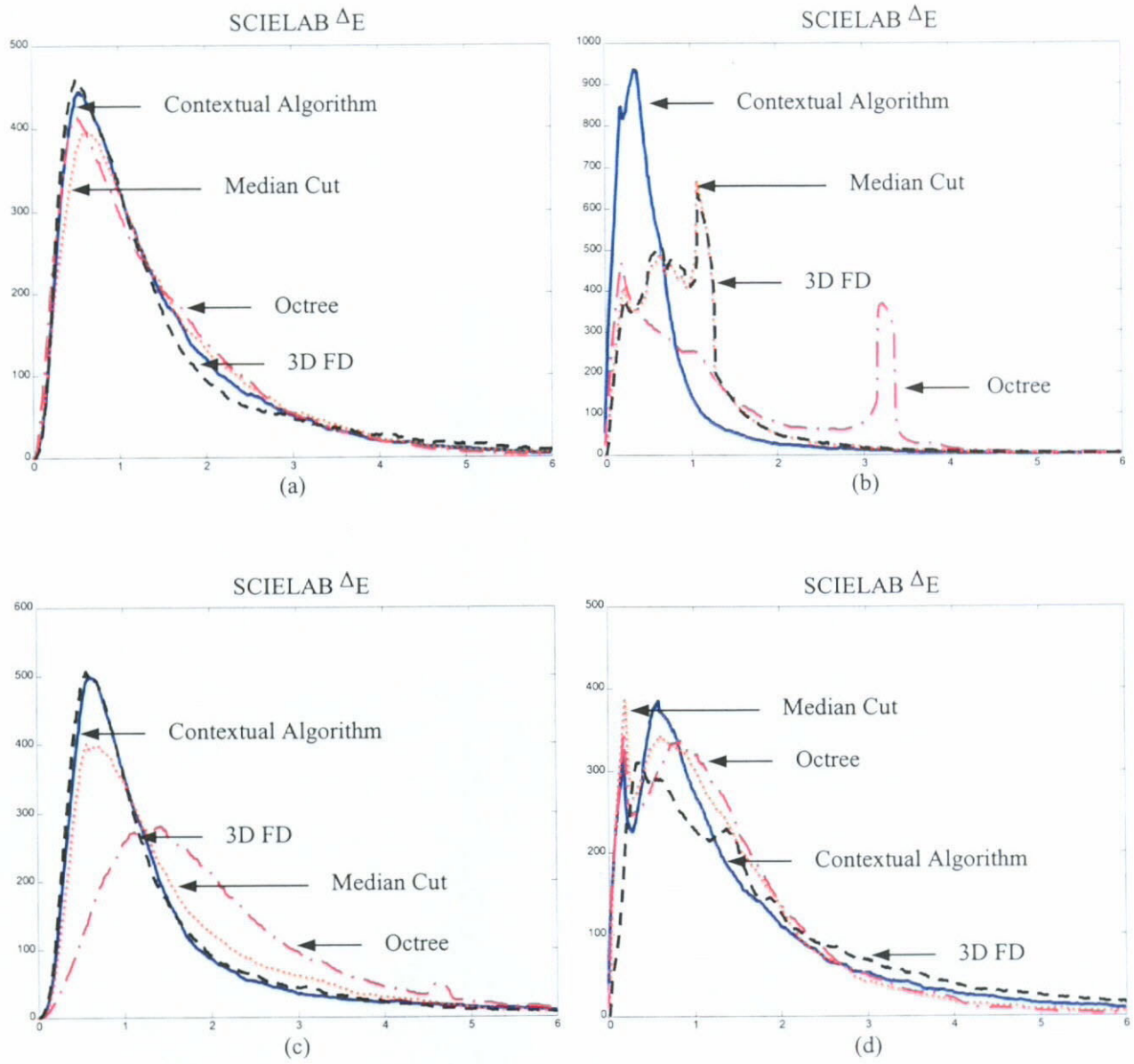


Figure 7-5: Distributions of the S-CIELAB ΔE of (a) "Woman" (b) "Pool" (c) "Shop" and (d) "Blythe" after color quantization and error diffusion.

7.2 Summary

- The Contextual algorithm, which performs color quantization based on the contextual information of the input image, is superior in focusing on the regions of an image having important color information. It is, however, very computational intensive and requires considerable execution time.

- The performance of the Contextual algorithm is affected by two factors: the size of the final subsegment at which the division process terminates, and the number of colors to be selected at a time from each subsegment for color extraction. For both factors, assumptions are made and the initial criteria are set arbitrarily.

- The process of division and color extraction is stopped at a level that, by assumption, is sufficient to include less popular colors that may be essential in contributing to the final image quality. On the other hand, further subdivision to extract colors from subregions of smaller size will be of no significance in contributing to the final image quality.

- From the experiment, it is found that best results are obtained by terminating the process of division and color extraction within the algorithm after the final input image array has reached a size 8×8. Terminating the division process at other

sizes such as 16×16 or 4×4 will not made any improvements to the overall performance of the algorithm.

- During color extraction, more than one color is selected at a time from each subsegment to ensure the selection of dominant colors in case the subsegment contains a large part of analogous background colors. At the same time, it is important to avoid selecting too many colors from a particular region, thus giving the subsequent regions insufficient prominence and neglecting some of the important colors in these regions.

- Best results are obtained when the colors among the first three highest frequency counts are selected. While decreasing the number of colors to be extracted will degrade the PSNR, increasing the number of colors to be selected will not improve the performance of the algorithm either.

- Instead of using fixed spatial image segments for operation, we use color segmentation as a means to identify regions of interest for quantization to take place. In this manner, priorities will be given to the colors segments that constitute a larger area of the image. Since colors of large coverage and colors of great importance are not equivalent, our aim to derive a mechanism to give priorities to areas having important color information might not be achieved.

- The 3D FD algorithm realizes frequency diffusion with the histogram of a color image. The algorithm is fast and effective. It is applicable to images of any size in any color space. All steps require only simple arithmetical calculations, but the algorithm performs consistently well with various types of pictures.

- The most time-consuming part of the algorithm is the preparation of the histogram, which has a complexity of N^2 (square class). For the part of frequency diffusion, the size of the histogram is fixed, i.e., independent of the image size, and so is the diffusion filter, which cannot be larger than the histogram itself. Hence the computational complexity of this part is a constant. Other algorithms, such as Median Cut, Principle Component Analysis and Octree also have an N^2 complexity. However, they require more complicated mathematical operations like finding statistical parameters or Eigenvectors of the images.

- By the use of a 3D frequency diffusion filter, error introduced during quantization is diffused to its neighbours accordingly. In this way, a self-correcting task is performed to penalize the neighbours of a color selected each iteration.

- The most convenient shape of the filter is a cube or a sphere contained within a cube. Quantization results show that cubes with 3 or 5 units per side are good choices and the best result is produced using SP5c filter for both high frequency and low frequency images.

- The algorithm works well in RGB color space. However, it is observed that images quantized in RGB tend to ignore atypical colors, especially near areas showing smooth color transition. This problem is solved when the operating color space has changed from RGB to YUV. However, the overall performance in YUV, even if the source image is given in YUV format, is not better than a quantization in RGB due to computational errors arises from conversion between color spaces.

- Instead of transforming the whole block of data between different color spaces, we achieve cross-color-space operation by transforming the coordinates of the filter from YUV to RGB and perform the frequency diffusion in the RGB space.

- By using a YUV-transformed filter in the RGB space, the algorithm is capable of producing similar effect as it does under the YUV color space, while at the same time retains similar PSNR as in RGB. Since no transformation of the image is involved, computational errors, such as rounding errors can be eliminated.

- From our experimental results, it is obvious that the Contextual algorithm is able to produce images with better perceived quality than the 3DFD as well as Median Cut and Octree.

- Comparing to the other well established algorithms such as Median Cut and Octree, 3D FD is simple, which does not involve any complicated calculations such as determination of eigenvector, and the performance is more consistent, whether with pictures consisting evenly of many different colors or a few dominant hues. Most importantly, it is able to perform cross-space operation without transforming between color spaces.

7.3 Further Development

The concept of quality in imaging is complex, especially when human factors are involved. The overall quality of a color image is the integrated result of the spatial pattern and color appearance. Since images are viewed by human, it is essential to consider the physical response of eyes to spatial patterns and the characteristics of the HVS to color perception in designing color quantization algorithms.

HVS models utilize human visual sensitivity and selectivity to model and improve perceived image quality. Many HVS models for luminance as well as for colors have been proposed to approximate the visual response of human eyes. The simplest model is very often the practice of spatial filtering that implement one of the CSFs. Besides, Weber's law [31][52][20], the modular transfer function (MTF) of a linear, shift invariant system [5], or Daley's angular dependence [58] are also commonly used. These models have been applied in every aspect of digital imaging. The application of Campbell's MTF and Daley's angular dependence in a direct binary search of digital halftoning by Analoui and Allebach [1] is an example. Other examples include the use of Kelly's CSF in color quantization of color image sequence by Atkins et al [4], or the combining of the Nasanen [38] and Sullivan et al [56] models in image analysis by Kolpatzik and Bouman[29].

While this work has provided a method to perform color quanitzation based on the contextual information of an input image, the incorporation of HVS models into the algorithm would be a desirable aspiration to improve performance. The design goal can simply be applying spatial filters to the color image to model the perceptual responses of the HVS in determining the regions of high interest. As color appearance is a complex phenomenon and involves a larger number of parameters and mechanisms to account for, the type of HVS effect to be simulated in order to produce the best result remains an open question. Further research and systematic investigations are needed to answer the question.

In the 3D FD algorithm, the color space is iteratively subdivided until the finest level containing 8 clusters is reached, and the single cluster representing one target color with the highest frequency is selected as a colormap entry. Since the original bit resolution of the image is scaled down, a single cluster might not possibly contain only one color in reality. Therefore, for development and optimization, we can further subdivide the cells with high frequency that exceed a predetermined threshold value such that more colors can be assigned to that cell by 3D FD.

Inspired by the methods developed for designing VQ codebook [14] [30] [39] [50], we suggest employing the principle of codebook training as an iterative optimization scheme after performing 3D FD. With the colormap obtained from the 3D FD

algorithm as the initial codebook, the final result is subject to the way at which the training data is selected. Suppose we make use of the mechanism that derived in giving priorities to different regions of the image from the Contextual algorithm to obtain the vectors for training. The advantages of the 3D FD algorithm can be retained while the inherent weakness as a statistical quantization method that neglects the spatial information of the input image can be eliminated. Such an approach would also benefit coding and compression algorithms.

REFERENCE

[1] Allebach, J.P., Flohr, T.J. and Hilgenberg, D. P. "Model-based halftoning via direct binary search", *Proc. IS&T's 47th Ann. Conf., ICPS'94: Physics and Chemistry of Imaging Systems*, Vol. 2, pp. 476–482 (1994).

[2] Analoui, M. and Allebach, J.P. "Model based halftoning using direct binary search". In Rogowitz, B.E., ed., *Proc. SPIE: Human Vision, Visual Processing, and Digital Display III*, 1992, Vol. 1666, pp. 96–108 (1992).

[3] Anastassiou, D. "Error diffusion coding for A/D conversion", *IEEE Trans. Circuits Syst.*, Vol. 36, pp. 1175–1186 (1989).

[4] Atkins, C.B., Flohr, T.J., Hilgenberg, D.P., Bouman, C.A. and Allebach, J. "Model-based color image sequence quantization", *Proc. SPIE*, Vol. 2179, pp. 310-317 (1994).

[5] Barnard, E. "Optimal error diffusion for computer-generated holograms", *J. Opt. Soc. Am. A*, Vol. 5, pp. 1803-1817 (1988).

[6] Braudaway, G. "A procedure for optimum choice of a small number of colors from a large color palette for color imaging", *Proc. Electronic Imaging'87*, San Francisco, CA, 1987, pp. 75–79 (1987).

[7] Chan, Y.H. "A modified multiscale error diffusion technique for digital halftoning", *IEEE Signal Processing Letters*, pp. 277-280 (1998).

[8] Evans, S. and Attaya, W.L. 'Research methods and strategies in the psychophysics of image quality', *Journal of Photographic Science and Engineering*, Vol. 22, No. 2 (1978).

[9] Fairchild, M.D. *Color Appearance Models*, Addison Wesley Longman, Inc., USA (1998).

[10] Flohr T.J. *et al.* "Model based color image quantization," *Proc. SPIE: Human Vision, Visual Processing, and Digital Display IV*, San Jose, CA, February, 1993, Vol. 1913, pp. 270–281 (1993).

[11] Floyd, R. and Steinberg, L. "An adaptive algorithm for spatial grey scale", *Proc. Soc. Inf. Disp.*, Vol.17, No.12, pp. 75-77 (1976).

[12] Gervautz, M. and Purgathofer, W. "A simple method for color quantization: Octree quantization". In Thalmann, H.M. and Thalmann, D., eds., *New Trends in Computer Graphics*, New York: Springer, 1988, pp. 219–231 (1988).

[13] Gervautz, M. and Purgathofer, W. "A simple method for color quantization: Octree quantization", Glassner, A.S., ed. *Graphics Gems*, New York: Academic, 1990, pp. 287–293 (1990).

[14] Gonzales, R.C. and Woods, R.E. *Digital Image Processing*, Addison-Wesley Publsishing Company, Inc. (1992).

[15] Granger, E.M. and Heurtley, J.C. "Visual chromaticity and modulation transfer function", *J. Opt. Soc. Am.*, Vol. 63, pp. 1173 – 1174 (1973).

[16] Heckbert, P.S. "Color image quantization for frame buffer display", *Comput. Graph.*, Vol. 16, pp. 297–307 (1982).

[17] Holst, G.C. *CCD Arrays, Cameras and Displays,* SPIE, USA (1996).

[18] Horowitz, S.L and Pavlidis, T. " Picture Segmentation by a Tree Traversal Algorithm", *Journal of the ACM*, Vol. 23, No. 2, 368-388 (1976).

[19] Hunt, R.W.G. *Measuring Color*, Ellis Horwood Ltd., 2nd Edition, Great Britain (1991).

[20] Jain, D.J. *Fundamentals of Digital image Processing*, Prentice-Hall, Englewood Cliffs, N.J., 1989, pp. 53-57 (1989).

[21] Joy, G. and Xiang, Z. "Center-cut for color image quantization", *The Visual Computer,* Vol.10, pp. 62-66 (1993).

[22] Kane, H.R. *Digital Color Halftoning*, IEEE Press, USA (1996).

[23] Kasson, M.J. and Ploaffe, W. "An analysis of selected computer interchange color spaces", *ACM Transactions of Graphics*, Vol. 11, No. 4, pp. 373 – 405 (1992).

[24] Katsavounidis, I. and Kuo, C.C.J. "A multiscale error diffusion technique for digital halftoning", *IEEE Trans. Image Processing*, Vol.6, pp. 483–490 (1997).

[25] Kim, C., Kim, S., Seo, Y. and Kweono, I. "Model based color halftoning techniques on perceptually uniform color spaces", *IS&T 47th Annual Conf.*, Rochester, NY, May 1994.

[26] Knox, K.T. "Evolution of error diffusion", *J. Electron. Imag.*, Vol. 8, No. 4, pp. 422-429 (1999).

[27] Knox, K.T. "Error image in error diffusion". In Sullivan, J.R. Rabbani, M. and Dawson, B.M., eds., *Proc. SPIE: Image Processing Algorithms and Techniques III*, 1992, Vol. 1657, pp. 268–279 (1992).

[28] Kolpatzik, B.W. and Bouman, C.A. "Optimized error diffusion for image display", *J. Electron. Imag.*, Vol. 1, pp. 277–292 (1992).

[29] Kolpatzik, B.W. and Bouman, C.A. "Optimized error diffusion based on a human visual model", *Proc. SPIE*, Vol. 1666, pp. 122-133 (1992).

[30] Linde, Y., Buzo, A. and Gray, R.M. "An algorithm for vector quantizer design", *IEEE Trans. Commun.*, Vol. COMM-28, pp. 84–89 (1980).

[31] Mannos, J.L. and Sakrison, D.J. "The effects of a visual fidelity criterion on the encoding of images", *IEEE Trans. Inf. Theory*, Vol. IT-20, pp. 525-536 (1974).

[32] Michelson, A.A. *Studies in Optics*, University of Chicago Press, Chicago, IL. (1927).

[33] Miller, R. and Sullivan, J.R. "Color halftoning using error diffusion and a human visual system model", *Proc. SPSE 43rd Annu. Meet.*, Rochester, NY, 1990, pp. 149–152 (1990).

[34] Mitsa, T. and Parker, K.J. "Digital halftoning technique using a blue-noise mask", *J. Opt. Soc. Amer.* A, Vol. 9, pp. 1920–1929 (1992).

[35] Mullen, K.T. "The contrast sensitivity if human color vision to red-green and blue-yellow chromatic gratings", *Journal of Physiology*, Vol. 359, pp. 381-400 (1985).

[36] Mulligan, J.B. and Ahumada Jr., A.J. "Principled halftoning based on human vision models". In Rogowitz, B.E. ed., *Proc. SPIE: Human Vision, Visual Processing, and Digital Display III*, 1992, Vol. 1666, pp. 109–120 (1992).

[37] Mulligan, J.B. and Ahumada Jr., A.J. "Principled methods for color dithering based on models of the human visual system", *SID Dig. Tech. Papers*, Boston, MA, May, 1992, pp. 194–197 (1992).

[38] Nasanen, R. "Visibility of halftone dot textures", *IEEE Trans. Sys. Man Cybern.*, Vol. SMC-14, pp.920-924 (1984).

[39] Nasrabadi, N.M. and King, R.A. "Image coding using vector quantization: A Review", *IEEE Trans. Common.*, Vol. COM-36, pp. 957-971 (1980).

[40] Netravali, A.N. and Haskell, B.G. *Digital Pictures: Representation, Compression, and Standards,* 2nd Edition, Plenum Press, New York, NY (1995).

[41] Orchard, M.T. and Bouman, C.A. "Color quantization of images", *IEEE Trans. Signal Processing*, Vol. 39, pp. 2677–2690 (1991).

[42] Pappas, T.N. and Neuhoff, D.L. "Model-based halftoning". In Rogowitz, B.E., Brill, M.H. and Allebach, J.P., eds., *Proc. SPIE: Human Vision, Visual Processing, and Digital Display II*, Vol. 1453, pp. 244–255 (1991).

[43] Pappas, T.N. "Model-based halftoning of color images", *Proc.IS&T's Eighth Int. Congress Advances in Non-Impact Printing Tech-nologies*, pp. 270–274 (1992).

[44] Pappas, T.N. "Digital halftoning: A model-based perspective", *J. Imag. Syst. Tech.*, Vol. 7, pp. 110–120 (1996).

[45] Pappas, T.N. "Model-based halftoning of color images," *IEEE Trans. Image Processing*, Vol. 6, pp. 1014–1024 (1997).

[46] Plataniotis, K.N. and Venetsanopoulos, A.N. *Color Image Processing and Application*, Springer, New York (2000).

[47] Poirson, A.B. and Wandell, B.A. "Appearance of colored patterns: pattern-color separability", *J. Opt. Soc. Am.*, Vol. 10, no. 12, pp. 2458- 2470 (1993).

[48] Poirson, A.B. and Wandell, B.A. "Pattern-color separable pathways predict sensitivity to simple colored patterns", *Vision Research,* Vol. 36, No. 4, pp. 515 – 526 (1996).

[49] Poynton, C. *A Technical Introduction to Digital Video,* John Wiley & Son Inc. (1996).

[50] Rabbani, M. and Jones, P.W. *Digital Image Compression Techniques* (TT7), SPIE Optical Engineering Press, Bellvue, Washington (1991).

[51] Rohally, A.M. and Buchsbaum, G. "Inference of global spatiochromatic mechanisms from contrast sensitivity functions", *J. Opt. Soc. Am.*, Vol. 5, pp. 572-576 (1988).

[52] Sakrison, D.J. "On the role of the observer and a distortion measure in image transmission", *IEEE Trans. Commun.*, Vol. COM-25, pp.1251-1267 (1977).

[53] Sekiguchi, N., Williams, D.R. and Brainard, D.H. "Efficiency in detection of isoluminant and isochromatic interference fringes", *J. Opt. Soc. Am.*, Vol. 10, pp. 2118 – 2133 (1993).

[54] Sharma, G. and Trussell, H.J. "Digital color imaging" *IEEE Trans. Image Processing*, Vol. 6, No. 7, pp. 901-932 (1997).

[55] Smith, A.R. "Color gamut transform pairs", *Proc. SIG-GRAPH'78*, Vol. 12, No. 3, pp. 12-19 (1978).

[56] Sullivan, J., Ray L. and Miller, R. "Design of minimum visual modulation halftone patterns", *IEEE Trans. Syst., Man, Cybern.*, Vol. 21, pp. 33–38 (1991).

[57] Sullivan, J., Miller, R. and Pios, G. "Optimized error diffusion based on a human vision model". In Rogowitz, B.E., ed., *Proc. SPIE: Human Vision, Visual Processing, and Digital Display III*, 1992, Vol. 1666, pp. 152–164 (1992).

[58] Sullivan, J., Miller, R. and Pios, G. "Image halftoning using a visual model in error diffusion", *J. Opt. Soc. Am. A*, Vol. 10, pp. 1714-1724 (1993).

[59] Ulichney, R.A. "Frequency analysis of ordered dither", *Proc. SPIE*, Vol. 1079, pp. 361-373 (1989).

[60] Ulichney, R.A. "Dithering with blue noise" *Proc. IEEE*, Vol. 76, pp. 56–79 (1988).

[61] Ulichney, R.A. "The void-and-cluster method for generating dither arrays". In Allebach, J. and Rogowitz, B., eds., *Proc. SPIE: Human Vision, Visual Processing, and Digital Display IV,* 1993, Vol. 1913, pp. 332-343 (1993).

[62] van der Horst, G.J.C. and Bouman, M.A. "Spatiotemporal chromaticity discrimination", *J. Opt. Soc. Am.*, Vol. 59, pp. 1482 – 1488 (1969).

[63] van Nes, F.L. and Bouman, M.A. "Spatial modulation transfer in the human eye", *J. Opt. Soc. Am.*, Vol. 57, pp. 401 – 406 (1976).

[64] Wandell, B.A. *Foundation of Vision,* Sinauer Associates, Sunderland, MA (1995).

[65] Wan, S.J., Wong, S.K.M. and Prusinkiewicz, P. "An algorithm for multi-dimensional data clustering", *ACM Trans. Math. Softw.*, Vol. 14, pp. 153–162 (1988).

[66] Wan, S.J., Prusinkiewicz, P. and Wong, S.K.M. "Variance-based color image quantization for frame-buffer display", *Color Res. Appl.*, Vol. 15, pp. 52–58 (1990).

[67] Wu, X. "Color quantization by dynamic programming and principal analysis", *ACM Trans. Graph.*, Vol. 11, pp. 348–372 (1992).

[68] Wyszecki, G. and Stiles, W.S. *Color Science, Concepts and Methods, Quantitative Data and Formulas*, John Wiley, 2nd Edition, N.Y.(1989).

[69] Yao, M. and Parker, K.J. "Modified approach to the construction of a blue-noise mask", *J. Electron. Imaging*, Vol. 3, pp. 92–97 (1994).

[70] Zhang, X. and Wandell, B.A. "A spatial extension of CIELAB for digital color reproduction", *SID Dig. Tech. Papers*, pp. 731-734 (1996).

[71] Zhang, X., Silverstein, D.A., Farrel, J.E. and Wandell, B.A. "Color Image quality metric S-CIELAB and its application on halftone texture visibility", *IEEE Compcon.*, pp. 44-48 (1997).

[72] Zhang, X., Farrel, J.E., and Wandell, B.A. "Application of a spatial extension to CIELAB", *Proc. SPIE*, Vol. 3025, pp. 154-157 (1997).

AUTHOR'S PUBLICATIONS

Journal Papers:

1. M. P. Y and K. C. Lo, "Contextual Color Quantization", *Journal of Electronic Imaging*, USA (accepted)

2. K. C. Lo, Y.H. Chan and M. P. Yu, "Colour Quantization by Three-dimensional Frequency Diffusion", submitted to *Pattern Recognition Letters*, the Netherlands (accepted)

Conference Papers:

1. M.P. Yu and K.C. Lo, "Contextual Color Quantization Algorithm", Proc., *11th International Conference on Image Analysis and Processing* (ICIAP 2001), pp.596-601, Palermo, Italy, Sept. 2001

2. M.P. Yu and K.C. Lo, "Object Recognition by Com bining Viewpoint Invariant Fourier Descriptor and Convex Hull", Proc., *2001 International Symposium on Intelligent Multimedia, Video and Speech Processing* (ISIMP 2001), pp 401-404, Hong Kong, 2001

3. K.C. Lo, Y.H. Chan and M. P. Yu, "A Novel Algorithm for Color Quantization by 3D Diffusion", submitted to *IEEE International Symposium on Circuits and Systems* (ISCAS 2003), Bangkok, Thailand, May 23 – 28, 2003