A MULTISCALE COLOR QUANTIZATION ALGORITHM FOR PRODUCING SCALABLE MEDIA

Yik-Hing Fung, Yuk-Hee Chan and Ka-Chun Lui

Centre for Multimedia Signal Processing
Department of Electronic and Information Engineering
The Hong Kong Polytechnic University, Hong Kong

ABSTRACT

To reliably and efficiently deliver media information to diverse clients over heterogeneous networks, the media involved must be scalable. In this paper, a color quantization algorithm for generating scalable images is proposed based on a multiscale error diffusion framework. Images of lower resolutions are embedded in the outputs such that a simple downsampling process can extract images of any desirable resolutions. Images possessing this scalable property support transmission over the Internet which contains clients with different display resolutions, systems with different caching resources and networks with varying bandwidths and QoS capabilities.

1. INTRODUCTION

Color quantization [1-2] is widely used in many multimedia applications to save data storage requirement, save transmission bandwidth and display images with a color display device that allows only a limited number of colors. When color quantization is performed, digital halftoning [3] would be helpful to improve the quality of the output by making use of the lowpass filtering property of human eyes. At the moment, the most popular halftoning method is error diffusion and several well-known error diffusion filters such as Floyd-Steinberg filter [4] are generally used to achieve the goal.

When one delivers media information to diverse clients over heterogeneous networks, clients may support different display resolutions and systems may have different caching capabilities. In that case, it is desirable to make media information scalable such that it can be delivered efficiently and reliability. Since color-quantized images are widely used in multimedia applications nowadays, it is desirable to make them scalable such that their downscaled versions can be obtained directly with the images through some simple operations.

The most straightforward approach to obtain a downscaled version of a halftoned color-quantized image on hand is downsampling. However, this approach does not work because such an image contains a lot of high frequency energy. Figure 1 shows the effect of directly downsampling such an image which is produced with a conventional color quantization algorithm[1] in which error diffusion is involved.

Two better approaches can be used to provide downscaled versions of a halftoned color-quantized image to a client over heterogeneous networks. The first one is to generate several scaled original color images of desired sizes, color-quantize each of them and store all of them in the server for future use. This is very memory-consuming since one has to store several halftoned color-quantized versions of the original color image for one single application. The second approach is to make use of post-processing technique. In this approach, it is not necessary to generate a set of halftoned color-quantized images of various resolutions beforehand. Only one halftoned color-quantized image is stored. To obtain a halftoned color-quantized image of desired size, the available halftoned color-quantized image is first restored to its original [5] or lowpass filtered to remove the high frequency noise. The processed image is then downscaled and color-quantized again to produce the image of desirable resolution. This approach is computation-demanding since a sequence of image processing steps has to be carried out.

In this paper, we proposed an efficient approach to generate a halftoned color-quantized image that can be displayed at several resolutions. It is generated in such a way that, when downsampling is performed, a halftoned color-quantized image with a high quality rendition of the original color image at reduced resolution can result. With such a scalable property, the generated halftoned color-quantized image is also suitable for progressive transmission. The advantages of the proposed approach are obvious. First, the generated result does not require extra memory to store a set of halftoned color-quantized images of different resolutions. Besides, no postprocessing is required to produce the downscaled versions and hence no computational effort is required to generate this set of halftoned color-quantized images.

2. A FRAMEWORK OF MULTISCALE ERROR DIFFUSION

This Section presents a framework of multiscale error diffusion. In this proposed framework, color quantization is performed in YIQ color space so as to reduce the correlation among different color components. Another reason for doing so is that Euclidean distance in YIQ space matches HVS response more closely as compared with that in RGB space. This allows the color quantizer to select a visually more appropriate palette color with a given input. Without lose of generality, hereafter, we assume the color palette and the input image are defined in YIQ space. If they are not, color transformation will be required to transform their colors from their original domain to YIQ domain before color quantization.

Let $\mathbf{X}$ be a 24-bit $N \times N$ true-color image each pixel of which is represented as $\vec{\mathbf{X}}_{(i,j)}=(X_{(i,j)Y}, X_{(i,j)I}, X_{(i,j)Q})$, where $X_{(i,j)c}$ for $c \in \{Y,I,Q\}$ is the intensity value of the c^{th} primary color component of the $(i,j)^{th}$ pixel of the image.

The proposed algorithm is an iterative algorithm. Let $\mathbf{U}$ be an image which reports the current status of the image being processed at the beginning of a particular iteration. At each iteration, the algorithm first locates a pixel location based on the maximum energy guidance with an energy pyramid $\mathbf{E}$ associated with $\mathbf{U}$. The details of the pyramid will be elaborated later. The selected pixel is then color-quantized with a predefined set of colors (palette). The quantization error is diffused with a non-casual filter to neighboring pixels to update $\mathbf{U}$. These procedures are repeated until all pixels are color-quantized. At the start of the first iteration, $\mathbf{U}$ is initialized to be $\mathbf{X}$.

A. Constructing energy pyramid $\mathbf{E}$

Let $\mathbf{M}$ be a mask of size $N \times N$ which defines which pixels have been color-quantized. Specifically, its element $M_{(i,j)}$ is 0 if $\vec{\mathbf{X}}_{(i,j)}$ has been color-quantized or else it is 1.

A multiscale representation of a given color image $\mathbf{U}$ is defined as a sequence of matrices $\{\mathbf{U}^0, \cdots \mathbf{U}^l, \cdots \mathbf{U}^L\}$, where $L = \log_2 N$ and $\mathbf{U}^L = \mathbf{U}$. $\mathbf{U}^l$ is of size $2^l \times 2^l$ and its $(i,j)^{th}$ element is a triplet $(U^l_{(i,j)Y}, U^l_{(i,j)I}, U^l_{(i,j)Q})$ for $i,j = 0,1,\ldots,2^l - 1$. Elements of $\mathbf{U}^l$ for $l = 0, 1 \cdots L-2$ is defined as

$$U^l_{(i,j)c} = \sum_{m=0}^{1}\sum_{n=0}^{1} U^{l+1}_{(2i+m,2j+n)c} \qquad \text{for } c \in \{Y,I,Q\} \quad (1)$$

while elements of $\mathbf{U}^{L-1}$ is defined as

$$U^{L-1}_{(i,j)c} = \begin{cases} \dfrac{1}{S}\sum_{m=0}^{1}\sum_{n=0}^{1} M_{(2i+m,2j+n)} U^L_{(2i+m,2j+n)c} & \text{if } S \neq 0 \\ 0 & \text{else} \end{cases} \qquad \text{for } c \in \{Y,I,Q\} \quad (2)$$

$$\text{where} \quad S = \sum_{m=0}^{1}\sum_{n=0}^{1} M_{(2i+m,2j+n)} \quad (3)$$

The energy pyramid $\mathbf{E}$ associated with image $\mathbf{U}$ is then constructed with $\{\mathbf{E}^l \mid l = 0,1,\cdots L\}$, where $\mathbf{E}^l$ is the energy plane of matrix $\mathbf{U}^l$. The $(i,j)^{th}$ element of $\mathbf{E}^l$ is defined as

$$E^l_{(i,j)} = \begin{cases} \left| U^l_{(i,j)Y} + U^l_{(i,j)I} + U^l_{(i,j)Q} \right| & \text{if } 0 \le l < L \\ \left| M_{(i,j)}\left(U^L_{(i,j)Y} + U^L_{(i,j)I} + U^L_{(i,j)Q} \right) \right| & \text{if } l = L \end{cases} \qquad \text{for } i,j = 0,1,\ldots,2^l - 1 \quad (4)$$

B. Searching the pixel for color quantization

The location of a pixel to be color-quantized is determined via maximum energy guidance with energy pyramid $\mathbf{E}$. The location is obtained by searching the energy pyramid from the coarsest level $\mathbf{E}^0$ to the finest level $\mathbf{E}^L$. Note that $\mathbf{E}^0$ contains only one element $E^0_{(0,0)}$.

Assume that we are now at position $(l,(i,j))$ which corresponds to the $(i,j)^{th}$ element of a particular level l. We check $\{E^{l+1}_{(2i+m,2j+n)} \mid m,n = 0,1\}$ and proceed to the position $(l+1,(2i+p,2j+q))$ such that $E^{l+1}_{(2i+p,2j+q)}$ is maximum in $\{E^{l+1}_{(2i+m,2j+n)} \mid m,n = 0,1\}$ and $p,q \in \{0,1\}$. If more than one position satisfies the criterion, one of them will be randomly selected.

C. Color quantization and error diffusion

Let $(L,(m,n))$ be the position that we finally reach at the finest level of the pyramid $\mathbf{E}$ in the search and $C = \{\hat{\mathbf{v}}_i : i = 1,2,\cdots N_c\}$ be the given color palette. $\vec{\mathbf{U}}_{(m,n)} = (U_{(m,n)Y}, U_{(m,n)I}, U_{(m,n)Q})$ is then color-quantized. The best-matched color in the palette, say $\hat{\mathbf{v}}_k$, is selected based on the minimum Euclidean distance criterion in YIQ color space as follows.

$$\left\| \vec{\mathbf{U}}_{(m,n)} - \hat{\mathbf{v}}_k \right\| \le \left\| \vec{\mathbf{U}}_{(m,n)} - \hat{\mathbf{v}}_l \right\| \qquad \forall\ \hat{\mathbf{v}}_l \in C \quad (5)$$

The quantization error $\vec{\varepsilon} = \hat{\mathbf{v}}_k - \vec{\mathbf{U}}_{(m,n)}$ is then diffused to $\vec{\mathbf{U}}_{(m,n)}$'s neighborhood to update image $\mathbf{U}$ with a non-causal filter. In formulation, it is given as

$$\vec{\mathbf{U}}_{(i,j)} = \vec{\mathbf{U}}_{(i,j)} - W_{(m-i,n-j)}\vec{\varepsilon} \qquad \text{for} \quad i = m \pm 1 \text{ and } j = n \pm 1 \quad (6)$$

where W is defined as $W = \begin{bmatrix} W_{(-1,-1)} & W_{(-1,0)} & W_{(-1,1)} \\ W_{(0,-1)} & W_{(0,0)} & W_{(0,1)} \\ W_{(1,-1)} & W_{(1,0)} & W_{(1,1)} \end{bmatrix} = \frac{1}{12}\begin{bmatrix} 1 & 2 & 1 \\ 2 & -12 & 2 \\ 1 & 2 & 1 \end{bmatrix}$. To handle the boundary and the corner pixels, W is modified to be $\frac{1}{8}\begin{bmatrix} 0 & 0 & 0 \\ 2 & -8 & 2 \\ 1 & 2 & 1 \end{bmatrix}$ and $\frac{1}{5}\begin{bmatrix} 0 & 0 & 0 \\ 0 & -5 & 2 \\ 0 & 2 & 1 \end{bmatrix}$ respectively to avoid energy leakage.

3. PROPOSED MULTISCALE MULTIRESOLUTION VECTOR ERROR DIFFUSION ALGORITHM

With the framework presented in the previous Section, a color quantization algorithm for generating scalable color quantization images is proposed in this Section.

Consider the case that one wants to produce a color quantization result of a given image $\mathbf{I}$ which embeds a set of color quantization results of downscaled versions of $\mathbf{I}$. Let

$\mathbf{I}^r$ be one of the downscaled versions of $\mathbf{I}$. Without loss of generality, we assume that $\mathbf{I}$ is of size $N \times N$ and $\mathbf{I}^r$ is of size $(N/s_r)\times(N/s_r)$, where $s_r \in \{2^r \mid r=1, 2 \ldots R; R<L=\log_2 N\}$ is a desirable scaling factor. The objective of the proposed algorithm is to produce an output such that all $\mathbf{Y}^r$ can be obtained by simply downsampling $\mathbf{Y}$, where $\mathbf{Y}$ and $\mathbf{Y}^r$ are, respectively, the color quantization results of $\mathbf{I}$ and $\mathbf{I}^r$.

Note $\mathbf{I}$ can be downscaled with any approach to obtain $\mathbf{I}^r$, producing different results. In our proposed algorithm, $\mathbf{I}^r$ is obtained by averaging $\mathbf{I}$ as follows.

$$I^r_{(i,j)c} = \frac{1}{s_r \times s_r} \sum_{m=0}^{s_r-1} \sum_{n=0}^{s_r-1} I_{(s_r i+m, s_r j+n)c}$$

for $i, j = 0,1,\ldots,(N/s_r)-1$ and $c \in \{Y, I, Q\}$ (7)

where $I^r_{(i,j)c}$ and $I_{(i,j)c}$ are, respectively, the c^{th} color components of the $(i,j)^{th}$ pixels of $\mathbf{I}^r$ and $\mathbf{I}$.

In the proposed algorithm, starting with $r = R$, we iteratively generate $\mathbf{Y}^r$ with $\mathbf{I}^r$ and then use $\mathbf{Y}^r$ as a constraint to produce $\mathbf{Y}^{r-1}$ in the next iteration until $\mathbf{Y}$ is eventually obtained.

As selected by the user, $\mathbf{Y}^R$ is of the lowest resolution to be supported in the scalable $\mathbf{Y}$. There is no constraint to generate it and one can make use of the multiscale error diffusion algorithm presented in Section 2 to generate it with $\mathbf{X}=\mathbf{I}^R$.

To obtain $\mathbf{Y}^r$ with $\mathbf{I}^r$ for $0<r<R$, the same multiscale error diffusion algorithm presented in Section 2 can be used by embedding a constraint in the initialization stage. Suppose one has already obtained $\mathbf{Y}^r$ with $\mathbf{I}^r$ and starts to produce $\mathbf{Y}^{r-1}$ with $\mathbf{I}^{r-1}$. At the start of the first iteration, after initializing $\mathbf{U}$ to be $\mathbf{X}=\mathbf{I}^{r-1}$, we force the downsampled elements of $\mathbf{Y}^{r-1}$ to be

$$Y^{r-1}_{(2i,2j)c} = Y^r_{(i,j)c} \qquad \text{for } i, j = 0,1,\ldots,(N/s_r)-1 \quad (8),$$

where $Y^r_{(k,l)c}$ is the c^{th} color component of the $(k,l)^{th}$ element of $\mathbf{Y}^r$, and then diffuse the quantization error at positions $(2i,2j)$'s with eqn.(6) to update $\mathbf{U}$. Note assignment (8) guarantees that $\mathbf{Y}^r$ can be obtained by simply downsampling $\mathbf{Y}^{r-1}$. This completes the first iteration and the following iterations are carried out as usual as it is presented in Section 2 until $\mathbf{Y}^{r-1}$ is obtained.

4. SIMULATION AND COMPARATIVE STUDY

Simulation was carried out to evaluate the performance of the algorithm on a number of *de facto* standard 24-bit full color images. Each of them is of size 256×256. The proposed algorithm was applied to these images to obtain their corresponding halftoned color quantization results. Color palettes of different size were used in the simulation. In the realization of the proposed algorithm, parameter R was selected to be 4.

For comparison, halftoned color quantization results were also generated with some other CQ algorithms[1,6,7] and then downsampled to produce various downscaled versions. Unlike most color halftoning algorithms which are dedicated for printing applications, these evaluated algorithms [1,6,7] are not straightforward extension of binary halftoning and are able to handle color quantization in which any arbitrary palettes can be used. Floyd-Steinberg filter [4] was used in the realization of [1].

Table 1 shows the average S-CIELAB difference ($\Delta\mathbf{E}$) values [8] of the color quantization results of different algorithms and their corresponding downscaled versions. The set of testing Images include *Lenna, Baboon, Peppers, Fruits, Cycles, Airplane, Parrots, Caps, Windows* and *Pool*. Our proposed algorithm is obviously better and the superiority is very significant when the scaling factor is large.

Algorithm	Palette size	Image			
		Full-scale (256×256)*	Downsampled versions		
			s_r=2 (128×128)*	s_r=4 (64×64)*	s_r=8 (32×32)*
Orchard [1]	16	35.38	36.70	38.82	42.93
	32	26.85	28.85	32.13	37.41
	64	20.06	22.46	26.56	33.17
	128	16.27	19.06	23.75	30.87
Akarun [6]	16	34.78	36.42	38.62	42.46
	32	26.84	28.84	32.14	37.59
	64	19.92	22.39	26.42	32.97
	128	16.08	18.90	23.58	30.56
Özdemir [7]	16	37.38	38.63	40.15	43.59
	32	30.97	32.16	34.43	38.28
	64	24.51	25.85	28.54	33.18
	128	22.37	23.55	26.35	30.68
Proposed	16	34.27	34.09	33.67	31.92
	32	26.27	26.24	26.18	24.42
	64	19.74	19.86	19.95	18.59
	128	16.09	16.38	16.60	15.27

* image size

Table 1 Average S-CIELAB difference $\Delta\mathbf{E}$ of the simulation results

Figures 1 and 2, respectively, show the results produced by Orchard's algorithm [1] and the proposed algorithm. The palette used for producing Figures 1(a) and 2(a) is of size 32 and was generated with median-cut algorithm [2]. One can see that the downscaled results of Figure 1(a) are very poor while those of Figures 2(a) can maintain the color feature of the image.

5. CONCLUSION

In this paper, we proposed a multiscale error diffusion framework for color quantization. Based on this framework, we proposed a color quantization algorithm which is able to produce scalable color-quantized images for delivering media information to diverse clients over heterogeneous networks reliably and efficiently. In particular, for any given image, this algorithm can produce a high-quality directional-hysteresis-free output and simultaneously embed a set of color quantization results of the downscaled versions of the given image without any memory overhead. Images of desirable resolutions can be extracted by simply downsampling the output.

6. ACKNOWLEDGEMENT

This work was supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region (PolyU 5205/04E).

REFERENCES

[1] M. T. Orchard and C. A. Bouman, “Color quantization of images,” *IEEE Trans.* On Signal Processing, Vol.39, No.12, pp.2677-2690, Dec 1991.

[2] P. Heckbert, “Color image quantization for frame buffer displays,” Comput. Graph., vol. 16, no. 4, pp. 297-307, 1982.

[3] R. S. Gentile, E. Walowit, and J. P. Allebach, “Quantization and multi-level halftoning of color images for near original image quality,”Proc. SPIE, vol. 1249, pp. 249-259, 1990.

[4] R. W. Floyd and L. Steinberg, “An Adaptive Algorithm for spatial Grayscale,”Proceedings of the society for Information Display, Vol.17, No.2, pp. 75-77, 1976

[5] Y.H. Fung and Y.H. Chan, "Restoration of halftoned color-quantized images using projection onto convex sets," Proc. IEEE ICIP'04, Singapore, Oct 24-27, 2004, pp.325-328.

[6] L. Akarun, Y. Yardımcı and A.E. Çetin, "Adaptive Methods for Dithering of Color Images", *IEEE Trans. on Image Processing*, July 1997, Vol. 6, no.7, pp. 950-955.

[7] D. Özdemir and L. Akarun, “Fuzzy Error Diffusion”, *IEEE Trans. on Image Processing*, April 2000, Vol. 9, no 4, pp. 683-690.

[8] X. Zhang and B. Wandell. “A spatial extension of cielab for digital color image reproduction,” Proc. Soc. Inform. Display 96 Digest, San Diego, 1996, pp. 731-734.

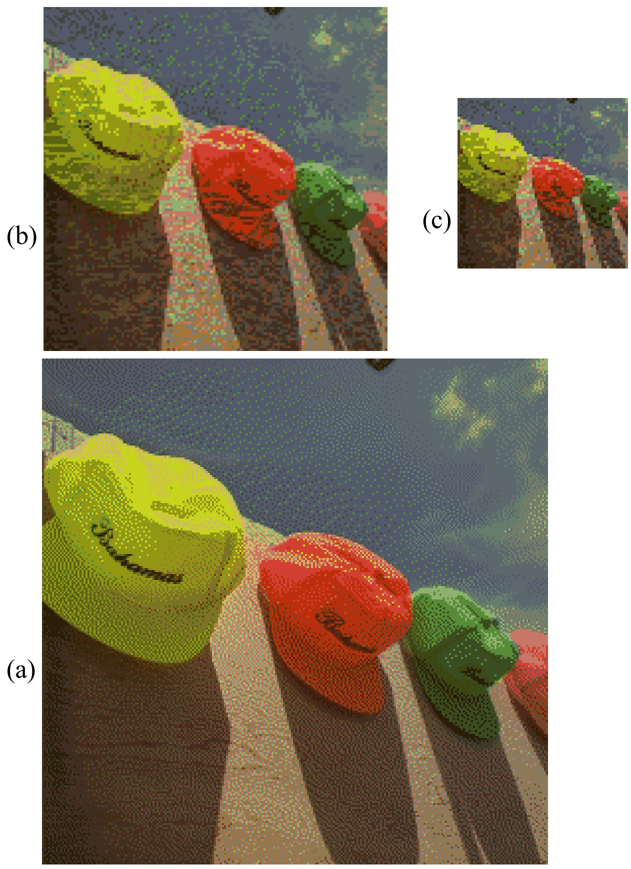

Figure 1. Color quantization result of Orchard’s algorithm [1]: (a) full-scale version and (b-c) downsampled versions. (b) s_r =2 and (c) s_r =4

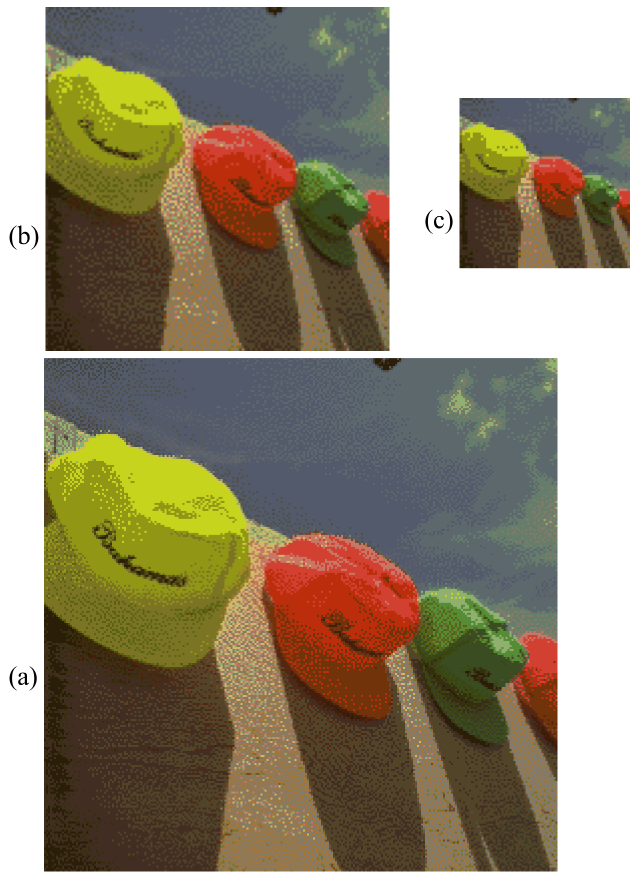

Figure 2. Color quantization result of the proposed algorithm: (a) full-scale version and (b-c) downsampled versions. (b) s_r =2 and (c) s_r =4