AN EFFECTIVE COLOR QUANTIZATION METHOD USING COLOR IMPORTANCE-BASED SELF-ORGANIZING MAPS

Hyun Jun Park, Kwang Baek Kim*†, Eui Young Cha**

Abstract: Color quantization is an important process for image processing and various applications. Up to now, many color quantization methods have been proposed. The self-organizing maps (SOM) method is one of the most effective color quantization methods, which gives excellent color quantization results. However, it is slow, so it is not suitable for real-time applications. In this paper, we present a color importance–based SOM color quantization method. The proposed method dynamically adjusts the learning rate and the radius of the neighborhood using color importance. This makes the proposed method faster than the conventional SOM-based color quantization method. We compare the proposed method to 10 well-known color quantization methods to evaluate performance. The methods are compared by measuring mean absolute error (MAE), mean square error (MSE), and processing time. The experimental results show that the proposed method is effective and excellent for color quantization. Not only does the proposed method provide the best results compared to the other methods, but it uses only 67.18% of the processing time of the conventional SOM method.

Key words: *SOM, color quantization, image processing*

Received: August 21, 2013 **DOI:** 10.14311/NNW.2015.25.006
Revised and accepted: March 30, 2015

1. Introduction

The purpose of color quantization is to represent the many colors in the original image with a reduced number of distinct colors and with minimal distortion. True-color images contain thousands of colors and can contain up to 16,777,216 colors. More colors representing an image can make a better output to look at.

However, more colors can be a problem for most image-processing applications. For example, colors can be used for object detection, object extraction, and to compare features. In image-processing, a single object represented with one color is an ideal case, but unfortunately, even if it is a single object, it is represented with many colors, which becomes a serious problem.

*Hyun Jun Park, Eui Young Cha – Corresponding author, Pusan National University, Busan, South Korea, E-mail: `hyunjun@pusan.ac.kr`, `eycha@pusan.ac.kr`

†Kwang Baek Kim, Silla University, Busan, South Korea, E-mail: `gbkim@silla.ac.kr`

Therefore, image-processing applications such as text detection [36], compression [43], segmentation [13], content-based searches [14], watermarks [25], and color-texture analysis [35] perform color quantization as a preprocessing step to reduce the number of colors.

Color quantization consists of palette design and pixel-mapping phases. The palette design phase is the selection of colors that represent the original colors, but with minimal distortion. The pixel-mapping phase is the assignment of each pixel in the original image to one of the colors in the designed palette. Color quantization methods perform a clustering process to design the palette by using one of the clustering algorithms, and perform pixel-mapping with the designed palette. Therefore, the degree of distortion is determined by the clustering algorithm that is used for palette design.

A self-organizing maps (SOM)-based color quantization method is one of the most effective methods. It shows natural output with little distortion. However, SOM is composed of two layers (the input layer and the competitive layer), fully connected. Due to the structure and many colors in an image used for SOM learning, many repetitive computations occur, and it takes too much computation time.

Recently, MFD-SOM method was proposed [10]. It uses a dynamic learning rate and neighborhood radius. User-defined constant values tune the learning rate and neighborhood radius to determine a winner. Also, a new way to update weight vectors is proposed. However MFD-SOM still needs a lot of time for color quantization.

Therefore, we propose a new color quantization method using SOM, which we call color importance–based SOM. The proposed method maintains the results of the conventional SOM-based color quantization method but is faster.

The proposed method uses sampled data for SOM learning, because the SOM learning result can be changed by the sequence of training data, and to minimize repeated learning with similar colors. Also, the proposed method defines the color importance, and uses it for learning. The color importance dynamically adjusts the learning rate and neighborhood radius. For example, if the color importance is high then the learning rate and neighborhood radius are increased. These mechanisms make SOM learning faster than conventional SOM learning.

This paper is organized as follows. Section 2 explains the existing color quantization methods. Section 3 describes the color quantization method using color importance–based SOM. Section 4 evaluates the performance of the proposed method using publicly available images and compares the proposed method to other well-known methods. Finally, Section 5 presents the conclusions.

2. Related works

Color quantization methods are classified by whether the distribution of colors in the image is used or not. Image-independent methods generate a palette that is unconcerned with color distribution [2]. Therefore, these are fast but give poor results.

By contrast, image-dependent methods generate a palette using color distribution. These are slower than image-independent methods but give better results.

Designing the palette in image-dependent methods is equivalent to the clustering of colors in an image. Therefore image-dependent methods can be classified into two categories: hierarchical clustering and partitional clustering [2].

Hierarchical clustering methods perform color quantization based on a statistical analysis of the color distribution. There are two approaches to hierarchical clustering. One is the divisive (top-down) approach, which repeatedly subdivides the initial cluster until K clusters are obtained. The other is the agglomerative (bottom-up) approach. It starts with N clusters and repeatedly merges the clusters until K clusters are obtained [2].

Partitional clustering methods typically know the expected number of clusters. They calculate all the clusters at each iteration, and repeatedly update the clusters to reduce the differences in the original image.

In other words, hierarchical clustering methods calculate the palette once, but partitional clustering methods calculate the palette and repeatedly update it to minimize distortion of the original image. Therefore, partitional clustering methods give higher quality results, but they need much more computation time.

Tab. I shows various color quantization methods mentioned above.

Image-independent methods		• Unconcerned with color distribution • Fast, but poor results
Image-dependent methods	**Hierarchical clustering:** median-cut [18], octree [16], greedy orthogonal bipartioning [40], variance-based method [38], binary splitting [27], center cut [26], RWM cut [44]	• Divide (or merge) initial clusters until K clusters are obtained • Faster than partitional clustering methods
	Partitional clustering k-means [19–22], weighted sort-means [7], fuzzy C-means [3,23,28,33,39], self-organizing maps [9, 11, 12, 29,30,42], maxmin [17,41], k-harmonic means [15], competitive learning [4,6,8,34, 37], rough C-means [31], BIRCH [1]	• Update K clusters repeatedly • Good results, but slower than hierarchical clustering methods

Tab. I *Various color quantization methods.*

Popularity (POP) [18] POP is one of the simplest methods. First, build a $16 \times 16 \times 16$ color histogram using four bits per channel uniform quantization. The palette color comprises the K most-frequent colors in the color histogram. This method is fast, but gives poor results.

Octree (OCT) [16] Octree is a tree structure with up to eight nodes as children. Because the colors are represented with 8 bits, the octree can represent all colors in an image within an eight-level tree. At first, color distribution in the image is represented using octree, which then prunes the nodes until K nodes remain. The palette colors are chosen from the remaining K nodes. This method is fast and gives good results.

MedianCut (MC) [18] MC starts by building a $32 \times 32 \times 32$ color histogram using five bits per channel uniform quantization. It makes cubes that include all of the histogram and then repeatedly splits the cubes that have the greatest number of colors until K cubes are obtained. The palette colors are chosen from the centroids of the K cubes.

Greedy Orthogonal Bipartitioning (GOB) [40] This method is similar to MC but uses the greatest sum of squared error to minimize the sum of the variances on both sides. The palette colors are again chosen from the centroids.

Adaptive Distributing Unit (ADU) [4] ADU quantizes the colors using Uchiyama and Arbib's clustering algorithm. It starts with a cluster, which is assigned as the centroid to the mean of all input data. Each cluster is split when the amount of data with a minimum distance is above a certain threshold. It continues splitting until K clusters are obtained. The palette is chosen from the centroids of the final clusters.

***k*-means (KM) [19–22]** k-means clustering is a well-known clustering method. It starts with K random clusters. In each iteration, all of the input data are assigned to the cluster that has the minimum distance within the data. The centroid of the cluster is calculated as the average of the assigned data, and it is repeated until the centroid of the cluster does not change. The palette colors are chosen from the centroids of the final clusters. In this paper, the k-means algorithm described by Hu and Lee [19] is used.

Weighted Sort-Means (WSM) [7] WSM is an adaptation of the conventional k-means clustering algorithm for color quantization. This method performs the data sampling step and sample weighting step, and uses the sort-means algorithm to reduce computation time.

Fuzzy *C*-means (FCM) [3,23,28,33,39] FCM starts with K clusters. In fuzzy clustering, each datum has a degree of belonging (membership) to the clusters. FCM calculates the centroid of the clusters using the degree of belonging and repeats the calculations until the algorithm has converged. If the image has a large amount of data, calculating the degrees of belonging takes a lot of time. Therefore, this method is very slow. In this paper, the FCM algorithm described by Kim et al. [23] is used.

Adaptive Resonance Theory 2 (ART2) [24] This method is an unsupervised learning model and starts with a cluster. ART2 creates new clusters based on a vigilance test. If the result of the vigilance test is larger than the vigilance parameter, then ART2 creates a new cluster or assigns the data to a cluster. The centroid of the clusters is defined as the average of the assigned data, and ART2 continues testing until the centroid of the clusters converge. The palette colors are chosen from the centroids of the K most-frequent clusters.

Self-Organizing Maps (SOM) [9, 11, 12, 29, 30, 42] SOM is also an unsupervised learning model and uses a one-dimensional self-organizing map with K neurons. It designates the minimum distance node as the winner node, and then updates the weights of the winner node and neighbor nodes. It repeats the process until the sum of the weight change is less than a certain threshold. The palette color is chosen from the final weights. In this paper, the SOM algorithm described by Dekker [12] is used.

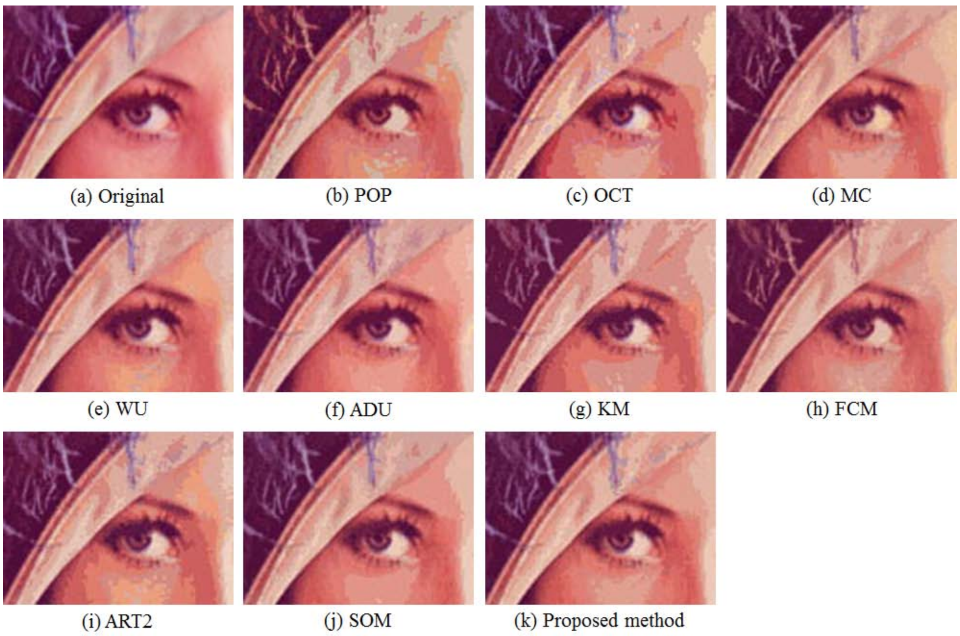

Fig. 1 *Lenna output images.* ($K = 32$).

3. Color quantization using color importance-based SOM

We need to consider the features of color distribution in natural images for efficient color quantization. Initially, one region in an image has similar colors. In other words, adjacent pixels in an image have similar colors. This means that SOM's learning of sequential pixels is equivalent to the same operation being repeated. Therefore, more efficient color quantization can be performed by reducing these repetitive operations. Second, high-frequency colors in an image should be assigned to many of the colors in the palette in order to reduce distortion of the original image. This gives a more natural quantization result.

We present an efficient color quantization method using color importance-based SOM by using the above two features. The proposed method is faster but maintains the performance of conventional SOM color quantization.

3.1 Training data sampling

SOM learning using all the pixels in an image requires a lot of processing time. As mentioned above, SOM learning using sequential pixels means similar pixels are learned repeatedly. Therefore, the proposed method for SOM uses a subset of the pixels in the image. The proposed method collects the pixels at regular intervals as training data from one-dimensional image data vector $\mathbf{x}$. It guarantees a variety of colors are selected and eliminates the repetition. The vector of training data for t-th iteration, $\mathbf{d}_t$, is constructed as

$$\mathbf{d}_t = (x_t, x_{t+\Delta}, x_{t+2\Delta}, \dots),$$

where Δ is the collection interval length. In this paper, the interval is set to 100, which means that 1 % of all the pixels are used in each iteration. The interval length also sets an upper bound on the number of algorithm iterations, as after Δ iterations the training data starts to repeat itself, but it usually does not need that much iterations.

3.2 Color importance

When conventional SOM learns the sampled training data, similar colors tend to determine the same neuron as a winner, and it updates the weights using the same learning rate and neighborhood. This raises an issue. Because every color has the same color importance, the colors are assigned to the palette regardless of the actual frequency of the colors in the original image. The learning rate and neighborhood radius should be changed by color importance.

The proposed method defines color importance based on color frequency to solve this problem. The SOM learning rate and the radius of the neighborhood are adjusted by color importance. This adjustment speeds up the results of SOM learning. At first, the proposed method builds the color distribution to define color importance. It uses a $32 \times 32 \times 32$ color histogram using 5 bits per channel uniform quantization. Color importance is defined by the frequency of the colors in the color histogram.

3.3 The color importance-based SOM learning algorithm

The learning algorithm for color importance-based SOM is similar to conventional SOM with two exceptions: how to set the learning rate α, and the radius of the neighborhood γ [9]. The learning algorithm is described below.

Algorithm 1 Learning algorithm of color importance-based SOM

Require: image $\mathbf{x} = (x_1, x_2, \ldots, x_N)$ as a vector of N pixels, interval Δ, number of neurons K
Ensure: weight $\mathbf{w} = (w_1, w_2, \ldots, w_K)$
  Build a $32 \times 32 \times 32$ color histogram using 5 bits per channel uniform quantization
  $\mathbf{c} \leftarrow (c_1, c_2, \ldots, c_L)$, where L is the number of colors in the histogram
  $\mathbf{i} \leftarrow (i_1, i_2, \ldots, i_L)$, such that $i_j = \sqrt{\text{frequency of } c_j/\text{maximum frequency}}$
  $\mathbf{w} \leftarrow K$ random values
  $t \leftarrow 1$
  repeat
    {process all training pixels in sequence}
    $\mathbf{d}_t \leftarrow (x_t, x_{t+\Delta}, x_{t+2\Delta}, \ldots)$ {generate subset of training pixels}
    $N' \leftarrow \|\mathbf{d}_t\|$ {number of training data}
    for $j \leftarrow 1$ **to** N' **do**
      $\text{winner} \leftarrow \underset{k=1,\ldots,K}{\operatorname{argmin}} \|d_{t,j} - w_k\|^2$ {determine the winner (nearest) neurons}
      $\gamma \leftarrow \frac{K}{2}\sqrt{i_{\text{winner}}}$ {calculate the update radius}
      $\alpha \leftarrow \mathrm{e}^{-0.25t}\sqrt{i_{\text{winner}} + 0.25}$ {calculate the learning rate}
      {update the winner's weight and its neighbor's}
      for $k = \text{winner} - \gamma$ **to** $\text{winner} + \gamma$ **do**
        $w_k \leftarrow w_k + \frac{\alpha\,(d_{t,j} - w_k)}{1 + \|k - \text{winner}\|^2}$
      end for
      Reduce the importance of i_{winner}
    end for
    $t \leftarrow t + 1$
  until weights converged

At first, the proposed method initializes the weights with random values. The weights are updated until they converge. We assumed the weights are converged when an average of weight variation is less than 0.01.

To update the weights, the proposed method finds the winner node with minimum distance, and then the weights are updated using the input values and the learning rate.

The learning rate α is defined by color importance of the winner, i_{winner}, and can be computed as

$$\alpha = \mathrm{e}^{-0.25t}\sqrt{i_{\text{winner}} + 0.25}.$$

It results in the weights converging faster by increasing the high-importance color's learning rate. The color importance is defined experimentally. The constants in the definition are set to 0.25. It adjusts the reduced rate for the learning rate and 0.25 shows good to color quantization.

The weights are updated by following formula.

$$w_k = w_k + \frac{\alpha\,(d_{t,j} - w_k)}{1 + \|k - \text{winner}\|^2}.$$

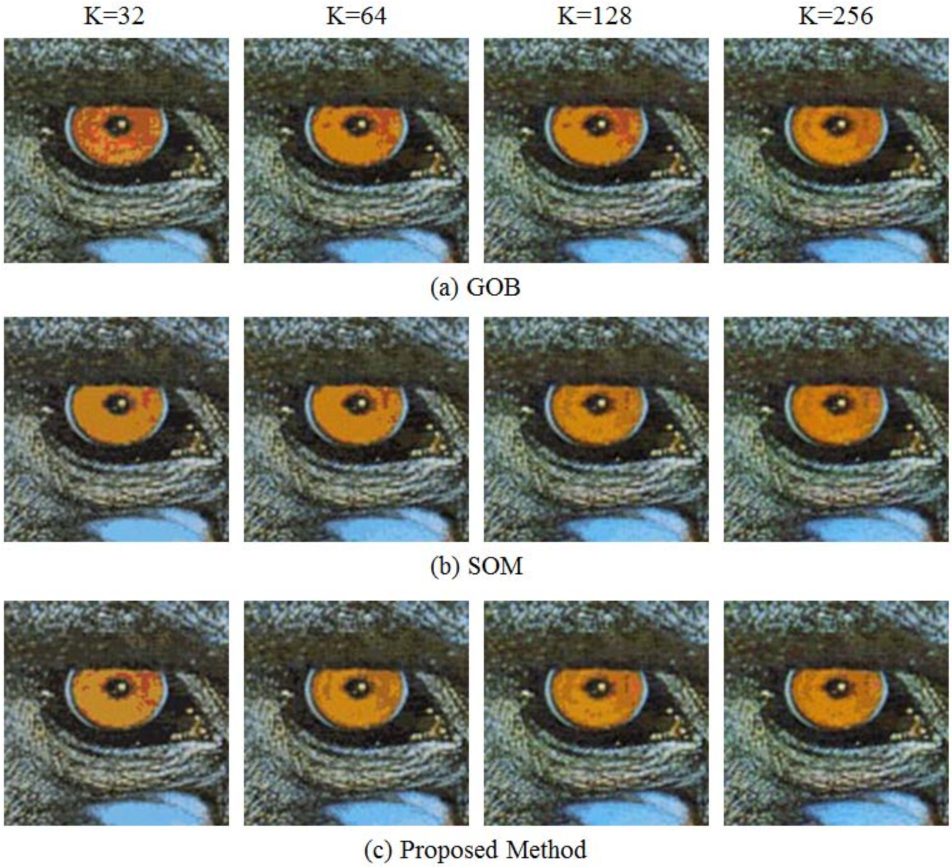

Fig. 2 *Mandrill output images.*

where k denotes a target node and $d_{t,j}$ is a j-th element in training data d_t. It updates the weights depend on the learning rate α and distance between winner and target node. The distance and updating value are inversely related.

When the weights of the winner are updated, the weights of the winner's neighbors are also updated. The neighboring nodes are located within the radius of neighborhood γ, which is given as

$$\gamma = \frac{K}{2}\sqrt{i_{\text{winner}}}.$$

This results in the high-importance colors being assigned to the palette by increasing the high-importance color's neighborhood radius.

Algorithm 1 shows the pseudo code of the learning algorithm of the color importance-based SOM.

Image	Source	Resolution	Colors
Airplane	USC-SIPI Image Database	512 × 512	77,041
Lenna	USC-SIPI Image Database	512 × 512	148,279
Mandrill	USC-SIPI Image Database	512 × 512	230,427
Peppers	USC-SIPI Image Database	512 × 512	183,525
Girl	Kodak Lossless True Color Image Suite	768 × 512	44,576
Hats	Kodak Lossless True Color Image Suite	768 × 512	34,871
Motocross	Kodak Lossless True Color Image Suite	768 × 512	63,558
Parrots	Kodak Lossless True Color Image Suite	768 × 512	72,079

Tab. II *Information of the test images.*

4. Experimental results

To evaluate performance, the proposed method was tested on a set of eight true-color images commonly used in color quantization papers. Images are shown in Fig. 3 and information of the test images are described in Tab. II. All of the color quantization methods were tested on an Intel i7-2640M 2.8 GHz, 8GB machine and were implemented in C++.

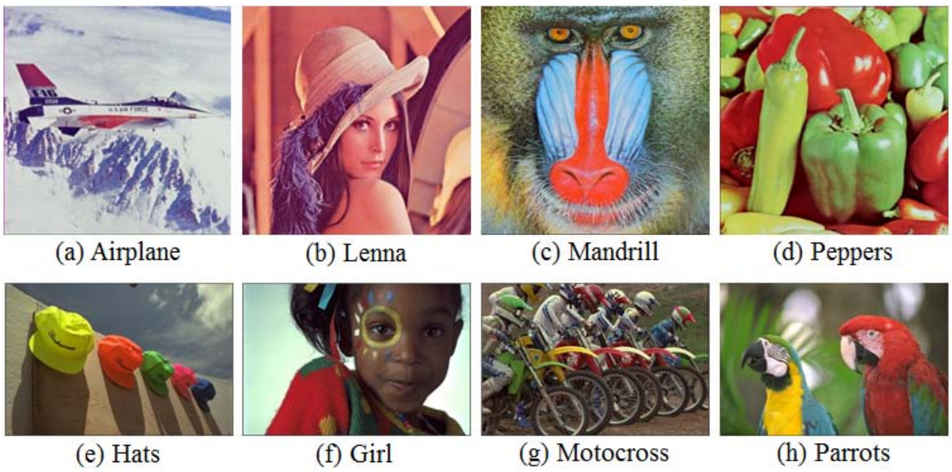

Fig. 3 *Test images.*

Fig. 1 shows the color quantized results of the Lenna image which is one of the test images. Compared with other methods, the proposed method generates less aliasing in the face and hat. Also, the area around the feathers of the hat is very similar to the original image.

The performance of the color quantization result was quantified by mean absolute error (MAE) and mean square error (MSE),

$$\mathrm{MAE}\left(X,\hat{X}\right)=\frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W}\left\|\mathbf{X}\left(h,w\right)-\hat{\mathbf{X}}\left(h,w\right)\right\|_{1}$$

$$\mathrm{MSE}\left(X,\hat{X}\right)=\frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W}\left\|\mathbf{X}\left(h,w\right)-\hat{\mathbf{X}}\left(h,w\right)\right\|_{2}^{2}$$

where H and W denote image height and width, respectively,

MAE measures the average magnitude of the errors in the same units as the data. This means that MAE represents the difference between the original image and the quantized image. MSE is a quadratic scoring rule that measures the average magnitude of the error. MSE is more sensitive than MAE to the occasional large error: the squaring process gives disproportionate weight to large errors.

Tabs. III and IV show the average MAE and MSE of five experimental results for other well-known color quantization methods. The top two methods are indicated in bold. Tab. V shows the average processing time of five experimental results. In this paper, for a more accurate comparison, processing time does not include time for the pixel-mapping phase, because pixel mapping is required regardless of the color quantization algorithm. Therefore, processing time is measured as CPU time for the palette generation phase only.

As expected, the hierarchical clustering methods (POP, MC, OCT, GOB) are faster, but generally, color quantization results are poor. POP, the simplest algorithm, is fastest, but quantization results are also the poorest. MC requires similar, or more, processing time than POP, and its results are better. OCT needs from 2.3 to 3.1 times more processing time than POP. Experimental results show GOB is one of the most effective methods. GOB gives the smallest distortion among the hierarchical clustering methods. It takes linear time and is also fast.

On the other hand, the partitional clustering methods (KM, FCM, ADU, ART2, SOM, and PM [the Proposed Method]) are slow, but the results are better. As stated above, the hierarchical clustering methods start with an initial N clusters and repeatedly merge or subdivide until K colors are obtained. By contrast, the partitional clustering methods calculate all the clusters in each iteration, and repeatedly update the clusters. Therefore, they typically take more processing time, but the generated palette tends to contain a greater variety of colors than a palette generated by the hierarchical clustering methods. If there are many similar colors in the palette, it generates the results without a variety of colors and makes them unnatural. Therefore, as shown in Fig. 2, even if MAE and MSE are similar (K = 128, MAE = GOB 17.0; SOM 16.7; PM 16.9, MSE = GOB 153.2; SOM 149.1; PM 149.7), the partitional clustering methods give more natural results than the hierarchical clustering methods. Nevertheless, KM, ADU, FCM, and ART2 require too much processing time, so they are inappropriate for use in applications.

However, not only does the proposed method provide the best results from among the color quantization methods, but it also only needs the processing time that is available in real-time applications. That means the proposed method makes better quality results than the conventional SOM method, and minimizes the distortion between the original image and the quantized image. Plus, it gives more

	Airplane				Girl				Hats				Lenna			
K	32	64	128	256	32	64	128	256	32	64	128	256	32	64	128	256
POP	19.2	14.9	12.6	12.1	23.4	19.1	15.7	14.5	33.6	19.6	14.1	12.3	22.0	16.1	12.7	11.9
MC	14.3	10.6	8.9	7.4	21.1	15.6	12.6	9.8	27.3	19.7	15.2	9.7	17.4	13.1	10.1	8.3
OCT	19.3	14.1	11.3	7.7	29.0	25.3	14.1	12.4	28.6	16.9	13.0	9.6	23.3	15.5	12.9	9.6
GOB	**9.9**	**7.8**	**6.3**	5.4	22.5	17.1	13.5	11.9	**16.4**	**11.2**	8.0	5.9	14.8	11.4	9.0	7.3
ADU	15.3	9.0	6.6	4.8	**14.7**	**11.0**	**8.4**	**6.5**	29.1	14.5	11.8	6.5	15.5	12.0	9.8	7.5
KM	15.0	12.7	9.4	8.8	20.0	13.1	10.1	6.8	23.1	16.1	12.5	10.7	20.9	16.9	14.8	11.9
FCM	14.2	9.5	7.4	8.5	21.6	21.7	14.8	14.3	20.5	13.7	11.5	8.3	19.2	14.9	13.7	10.2
ART2	16.1	10.5	7.5	6.2	20.9	14.5	11.5	8.5	21.0	15.4	11.0	8.9	18.5	12.4	10.3	8.5
SOM	10.1	**7.6**	**6.0**	**4.7**	16.1	**10.9**	**8.0**	**6.1**	18.3	11.6	**8.0**	**5.5**	**14.7**	**11.3**	**8.7**	**7.0**
PM	**10.0**	8.3	6.4	**5.0**	**14.4**	11.4	8.6	6.9	**15.9**	**10.9**	**7.8**	**5.7**	**14.1**	**11.0**	**8.6**	**7.0**
	Mandrill				Motocross				Parrot				Peppers			
K	32	64	128	256	32	64	128	256	32	64	128	256	32	64	128	256
POP	40.9	31.0	21.1	15.7	33.2	19.3	15.9	13.2	58.8	22.5	16.5	13.6	35.3	21.4	16.0	13.3
MC	59.0	46.2	33.8	21.6	21.9	17.8	14.5	11.0	36.9	25.1	17.8	12.8	24.8	21.1	17.0	13.1
OCT	39.0	27.2	22.0	17.9	31.8	23.4	16.8	13.5	28.6	20.3	16.3	12.6	32.6	22.8	17.7	15.1
GOB	**26.8**	21.3	17.0	13.7	**18.9**	14.1	10.8	8.5	**21.1**	**15.4**	11.7	**9.0**	**20.1**	**15.5**	**12.2**	**9.6**
ADU	28.5	22.4	17.8	13.7	21.3	14.7	11.3	8.7	36.2	19.0	15.4	10.1	21.8	17.0	13.5	10.1
KM	28.7	23.2	19.7	16.7	22.4	19.0	16.2	13.4	25.8	21.0	16.0	13.1	25.3	20.9	16.9	14.3
FCM	30.9	25.9	20.3	16.6	22.1	21.9	17.6	13.9	27.2	21.7	16.1	12.9	24.1	19.3	15.5	13.2
ART2	29.2	24.8	20.0	16.3	23.7	19.2	14.4	11.7	25.9	18.3	14.2	11.3	23.1	18.9	14.5	12.2
SOM	**27.1**	**20.9**	**16.7**	**13.4**	19.6	**13.8**	**10.4**	**8.1**	22.8	16.0	**11.5**	**8.7**	20.9	**15.3**	**11.8**	**9.4**
PM	28.3	**20.7**	**16.9**	**13.6**	**18.8**	**13.9**	**10.6**	**8.4**	**20.4**	**15.1**	**11.6**	9.3	**19.7**	**15.5**	**12.2**	9.8

Tab. III *MAE comparison of the color quantization methods.*

	Airplane				Girl				Hats				Lenna			
K	32	64	128	256	32	64	128	256	32	64	128	256	32	64	128	256
POP	382.2	174.1	85.9	67.5	428.2	284.5	148.5	109.4	1301.1	447.7	131.0	75.4	336.1	175.1	79.8	64.8
MC	314.5	164.7	125.3	52.0	398.6	233.9	164.8	86.3	916.5	559.6	392.1	110.5	191.6	116.5	68.2	44.3
OCT	228.2	120.8	66.7	36.3	450.4	362.2	112.0	85.3	516.2	177.2	101.8	54.1	273.5	126.1	85.4	47.5
GOB	**69.4**	**45.1**	**28.6**	**19.1**	272.6	161.3	103.6	80.1	**175.7**	**82.6**	**42.9**	**24.3**	122.9	73.1	45.8	30.3
ADU	274.8	75.6	41.8	22.7	**137.5**	**80.5**	**46.7**	**28.0**	717.3	186.7	155.5	38.4	132.7	80.7	54.9	32.4
KM	126.7	90.1	56.4	47.2	341.7	112.3	71.1	32.9	324.2	158.6	93.6	67.3	230.4	154.0	120.9	76.6
FCM	151.9	82.8	48.5	59.4	372.9	398.8	168.8	191.5	392.2	180.0	133.8	69.6	253.5	144.0	131.8	69.2
ART2	140.9	64.4	35.1	23.2	239.0	116.3	74.2	40.3	234.7	129.3	67.4	42.8	183.0	82.9	56.0	37.8
SOM	94.4	48.1	30.2	19.9	156.8	**81.9**	**46.4**	**28.8**	217.9	107.8	47.6	26.0	**118.1**	**70.5**	**43.7**	**28.8**
PM	**68.6**	**45.2**	**29.4**	**19.4**	**130.5**	**81.9**	47.4	29.0	**175.8**	**87.9**	**45.8**	**24.4**	**110.2**	**67.1**	**42.3**	**27.9**
	Mandrill				Motocross				Parrot				Peppers			
K	32	64	128	256	32	64	128	256	32	64	128	256	32	64	128	256
OP	1200.6	740.0	305.7	151.7	1278.7	334.4	213.7	97.0	4102.2	362.6	179.5	104.1	1313.9	354.4	214.1	127.2
MC	2614.9	1619.9	801.5	347.2	410.8	304.8	224.1	136.0	999.7	508.6	274.4	142.9	443.3	339.8	243.9	150.6
OCT	841.2	379.1	245.6	153.7	614.2	305.4	159.4	96.1	469.9	244.6	144.7	85.4	595.9	281.1	166.3	116.6
GOB	**378.0**	237.8	153.2	98.6	**214.8**	123.6	72.7	**44.2**	**252.2**	**140.4**	79.7	**48.1**	**229.4**	**135.9**	**83.1**	**52.9**
ADU	472.8	278.9	171.0	100.5	375.7	148.8	82.9	48.9	1225.1	231.4	144.2	63.2	310.1	159.1	100.3	57.0
KM	431.5	281.7	203.5	144.3	267.8	196.1	138.6	97.6	369.0	244.6	140.3	93.8	367.2	248.9	158.6	110.8
FCM	581.7	411.7	244.5	160.1	354.6	376.2	244.4	162.9	507.9	300.4	177.1	121.8	397.1	240.6	190.2	122.9
ART2	437.0	319.6	206.3	135.4	294.7	197.3	109.1	73.0	353.4	179.7	105.8	68.0	291.8	197.9	113.2	77.4
SOM	**381.4**	**232.1**	**149.1**	**96.7**	219.2	**118.9**	**70.8**	46.0	291.9	147.5	**79.1**	**48.9**	252.9	**139.2**	**81.7**	**53.9**
PM	418.1	**224.7**	**149.7**	**96.8**	**201.3**	**114.4**	**69.6**	**43.3**	**245.2**	**131.3**	**78.3**	49.2	**225.5**	139.9	89.8	54.7

Tab. IV *MSE comparison of the color quantization methods.*

	Airplane				Girl				Hats				Lenna			
K	32	64	128	256	32	64	128	256	32	64	128	256	32	64	128	256
POP	2	3	3	4	4	5	5	8	5	5	5	8	3	4	4	4
MC	2	3	3	3	4	4	4	5	4	4	5	5	3	3	4	4
OCT	8	9	8	9	12	14	13	18	11	12	15	17	10	10	11	11
GOB	22	22	22	22	33	33	34	33	33	34	34	34	21	21	21	22
ADU	81	381	2067	11356	420	670	1117	2208	95	398	2071	11504	91	384	2052	11375
KM	200	374	727	860	93	395	2081	11413	314	596	1124	1950	210	381	681	875
FCM	4528	14671	58961	296074	6628	22668	81003	426836	6826	22664	90481	433151	4684	16105	58019	291207
ART2	121	231	884	1823	521	975	1850	3730	530	1001	1962	3710	118	248	986	2113
SOM	83	119	165	293	123	173	231	435	178	164	271	425	85	116	176	270
PM	61	66	88	134	95	152	129	213	98	106	163	180	152	102	134	182
	Mandrill				Motocross				Parrot				Peppers			
K	32	64	128	256	32	64	128	256	32	64	128	256	32	64	128	256
POP	4	3	4	5	4	6	5	6	5	5	5	6	3	3	4	4
MC	3	3	3	3	6	4	4	7	4	5	4	4	3	4	3	4
OCT	7	10	8	10	11	14	14	18	12	14	13	16	9	8	9	10
GOB	21	21	22	22	33	33	33	33	33	34	34	33	22	23	22	23
ADU	84	369	2029	11198	92	392	2042	11271	92	391	2063	11363	85	393	2104	12307
KM	206	372	740	1398	311	587	1087	2137	321	576	1149	2113	229	412	787	1548
FCM	4439	14441	56354	298427	6498	22207	89159	396676	6747	22554	94258	400386	5268	17227	63570	304416
ART2	493	1033	2357	5070	667	1402	2844	6106	463	855	1684	3369	395	788	1715	3582
SOM	77	103	150	238	127	175	242	426	127	164	250	423	81	115	162	267
PM	152	102	134	182	109	120	150	202	112	147	195	220	84	95	121	141

Tab. V *Processing time (ms) comparison of the color quantization methods.*

natural results even though it requires only 67.18% of the processing time of the conventional SOM method.

When comparing the proposed method with GOB, which is the most efficient algorithm for the processing time, the proposed method gives better results, but about 3 to 8 times the processing time is required. However, SOM methods do not need to repeat learning when there is new, similar input. They can just update the existing learned results. This process does not require a lot of time. On the other hand, the results from GOB cannot be updated, and it needs to create a new palette for every input. This characteristic of SOM should allow it to obtain faster and better results than GOB in the applications that require repeated color quantization.

In addition, hierarchical methods require a certain amount of time to obtain results. However, SOM methods can adjust the processing time and the quality of the results. This means that depending on system performance and requirements, the proposed method is available in a variety of environments by changing the learning termination condition.

5. Conclusions

In this paper, we propose a new color quantization method using color importance–based SOM. It improves on the conventional SOM color quantization method.

The proposed method defines color importance using the frequency of colors, and dynamically adjusts the learning rate and the radius of the neighborhood based on color importance. In other words, the proposed method uses color importance to speed up SOM learning.

To evaluate the performance of the proposed method, we quantified MAE, MSE, and processing time on a set of eight true-color images commonly used in color quantization papers. The lower MAE and MSE values mean there is less distortion of the original image. The proposed method has the lowest MAE and MSE with most of the test images, as shown in Tab. III and Tab. IV. So, we conclude the proposed method minimizes the distortion of the original image. Also, the proposed method is the fastest among the partitional clustering color quantization methods, as shown in Tab. V. It takes only 67.18% of the conventional SOM processing time. Therefore, the experimental results prove that the proposed method is effective for color quantization.

In this paper, the most important thing is color importance. We defined color importance, and used it to improve the conventional SOM color quantization method. Therefore, if there is a more effective method for defining color importance, then it makes for a more effective color importance–based quantization method than the proposed method. Color importance is currently defined by using color frequency, but the distance between pixels, clustering, component analysis, and so on, can also be used. The processing time for color quantization will slightly increase, but it will be able to reduce MAE and MSE by calculating more effective color importance. Therefore, for future work, we will study how to improve the definition of color importance.

References

[1] BING Z., JUNYI S., QINKE P. An adjustable algorithm for color quantization. *Pattern Recognition Letters.* 2004, 25(16), pp. 1787–1797, doi: 10.1016/j.patrec.2004.07.005.

[2] BRUN L., TRÉMEAU A. Color quantization. In: G. SHARMA, ed. *Digital Color Imaging Handbook.* CRC Press, 2002, pp. 589–638, doi: 10.1201/9781420041484.ch9.

[3] CAK S., DIZDAR E.N., ERSAK A. A fuzzy colour quantizer for renderers. *Displays.* 1998, 19(2), pp. 61–65, doi: 10.1016/s0141-9382(98)00038-9.

[4] CELEBI M.E. An effective color quantization method based on the competitive learning paradigm. In: *Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition.* 2009, pp. 876–880.

[5] CELEBI M.E., et al. Batch Neural Gas with Deterministic Initialization for Color Quantization. *Computer Vision and Graphics.* Springer Berlin Heidelberg, 2012, pp. 48–54, doi: 10.1007/978-3-642-33564-8_6.

[6] CELEBI M.E., HWANG S., WEN Q. Color Quantization Using the Adaptive Distributing Units Algorithm. *The Imaging Science Journal.* 2014, 62(2), pp. 80–91, doi: 10.1179/1743131x13y.0000000059.

[7] CELEBI M.E. Improving the Performance of K-means for Color Quantization. *Image and Vision Computing.* 2011, 29(4), pp. 260–271, doi: 10.1016/j.imavis.2010.10.002.

[8] CELEBI M.E., SCHAEFER G. Neural gas clustering for color reduction. In: *Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV 2010)*, Las Vegas, Nevada, USA. 2010, pp. 429–432.

[9] CHANG C.H., et al. New adaptive color quantization method based on self-organizing maps. *IEEE Trans. Neural Netw.* 2005, 16(1), pp. 237–249, doi: 10.1109/tnn.2004.836543.

[10] CHEN L.P., et al. An improved SOM algorithm and its application to color feature extraction. *Neural Computing and Applications.* 2014, 24(7–8), pp. 1759–1770, doi: 10.1007/s00521-013-1416-9.

[11] CHUNG K.L., et al. Speedup of color palette indexing in self-organization of Kohonen feature map. *Expert Systems with Applications.* 2012, 39(3), pp. 2427–2432, doi: 10.1016/j.eswa.2011.08.092.

[12] DEKKER A. Kohonen neural networks for optimal colour quantization. *Network: Computation in Neural Systems.* 1994, 5(3), pp. 351–367, doi: 10.1088/0954-898x/5/3/003.

[13] DENG Y., MANJUNATH B. Unsupervised segmentation of color texture regions in images and video. *IEEE Trans. Pattern Anal. Machine Intell.* 2001, 23(8), pp. 800–810, doi: 10.1109/34.946985.

[14] DENG Y., et al. An efficient color representation for image retrieval. *IEEE Trans. on Image Process.* 2001, 10(1), pp. 140–147, doi: 10.1109/83.892450.

[15] FRACKIEWICZ M., PALUS H. KM and KHM clustering techniques for colour image quantisation. In: R. JOAO MANUEL, S. TAVARES, R.M. NATAL JORGE, eds. *Computational Vision and Medical Image Processing: Recent Trends.* Berlin: Springer, 2011, pp. 161–174, doi: 10.1007/978-94-007-0011-6_9.

[16] GERVAUTZ M., PURGATHOFER W. Simple method for color quantization: octree quantization. In: N. MAGNENAT-THALMAN, D. THALMANN, eds. *New Trends in Computer Graphics.* Berlin: Springer, 1988, pp. 219–231, doi: 10.1007/978-3-642-83492-9_20.

[17] GOLDBERG N. Colour image quantization for high resolution graphics display. *Image and Vision Computing.* 1991, 9(5), pp. 303–312, doi: 10.1016/0262-8856(91)90035-n.

[18] HECKBERT P. Color image quantization for frame buffer display. *Computer Graphics.* 1982, 16(3), pp. 297–307, doi: 10.1145/965145.801294. ACM SIGGRAPH proceedings.

[19] HU Y.C., LEE M.G. K-means based color palette design scheme with the use of stable flags. *J. Electron. Imaging.* 2007, 16(3), pp. 033003, doi: 10.1117/1.2762241.

[20] HU Y.C., SU B.H. Accelerated K-means clustering algorithm for colour image quantization. *The Imaging Science Journal.* 2008, 56(1), pp. 29–40, doi: 10.1179/174313107x176298.

[21] HUANG Y.L., CHANG R.F. A fast finite-state algorithm for generating RGB palettes of color quantized. *Journal of Information Science and Engineering.* 2004, 20(4), pp. 771–782.

[22] KASUGA H., YAMAMOTO H., OKAMOTO M. Color quantization using the fast k-means algorithm. *Systems and Computers in Japan.* 2000, 31(8), pp.33–40.

[23] KIM D.W., LEE K., LEE D. A novel initialization scheme for the fuzzy C-means algorithm for color clustering. *Pattern Recognition Letters.* 2004, 25(2), pp. 227–237, doi: 10.1016/j.patrec.2003.10.004.

[24] KIM K.B., KIM M., WOO Y.W. Recognition of Shipping Container Identifiers Using ART2-Based Quantization and a Refined RBF Network. In: BELICZYNSKI B., DZIELINSKI A., IWANOWSKI M., RIBEIRO B., eds. *Adaptive and Natural Computing Algorithms.* Proceedings of the 8th International Conference ICANNGA 2007, Warsaw, Poland. Berlin, Heidelberg: Springer, 2007, pp. 572–581, doi: 10.1007/978-3-540-71629-7_64.

[25] KUO C.T., CHENG S.C. Fusion of color edge detection and color quantization for color image watermarking using principal axes analysis. *Pattern Recognition.* 2007, 40(12), pp. 3691–3704, doi: 10.1016/j.patcog.2007.03.025.

[26] JOY G., XIANG Z. Center-cut for color image quantization. *The Visual Computer.* 1993, 10(1), pp. 62–66, doi: 10.1007/bf01905532.

[27] ORCHARD M., BOUMAN C. Color quantization of images. *IEEE Trans. Signal Process.* 1991, 39(12), pp. 2677–2690, doi: 10.1109/78.107417.

[28] OZDEMIR D., AKARUN L. Fuzzy algorithm for color quantization of images. *Pattern Recognition.* 2002, 35(8), pp. 1785–1791, doi: 10.1016/s0031-3203(01)00170-4.

[29] PAPAMARKOS N., ATSALAKIS A., STROUTHOPOULOS C. Adaptive color reduction. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics).* 2002, 32(1), pp. 44–56, doi: 10.1109/3477.979959.

[30] RASTI J., MONADJEMI A., VAFAEI A. Color reduction using a multi-stage Kohonen self-organizing map with redundant features. *Expert Systems with Applications.* 2011, 38(10), pp. 13188–13197, doi: 10.1016/j.eswa.2011.04.132.

[31] SCHAEFER G. Intelligent approaches to colour palette design. In: H. KWASNICKA, L.C. JAIN, eds. *Innovations in Intelligent Image Analysis.* Berlin: Springer, 2011, pp. 275–289, doi: 10.1007/978-3-642-17934-1_12.

[32] SCHAEFER G., et al. Rough Colour Quantisation. *International Journal of Hybrid Intelligent Systems.* 2011, 8(1), pp. 25-–30.

[33] SCHAEFER G., ZHOU H. Fuzzy clustering for colour reduction in images. *Telecommunication Systems.* 2009, 40(1–2), pp. 17–25, doi: 10.1007/s11235-008-9143-8.

[34] SCHEUNDERS P. Comparison of clustering algorithms applied to color image quantization. *Pattern Recognition Letters.* 1997, 18(11–13), pp.1379–1384, doi: 10.1016/s0167-8655(97)00116-5.

[35] SERTEL O., et al. Histopathological image analysis using modelbased intermediate representations and color texture: Follicular lymphoma grading. *J. Signal Process System Sign Image Video Technology.* 2009, 55(1–3), pp. 169–183, doi: 10.1007/s11265-008-0201-y.

[36] SHERKAT N., ALLEN T., WONG S. Use of colour for hand-filled form analysis and recognition. *Pattern Analysis and Applications.* 2005, 8(1), pp. 163–180, doi: 10.1007/s10044-005-0253-6.

[37] VEREVKA O., BUCHANAN J. Local k-means algorithm for colour image quantization. In: DAVIS W.A., PRUSINKIEWICZ P., eds. *Proceedings of Graphics Interface '95*, Quebec, Canada. Toronto: Canadian Information Processing Society, 1995, pp. 128–135.

[38] WAN S.J. PRUSINKIEWICZ P., WONG, S.K.M. Variance-based color image quantization for frame buffer display. *Color Research and Application.* 1990, 15(1), pp. 52–58, doi: 10.1002/col.5080150109.

[39] WEN Q., CELEBI M.E. Hard versus Fuzzy c-means clustering for color quantization. *EURASIP Journal on Advances in Signal Processing.* 2011(1), 2011, pp. 118–129, doi: 10.1186/1687-6180-2011-118.

[40] WU X. Efficient statistical computations for optimal color quantization. In: J. ARVO, ed. *Graphics Gems*, vol. II. London: Academic Press, 1991, pp. 126–133, doi: 10.1016/b978-0-08-050754-5.50035-9.

[41] XIANG Z. Color image quantization by minimizing the maximum intercluster distance. *ACM Trans. Graph.* 1997, 16(3), pp. 260–276, doi: 10.1145/256157.256159.

[42] XIAO Y., et al. Self-organizing map-based color palette for high-dynamic range texture compression. *Neural Comput. and Appl.* 2012, 21(4), pp. 639–647, doi: 10.1007/s00521-011-0654-y.

[43] YANG C.K., TSAI W.H. Color image compression using quantization, thresholding, and edge detection techniques all based on the moment-preserving principle. *Pattern Recognition Letters.* 1998, 19(2), pp. 205–215, doi: 10.1016/s0167-8655(97)00166-9.

[44] YANG C.Y., LIN J.C. RWM-cut for color image quantization. *Computers and Graphics.* 1996, 20(4), pp. 577–588, doi: 10.1016/0097-8493(96)00028-3.