International Journal of Computational Intelligence Systems (2023) 16:7
https://doi.org/10.1007/s44196-023-00185-x

RESEARCH ARTICLE

Adaptive Color Quantization Method with Multi-level Thresholding

Mahmut Kılıçaslan[1] · **Mürsel Ozan İncetaş**[2]

Received: 28 April 2022 / Accepted: 17 January 2023

Abstract
In this study, a novel color quantization approach which automatically estimates the number of colors by multi-level thresholding based on the histogram is proposed. The method consists of three stages. First, red–green–blue is clustered by threshold values. Thus, the pixels are positioned in a cluster or sub-prism. Second, the color palette is produced by determining the centroids of the clusters. Finally, the pixels are reassigned to clusters based on their distance from each centroid. The average of the pixels included in each cluster also represents the color of that cluster. While conventional methods are user-dependent, the proposed algorithm automatically generates the number of colors by considering the pixels assigned to the clusters. Additionally, the multi-level thresholding approach is also a solution to the initialization problem, which is another important issue for quantization. Consequently, the experimental results of the method tested with various images show better performance than many frequently used quantization techniques.

Keywords Multi-level thresholding · Cluster · Centroid · Histogram · Color quantization

Abbreviations

RGB	Red–green–blue
BSDS	Berkeley segmentation dataset
CoG	Center of gravity of histogram
FCM	Fuzzy C-means
MC	Median cut
PSNR	Peak signal–noise ratio
SSIM	Structural Similarity Index

1 Introduction

With the development of technology in recent years, digital images are used in many parts of our lives. Additionally, high-quality digital images are created thanks to advanced hardware. RGB is a typical color space in which related images are described. The color components called red (R), green (G), and blue (B) are 8-bit colors. Therefore, each pixel consists of 24 bits in the RGB color space [1]. Since each color channel is in the range of 0–255, there are approximately 16 million possible colors. The high number of colors in images can cause problems in data transfer and storage [2]. Color reduction is a procedure that represents images with limited color and plays an important role in many image-processing applications, such as segmentation, compression, and image retrieval [3–5]. The main purpose of color reduction is to eliminate a substantial number of details from the image and create a representation close to the original image with minimal distortion. Clearly, this will reduce the storage requirements and transfer time of the image. Color reduction consists of two stages: color palette or codebook design and pixel mapping. In the first stage, the appropriate number of colors is determined (usually 8–256 colors are used), while in the second stage, the pixels in the original image are reassigned according to the color palette [6, 7].

There are different studies on color quantization in the literature. Quantization approaches can generally be categorized into two groups: splitting [8–13] and clustering [14–16]. Splitting methods are iterative approaches that divide the color space into sub-prisms. Median Cut (MC) [8], Center-Cut (CC) [9], Octree (OCT) [10], Variance Cut (VC), Variance Cut with Lloyd-Max (VCL), Variance-Based (WAN) [17, 18], Popularity (POP) and Modified Popularity (MPOP) [19, 20], Greedy Orthogonal

✉ Mahmut Kılıçaslan
m.kilicaslan@ankara.edu.tr

Mürsel Ozan İncetaş
ozan.incetas@alanya.edu.tr

[1] Computer Technologies Department, Ankara University, Ankara, Turkey

[2] Computer Technologies Department, Alanya Alaaddin Keykubat University, Antalya, Turkey

Published online: 03 February 2023

bi-Partitioning (WU) [12], Radius-Weighted Mean-Cut (RWM) [21], Split and Merge (SAM) [20], Self-Organization Map (SOM) [22], and Modified Max–Min (MMM) [23] are commonly used quantization methods. The MC method [8] repeatedly separates the three-dimensional color space into smaller regions. The method is based on the distribution of colors in the image. In other words, the boxes with the most pixels counted at each step are split along the axis. The process is repeated until the number of different colors in the original image is reduced to the desired number of colors. The average colors of each separated region generate the color palette. After the splitting process, the positions of the pixels become close to the colors with more pixels. For this reason, the process leads to information loss. The CC [9] algorithm is an improved form of the MC method. In the aforementioned method, the division process is carried out from the center points of the prisms, and two prisms with almost equal numbers of pixels are obtained. Another reduction method is the OC (octal tree) [10] color quantization algorithm based on an 8-level tree data structure. It uses the pixels of the image to create the octal tree. Moreover, it divides the color cube into 8 sub-cubes. Each sub-cube is divided by 8 until the pre-defined number of colors is formed. In the binary splitting method, the middle point is split by considering the box with the dominant pixel value. While the octal tree generates codebooks with the color distribution in the image, it neglects the frequency of the colors. Therefore, if the images consist of similar colors and different low-frequency colors, the effect of the octal tree will be reduced as expected. The CC, WAN, and RWM approaches are similar to MC. Although these methods produce fast results, color distributions affect the effectiveness of the methods. This is because MC is a technique that focuses on the median of pixels. Among other approaches, POP and MPOP are uniformity-based quantization methods. The division process continues iteratively until the desired number of colors is obtained with a uniform color palette. The method neglects colors with sparse distributions in the color space. The MMM method selects the first palette color randomly. The colors of the palette are chosen to be the colors that have the greatest minimum weighted Euclidean distance. MMM [23] is iterative and has an initial value problem. WU is also a variance-based splitting method. In this approach, the division operation is performed at the point that minimizes the variance between pixels. The SAM approach consists of two stages. The first of these is the splitting stage. At this stage, the colors are uniformly divided into a specified number of sets. The number that is determined here is randomly chosen by the user. The second stage is merging. At this stage, the pairs with the lowest reduction errors are combined. The disadvantages of the method are the use of a uniform structure and its user-dependent nature. SOM [22] is a model proposed for artificial neural networks. SOM is an effective color reduction approach that is used in many fields. However, it has some disadvantages. It needs sufficient data to form meaningful clusters. Therefore, it is expected to have poor performance in images with a low degree of color distribution. The VC and VCL [17] methods divide the image's colors into subsets and determine the maximum sum of squared error between sets. The aforementioned techniques use a hash function while executing the operation. Some parameters in the function are user-dependent. Additionally, the methods have an iterative perspective.

K-means is a well-known clustering method, and it has been proposed to solve the color quantization problem [24–27]. K-means considers the initial centers for K clusters and applies an iterative process to improve these centroids. In other words, initial cluster centers are created to reduce the color palette. Each iteration correlates pixels with their distance from cluster centers. The process is continued until the user-defined iteration. The major drawback of this method is that the reduced images are affected by the initial cluster centers. Moreover, it is a user-dependent and iterative method. Fuzzy C-means is another clustering method and is similar to K-means. While K-means associates each pixel with a single cluster, Fuzzy C-means determines the membership degree of each pixel with several clusters. Researchers have proposed various color reduction approaches based on the approach mentioned above [28–30]. Recently, it has been seen that the success of the quantization of various techniques developed especially based on optimization or K-means approaches is also relatively high. In 2021, the iterative Ant-tree color quantization technique (ITATCQ-n) was developed using the ant-tree optimization approach [31]. In 2022, an incremental technique (IOKM) was also developed based on the K-means algorithm [32]. Comparative results of the aforementioned techniques are given in Chapter 4.

In this study, a novel color reduction approach is proposed. First, the histogram of each color component in the image is subjected to multi-level thresholding using the center of gravity (CoG). In the last 10 years, the CoG approach has been used in many studies for thresholding. Contrary to variance or entropy calculations in histograms, CoG can be calculated much more easily, and it can perform thresholding quite simply. Additionally,

it has been observed that the CoG thresholding method achieves very successful results in many areas from the analysis of medical images [33] and filtering [34] to CBIR approaches [35]. Due to its low computational complexity and easy application, the CoG technique was used in this study for thresholding the histogram with by multi-thresholding. Each color component of the RGB color space is divided into sub-prisms according to the threshold values obtained with CoG from their histograms. After the multi-thresholding process in the proposed model with CoG, each pixel is assigned to the closest cluster by calculating its distance from the center of the prism. This way, a pixel is assigned to a closer sub-prism, even if its color value is inside another sub-prism. Thus, it is ensured that pixels with similar color values remain in the same sub-prism. Finally, the color palette is created by the average of the pixels included in the sub-prisms.

The rest of this manuscript is organized as follows: Sect. 2 describes multi-level image thresholding, and Sect. 3 describes the proposed color reduction algorithm. Section 4 includes comparative experimental results. Finally, the conclusions are presented in Sect. 5.

2 Image Thresholding by Using Center of Gravity of Histogram (CoG)

The main purpose of image thresholding is dividing the image into distinct regions, so that the same regions have the same attributes, and different regions have dissimilar attributes [36]. This way, homogeneity is confirmed within the clusters. A histogram is an important feature vector that shows the probability distribution of the colors in an image. The execution of thresholding operations on the histogram is a frequently used method [37–39]. The histogram is defined as follows:

$$p_i = n_i/(MxN), \tag{1}$$

where n_i is the number of pixels at the ith level, and MxN is the size of the image. Demirci proposed an algorithm for image thresholding based on image histogram [40, 41]. In this approach, the center of gravity of the histogram is used for the estimation of threshold values. Initially, the global mean of the histogram, μ_τ, is calculated according to Eq. 2. Subsequently, the center of gravity is determined for the ranges $0-\mu_\tau$ and $\mu_\tau-255$. At each level, the previous threshold values are repeatedly used. Thus, the number of threshold values increases with each step. These algorithms based on the sum of partial probability distributions (ω) and partial means (μ_{xy}) are described as in Eq. 3

$$\mu_\tau = \sum_{i=0}^{L} ip_i, \tag{2}$$

$$\begin{aligned}
\omega_0 &= \sum_{i=0}^{\mu_T-1} p_i,\ \mu_0 = \sum_{i=0}^{\mu_T-1} \frac{ip_i}{\omega_0},\\
\omega_1 &= \sum_{i=\mu_T}^{255} p_i,\ \mu_1 = \sum_{i=\mu_T}^{255} \frac{ip_i}{\omega_1},\\
\omega_{00} &= \sum_{i=0}^{\mu_0-1} p_i,\ \mu_{00} = \sum_{i=0}^{\mu_0-1} \frac{ip_i}{\omega_{00}},\\
\omega_{01} &= \sum_{i=\mu_0}^{\mu_T-1} p_i,\ \mu_{01} = \sum_{i=\mu_0}^{\mu_T-1} \frac{ip_i}{\omega_{01}}\\
\omega_{10} &= \sum_{i=\mu_T}^{\mu_1-1} p_i,\ \mu_{10} = \sum_{i=\mu_T}^{\mu_1-1} \frac{ip_i}{\omega_{10}},\\
\omega_{11} &= \sum_{i=\mu_1}^{255} p_i,\ \mu_{11} = \sum_{i=\mu_1}^{255} \frac{ip_i}{\omega_{11}}.
\end{aligned} \tag{3}$$

$L=255$ is the maximum gray level in Eq. 2. As seen here, each stage depends on the values obtained in the previous stage. The estimated threshold values are shown as a representation in Table 1. While using the general average of the image for a single threshold, recursive means are taken into account for more thresholds.

3 Proposed Color Quantization Method

The main purpose of the color quantization procedure is to reconstruct the image with a limited number of colors. The similarity of the reduced image to the original and the preservation of important information are a substantial challenge in quantization. The problem with splitting methods is that the splitting decision cannot be restarted at every level. Nevertheless, clustering-based methods are often affected by the initial cluster centroids. The developed method focuses on solving the aforementioned problems.

In the RGB color space, each gray level is in the range of $L=\{0,1,\ldots 255\}$. The aforementioned process reduces each gray level to k ($k<L$). In the proposed strategy, the reduction operation is realized by multi-level thresholding. Each pixel in digital images represents a point in the 3D color space. Figure 1a, b shows randomly selected images from

Table 1 Number and values of thresholds with center of gravity of histogram

Levels	Number of thresholds	Values of thresholds
1st	t_1	μ_T
2nd	$t_1\ t_2$	$\mu_0\ \mu_1$
3rd	$t_1\ t_2\ t_3$	$\mu_0\ \mu_T\ \mu_1$
4th	$t_1\ t_2\ t_3\ t_4$	$\mu_{00}\ \mu_{01}\ \mu_{10}\ \mu_{11}$
5th	$t_1\ t_2\ t_3\ t_4\ t_5$	$\mu_{00}\ \mu_{01}\ \mu_T\ \mu_{10}\ \mu_{11}$

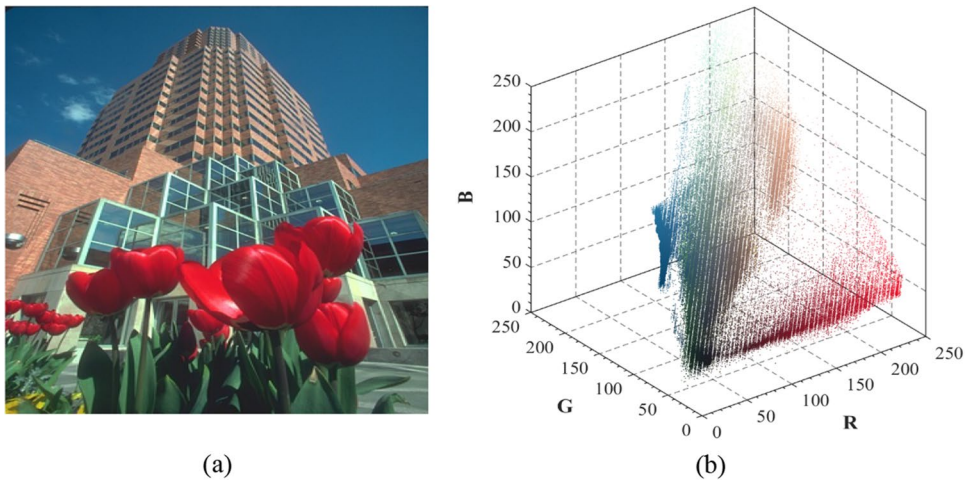

Fig. 1 **a** Original image. **b** Color distribution of original image

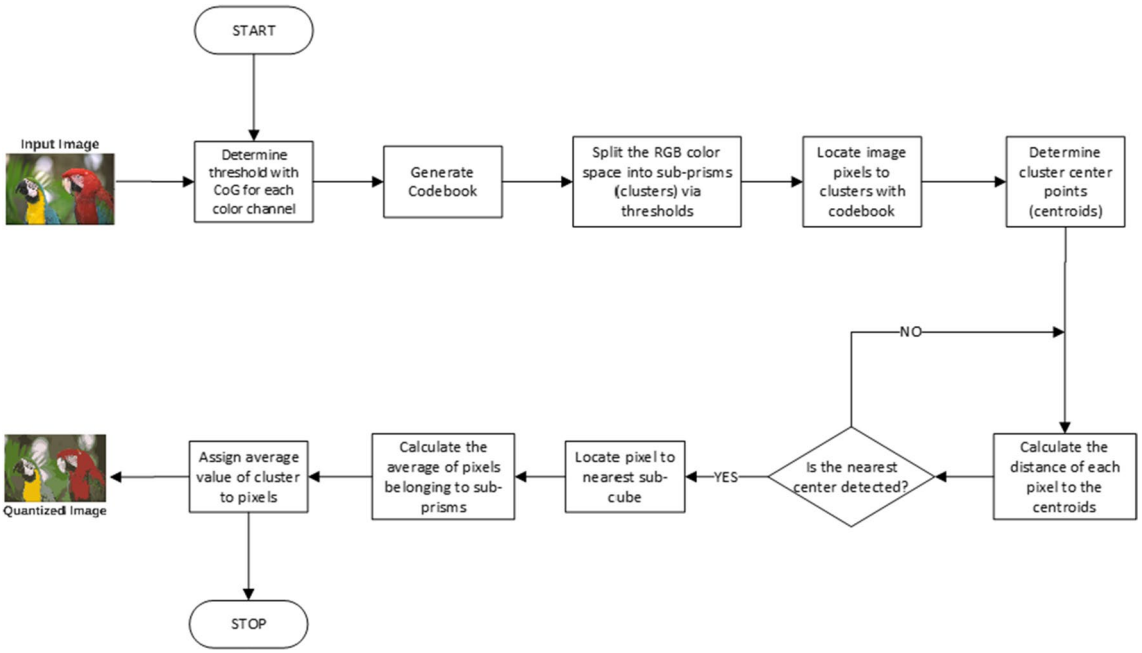

Fig. 2 Flowchart of proposed algorithm

the Berkeley Segmentation Database (BSDS) and its color distribution, respectively.

In the developed quantization approach, as also shown in the flowchart in Fig. 2, first, the Center of Gravity (CoG) of the histogram and the thresholds for each color channel are determined. Additionally, the threshold values determine which sub-cube the image's pixels will be assigned to, which is also called the codebook. Table 2 represents a codebook for a single threshold. Afterward, the RGB space is divided into sub-regions by means of threshold values. Kılıçaslan et al. proposed an approach that reduces color images using the CoG multi-thresholding method in a previous study of theirs [35]. They selected the cluster centers as threshold values. However, in this study, the color palette is created by

Table 2 Codebook for single threshold

Class label (c_i)	Partition rules	Binary code
s_0	if($R <= t_{r,1}$ && $G <= t_{g,1}$ && $B <= t_{b,1}$)	*000*
s_1	if($R <= t_{r,1}$ && $G <= t_{g,1}$ && $B >= t_{b,1}$)	*001*
s_2	*if*($R <= t_{r,1}$ && $G >= t_{g,1}$ && $B <= t_{b,1}$)	*010*
s_3	*if*($R <= t_{r,1}$ && $G >= t_{g,1}$ && $B >= t_{b,1}$)	*011*
s_4	if($R >= t_{r,1}$ && $G <= t_{g,1}$ && $B <= t_{b,1}$)	*100*
s_5	if($R >= t_{r,1}$ && $G <= t_{g,1}$ && $B >= t_{b,1}$)	*101*
s_6	if($R >= t_{r,1}$ && $G >= t_{g,1}$ && $B <= t_{b,1}$)	*110*
s_7	if($R >= t_{r,1}$ && $G >= t_{g,1}$ && $B >= t_{b,1}$)	*111*

Table 3 Center points of color components

Center points of R	Center points of G	Center point of B
$c_{r1} = 0 + (t_{r,1} - 0)/2$	$c_{g1} = 0 + (t_{g,1} - 0)/2$	$c_{b1} = 0 + (t_{b,1} - 0)/2$
$c_{r2} = t_{r,1} + (t_{r,2} - t_{r,1})/2$	$c_{g2} = t_{g,1} + (t_{g,2} - t_{g,1})/2$	$c_{b2} = t_{b,1} + (t_{b,2} - t_{b,1})/2$
$c_{r3} = t_{r,2} + (L - t_{r,2})/2$	$c_{g3} = t_{g,2} + (L - t_{g,2})/2$	$c_{b3} = t_{b,2} + (L - t_{b,2})/2$

Table 4 Cluster centers and points corresponding to centers

Cluster index	Cluster centers	Points of cluster centers
s_1	C_{s1}	$P(c_{r1},c_{g1},c_{b1})$
s_2	C_{s2}	$P(c_{r1},c_{g1},c_{b2})$
s_3	C_{s3}	$P(c_{r1},c_{g1},c_{b3})$
…	…	…
s_{13}	C_{s13}	$P(c_{r2},c_{g2},c_{b1})$
s_{14}	C_{s14}	$P(c_{r2},c_{g2},c_{b2})$
s_{15}	C_{s15}	$P(c_{r2},c_{g2},c_{b3})$
…	…	…
s_{25}	C_{s25}	$P(c_{r3},c_{g3},c_{b1})$
s_{26}	C_{s26}	$P(c_{r3},c_{g3},c_{b2})$
s_{27}	C_{s27}	$P(c_{r3},c_{g3},c_{b3})$

taking into account the center of the sub-cubes or regions. Unlike previous studies, in the proposed method, it is evaluated that pixels in a cluster separated by threshold values could be close to the center of another cluster. Especially, pixels close to threshold values can be included in another sub-prism. With this remarkable situation, the details in the image are preserved to a higher extent. The maximum number of clusters, s, is calculated as

$$s = (t + 1)^3, \tag{4}$$

where t is the number of threshold values. After the sub-cubes are created, the centers of each are determined for each color component. Table 3 demonstrates the center points for each of the R, G, and B components for two threshold values.

The t_r, t_g, and t_b values given in Table 3 are the threshold values calculated for the red, green, and blue color channels. For two threshold values, three centers are defined for each color channel. Combinations of center points produce cluster centroids. Table 4 shows the points of the centroid for each cluster in the color space depending on the values in Table 3. In fact, the combination of threshold values creates sub-regions. Thresholds are critical points for regions, but the distance of pixels from the center increases, especially in regions with large volumes. For this reason, calculating the Euclidean distances of the pixels to the cluster centers will lead to a more appropriate pixel mapping process. Pixels

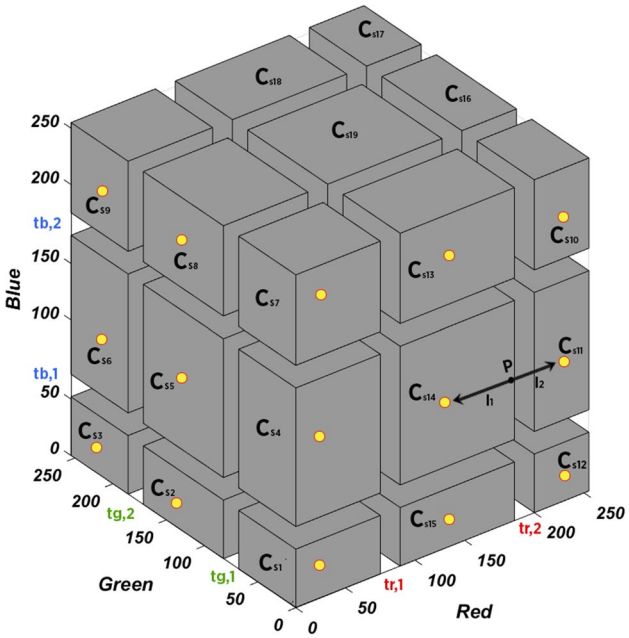

Fig. 3 Clustered RGB color space and cluster center of sub-prism

are included in the cluster where they are the closest to the center. Then, the pixels in the cluster are assigned the average value of the pixels in the same cluster. Figure 3 represents the clustered RGB color space and the center points of all clusters (yellow points).

Point P shown in Fig. 3 represents any pixel of the image. Therefore, it can be expressed as $P(r_x,g_y,b_z)$. With multilevel thresholding, the corresponding pixel is positioned in the s_{14} region, while it is closer to the center of the s_{11} region. Therefore, it is reassigned to the s_{11} region. In Fig. 3, the l_1 values show the Euclidean distances from the center of C_{s14}, and the l_2 values show these distances from the center of C_{s11}. l_1 and l_2 are calculated as follows:

$$\begin{aligned} l_1 &= \sqrt{(r_2 - r_x) + (g_2 - g_y) + (b_2 - b_z)} \\ l_2 &= \sqrt{(r_2 - r_a) + (g_1 - g_b) + (b_2 - b_c)}. \end{aligned} \tag{5}$$

In the proposed method, the distances of each pixel in the image from all sub-prism centers are calculated. Pixels are assigned to the closest region. Figure 4a, b demonstrates

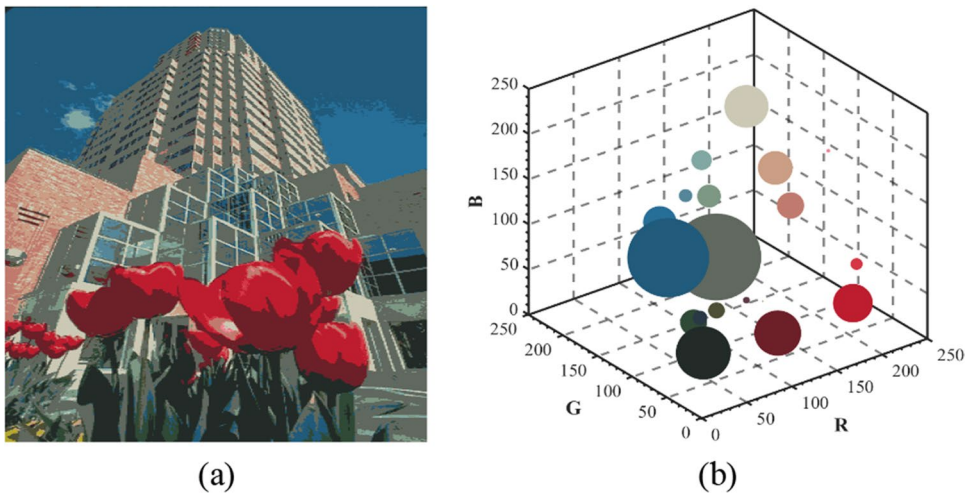

Fig. 4 **a** Quantized image: $t=2$, $s=27$, $m=21$. **b** Color distribution of **a**

a quantized image and color distribution, respectively. 27 clusters are generated using two thresholds. However, pixels cannot be assigned to any cluster. As seen in Fig. 3, 27 clusters are created for two thresholds. On the other hand, the pixels are positioned in only 21 clusters. In other words, the number of colors (m) in the reduced image is 21.

4 Experimental Results and Discussion

Our experimental results are obtained using Visual Studio 2015 and MATLAB 2017 on a computer with a Core i7-8750H 2.20 GHz CPU and 16 GB of RAM. The proposed method and the compared approaches are tested on the BSDS. To evaluate the performance of the developed method, 20 randomly selected images from the related dataset are quantized using different threshold numbers. Additionally, comparative results are gathered for the eight most frequently used images in quantization studies. PSNR (Peak Signal–Noise Ratio), MSE (Mean Squared Error), SSIM (Structural Similarity Index), and computation time are considered as evaluation criteria. PSNR, MSE, and SSIM are calculated as follows:

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{255^2}{\mathrm{MSE}}\right), \tag{6}$$

$$\mathrm{MSE} = \frac{1}{M\times N}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[I(x,y)-Q(x,y)\right]^2, \tag{7}$$

$$\mathrm{SSIM}(I,Q) = \frac{\left(2\mu_I\mu_Q+c_1\right)\left(2\sigma_{IQ}+c_2\right)}{\left(\mu_I^2\mu_Q^2+c_1\right)\left(\sigma_I^2+\sigma_Q^2+c_2\right)}, \tag{8}$$

where I and Q are the original and quantized images, respectively. μ_I, μ_Q represent the averages of I and Q, and σ_I^2, σ_Q^2 are the variances of I and Q. The covariances of the original and quantized images are expressed as σ_{IQ}.

The proposed method automatically determines the number of colors. For this reason, to make the experimental comparisons compatible, the maximum iteration number of FCM is set to 100, and the number of colors of the proposed method is balanced. However, some approaches, such as the Median Cut method, reduce the number of colors to powers of two. For such techniques, the number of colors is set close to the values estimated by the proposed method. The original and quantized images are shown in Fig. 5. The outputs of the proposed method are obtained by dividing the color space into 64 sub-regions with the threshold number $t=3$. As seen in Fig. 5, the developed technique estimated fewer than 64 colors (m). The fact that the number of colors is lower than 64 ($m<64$) indicates that there are no pixels in some sub-regions as a result of the quantization process performed with the proposed method.

The numerical results of the visual outputs shown in Fig. 5 are reported in Table 5. For each image in Fig. 5, the PSNR/SSIM results of the FCM, MC, MLQ, and proposed method, as well as the number of colors obtained with the proposed method, are given in detail in Table 5. Figure 5 and Table 5 clearly prove that the proposed approach has satisfactory quantization performance. The most important advantage of the proposed method is that it does not require a pre-defined number of representative colors. In other words, it automatically determines the number of representative colors. Additionally, it eliminates the initialization problem. Determining the number of colors is a crucial issue in both the FCM and Median Cut quantization techniques. The Median Cut also represents the number of colors based on powers of two.

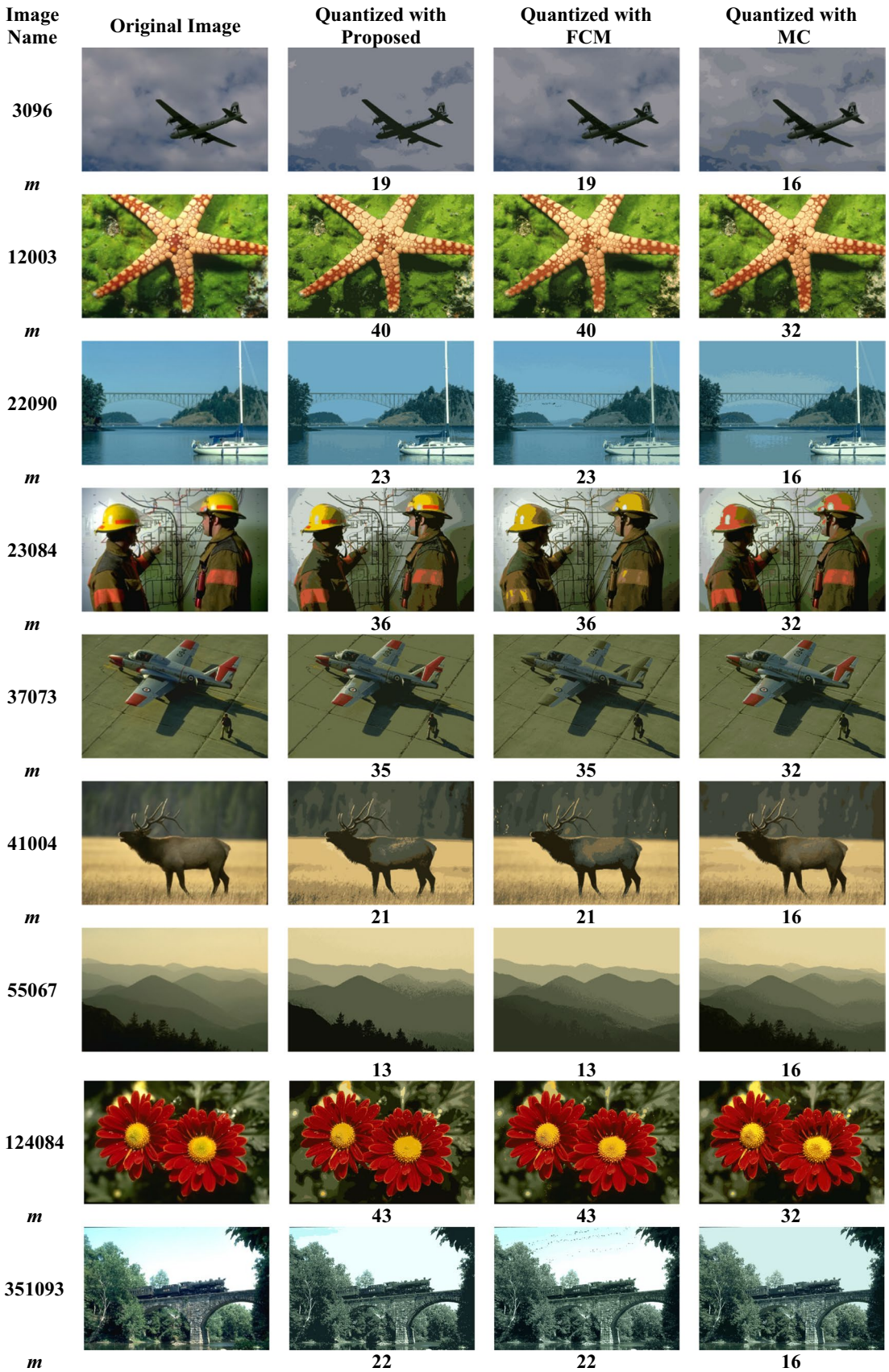

Fig. 5 Original test images and quantized output image with different algorithms

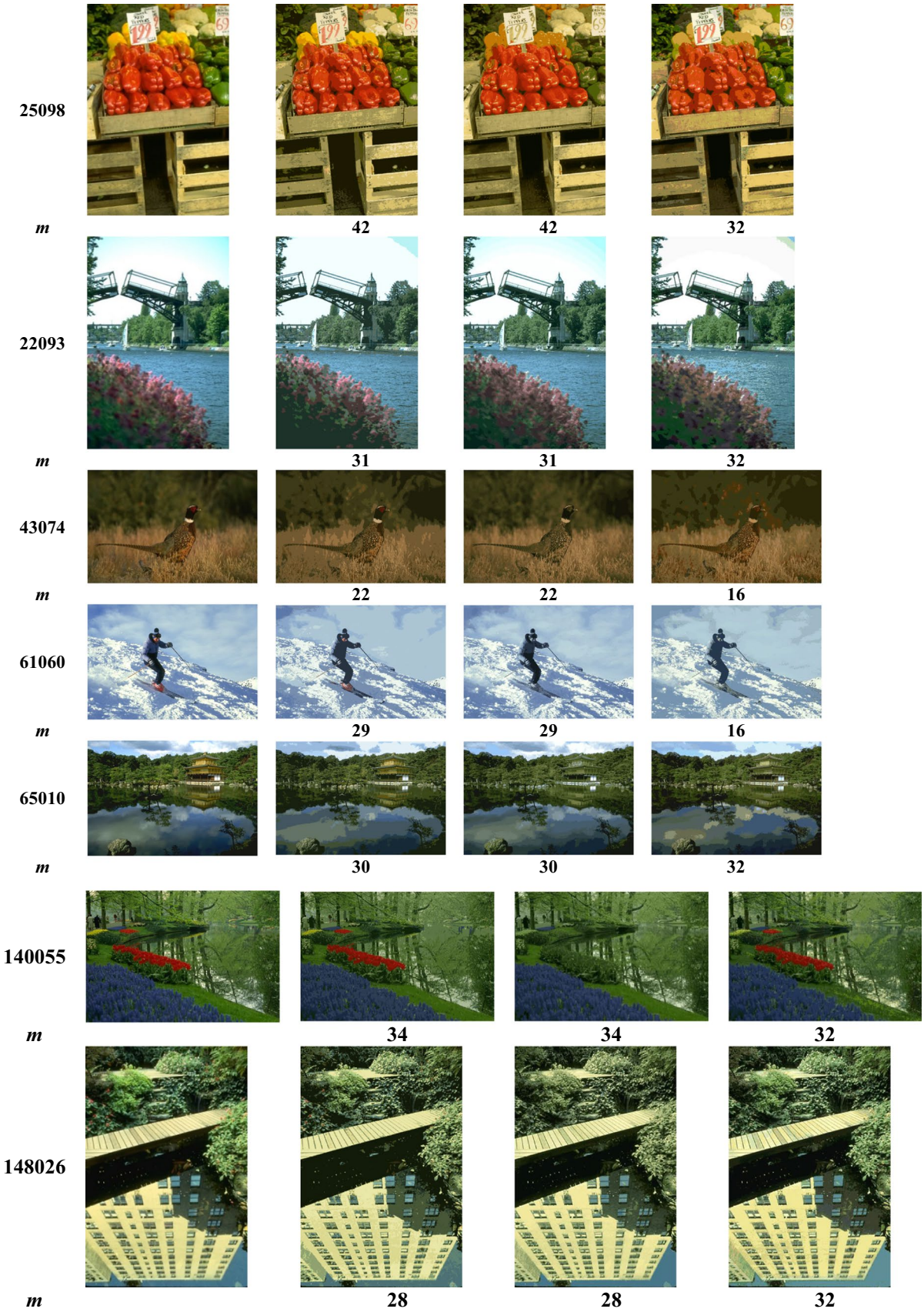

Fig. 5 (continued)

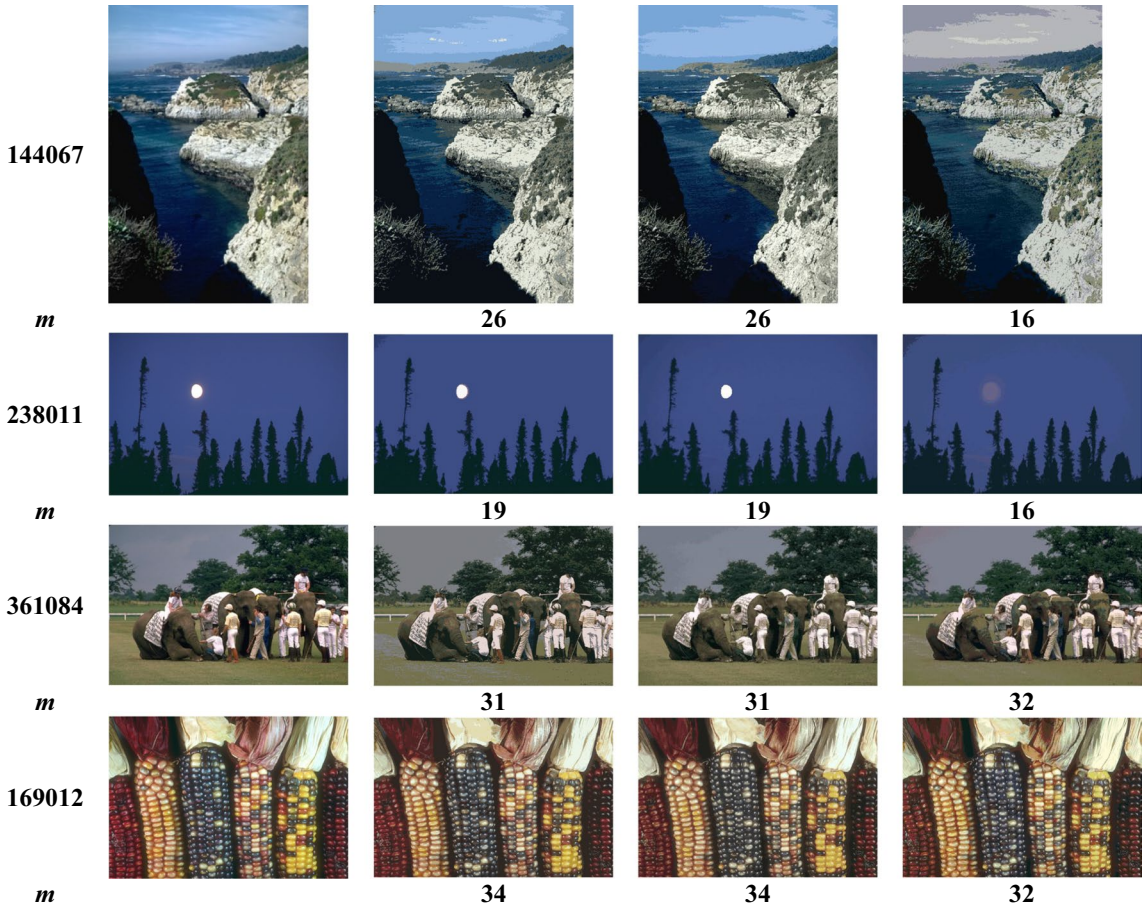

Fig. 5 (continued)

While the results in Table 5 are calculated for three thresholds ($t=3$), Table 6 includes comparisons of the quantization results obtained with 5 thresholds ($t=5$). Considering the SSIM and PSNR results in Tables 5 and 6, all methods show parallel results. For example, the proposed technique for '55,067' outperforms other methods in terms of all performance criteria. However, FCM appears to be more successful than others for '3096' in all assessments except for the computation period. On the other hand, MC is still faster than FCM in terms of competition time, although it is seen that the information loss is higher compared to those in the PSNR and SSIM values. It is clear that the computation time required to perform FCM is affected by the cluster size. Therefore, increasing the number of colors raises the computation time of FCM. The experimental results clearly prove that FCM is slower than the other two methods in terms of computation time. While the average calculation time of FCM is 35.67 in Table 5, it is 66.58 in Table 6. On the other hand, the proposed method and Median Cut are not significantly affected by the increase in the number of colors relative to the calculation time. All methods show that increasing the number of colors results in higher quality outputs. In Table 5, the mean SSIM value is approximately 0.89 for FCM, while the corresponding value for the recommended method and Median Cut is 0.88 and 0.84, respectively. However, the relevant criterion values in Table 6 reach 0.92 for the proposed method, outperforming the other methods. This situation clearly shows that the increase in the number of colors affects the technique developed in this study more positively. Additionally, there are reduced versions of the ten test images in different color numbers with the proposed algorithm in the Appendix. As seen in the figures in the Appendix, it is clearly seen that the more thresholds are used, the higher the quality of the output images. In general, it can be stated that the proposed method and FCM perform similarly to each other and provide satisfactory quantization results. Nevertheless, the automatic determination of the number of colors and the provision of a solution to the initialization problem with threshold values are important capabilities of the proposed algorithm.

Table 5 Overall performance of algorithms with various metrics for $t=3$

Image	m				PSNR				SSIM				Computation time			
	CoG	FCM	MC	MLQ	CoG	FCM	MC	MLQ	CoG	FCM	MC	MLQ	CoG	FCM	MC	MLQ
3096	19	19	16	21	29.96	33.04	30.35	29.90	0.9189	0.9545	0.9070	0.9099	1.81	17.30	1.66	1.77
12003	40	40	32	44	27.93	28.89	26.91	26.81	0.8650	0.8803	0.8211	0.8675	2.17	62.40	2.00	2.14
22090	23	23	16	27	28.68	24.57	27.12	28.84	0.8894	0.9081	0.8524	0.8994	1.48	20.54	1.15	1.45
23084	36	36	32	35	28.59	26.44	21.91	28.63	0.8524	0.8716	0.7724	0.8507	1.28	36.02	1.84	1.27
351093	22	22	16	23	30.09	27.69	24.97	30.29	0.8687	0.9225	0.7754	0.8707	1.01	32.86	1.53	0.97
37073	35	35	32	34	29.85	27.40	28.7	29.46	0.8808	0.9103	0.8857	0.8798	0.79	23.46	0.82	0.75
41004	21	21	16	25	29.80	31.39	28.86	30.14	0.8485	0.8946	0.8847	0.8685	1.06	19.73	1.59	1.02
55067	13	13	8	9	30.38	28.87	30.27	29.53	0.9783	0.8046	0.9241	0.9283	0.95	12.59	1.35	0.94
124084	43	43	32	45	28.34	29.64	26.87	28.97	0.8188	0.8654	0.7984	0.8210	1.23	82.79	1.65	1.18
25098	42	42	32	33	30.74	26.15	23.09	26.02	0.8228	0.8440	0.7901	0.7667	1.84	78.24	1.71	1.81
22093	31	31	32	28	29.98	31.66	29.06	23.90	0.8669	0.8967	0.8572	0.8458	1.12	39.55	1.75	1.10
43074	22	22	16	26	34.01	35.80	31.61	27.82	0.9353	0.9317	0.8348	0.9588	1.32	25.01	1.08	1.27
61060	29	29	16	27	32.41	33.34	31.79	26.97	0.9019	0.9164	0.8684	0.9088	1.20	32.97	1.07	1.16
65010	30	30	32	28	29.43	27.4	24.77	23.44	0.9001	0.8829	0.8394	0.8992	1.05	24.39	1.20	1.03
140055	34	34	32	42	30.16	28.74	27.65	27.19	0.8927	0.8678	0.8069	0.9250	1.66	32.56	1.78	1.65
148026	28	28	32	24	26.43	26.39	25.93	20.15	0.9237	0.9107	0.8289	0.8239	1.15	35.25	1.45	1.10
144067	26	26	16	26	31.88	31.32	23.95	25.83	0.8885	0.8830	0.8152	0.8798	1.46	60.46	1.35	1.45
238011	19	19	16	17	39.79	38.97	29.90	33.37	0.9421	0.9187	0.8591	0.9327	1.65	18.25	1.56	1.62
361084	31	31	32	29	31.06	32.27	31.22	24.72	0.8760	0.8815	0.8729	0.8869	1.35	24.66	1.60	1.31
169012	34	34	32	37	29.15	29.15	27.85	23.71	0.9041	0.8983	0.8524	0.9125	1.25	34.38	1.25	1.22
Mean value	–	–	–	–	30.43	29.96	27.64	27.28	0.8887	0.8922	0.8423	0.8818	1.34	35.67	1.47	1.31

Table 6 Overall performance of algorithms with various metrics for $t=5$

Image	m				PSNR				SSIM				Computation time			
	CoG	FCM	MC	MLQ	CoG	FCM	MC	MLQ	CoG	FCM	MC	MLQ	CoG	FCM	MC	MLQ
3096	31	31	32	31	36.80	38.08	32.42	36.74	0.9355	0.9451	0.9155	0.9390	1.92	26.88	1.99	1.89
12003	108	108	64	95	32.02	33.74	27.61	28.44	0.9427	0.9172	0.8943	0.9256	2.67	92.85	2.82	2.66
22090	48	48	32	45	34.07	29.25	28.68	32.15	0.9291	0.9049	0.8982	0.9195	1.97	36.47	1.48	1.94
23084	93	93	64	81	32.28	30.74	25.65	29.84	0.9218	0.9010	0.8614	0.8967	2.08	82.98	1.87	2.04
351093	45	45	32	34	33.95	33.23	28.75	27.79	0.9441	0.9530	0.8886	0.9298	1.56	57.38	1.59	1.54
37073	72	72	64	51	36.84	29.82	33.27	30.61	0.9089	0.9133	0.9088	0.8691	1.36	64.54	1.49	1.34
41004	45	45	32	35	34.96	35.97	30.65	31.15	0.9243	0.9207	0.8639	0.8958	1.79	46.96	1.69	1.74
55067	27	27	16	27	35.54	40.04	32.70	34.96	0.9405	0.9394	0.8950	0.9390	1.39	25.36	1.48	1.35
124084	105	105	64	92	33.32	33.32	30.34	32.63	0.8646	0.8662	0.8212	0.8236	2.36	102.01	2.02	2.35
25098	89	89	64	88	31.12	29.81	27.16	30.69	0.8991	0.8568	0.8533	0.8988	2.11	88.17	2.51	2.09
22093	77	77	64	74	32.75	32.12	30.54	26.73	0.9575	0.9729	0.9163	0.9598	1.59	80.65	1.25	1.55
43074	69	69	64	52	37.39	37.60	37.12	30.90	0.9376	0.9528	0.9272	0.8976	1.38	72.43	1.47	1.37
61060	69	69	64	60	34.62	34.48	34.45	28.94	0.9208	0.9515	0.9170	0.9104	1.48	70.69	1.59	1.46
65010	64	64	64	68	33.28	33.08	33.10	27.38	0.9115	0.9151	0.8887	0.8997	1.59	69.70	1.77	1.57
140055	80	80	64	72	33.25	29.06	29.52	28.09	0.9524	0.9323	0.8363	0.9457	2.18	80.66	2.10	2.16
148026	78	78	64	61	31.95	32.47	32.01	25.45	0.9317	0.9221	0.8810	0.9124	1.69	70.56	1.72	1.66
144067	80	80	64	82	33.83	33.68	32.01	27.80	0.9169	0.9263	0.8258	0.9024	2.14	82.65	2.25	2.13
238011	37	37	32	25	40.68	40.73	39.50	34.50	0.9574	0.9804	0.9258	0.9124	1.02	21.02	1.15	0.99
361084	62	62	64	60	33.48	33.80	33.70	27.66	0.8909	0.9049	0.8814	0.8988	1.45	77.55	1.65	1.43
169012	88	88	64	90	32.96	32.76	28.07	27.79	0.9052	0.9394	0.8547	0.9000	2.05	82.15	2.00	2.03
Mean value	–	–	–		34.25	33.69	31.36	30.01	0.9246	0.9258	0.8827	0.9088	1.79	66.58	1.79	1.76

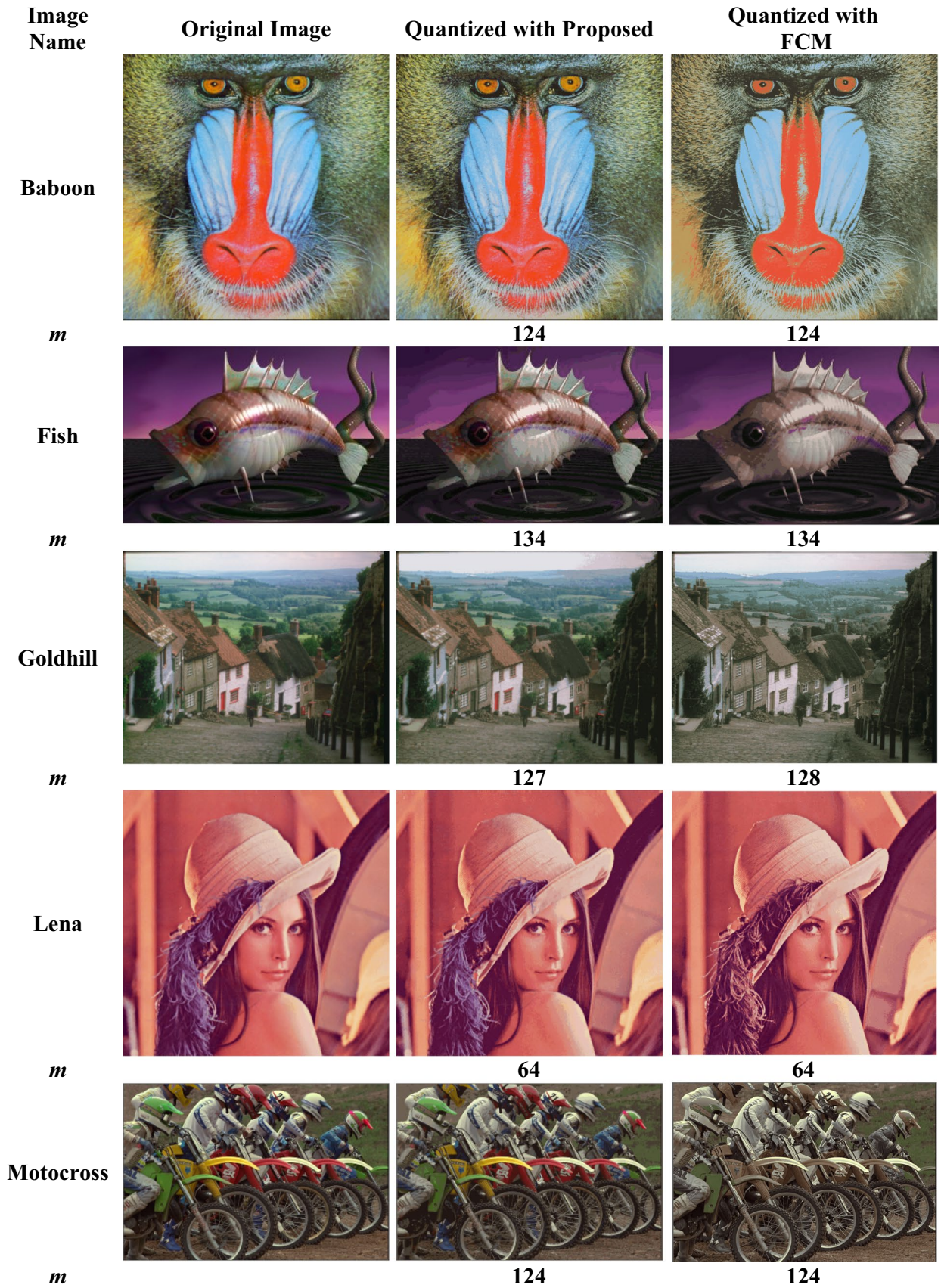

Fig. 6 Common color quantization test images and quantized outputs with FCM and proposed method

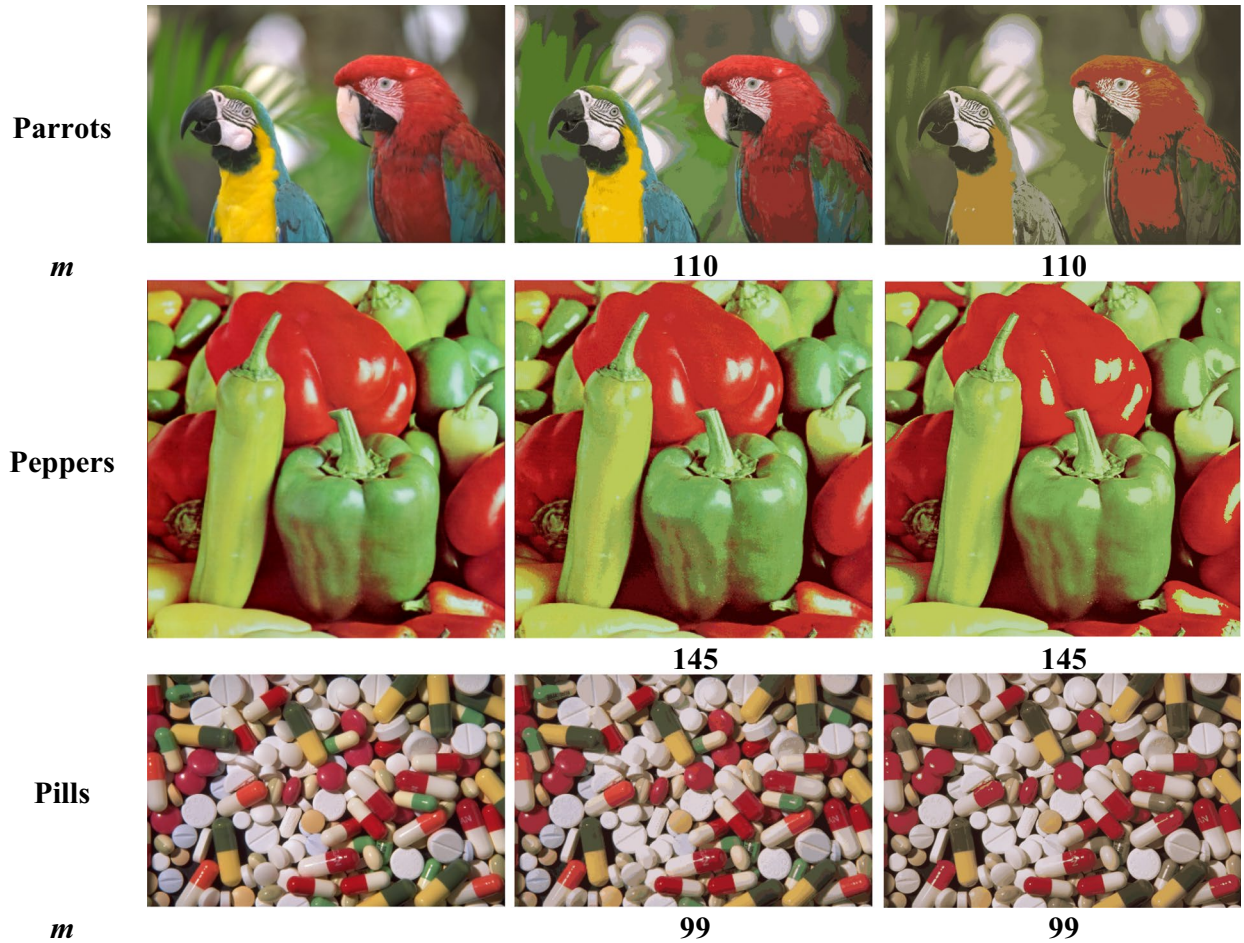

Fig. 6 (continued)

The results for the eight most commonly used images in the comparisons of quantization techniques are shown in Fig. 6. The quantized images obtained with FCM and the proposed method are included along with each original image. Visually, it is seen that the images obtained by both FCM and the proposed method are very close to the original image.

The results obtained with various techniques for eight images in Fig. 6, which are used quite frequently in quantization studies, are given in Tables 7 and 8. According to the PSNR and MSE results, it is seen that FCM, which is one of the conventional methods, is quite successful when low numbers of colors are used. However, when the number of colors increases, the success of the proposed method also increases significantly, and it is more successful than FCM and other techniques in almost all images. The main reason for the increase in the success of the proposed method is that known techniques choose cluster center color values among global values (0–255). On the other hand, the proposed method chooses the color values between the threshold values for the cluster center. As the number of thresholds and, accordingly, the number of colors increases, the probability of selecting more accurate cluster centers also increases. This is one of the important factors that increase the success of the proposed method. It is understood from the PSNR and MSE results that the proposed method is also highly competitive against state-of-the-art methods. Although it is slightly behind the IOKM or ITATCQ-n techniques for some images, its success seems satisfactory in general. Furthermore, despite the high computational costs of IOKM and ITATCQ-n techniques based on optimization and k-means, the most significant advantage of the proposed method is that it is based on a very simple calculation.

Another remarkable feature in the results is that the proposed method has a higher number of colors than that in the MLQ technique, according to the selected threshold values.

Table 7 MSE results of quantization methods

m	Baboon				Fish				Goldhill				Lena			
	16	32	64	128	16	32	64	128	16	32	64	128	16	32	64	128
POP	1859.6	1679.5	849.5	330.7	3378.9	2827.6	482.5	105.2	874.2	576.7	199.3	101.8	415.0	347.2	199.5	84.5
MC	1000.7	643.0	445.6	307.4	550.8	282.3	189.4	121.2	607.1	293.9	188.8	132.3	317.3	214.0	146.1	112.4
MPOP	501.7	453.1	290.4	195.0	295.1	198.4	145.5	66.2	262.2	200.2	140.7	66.7	226.2	194.5	138.9	60.0
OCT	587.1	530.2	306.6	203.6	314.2	218.4	125.1	77.8	313.5	230.3	130.3	79.0	217.1	186.7	110.0	66.0
WAN	717.1	528.3	385.7	266.0	387.5	311.6	209.0	124.5	299.9	229.0	141.2	94.5	251.8	216.5	140.8	87.6
WU	635.7	468.3	288.3	186.5	260.8	187.6	111.6	69.0	256.7	196.0	114.2	71.4	173.7	158.2	99.1	61.7
CC	540.7	473.1	299.7	202.5	208.3	189.8	127.3	82.3	285.4	202.0	134.9	87.9	244.5	189.1	125.5	80.6
RWM	475.4	459.0	301.6	188.1	262.8	176.7	109.0	68.9	235.5	179.8	118.3	71.0	192.7	161.2	94.6	60.1
SAM	631.1	464.9	293.9	188.8	275.9	198.5	120.1	74.0	234.8	179.3	111.2	70.4	183.7	158.0	102.0	65.0
VC	579.5	450.6	273.5	179.9	225.5	168.1	106.5	67.4	219.9	174.8	109.5	68.3	159.8	145.6	91.7	60.7
VCL	547.3	425.6	264.0	173.1	252.7	169.9	102.5	65.1	186.8	169.3	104.3	66.2	155.8	146.3	89.2	59.2
SOM	511.1	433.6	268.9	163.9	206.8	180.4	114.1	60.4	219.7	182.1	104.2	59.5	176.7	140.2	87.4	50.5
MMM	564.7	510.0	368.4	230.4	299.6	223.4	144.2	81.7	264.7	239.9	143.1	95.4	219.1	183.3	114.2	73.5
ITATCQ-n	–	411.0	258.3	163.9	–	156.6	99.4	62,9	–	149.9	86.9	54.7	–	136.37	78.19	49.22
IOKM	–	376.2	237.7	153.7	–	141.8	85.6	52.9	–	143.5	83.6	52.5	–	118.0	72.1	46.3
m	**8**	**22**	**50**	**116**	**14**	**31**	**42**	**121**	**20**	**41**	**59**	**93**	**13**	**24**	**54**	**–**
MLQ	1886.8	343.6	249.2	108.9	272.4	138.5	116.1	50.3	304.0	168.8	110.2	76.7	161.4	96.7	48.5	–
m	**8**	**24**	**54**	**124**	**16**	**31**	**50**	**134**	**18**	**41**	**63**	**128**	**18**	**31**	**64**	**–**
FCM	396.9	209.5	249.6	117.7	321.8	169.2	59.0	50.6	127.2	90.5	97.3	54.2	104.1	68.8	57.1	–
Prop	1885.3	342.1	175.1	93.0	277.3	158.4	102.6	42.1	305.4	168.9	96.8	53.0	100.7	65.9	24.1	–
m	Motocross				Parrots				Peppers				Pills			
	16	32	64	128	16	32	64	128	16	32	64	128	16	32	64	128
POP	1458.9	1288.6	474.3	201.6	4929.4	4086.8	371.7	180.6	1859.6	1389.3	367.7	218.3	1096.9	788.2	222.9	124.0
MC	653.5	437.6	254.0	169.4	671.9	441.0	265.1	153.6	731.3	377.6	238.9	173.8	574.0	324.2	233.8	159.5
MPOP	344.5	287.5	177.9	84.1	458.1	379.8	212.1	104.7	453.4	338.7	204.9	112.1	313.7	277.5	175.2	88.4
OCT	320.4	300.5	158.9	96.2	460.1	342.4	191.2	111.2	394.2	317.4	193.1	113.9	331.0	281.9	159.8	99.1
WAN	460.3	445.6	292.1	168.7	453.5	376.0	233.4	153.4	465.9	348.1	225.7	157.2	333.4	294.9	197.7	133.1
WU	277.0	268.1	147.2	86.7	391.7	299.2	167.3	95.4	359.8	278.9	165.5	102.2	340.8	261.2	150.1	89.5
CC	434.7	335.1	202.0	122.6	481.0	398.8	246.5	148.7	438.6	418.4	256.8	160.7	373.0	285.9	171.7	111.9
RWM	276.3	251.4	150.1	83.7	306.7	296.5	171.0	99.8	324.2	295.6	178.8	107.1	351.1	260.4	149.7	88.8
SAM	269.6	238.1	138.5	81.8	379.4	282.4	157.5	92.4	302.4	275.7	159.2	100.8	321.2	246.2	141.2	85.0
VC	278.3	253.2	144.5	79.6	380.5	290.6	166.4	98.0	323.3	294.8	169.3	108.0	254.8	234.4	146.6	90.2
VCL	264.4	240.6	131.5	77.1	281.8	263.7	157.5	96.6	286.4	261.1	160.3	103.8	319.8	229.8	141.4	85.7
SOM	391.4	301.7	134.7	70.3	337.0	279.4	151.5	82.2	323.3	270.9	160.5	89.9	275.7	226.4	137.8	72.4
MMM	434.9	407.9	276.9	138.2	412.6	352.1	194.8	128.7	490.1	341.5	213.3	136.5	288.2	276.2	174.9	117.2
ITATCQ-n	–	220.1	125.9	69.9	–	266.8	149.3	85.9	–	287.7	170.8	102.9	–	221.1	121.7	68.2
IOK*M*	–	191.5	108.1	62.9	–	241.8	127.1	73.4	–	230.3	134.0	83.7	–	199.1	112.4	66.6
m	**8**	**24**	**54**	**114**	**8**	**22**	**70**	**110**	**8**	**23**	**52**	**134**	**19**	**38**	**58**	**91**
MLQ	774.9	325.7	165.7	110.6	705.9	215.6	110.6	79.0	507.8	253.2	142.8	104.2	282.3	134.0	122.4	86.6
m	**8**	**24**	**62**	**124**	**8**	**22**	**76**	**110**	**8**	**23**	**54**	**145**	**19**	**38**	**63**	**99**
FCM	253.2	185.0	161.5	150.8	530.8	355.2	108.0	180.8	644.5	254.2	149.0	77.9	248.7	86.5	183.4	83.0
Prop	774.2	361.8	108.6	81.1	609.3	288.7	107.6	78.8	507.2	253.5	139.9	73.8	312.6	157.6	110.6	76.8

The *m* values for MLP and the proposed method are shown in bold

Table 8 PSNR results of quantization methods

m	Baboon				Fish				Goldhill				Lena			
	16	32	64	128	16	32	64	128	16	32	64	128	16	32	64	128
POP	15.44	15.88	18.84	22.94	12.84	13.62	21.30	27.91	18.71	20.52	25.14	28.05	21.95	22.73	25.13	28.86
MC	18.13	20.05	21.64	23.25	20.72	23.62	25.36	27.30	20.30	23.45	25.37	26.92	23.12	24.83	26.48	27.62
MPOP	21.13	21.57	23.50	25.23	23.43	25.16	26.50	29.92	23.94	25.12	26.65	29.89	24.59	25.24	26.70	30.35
OCT	20.44	20.89	23.27	25.04	23.16	24.74	27.16	29.22	23.17	24.51	26.98	29.15	24.76	25.42	27.72	29.94
WAN	19.57	20.90	22.27	23.88	22.25	23.19	24.93	27.18	23.36	24.53	26.63	28.38	24.12	24.78	26.64	28.71
WU	20.10	21.43	23.53	25.42	23.97	25.40	27.65	29.74	24.04	25.21	27.55	29.59	25.73	26.14	28.17	30.23
CC	20.80	21.38	23.36	25.07	24.94	25.35	27.08	28.98	23.58	25.08	26.83	28.69	24.25	25.36	27.14	29.07
RWM	21.36	21.51	23.34	25.39	23.93	25.66	27.76	29.75	24.41	25.58	27.40	29.62	25.28	26.06	28.37	30.34
SAM	20.13	21.46	23.45	25.37	23.72	25.15	27.34	29.44	24.42	25.60	27.67	29.66	25.49	26.14	28.04	30.00
VC	20.50	21.59	23.76	25.58	24.60	25.88	27.86	29.84	24.71	25.71	27.74	29.79	26.09	26.50	28.51	30.30
VCL	20.75	21.84	23.91	25.75	24.10	25.83	28.02	29.99	25.42	25.84	27.95	29.92	26.20	26.48	28.63	30.41
SOM	21.05	21.76	23.83	25.99	24.98	25.57	27.56	30.32	24.71	25.53	27.95	30.39	25.66	26.66	28.72	31.10
MMM	20.61	21.06	22.47	24.51	23.36	24.64	26.54	29.01	23.90	24.33	26.57	28.34	24.72	25.50	27.55	29.47
ITATCQ-n	–	21.99	24.01	25.99	–	26.18	28.16	30.14	–	26.37	28.74	30.75	–	26.78	29.20	31.21
IOKM	–	22.38	24.37	26.26	–	26.61	28.81	30.90	–	26.56	28.91	30.93	–	27.41	29.55	31.47
m	**8**	**22**	**50**	**116**	**14**	**31**	**42**	**121**	**20**	**41**	**59**	**93**	**13**	**24**	**54**	**–**
MLQ	15.37	22.77	24.16	27.76	23.78	**26.72**	27.48	31.11	23.30	25.86	27.71	29.29	26.05	28.28	31.27	–
m	**8**	**24**	**54**	**124**	**16**	**31**	**50**	**134**	**18**	**41**	**63**	**128**	**18**	**31**	**64**	**–**
FCM	22.14	24.92	24.16	27.42	23.06	25.85	30.42	31.09	27.09	28.56	28.25	30.79	27.96	29.76	30.57	–
Prop	15.38	22.79	25.70	28.45	23.70	26.13	28.02	31.89	23.28	25.86	28.27	30.89	28.10	29.94	34.31	–
m	Motocross				Parrots				Peppers				Pills			
	16	32	64	128	16	32	64	128	16	32	64	128	16	32	64	128
POP	16.49	17.03	21.37	25.09	11.20	12.02	22.43	25.56	15.44	16.70	22.48	24.74	17.73	19.16	24.65	27.20
MC	19.98	21.72	24.08	25.84	19.86	21.69	23.90	26.27	19.49	22.36	24.35	25.73	20.54	23.02	24.44	26.10
MPOP	22.76	23.54	25.63	28.88	21.52	22.34	24.87	27.93	21.57	22.83	25.02	27.63	23.17	23.70	25.70	28.67
OCT	23.07	23.35	26.12	28.30	21.50	22.79	25.32	27.67	22.17	23.11	25.27	27.57	22.93	23.63	26.10	28.17
WAN	21.50	21.64	23.48	25.86	21.56	22.38	24.45	26.27	21.45	22.71	24.60	26.17	22.90	23.43	25.17	26.89
WU	23.71	23.85	26.45	28.75	22.20	23.37	25.90	28.34	22.57	23.68	25.94	28.04	22.81	23.96	26.37	28.61
CC	21.75	22.88	25.08	27.25	21.31	22.12	24.21	26.41	21.71	21.91	24.03	26.07	22.41	23.57	25.78	27.64
RWM	23.72	24.13	26.37	28.90	23.26	23.41	25.80	28.14	23.02	23.42	25.61	27.83	22.68	23.97	26.38	28.65
SAM	23.82	24.36	26.72	29.00	22.34	23.62	26.16	28.47	23.33	23.73	26.11	28.10	23.06	24.22	26.63	28.84
VC	23.69	24.10	26.53	29.12	22.33	23.50	25.92	28.22	23.03	23.44	25.84	27.80	24.07	24.43	26.47	28.58
VCL	23.91	24.32	26.94	29.26	23.63	23.92	26.16	28.28	23.56	23.96	26.08	27.97	23.08	24.52	26.63	28.80
SOM	22.20	23.34	26.84	29.66	22.85	23.67	26.33	28.98	23.03	23.80	26.08	28.59	23.73	24.58	26.74	29.53
MMM	21.75	22.03	23.71	26.73	21.98	22.66	25.23	27.04	21.23	22.80	24.84	26.78	23.53	23.72	25.70	27.44
ITATCQ-n	–	24.70	27.13	29.69	–	23.87	26.39	28.79	–	23.54	25.81	28.01	–	24.68	27.28	29.79
IOKM	–	25.31	27.79	30.14	–	24.30	27.09	29.47	–	24.51	26.86	28.90	–	25.14	27.62	29.90
m	**8**	**24**	**54**	**114**	**8**	**22**	**70**	**110**	**8**	**23**	**52**	**134**	**19**	**38**	**58**	**91**
MLQ	19.24	23.00	25.94	27.69	19.64	24.79	27.69	29.15	21.07	24.10	26.58	27.95	23.62	26.86	27.25	28.75
m	**8**	**24**	**62**	**124**	**8**	**22**	**76**	**110**	**8**	**23**	**54**	**145**	**19**	**38**	**63**	**99**
FCM	24.10	25.46	26.05	26.35	20.88	22.63	27.80	25.56	20.04	24.08	26.40	29.22	24.17	28.76	25.50	28.94
Prop	19.24	22.55	**27.77**	29.04	20.28	23.53	27.81	29.16	21.08	24.09	26.67	29.45	23.18	26.15	27.69	29.28

The m values for MLP and the proposed method are shown in bold

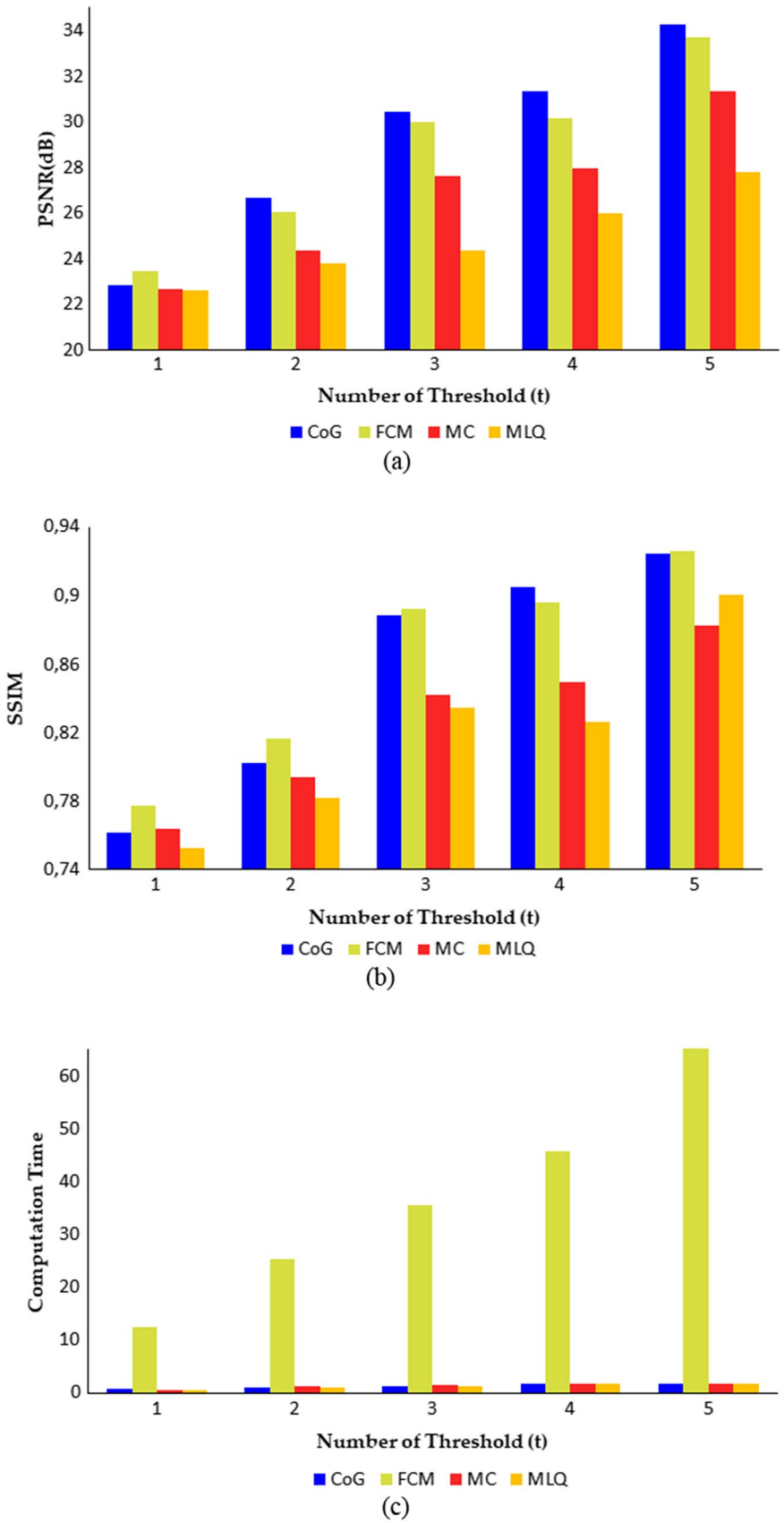

Fig. 7 Average performance of algorithms with different numbers of color on BSDS images: **a** PSNR, **b** SSIM, and **c** computation time

The most important difference from the MLQ technique, which is subjected to thresholding with the help of CoG with a similar approach, is that the proposed method assigns the pixels to the clusters closest to their centers, instead of assigning them directly to the remaining clusters between the thresholds. Thus, the number of empty clusters decreases, and the number of colors that are determined increases. Therefore, the quantization success of the proposed method also increases.

In general, the highest results are obtained by the FCM and the proposed method. In Table 5 where five thresholds are used, and the number of colors is higher, it is seen that the success of the proposed method increased. Another important advantage of the proposed method, which achieves very competitive results against FCM, is its computation time. The proposed method is about 37 times faster than FCM on average. The fact that the proposed method is so fast is based on the fact that the threshold values can be calculated very simply with CoG. Although the MLQ technique based on a similar calculation is slightly faster, the quantization success of the proposed method is higher.

As in the Lena image, although the number of thresholds increases due to the quantization process performed with the proposed method, there is either no change in the number of colors or the number of colors increases slightly. Though it is an advantage that the number of colors does not increase by a large amount, it still turns into a disadvantage in cases where the image quality needs to be improved by increasing the number of colors. Therefore, for the future, a new quantization study in which the minimum color value parameter is used as well as the threshold number in the selection of cluster centers is planned. Additionally, research will be performed to investigate the effects of the success of the proposed method, especially on the success of CBIR and segmentation techniques.

Figure 7 visualizes the performance of the algorithms on various color numbers. The x-axes of the plots are the threshold numbers in the proposed method. Since there are no threshold values for the FCM and MC approaches, the results are obtained with the threshold numbers of the developed technique and the color numbers that are obtained. Thus, it is ensured that the comparisons are appropriate. The average PSNR, SSIM, and computational costs of the color numbers corresponding to the threshold numbers are highlighted. Figure 7a, b clearly shows that the algorithms produce superior results as the number of colors increases. However, the success of the proposed method becomes more evident when the number of colors increases. While the proposed method is insufficient for a single threshold number, it outperforms FCM in the PSNR results when the threshold number increases. When three or more thresholds are used ($t > 2$), the SSIM results of the FCM and the proposed method are quite similar. The computation times in Fig. 7c also prove that the proposed method is quite fast. Especially with the increase in the number of colors, the computational cost of FCM increases tremendously compared to the computational costs of MC, MLQ, and the proposed method.

5 Conclusion

In this paper, a novel color quantization algorithm is proposed. The proposed color reduction technique is based on multi-level thresholding on the histogram. The color space is clustered by thresholding each color component in the RGB color space. A unique color palette is generated for each image by detecting cluster centers based on threshold values. Color quantization is achieved based on the distances of the pixels from the cluster centers. The pixels assigned to the clusters and the number of colors are determined automatically. In many methods, the number of colors to be reduced is a parameter that must be determined by the user. Determining the initial values is a significant problem. The proposed method automatically estimates the number of colors. Moreover, unlike traditional methods, it offers a solution to the initial value problem with threshold values. To evaluate the performance of the method, images are randomly selected from BSDS and tested based on different evaluation criteria. The visual and numerical results show that the proposed method yields satisfactory color quantization outputs.

Appendix

See Figs. 8, 9, 10, 11, 12, 13, 14, 15, 16 and 17.

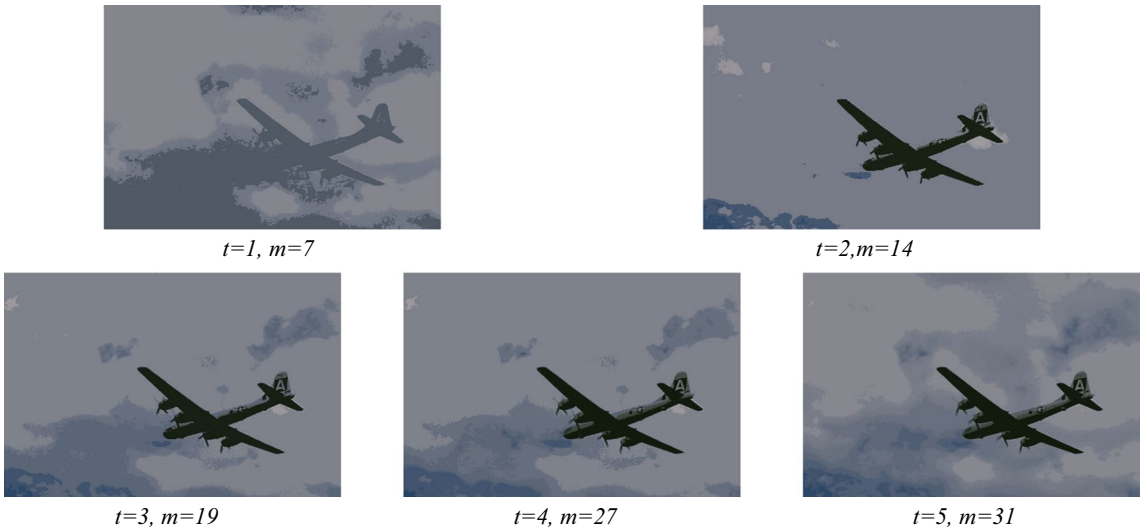

Fig. 8 Quantized 3096 image using CoG

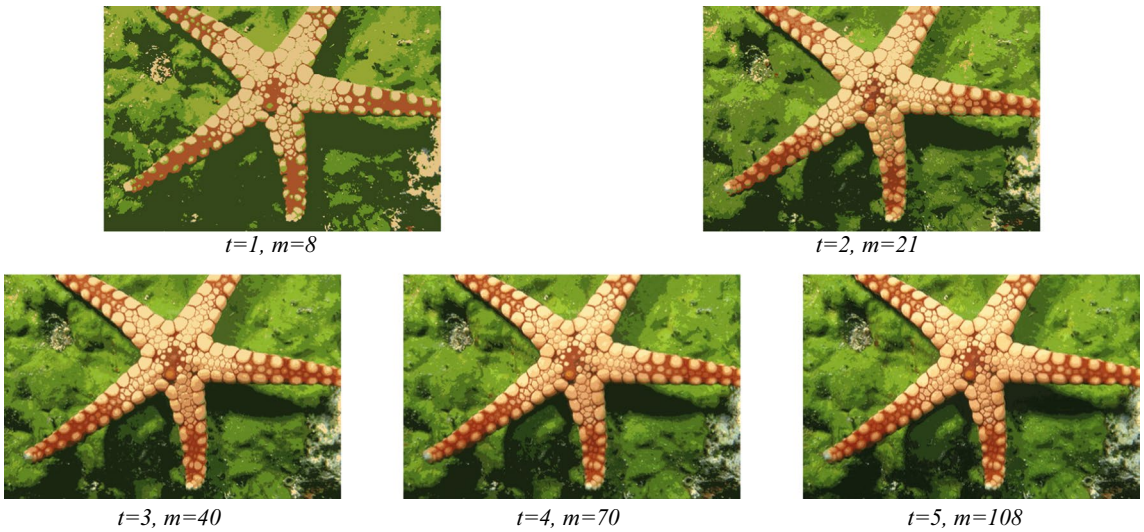

Fig. 9 Quantized 12003 image using CoG

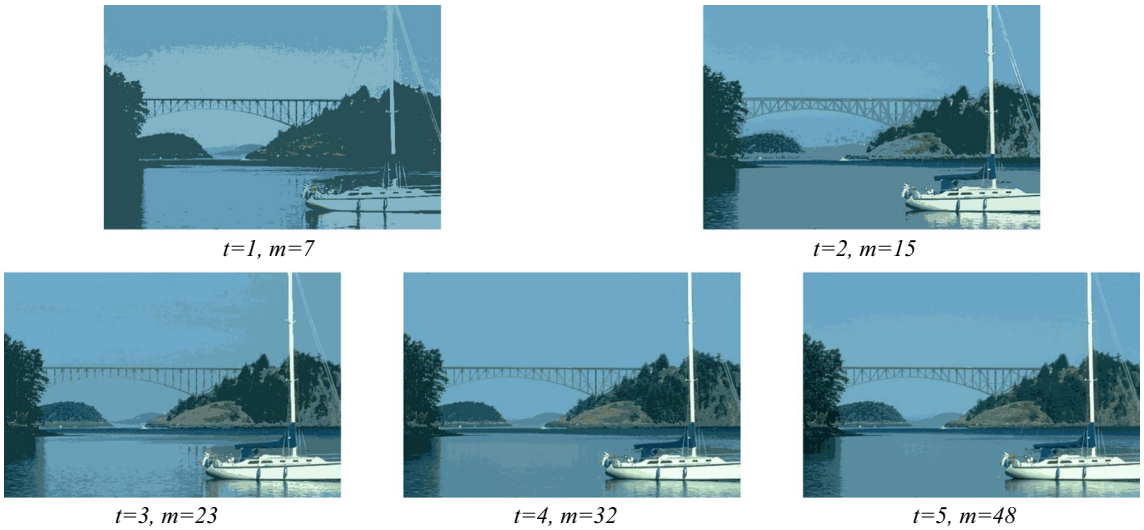

Fig. 10 Quantized 22090 image using CoG

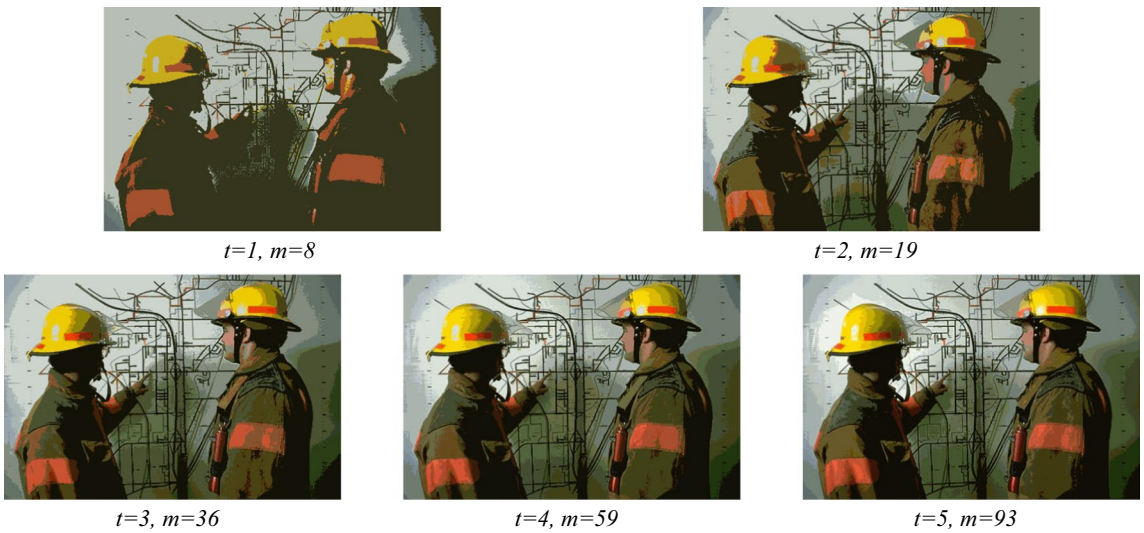

Fig. 11 Quantized 23084 image using CoG

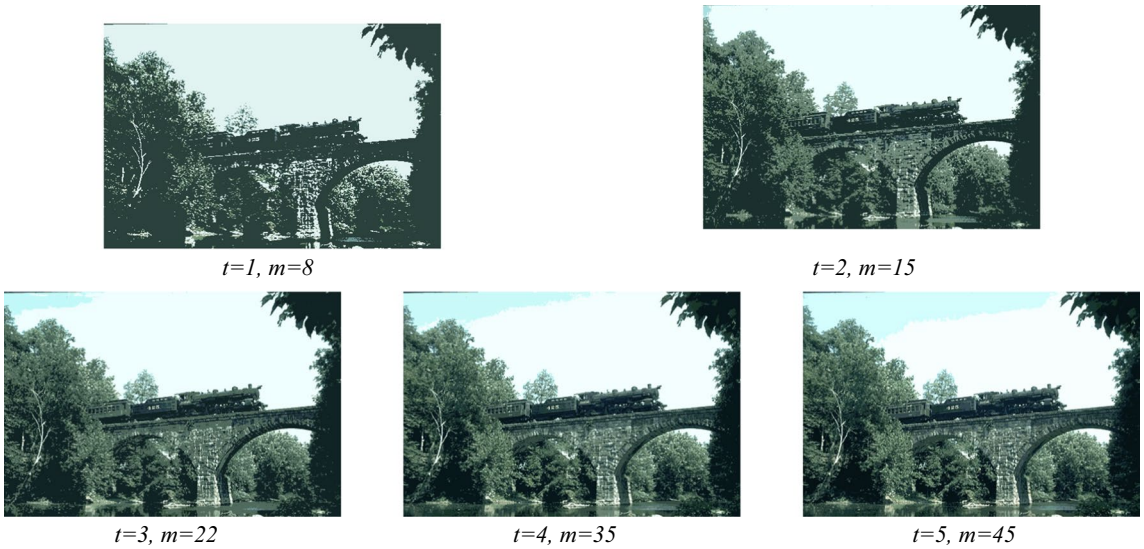

Fig. 12 Quantized 351093 image using CoG

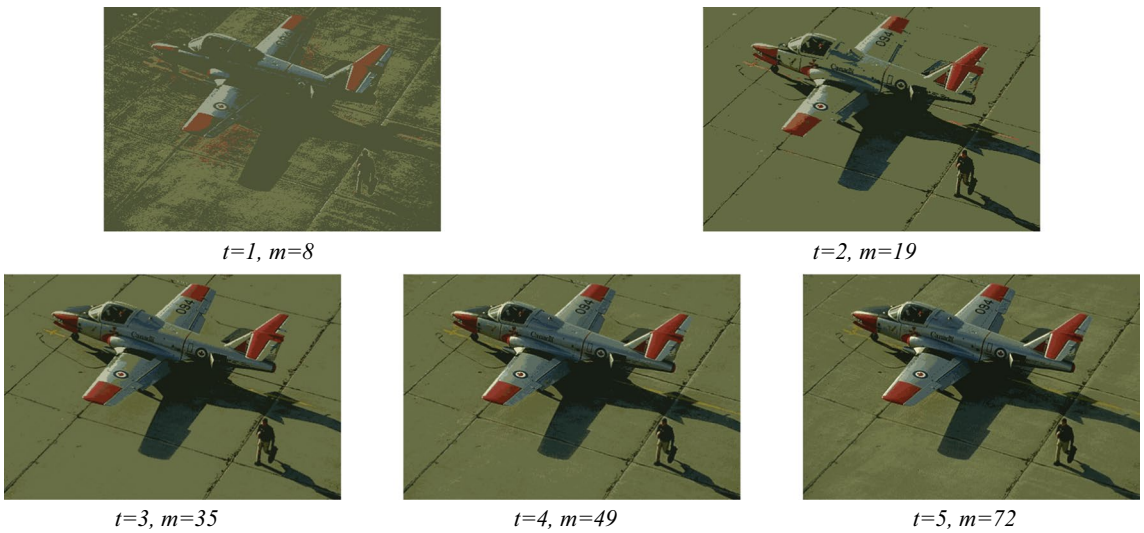

Fig. 13 Quantized 37073 image using CoG

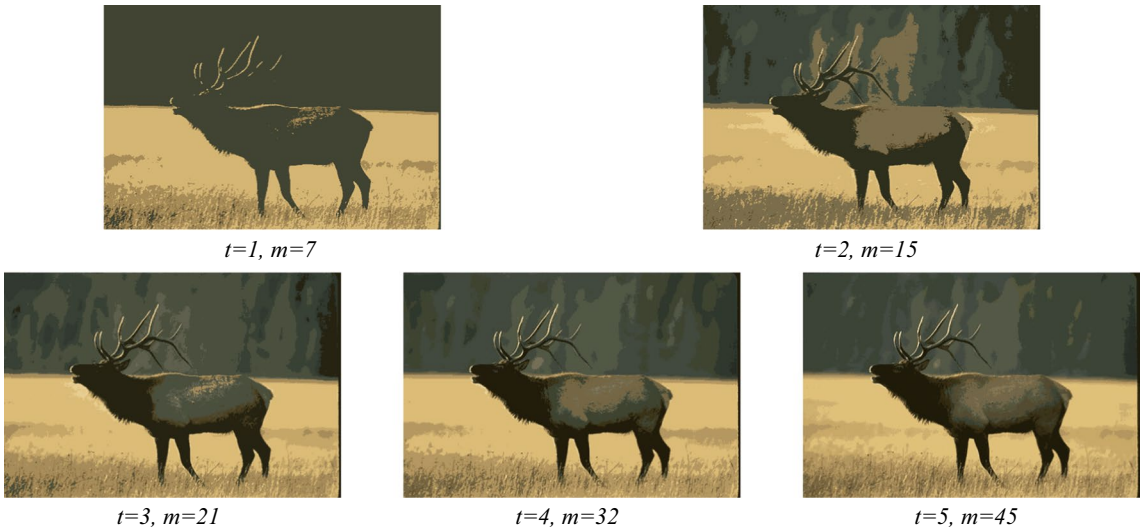

Fig. 14 Quantized 41004 image using CoG

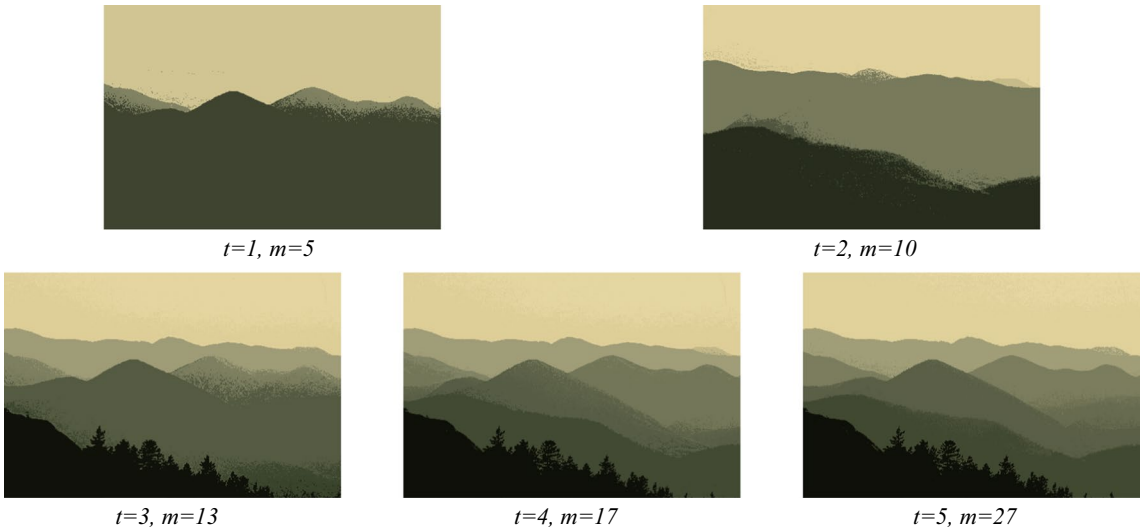

Fig. 15 Quantized 55067 image using CoG

Fig. 16 Quantized 124084 image using CoG

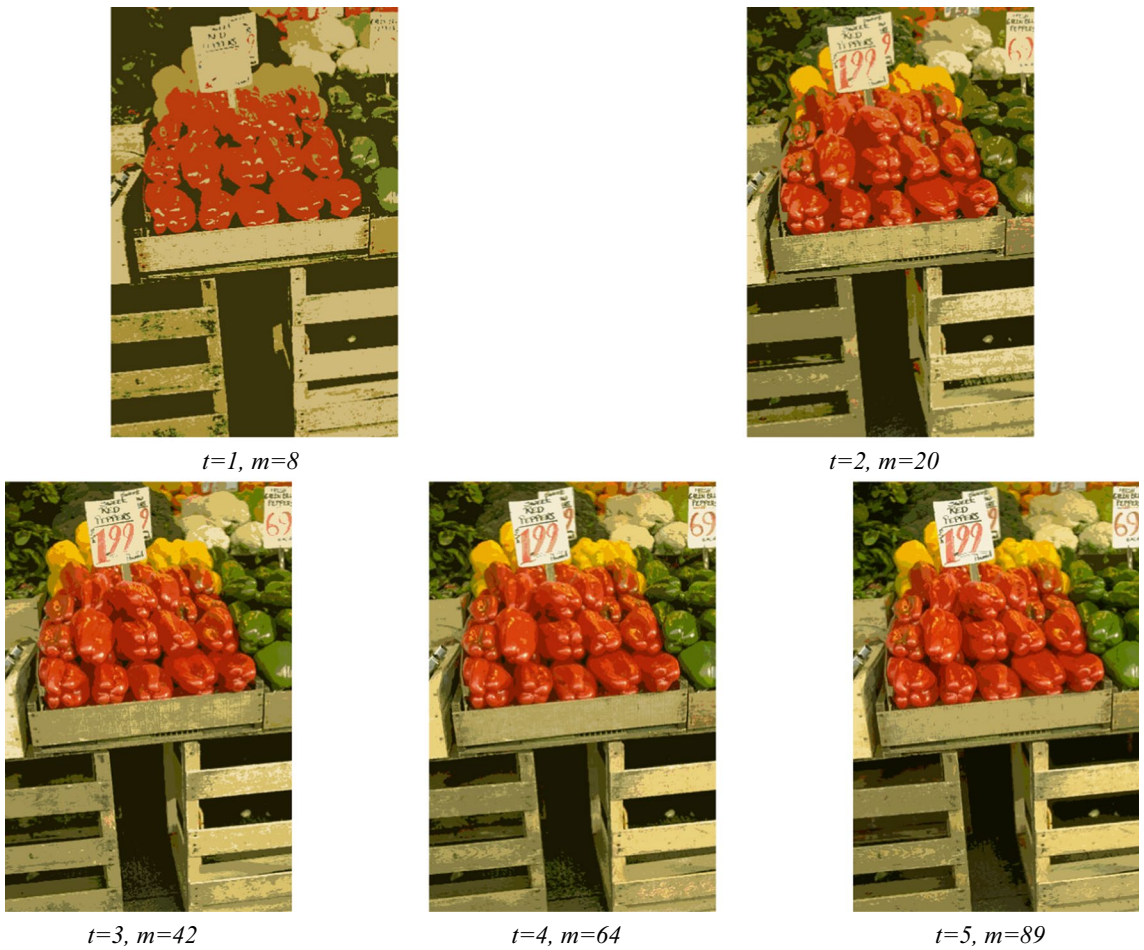

Fig. 17 Quantized 12003 image using CoG

Acknowledgements The authors would like to thank the anonymous reviewers and editor for their insightful comments.

Author contributions MK conceives the idea and writes the draft of the paper, implements the algorithm, and performs analysis. MOI writes the revision of the article, analyzes the algorithm results in detail, and obtains the results for comparison. All authors read and approved the final manuscript.

Funding This article is not supported by any funding.

Availability of data and materials All data have appeared in this paper.

Declarations

Conflicts of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Consent for publication All authors appeared in this paper agreed to publication in this journal.

References

1. Lo, K.C., Chan, Y.H., Yu, M.P.: Colour quantization by three-dimensional frequency diffusion. Pattern Recognit. Lett. **24**(14), 2325–2334 (2003)
2. Lei, M.Y., Zhou, Y.Q., Luo, Q.F.: Color image quantization using flower pollination algorithm. Multim. Tools Appl. **79**(43–44), 32151–32168 (2020)
3. Perez-Delgado, M.L.: Color quantization with Particle swarm optimization and artificial ants. Soft. Comput. **24**(6), 4545–4573 (2020)
4. Yeung, C.W., et al.: Restoration of half-toned color-quantized images using particle swarm optimization with wavelet mutation. In: 2008 IEEE Region 10 Conference: Tencon 2008, vol. 1–4, p. 1844 (2008)
5. Ahmad, J., et al.: Saliency-weighted graphs for efficient visual content description and their applications in real-time image retrieval systems. J. Real Time Image Proc. **13**(3), 431–447 (2017)
6. Qin, X.H., et al.: A novel steganography for spatial color images based on pixel vector cost. IEEE Access **7**, 8834–8846 (2019)
7. Ozturk, C., Hancer, E., Karaboga, D.: Color image quantization: a short review and an application with artificial bee colony algorithm. Informatica **25**(3), 485–503 (2014)
8. Heckbert, P.: Color image quantization for frame buffer display. ACM Siggraph Comput. Graphics **16**(3), 10 (1982)
9. Joy, G., Xiang, Z.: Center-cut for color-image quantization. Vis. Comput. **10**(1), 62–66 (1993)
10. Gervautz, M., Purgathofer, W.: A simple method for color quantization: octree quantization. In: In New Trends in Computer Graphics. Springer (1988)
11. Orchard, M.T., Bouman, C.A.: Color quantization of images. IEEE Trans. Signal Process. **39**(12), 2677–2690 (1991)
12. Wu, X.: Efficient statistical computations for optimal color quantization. Graphics gems II, pp. 126–133. Academic Press, Cambridge (1991)
13. Cheng, S.C., Yang, C.K.: A fast and novel technique for color quantization using reduction of color space dimensionality. Pattern Recognit. Lett. **22**(8), 845–856 (2001)
14. Velho, L., Gomes, J., Sobreiro, M.V.R.: Color image quantization by pairwise clustering. In: X Brazilian Symposium on Computer Graphics and Image Processing, Proceedings, pp. 203–210 (1997)
15. Goldberg, N.: Color image quantization for high-resolution graphics display. Image Vis. Comput. **9**(5), 303–312 (1991)
16. Celenk, M.: A color clustering technique for image segmentation. Comput. Vis. Graphics Image Process. **52**(2), 145–170 (1990)
17. Celebi, M.E., Wen, Q., Hwang, S.: An effective real-time color quantization method based on divisive hierarchical clustering. J. Real Time Image Proc. **10**(2), 329–344 (2015)
18. Wan, S.J., Prusinkiewicz, P., Wong, S.K.M.: Variance-based color image quantization for frame buffer display. Color. Res. Appl. **15**(1), 52–58 (1990)
19. Heckbert, P.: Color image quantization for frame buffer display. ACM SIGGRAPH Comput. Graphics **16**(3), 11 (1982)
20. Brun, L., Mokhtari, M.: Two high speed color quantization algorithms. In: Proceedings of the 1st International Conference on Color in Graphics and Image Processing, pp. 116–121 (2000)
21. Yang, C.Y., Lin, J.C.: RWM-cut for color image quantization. Comput. Graph **20**(4), 577–588 (1996)
22. Dekker, A.H.: Kohonen neural networks for optimal color quantization. Netw. Comput. Neural Syst. **5**(3), 351–367 (1994)
23. Xiang, Z.G.: Color image quantization by minimizing the maximum intercluster distance. ACM Trans. Graphics **16**(3), 260–276 (1997)
24. Celebi, M.E.: Improving the performance of k-means for color quantization. Image Vis. Comput. **29**(4), 260–271 (2011)
25. Yue, X.D., et al.: An efficient color quantization based on generic roughness measure. Pattern Recognit. **47**(4), 1777–1789 (2014)
26. Khaled, A., Abdel-Kader, R.F., Yasein, M.S.: A hybrid color image quantization algorithm based on k-means and harmony search algorithms. Appl. Artif. Intell. **30**(4), 331–351 (2016)
27. Hu, Y.C., Su, B.H.: Accelerated k-means clustering algorithm for colour image quantization. Imaging Sci. J. **56**(1), 29–40 (2008)
28. Ozdemir, D., Akarun, L.: A fuzzy algorithm for color quantization of images. Pattern Recognit. **35**(8), 1785–1791 (2002)
29. Wen, Q., Celebi, M.E.: Hard versus fuzzy c-means clustering for color quantization. Eurasip J. Adv. Signal Process. 2011:118, 1–12 (2011)
30. Schaefer, G., Zhou, H.Y.: Fuzzy clustering for colour reduction in images. Telecommun. Syst. **40**(1–2), 17–25 (2009)
31. Perez-Delgado, M.L.: Revisiting the Iterative Ant-tree for color quantization algorithm. J. Vis. Commun. Image Represent. **78**, 103180 (2021). https://doi.org/10.1016/j.jvcir.2021.103180
32. Abernathy, A., Celebi, M.E.: The incremental online k-means clustering algorithm and its application to color quantization. Expert Syst. Appl. **207**, 117927–117938 (2022). https://doi.org/10.1016/j.eswa.2022.117927

33. R. Demirci, et al.: Automatic background extraction for Prostate Biopsy images, In: 16th International Conference on System Theory, Control and Computing (ICSTCC), Sinaia, Romania, pp. 1–6 (2012).
34. Demirci, R., Tanyeri, U.: Anisotropic diffusion filter using Haar wavelet, In: 2012 16th International Conference on System Theory, Control and Computing (ICSTCC), Sinaia, Romania, pp. 1–6 (2012).
35. Kılıçaslan, M., Tanyeri, U., Demirci, R.: Tekrarlı Ortalama Yardımıyla Renk İndirgeme ve Görüntü Erişimi. Düzce Üniversitesi Bilim ve Teknoloji Dergisi **8**(1), 1042–1057 (2020)
36. Zhao, D., et al.: Chaotic random spare ant colony optimization for multi-threshold image segmentation of 2D Kapur entropy. Knowl. Based Syst. **216**, 106510–106570 (2021). https://doi.org/10.1016/j.knosys.2020.106510
37. Mahajan, S., Mittal, N., Pandit, A.K.: Image segmentation using multilevel thresholding based on type II fuzzy entropy and marine predators algorithm. Multim. Tools Appl. **80**(13), 19335–19359 (2021)
38. Farshi, T.R., Demirci, R.: Multilevel image thresholding with multimodal optimization. Multim. Tools Appl. **80**(10), 15273–15289 (2021)
39. RahkarFarshi, T., Demirci, R., Feizi-Derakhshi, M.-R.J.E.: Image clustering with optimization algorithms and color space. Entropy **20**(4), 296 (2018)
40. Demirci, R., Okur, Ü.: Renkli Görüntülerin Ortalama Tabanlı Çok Seviyeli Eşiklenmesi. Düzce Üniversitesi Bilim ve Teknoloji Dergisi **7**(1), 13 (2019)
41. Demirci, R.: Adaptive threshold selection for edge detection in colour images. In: SIU 2010—IEEE 18th Signal Processing and Communications Applications Conference. 2010, IEEE: Turkey, pp. 677–679