

Dual Lookup Table Algorithm: An Enhanced Method of Displaying 16-Bit Gray-Scale Images on 8-Bit RGB Graphic Systems

Thurman Gillespy III and Alan H. Rowberg

Most digital radiologic images have an extended contrast range of 9 to 13 bits, and are stored in memory and disk as 16-bit integers. Consequently, it is difficult to view such images on computers with 8-bit red-green-blue (RGB) graphic systems. Two approaches have traditionally been used: (1) perform a one-time conversion of the 16-bit image data to 8-bit gray-scale data, and then adjust the brightness and contrast of the image by manipulating the color palette (palette animation); and (2) use a software lookup table to interactively convert the 16-bit image data to 8-bit gray-scale values with different window width and window level parameters. The first method can adjust image appearance in real time, but some image features may not be visible because of the lack of access to the full contrast range of the image and any region of interest measurements may be inaccurate. The second method allows "windowing" and "leveling" through the full contrast range of the image, but there is a delay after each adjustment that some users may find objectionable. We describe a method that combines palette animation and the software lookup table conversion method that optimizes the changes in image contrast and brightness on computers with standard 8-bit RGB graphic hardware—the dual lookup table algorithm. This algorithm links changes in the window/level control to changes in image contrast and brightness via palette animation. The purpose of the algorithm is to use palette animation to mimic changes in image appearance performed by the software lookup table method after the window width and window level parameters have changed. The algorithm combines the advantages of both methods: rapid manipulation of image brightness and contrast by palette animation, and the ability to window and level on the full 16-bit image data using the software lookup table. This algorithm may be useful for applications that display 16-bit radiologic images on computers with standard 8-bit RGB graphic systems.

Copyright © 1994 by W.B. Saunders Company

KEY WORDS: window width, window level, palette animation, image display, personal computer.

RADIOGRAPHIC IMAGES obtained by the computed tomographic, magnetic resonance imaging, and computed radiography imaging modalities typically contain 9 to 13 bits of contrast resolution per pixel. Typically, each pixel is represented in memory and on disk storage as a 16-bit unsigned integer.¹ The display systems associated with these modalities allow the user to select a portion of the image

contrast range for viewing on a monitor. The user interface for this selection is typically the familiar "window" and "level" control. These display systems have special graphic hardware that can directly convert 9- to 16-bit pixels into gray shades for display on the monitor. When viewing and/or analyzing radiographic images on a display console, the full contrast resolution should be available to the user. Otherwise, not all of the image features might be visible, and any quantitative measurements obtained with region of interest analysis might not be accurate.

Displaying radiologic images on personal computers (PCs), on the other hand, presents special challenges because the graphic hardware associated with most PCs can only display gray scale images with at most 8 bits of color resolution per pixel, ie, 256 gray shades ($2^8 = 256$). Higher resolution red-green-blue (RGB) color graphic systems that use 16-, 24- or 32-bits of color resolution per pixel are now increasingly common; but while these systems can display far more than 256 colors, they are still incapable of displaying more than 256 gray shades per pixel.² There are two methods of displaying extended contrast range radiologic images on PCs. First, the 16-bit pixels can be converted to 8-bit gray-scale pixels before display. Then contrast and brightness adjustments to the image can be made via palette animation (described below). However, as noted above, because the full contrast range of the image is not available, not all image features may be visible, and any quantitative measurements of the image may contain errors. Second, a software lookup table can be used to interactively convert the 16-bit pixels to 8-bit gray-scale pixels. When coupled

From the Department of Radiology, University of Washington, Seattle, WA.

Address reprint requests to Thurman Gillespy III, MD, Department of Radiology, SB-05, 1956 NE Pacific St, University of Washington, Seattle, WA 98195.

*Copyright © 1994 by W.B. Saunders Company
0897-1889/94/0701-0006\$03.00/0*

with a window/level control, the images can be interactively viewed under different window and level settings. An optimized method for using such a software lookup table has been described.³ Although this second method retains access to the full contrast resolution of the image, there is a delay after each window and level adjustment that some users may find objectionable.

RGB GRAPHIC HARDWARE

On 8-bit RGB graphic systems, the determination of which color or gray shade is displayed on the monitor is determined by a hardware lookup table and a software data structure known as a color palette (Fig 1 and Table 1). The hardware lookup table matches the 8-bit image data value in the display memory with an RGB value: on Macintosh computers (Apple Computer, Cupertino, CA), this arrangement has been termed an indexed-pixel system.⁴ The color palette, on the other hand, determines which RGB values are in the hardware lookup table.

Adjustment of the brightness and contrast of an 8-bit gray scale image on computers with standard 8-bit RGB graphic hardware is typically accomplished by changing the RGB entries of the color palette, and then using the

Table 1. The Normal (Baseline) Color Palette

Index	Red	Green	Blue
0	0	0	0
1	257	257	257
2	514	514	514
3	771	771	771
4	1028	1028	1028
5	1285	1285	1285
6	1542	1542	1542
7	1799	1799	1799
8	2056	2056	2056
9	2313	2313	2313
10	2570	2570	2570
11	2827	2827	2827
12	3084	3084	3084
13	3341	3341	3341
14	3598	3598	3598
15	3855	3855	3855
↓	↓	↓	↓
240	61680	61680	61680
241	61937	61937	61937
242	62194	62194	62194
243	62451	62451	62451
244	62708	62708	62708
245	62965	62965	62965
246	63222	63222	63222
247	63479	63479	63479
248	63736	63736	63736
249	63993	63993	63993
250	64250	64250	64250
251	64507	64507	64507
252	64764	64764	64764
253	65021	65021	65021
254	65278	65278	65278
255	65535	65535	65535

The color palette determines which color/gray shade (RGB values) is associated with each 8-bit pixel value (the index value). The RGB values were determined from equation 4 where $P_i = i$. The RGB values (0,0,0) are converted to black on the monitor, and the values (65535, 65535, 65535) are converted to white. For grayscale images, the values for the RGB channels are equal for each index position, as in this example.

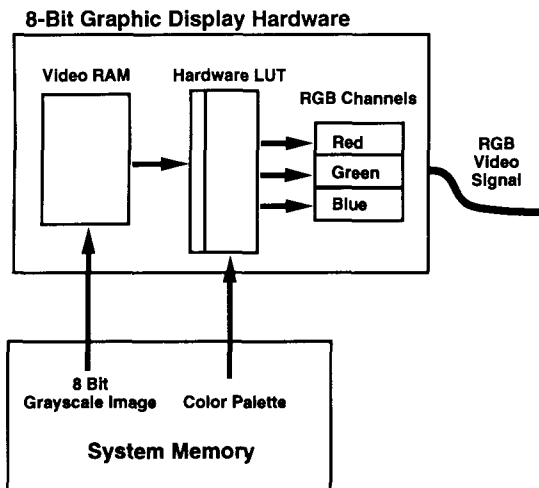


Fig 1. Indexed Color System. The 8-bit image is stored in video random access memory (RAM) from system memory. The hardware lookup table (LUT) matches the 8-bit image value with RGB intensity values. The color palette (Table 1), in system memory, determines which RGB values are stored in the hardware lookup table. The video circuitry converts the RGB values into an RGB signal, which is displayed on the monitor.²

palette to update the RGB entries of the hardware lookup table. On the Macintosh and other computers, this process is termed palette animation. The palette entries can be changed very quickly because, at most, 256 RGB values have to be altered, and each RGB value is composed of three 16-bit integers (1 each for the red, green, and blue channels). After the color palette and hardware lookup table have been altered, the displayed image reflects the changed palette at the next display monitor update, which occurs roughly 60 times per second. Palette animation allows essentially

real time adjustment of brightness and contrast of 8-bit gray-scale images.

DUAL LOOKUP TABLE ALGORITHM

In this article, we describe an algorithm that combines palette animation and the software lookup table 16- to 8-bit conversion method to optimize the display of 16-bit radiographic images on computers with standard 8-bit graphic hardware—the dual lookup table algorithm. This algorithm links changes in the window/level control to changes in image contrast and brightness via palette animation. The purpose of the algorithm is to use palette animation to mimic changes in image appearance performed by the software lookup table method after the window width and window level parameters have changed. This algorithm combines the advantages of both methods: rapid manipulation of image brightness and contrast by palette animation and the ability to window and level on the full 16-bit image data using the software lookup table.

The following equation describes a software lookup table that can convert 16-bit image values into 8-bit gray-scale values for a given window width and window level setting of the window/level control:

$$LUT_i = \left[\left[\left(\frac{256}{W} \right) (i - L) \right] + 128 \right]_0^{255} \cdot (i = 0 \rightarrow [2^n - 1]) \quad (1)$$

where LUT is an array of 8-bit gray-scale integer values bounded from 0 to 255, W is the window width, L is the window level, and n is the bit depth of the image¹ and $[2^n - 1]$ is the largest possible image data value for the image. By using the image data value of each pixel as an index for the array LUT , the 16-bit image is converted to an 8-bit gray-scale image as follows:³

$$I_{gk} = LUT[I_k] (k = 0 \rightarrow N - 1) \quad (2)$$

where I_k is the 16-bit image data value, I_{gk} is the converted 8-bit gray-scale value, N is the number of pixels in the image and LUT is the array derived in equation 1.

To link the window/level control with palette animation, we can derive the following equations, which determine new color palette RGB

values given a change in the window width and window level setting:

$$P_i = \left[\left[\frac{(W_1)(i - 128) - (256)(L_2 - L_1)}{W_2} \right] + 128 \right]_0^{255} (i = 0 \rightarrow 255) \quad (3)$$

where P_i is an 8-bit integer bounded from 0 to 255 that is related to the RGB values in the color palette (see equation 4 below), W_1 and L_1

Table 2. The Effect of Different Window Settings on the Color Palette

Index	Normal	A	B
0	0	0	0
1	257	0	0
2	514	0	0
3	771	0	0
4	1028	0	0
5	1285	0	0
6	1542	0	0
7	1799	257	0
8	2056	514	0
9	2313	771	0
10	2570	1028	0
11	2827	1285	0
12	3084	1542	0
13	3341	1799	0
14	3598	2056	257
15	3855	2570	514
↓	↓	↓	↓
235	60395	61937	63479
236	60652	62194	63736
237	60909	62451	63993
238	61166	62708	64250
239	61423	62965	64507
240	61680	63222	64764
241	61937	63479	65021
242	62194	63736	65278
243	62451	63993	65535
244	62708	64250	65535
245	62965	64507	65535
246	63222	64764	65535
247	63479	65021	65535
248	63736	65278	65535
249	63993	65535	65535
250	64250	65535	65535
251	64507	65535	65535
252	64764	65535	65535
253	65021	65535	65535
254	65278	65535	65535
255	65535	65535	65535

Each of the right three columns represents one of the RGB channels under different window/level settings. Initial values (equation 3): window (W_1) = 1000, level (L_1) = 500. Column Normal: baseline values (same as Table 1); Column A: $W_2 = 950$, $L_2 = 1000$; Column B: $W_2 = 900$, $L_2 = 1000$.

are the initial window width and window level settings (after the last 16-bit to 8-bit conversion), and W_2 and L_2 are the new window width and window level settings obtained from the window/level control. The values of P_i allow us to change the color palette so that image contrast and brightness are appropriately changed to reflect the new window width and window level settings.

$$\text{red}_i = \text{green}_i = \text{blue}_i = \frac{P_i * 65,535}{255} \quad (i = 0 \rightarrow 255) \quad (4)$$

where red_i , green_i , blue_i are the 16-bit RGB values associated with the color palette (Table 1), and P_i is defined from equation 3. The value 65,535 is the largest number that can be represented by a 16-bit integer. Finally, the values in the new color palette are transferred to the hardware lookup table, and the image appearance is altered at the next monitor refresh (palette animation).

IMPLEMENTATION

The dual lookup table algorithm has been implemented in an Macintosh application written in Think C 5.0 (Symantec Corp, Cupertino, CA). The following steps show the algorithm in practice.

1. A 16-bit image is read from disk into memory, and stored in an array. The color palette is set to the default values (Table 1).
2. A software lookup table is used to convert the 16-bit image values into 8-bit gray-scale values using default window width and window level settings (equations 1 and 2).
3. The 8-bit gray-scale values are then displayed on the monitor as gray shades using the hardware lookup table and color palette (Fig 1).
4. The user adjusts the window width and/or the window level control in the program. The original settings correspond to W_1 and L_1 in equation 3, and the new settings correspond to W_2 and L_2 .
5. The RGB values of the color palette associated with the image are changed per equations 3 and 4.
6. This altered color palette is used to change

the RGB entries in the hardware lookup table (palette animation).

7. At the next display monitor update, the image is altered.
8. The process is repeated until the user stops changing the window/level control. After a brief delay, a new set of 8-bit gray-scale values are obtained from the 16-bit image data values using the latest window width and window level settings (equations 1 and 2).

Table 3. The Effect of Different Level Settings on the Color Palette

Index	Normal	A	B
0	0	0	0
1	257	0	0
2	514	0	0
3	771	0	0
4	1028	0	0
5	1285	0	0
6	1542	0	0
7	1799	257	0
8	2056	514	0
9	2313	771	0
10	2570	1028	0
11	2827	1285	0
12	3084	1542	0
13	3341	1799	0
14	3598	2056	257
15	3855	2313	514
16	4112	2570	771
17	4369	2827	1028
18	4626	3084	1285
19	4883	3341	1542
20	5140	3598	1799
	↓	↓	↓
240	61680	60138	58339
241	61937	60395	58596
242	62194	60652	58853
243	62451	60909	59110
244	62708	61166	59367
245	62965	61423	59624
246	63222	61680	59881
247	63479	61937	60138
248	63736	62194	60395
249	63993	62451	60652
250	64250	62708	60909
251	64507	62965	61166
252	64764	63222	61423
253	65021	63479	61680
254	65278	63736	61937
255	65535	63993	62194

Each of the right three columns represents one of the RGB channels under different window/level settings. Initial values (equation 3): window (W_1) = 1000, level (L_1) = 500. Normal: baseline values (same as Table 1); Column A: $W_2 = 1000$, $L_2 = 525$; Column B: $W_2 = 1000$, $L_2 = 550$.

9. The color palette is then reset to its normal (baseline) values (Table 1).

The effect of different window/level settings on the RGB values of the color palette are further shown in Tables 2 and 3.

DISCUSSION

Our implementation of the dual lookup algorithm has met the original design goals: the change in the image appearance via palette animation closely mimics the change from the window width/window level adjustment via the software lookup table method. We believe this algorithm may be useful for applications that display 16-bit radiologic images on computers with standard 8-bit RGB graphic systems. Of

course, because palette animation only works on an 8-bit subset of the original image data, at times a banding or thresholding artifact can be noted in the image. These artifacts occurs because fewer and fewer gray scales are available to display as the adjustment of the window width/window level control continues. These artifacts are most prominent when the change in window width or window level setting is relatively large, and if the window width or the window level setting has increased in value. We have also noted a brief flash in the image when a new 16- to 8-bit conversion and color palette resetting occurs (steps 8 and 9, above). This flash can be minimized if the "black" index entry (RGB value = 0, 0, 0) is kept the same at all times.

REFERENCES

1. Gillespy T III, Rowberg AH: Radiologic images on personal computers: Introduction and fundamental principles of digital images. *J Digit Imaging* 6:81-87, 1993
2. Gillespy T III, Rowberg AH: Displaying radiologic images on personal computers. *J Digit Imaging* 6:151-163, 1993

3. Gillespy T III: Optimized algorithms for displaying 16-bit gray scale images on 8-bit computer graphic systems. *J Digit Imaging* 6:25-29, 1993
4. Graphics Overview: Inside Macintosh, Volume VI, Apple Computer Company, Addison-Wesley 1991, pp 16-16 and 16-17