Image Approximation by Preserving Color Features

Wu-Ja Lin[1, 2], Wei-Yu Han[1], Kuang-Shyr Wu[1], and Yen-Po Lee[1]

[1]Department of Computer Science and Information Engineering, Ching Yun University
229, Chien-Hsin Rd., Jung-Li, Taiwan 320, R.O.C., [2]Email: wj.lin@cyu.edu.tw

Abstract

In this paper, an image approximation method is proposed. The method approximates the given image by preserving image's average color and color contrast. The experimental results show that the proposed method usually yields satisfactory results.

Keywords: average color, color contrast, feature-preserving, image approximation.

1. Introduction

Image approximation provides users a quick overview of images when the local storage size or network bandwidth resources are limited. It is widely used in many applications. For example, low-resolution roadmap images are usually transmitted over the network from the map web site when users are surfing the maps and has not yet identify the location of interest. This strategy could effectively keep the network bandwidth consumption low and thus provide more users to surf the same web site. Image compression using vector quantization (VQ) [1-4], on the other hand, is another application within which the image approximation is used. Baker and Gray propose a Mean Residual VQ (MRVQ) algorithm [5-6] which had been shown to be better than the direct VQ [7]. Hang and Haskell also proposed the interpolative VQ (IVQ) [7] which could obtain good encoded image at low bit-rate.

Many approximation methods were proposed in the past. Among these methods, mean-filled and the bilinear interpolation [8-9] are two common methods which are simple and easy to be implemented. The B-spline function is also used to interpolate images [10-12]. Besides that, as traditional approximation methods suffer from the zigzagging artifacts, some methods [13-17] were proposed to get rid of zigzagging effects. These methods use orientation of object contours or edges in the source image to direct their interpolations. Since human's eyes are sensitive to edges or contours, the directional interpolation methods usually yield better interpolation results than the omni-directional bi-cubic and bi-linear interpolation methods.

In this paper, we propose a new algorithm to obtain the approximated image. The proposed Mean-Variance-Preserving (MVP) approximation algorithm divides an image into various blocks and it calculates the color of the top-left pixel of every block in the approximated image first. The calculated colors are then used to interpolate the remaining pixels of the approximated image. The color of the top-left pixel in every block is calculated by requiring the average color and color contrast in the approximated image are the same as those ones in the original image. Using this constrain, we develop analytical formulas to obtain the approximated image.

The organization of this paper is as follows. The proposed MVP approximation algorithm is introduced in Section 2. Experimental results are shown in Section 3. The conclusion is presented in Section 4.

2. Mean-Variance-Preserving (MVP) Image Approximation

Before approximating the images, we first transform the color space from RGB to YIQ. Then, we apply our approximation algorithm, separately, to the Y, I, and Q images. By combining the approximated Y, I, Q images and transforming from YIQ to RGB, we can generate our approximated image. Let S denote Y, I, or Q; and $\hat{S}$ the approximation of S. We first divided S into $G \times G$ blocks (the block size is $W \times W$). Let $f_{i,j}(1,1)$ denote the intensity value of the top-left pixel of block *(i, j)* in S ($1 \le (i,\ j) \le G$); and $\hat{f}_{i,j}(1,1)$ the corresponding intensity value of $\hat{S}$. The MVP algorithm first calculates $\hat{f}_{i,j}(1,1)$, $1 \le i,\ j \le G$, for $\hat{S}$, and then uses the calculated $\{\hat{f}_{i,j}(1,1)\}_{1\le i,j\le G}$ and the linear interpolation to fill in the intensity values of the remaining pixels of $\hat{S}$. For example, if $\hat{f}_{i,j}(1,1)$ and $\hat{f}_{i,j+1}(1,1)$ are obtained, then $\hat{f}_{i,j}(1,x),\ x = 2,\ldots,W$, can be generated using the line segment connecting $\hat{f}_{i,j}(1,1)$ and $\hat{f}_{i,j+1}(1,1)$; that is, $\hat{f}_{i,j}(1,x)$ can be calculated by

$$\hat{f}_{i,j}(1,x) = \frac{\hat{f}_{i,j+1}(1,1) - \hat{f}_{i,j}(1,1)}{(W+1)-1}(x-1) + \hat{f}_{i,j}(1,1). \tag{1}$$

All other intensity values in $\hat{S}$ can be filled in a similar way.

We summarize below the overall process of the MVP algorithm:

Step 1. Calculate $\hat{f}_{i,j}(1,1)$, $1 \le i,\ j \le G$, by requiring that the { $\hat{f}_{i,j}(1,1)$, ..., $\hat{f}_{i,j}(1,W)$; $\hat{f}_{i,j+1}(1,1)$ } in $\hat{S}$ preserve the mean m and variance σ^2 of the { $f_{i,j}(1,1)$, ..., $f_{i,j}(1,W)$; $f_{i,j+1}(1,1)$ } in S.

Step 2. Obtain $\hat{f}_{i,j}(1,x)$ and $\hat{f}_{i,j}(x,1)$, where $x \in \{2,\dots,W\}$ by linear interpolation, i.e., using the line segment connecting $\hat{f}_{i,j}(1,1)$ and $\hat{f}_{i,j+1}(1,1)$, and using the line segment connecting $\hat{f}_{i,j}(1,1)$ and $\hat{f}_{i+1,j}(1,1)$.

Step 3. Calculate $\hat{f}_{i,j}(r,s)$, $2 \le r, s \le W$, using the line segment connecting $\hat{f}_{i,j}(1,s)$ and $\hat{f}_{i+1,j}(1,s)$.

The key point of the MVP algorithm is to generate $\hat{f}_{i,j}(1,1)$, $1 \le i,\ j \le G$. We show below how to get them. Let $\hat{f}(x) = \hat{f}_{i,j}(1,x),\ x = 1,\dots,W$, and $\hat{f}(W+1) = \hat{f}_{i,j+1}(1,1)$. Equation (1) can be rewritten as

$$\hat{f}(x) = \frac{\hat{f}(W+1) - \hat{f}(1)}{(W+1)-1}(x-1) + \hat{f}(1) \cdot \tag{2}$$

To calculate $\hat{f}(1)$ and $\hat{f}(W+1)$ in Step 1, the MVP algorithm requires

$$\frac{1}{W+1}\sum_{x=1}^{W+1}\hat{f}(x) = \hat{m} = m ,$$

$$\frac{1}{W+1}\sum_{x=1}^{W+1}(\hat{f}(x) - \hat{m})^2 = \sigma^2.$$

Note that $\hat{f}(1), \hat{f}(2), \dots, \hat{f}(W+1)$ is an arithmetic sequence $(\hat{f}(1) - \hat{f}(2) = \hat{f}(2) - \hat{f}(3) = \cdots)$ because of Equation (2). Hence, $\hat{m} = (\hat{f}(1) + \hat{f}(W+1))/2$. Therefore, we only have to solve

$$\frac{\hat{f}(1) + \hat{f}(W+1)}{2} = m, \tag{3}$$

$$\frac{1}{W+1}\sum_{x=1}^{W+1}(\hat{f}(x) - \frac{\hat{f}(1) + \hat{f}(W+1)}{2})^2 = \sigma^2. \tag{4}$$

After certain derivation (presented in the next paragraph), we obtain the formulas to calculate $\hat{f}(1)$ and $\hat{f}(W+1)$, namely,

$$\hat{f}(1) = 2m - \hat{f}(W+1), \tag{5}$$

$$\hat{f}(W+1) = m \pm \sqrt{\frac{3W}{W+2}\sigma^2}. \tag{6}$$

(The way to determine the sign of $\hat{f}(W+1)$ will also be presented in the next paragraph.) Using the formulas, we can obtain $\hat{f}_{i,j}(1,1)$, $1 \le i,\ j \le G$, in Step 1. The remaining intensity values in $\hat{S}$ can then be filled with linear interpolation (Steps 2 and 3).

In the following, we present the detail derivation about how we derive Equations (5) and (6) from Equations (3) and (4). Equation (3) implies

$$\hat{f}(1) = 2m - \hat{f}(W+1), \tag{7}$$

i.e.,

$$\hat{f}(W+1) - \hat{f}(1) = 2(\hat{f}(W+1) - m). \tag{8}$$

Substituting Equation (2) in (4), we have

$$\sigma^2 = \frac{1}{W+1}\sum_{x=1}^{W+1}T_x^2, \tag{9}$$

where

$$T = \frac{\hat{f}(W+1) - \hat{f}(1)}{W}(x-1) - \frac{\hat{f}(W+1) - \hat{f}(1)}{2}.$$

Let $\alpha = \hat{f}(W+1) - m$, substitute Equation (8) in (9), we have,

$$\begin{aligned}
\sigma^2 &= \frac{1}{W+1}\sum_{x=1}^{W+1}(\frac{4\alpha^2}{W^2}(x-1)^2 - \frac{4\alpha^2}{W}(x-1) + \alpha^2) \\
&= (\frac{4\alpha^2}{(W+1)W^2}\sum_{x=1}^{W+1}(x-1)^2) - (\frac{4\alpha^2}{(W+1)W}\sum_{x=1}^{W+1}(x-1)) + (\frac{1}{W+1}\sum_{x=1}^{W+1}\alpha^2) \\
&= \frac{4\alpha^2}{(W+1)W^2}\sum_{x=1}^{W}x^2 - (\frac{4\alpha^2}{(W+1)W}\sum_{x=1}^{W}x) + \alpha^2 \\
&= \frac{4\alpha^2}{(W+1)W^2}\frac{W(W+1)(2W+1)}{6} - \frac{4\alpha^2}{(W+1)W}\frac{W(W+1)}{2} + \alpha^2 \\
&= \frac{(W+2)}{3W}\alpha^2.
\end{aligned}$$

That is,

$$\hat{f}^2(W+1) - 2m\hat{f}(W+1) + m^2 = \frac{3W}{(W+2)}\sigma^2$$

by the definition of α . Therefore,

$$\begin{aligned}
\hat{f}(W+1) &= \frac{2m \pm \sqrt{4m^2 - 4(m^2 - \frac{3W}{W+2}\sigma^2)}}{2} \\
&= m \pm \sqrt{\frac{3W}{W+2}\sigma^2}.
\end{aligned} \tag{10}$$

To determine the sign of Equation (10), we can examine the sign of

$$f(W+1) - f(1) \cdot \tag{11}$$

If the sign of Equation (11) is positive, we choose a plus sign for Equation (10); otherwise, we choose a minus sign for Equation (10).

3. Experiment Results

Four experiments were done for the proposed MVP approximation algorithm. The images that we used are Lena, Jet, Peppers, and Tiffany. The image size is 8×8. The approximated results of the B-spline, mean-filled, and bilinear methods are also included in this paper for comparisons. Figure 1 shows the original, and the approximated Lena of different methods (the block sizes used for the Y, I, Q images are all 8×8). Table 1 lists the APSNR of the approximated Lena using different block sizes. Table 2 lists the results of other images using block size 8×8 for Y, and 16×16

for both I and Q. The APSNR measurement used here were calculated by

$$\text{APSNR} = 10 \times \log \frac{3 \times 255^2}{\text{MSE}},$$

where MSE denotes the Mean Square Errors (MSE) of the approximated images. Figure 1 shows that our MVP method gives better approximation than the B-spline, mean-filled, and bilinear methods. Tables 1 and 2 show this fact in quantity values.

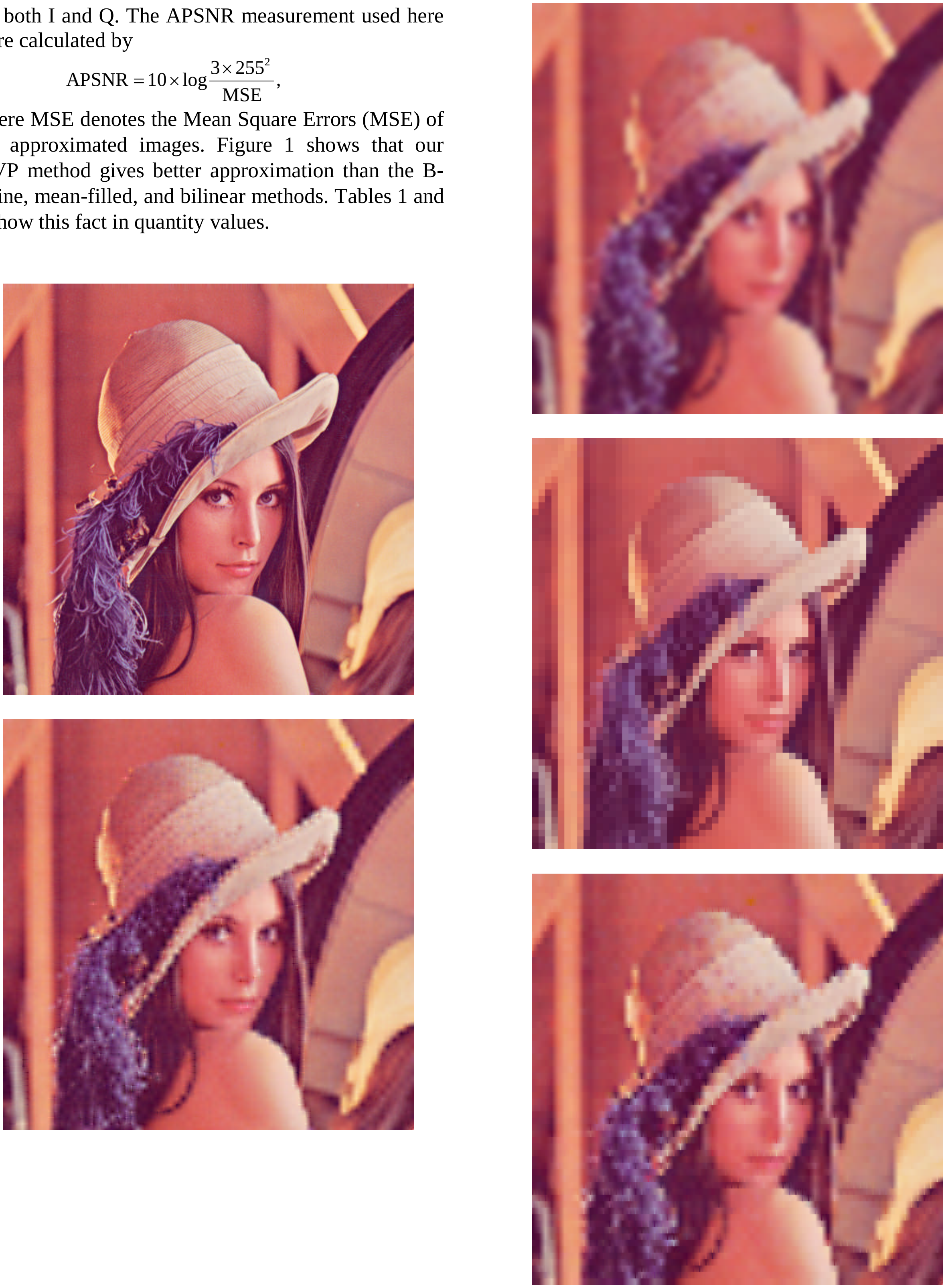

Fig. 1: (Left to Right, Top to Bottom) The source image Lena and the approximated images obtained by MVP, B-spline, mean-filled, and bilinear methods.

TABLE 1
The APSNR of the approximated image Lena of different methods. The term (4; 8; 8) under (Y, I, Q) indicates that the block size is 4×4 for Y and 8×8 for both I and Q.

(Y, I, Q)	our MVP method	B-spline method	mean-filled method	bilinear method
(4, 4, 4)	29.70	26.90	26.93	26.47
(4, 8, 8)	29.14	26.62	26.49	26.01
(4, 16, 16)	28.29	25.96	25.90	25.33
(8, 8, 8)	25.90	23.63	23.88	23.35

TABLE 2
The APSNR of the approximated images obtained by different methods. The block sizes are 8×8 for Y and 16×16 for I and Q.

(Y, I, Q) is (8, 16, 16)	our MVP method	B-spline method	mean-filled method	bilinear method
Lena	25.49	23.38	23.54	22.95
Jet	23.14	20.71	21.92	19.62
Peppers	23.19	20.67	21.57	19.42
Tiffany	25.05	22.65	24.82	22.38

4. Conclusion

In this paper, a MVP color image approximation method is proposed. Before approximating a color image, the method first translates the image from RGB color space to YIQ; then, it separately, applies approximation to the Y, I, and Q images; by combining the approximated Y, I, Q images and transforming from YIQ color space back to RGB, the proposed method generates the approximated color images. The experimental results show that the approximated images generated by the MVP method are superior to that generated by the B-spline, mean-filled, and bilinear methods.

5. Acknowledges

This work is supported by the National Science Council.

6. References

[1] R. M. Gray, "Vector quantization," *IEEE ASSP Mag.*, pp. 4-29, 1984.

[2] N. M. Nasrabadi and R A King, "Imaging coding using vector quantization: a review," *IEEE Trans. Commun.*, Vol. 36, pp. 957-971, 1988.

[3] T. Murakami, K. Asai, and E. Yamazaki, "Vector quantizer of video signals," *Electronic Letters*, pp. 1005-1006, 1982.

[4] A. Gersho and R. M. Gray, "Vector quantization and signal compression," Boston Kluwer Academic Publisher, 1992.

[5] R. L. Baker, Vector quantization of digital images, Ph.D. dissertation, Stanford Univ., Stanford, CA, June 1984.

[6] C. M. Huang and R. W. Harris, "A comparison of several vector quantization codebook generation approaches," *IEEE Trans. Image Processing*, Vol. 2, No. 1, pp. 108-112, 1993.

[7] H. M. Hang and B. G. Haskell, "Interpolative vector quantization of color images," *IEEE Trans. Communication.*, Vol. 36, No. 4, pp. 465-470, 1988.

[8] A. K. Jain, Fundamentals of Digital Image Processing, Prentice-Hall, Englewood cliffs, NJ, 1989.

[9] G. Farin, Curvesand Surfaces for Computer Aided Geometric Design, Academic Press, San Diego, CA, 1990.

[10] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing: Part I - Theory," *IEEE Tran. Signal Processing*, Vol. 41, No. 2, pp. 821-833, 1993.

[11] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing: Part II - efficient design and applications," *IEEE Tran. Signal Processing*, Vol. 41, No. 2, pp. 834-848, 1993.

[12] M. Unser, A. Aldroubi, M. Eden, "Fast B-spline transforms for continuous image representation and interpolation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 13, No. 3, pp. 277-285, 1991.

[13] Q. Wang and R. Ward, "A contour-preserving image interpolation method," *Proceeding of the International Conference on Image Processing*, Vol. 3, pp. 14-17, 2003.

[14] L. Demaret and A. Iske, "Adaptive image approximation by linear splines over locally optimal delaunay triangulations," *IEEE Trans. Signal Processing Letters*, Vol. 13, Issue 5, pp. 281-284, 2006.

[15] S. E. EI-Khamy, M. M. Hadhoud, M.I. Dessouky, B. M. Salam, and F.E.A. EI-Samie, "A new edge preserving pixel-by-pixel (PBP) cubic image interpolation approach," *Proceedings of the Twenty-First National Radio Science Conference*, pp. C11-1-C11-9, 2004.

[16] N. Toronto, D. Ventura, and B. S. Morse, "Edge inference for image interpolation," *IEEE International Joint Conference on Neural Networks*, Vol. 3, pp. 1782-1787, 2005.

[17] Y. Tian, B. Liu, and T. Li, "A local image interpolation method based on gradient analysis," *Proceedings of the International Conference on Neural Networks and Brain,* Vol. 2, pp. 1202-1205, 2005.