Pattern Recognition 47 (2014) 1777–1789

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

An efficient color quantization based on generic roughness measure

X.D. Yue [a,c,*], D.Q. Miao [b], L.B. Cao [c], Q. Wu [d], Y.F. Chen [b]

[a] School of Computer Engineering and Science, Shanghai University, Room 919, Nanchen Road 333, Baoshan District, Shanghai 200444, China
[b] School of Electronics and Information Engineering, Tongji University, Shanghai 201804, China
[c] Advanced Analytics Institute, University of Technology, Sydney, NSW 2007, Australia
[d] School of Computing and Communications, University of Technology, Sydney, NSW 2007, Australia

ARTICLE INFO

Article history:
Received 4 August 2012
Received in revised form
6 October 2013
Accepted 15 November 2013
Available online 25 November 2013

Keywords:
Color quantization
Generic roughness measure
Thresholding
Clustering

ABSTRACT

Color quantization is a process to compress image color space while minimizing visual distortion. The quantization based on preclustering has low computational complexity but cannot guarantee quantization precision. The quantization based on postclustering can produce high quality quantization results. However, it has to traverse image pixels iteratively and suffers heavy computational burden. Its computational complexity was not reduced although the revised versions have improved the precision. In the work of color quantization, balancing quantization quality and quantization complexity is always a challenging point. In this paper, a two-stage quantization framework is proposed to achieve this balance. In the first stage, high-resolution color space is initially compressed to a condensed color space by thresholding roughness indices. Instead of linear compression, we propose generic roughness measure to generate the delicate segmentation of image color. In this way, it causes less distortion to the image. In the second stage, the initially compressed colors are further clustered to a palette using Weighted Rough *K*-means to obtain final quantization results. Our objective is to design a postclustering quantization strategy at the color space level rather than the pixel level. Applying the quantization in the precisely compressed color space, the computational cost is greatly reduced; meanwhile, the quantization quality is maintained. The substantial experimental results validate the high efficiency of the proposed quantization method, which produces high quality color quantization while possessing low computational complexity.

1. Introduction

In a high quality color image, there are millions of different colors. Color quantization is a process that reduces the number of distinct colors in a digital image, usually with the intention that the reconstructed image should be as visually similar as possible to the original image. By reducing the complexity of color space, color quantization will benefit image storage and image transfer on the internet. Moreover, color quantization also simplifies feature spaces, which is helpful for image recognition and retrieval.

A color quantization algorithm generally consists of two parts. The first is color palette design and the second part is pixel mapping. There are two kinds of methods for creating color palette: image-independent methods and image-dependent methods. Image-independent methods determine a generic palette without considering any specific image contents, while image-dependent methods generate palettes based on the color distribution of images. Although it is fast, the image-independent method often produces poor results because of not considering image contents. To maintain a quality of image representation, most of the recent works on color quantization rely on image-dependent methods. For image-dependent methods, there are two different strategies to build up color palette: preclustering and postclustering [1]. The preclustering strategy partitions the original image colors into multiple subspaces based on the statistics of color distribution [7,9,11,19,26,39,47,48]. Because palette construction is an once-off procedure, the preclustering strategy usually has low computational complexity while sacrificing quantization quality in some degree. The postclustering strategy starts color quantization with an initial palette and improves it iteratively [3,13,27,35,36,38,49]. Since this strategy involves an objective function to minimize color distortion through stochastic optimization, it has better quality than preclustering strategy. However, it greatly increases the computational complexity.

From the discussion above, the challenge to color image quantization is to balance the quantization quality and computational complexity. To achieve this balance, the postclustering

* Corresponding author. Permanent address: Room 604, Unit 1, No. 136, Benxi Road, Yangpu District, 200092, Shanghai, China. Tel.: +86 13585866176.
E-mail addresses: yswantfly@gmail.com, yswantfly@hotmail.com (X.D. Yue), miaoduoqian@163.com (D.Q. Miao), longbing.cao@uts.edu.au (L.B. Cao), qiang.wu@uts.edu.au (Q. Wu), april337@163.com (Y.F. Chen).

0031-3203/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.patcog.2013.11.017

quantization methods have been improved in different ways. Puzicha et al. proposed a color quantization method incorporating spatial and contextual information, in which the quantization was performed by an efficient multi-scale procedure to alleviate the computational burden [34]. Mojsilović and Soljanin proposed another quantization approach based on Fibonacci numbers and spiral lattices, in which the sampling scheme was used to generate color palettes [23]. Computational intelligence theories such as neural networks [29], PSO [25], GA [37], SOM [4] and competitive learning [2,45] were also used to optimize the color quantization process. Zhou et al. proposed an algorithm to adjust the color quantization results which tuned the palette by assigning weights to pixel clusters and color distances [50]. For the postclustering quantization based on clustering strategies, the quantization efficiency was improved by reducing the computational cost of pixel clustering. The modifications focused on speeding up clustering process, in the meantime, optimizing the clustering initialization [3,12,27,42]. The above improvements of postclustering methods could produce the higher quality of quantization results and alleviate the computational burden to some degree. However, these methods needed to heavily traverse pixels iteratively thus computational complexity was still high.

To tackle the problem above, we propose a two-stage color quantization framework based on Generic Roughness measure, which is abbreviated as *GR* framework. The basic idea of this framework is to integrate the low complexity of the histogram-based color space segmentation and the high quality of the clustering-based color quantization. First, the original image color is initially compressed to a condensed color space through thresholding color components. It is very important to form precise segmentation of color space to avoid the severe color distortion in final quantization results. However, the traditional histogram-like statistics cannot guarantee the segmentation precision. We propose generic roughness measure for color segmentation in the initial compression stage. Generic roughness can represent the spatial color homogeneity and thus generate the delicate color segmentation results. In the second stage, the initially compressed colors are further merged to a palette using clustering methods. Carrying out clustering in a compressed color space rather than on the pixel level, merging color in the second stage is very fast. Thus the computational cost of the framework mainly depends on the roughness thresholding in the first stage. The efficiency of the framework is analyzed in Section 5.3. Meanwhile, because of the precise segmentation of color space through generic roughness, the proposed framework causes little color distortion in the initial compression stage and guarantees the quantization quality. Therefore, the proposed framework can well balance the quantization quality and computational cost. The contributions of this paper are summarized as follows:

- *Propose a postclustering quantization strategy at color space level*: Common postclustering quantization methods are implemented at the pixel level. The proposed strategy applies the postclustering quantization in a precisely compressed color space. At this level, the quantization efficiency is greatly improved. In the meantime, the quantization quality is maintained.
- *Propose a generic roughness measure for color space segmentation*: Generic roughness measure is the key to precise color space compression. It can tolerate the disturbance of imbalanced color distribution and thus produces the accurate segmentation of image color space.
- *Design an efficient two-stage quantization framework*: In the first stage, a self-adaptive algorithm is designed to threshold roughness on color components to compress color space. In the second stage, the Rough *K*-means algorithm [18,33] is modified by integrating the color weights to merge the compressed colors to obtain the final quantization result.

The rest of this paper is organized as follows: Section 2 reviews the related work and analyzes the existing problems. Section 3 describes the basic framework and workflow of the roughness-based quantization. Section 4 introduces the construction of generic roughness measure. Section 5 presents the specific quantization method which includes the roughness thresholding algorithm and color merging algorithm. In Section 6, the comprehensive experimental results validate the high efficiency of the proposed color quantization framework. The work is concluded in Section 7.

2. Related work

The clustering technique is a key component in postclustering color quantization. To accelerate the clustering for quantization, Celebi proposed an improved *K*-means algorithm which simplified distance calculation and comparison in the clustering procedure through sorting cluster means [3]. Using partition indices, Özdemir and Akarun proposed a variant of fuzzy *C*-means algorithm to reduce the computational cost of the quantization based on soft clustering [27]. The initialization strategies based on color histograms were also utilized to speed up the clustering-based quantization. Tan and Isa presented a strategy for locating initial cluster centers through thresholding the histograms on color components [42]. Hsieh and Fan constructed 3D color histograms to prepartition pixels into bins and then merged colors on the color histograms to obtain the final quantization results [12]. Although these methods reduced the computational cost to some extent, they still needed to traverse the pixel set iteratively and possessed high computational complexity.

Segmentation of color space [5,28] is inevitable to color quantization. Through color space segmentation, the image is partitioned into homogenous regions depending on the color properties of pixels and numerous original colors can be concisely represented using only a small set of compressed colors. Existing segmentation methods for color compression can be roughly classified into the approaches as histogram based [6,24], edge based [43,44], region based [20], clustering based [8], and combination of several techniques [15,42]. As a popular segmentation technique, histogram thresholding has the advantages of low computational complexity and no requirement of prior information. However, since the bins in a histogram only count the number of pixels with the same color, the histogram thresholding usually produces over rough segmentation results. To refine the color segmentation based on histogram thresholding, the traditional histogram was improved based on rough sets.

As a soft computing technique, rough set theory was widely used in the area of image analysis to handle the data uncertainty [10,30–32]. Mohabey and Ray proposed the statistics histon as an approximation of traditional histogram through checking the neighborhood color similarity [22]. Compared with histogram, the segmentation based on histon considered the spatial color correlation. However, it is not robust to the imbalanced color distribution. To tackle this shortage, Mushrif and Ray proposed a roughness measure utilizing the boundary between histogram and histon to represent the color homogeneity [24]. For an RGB image F of size $M \times N$, its basic roughness measure is defined as follows:

$$roughness_i(l) = 1 - \frac{|histogram_i(l)|}{|histon_i(l)|}, \quad 0 \le l \le L,\ i \in \{R, G, B\} \tag{1}$$

$$histon_i(l) = \sum_{m=1}^{M} \sum_{n=1}^{N} (1 + x(m,n)) \cdot \delta(F(m,n,i) - l) \quad (2)$$

$$x(m,n) = \begin{cases} 1, & d_T(m,n) < threshold \\ 0 & \text{otherwise} \end{cases}$$

where L is the number of intensity level, $F(m,n,i)$ is the intensity of pixel $F(m,n)$ on color component i, δ is the impulse function which outputs 1 when the input equals 0, $d_T(m,n)$ is the neighborhood color difference and $x(m,n)$ is the homogeneity indicator. At the pixel level and neighborhood level, histogram and histon present the intensity distributions without and with uncertainty respectively. From the rough sets view, they can be regarded as the lower approximation and the upper approximation of color component respectively. The roughness index computed based on the approximation boundary denotes the homogeneity degree of color distribution at the corresponding intensity. The roughness measure can tolerate the disturbance of imbalanced color distribution for homogeneity detection. However, qualifying the homogeneity into {0, 1} and checking the color difference just in a small eight-pixel neighborhood, the basic roughness measure over focuses on the homogeneity of small regions. The roughness index of a trivial region, even a noisy point, may take over a big homogeneous region.

Fig. 1 illustrates the quantization results based on different strategies. In Fig. 1(b) and (c), the color space segmentation based on histogram and histon tends to merge significant small regions into big ones and thus produces the blurred quantization results. The basic roughness measure induces more delicate quantization results but it is sensitive to the noisy points. In Fig. 1(d), the yellow color hidden in background influences the whole quantization effect. Although the above methods cannot guarantee quantization precision, they are implemented by thresholding the histogram-like statistics and have low computational complexity. In Fig. 1(e), using a proper initialization strategy [48], the quantization based on K-means clustering obtains the most precise result. Suppose N is the pixel number, K is the number of compressed colors and $iter$ is the iteration times of clustering, the quantization based on K-means needs $iter \times K \times N$ distance calculations and comparisons. Because N is always a huge integer, this method suffers from the heavy computational cost caused by the iterative process in K-means. In this paper, we propose a superior quantization framework which improves the roughness measure to produce the accurate initial color segmentation and then clusters the compressed colors to reduce the quantization complexity.

3. Framework

Our objective is to design a hybrid quantization framework to balance the quantization quality and computational complexity. The solution involves two stages. In the first stage, the color space of original image is segmented based on the histogram-like thresholding to form the initial color compression. In the second stage, the initially compressed colors are merged using clustering to form a palette for quantization. The key of this solution is to guarantee the precision of color space segmentation in the initial compression stage. To achieve this, we propose the generic roughness measure which improves color homogeneity representation and is regarded as a quantified and flexible version of the basic roughness measure. Thresholding based on generic roughness measure can form the delicate color space segmentation and finally produce high quality quantization results.

Fig. 2 illustrates the workflow of the quantization framework based on generic roughness measure. First, the matrix of multi-scale color differences is established by filtering the neighborhood color differences with the multi-scale Gaussian kernel. Second, the homogeneity is measured based on the output of the previous step. The color differences are mapped to the homogeneity degree using a smoothed homogeneity function. Compared with the qualitative description of color similarity in the basic roughness measure, this homogeneity representation is more precisely quantified. Third, based on the homogeneity representation, the approximations of color components are constructed and then the roughness indices are computed. In the fourth step, the color space segmentation is carried out by thresholding the roughness indices to initially compress image color. A self-adaptive thresholding strategy is designed to make the segmentation more robust to various color distributions. Finally the initially compressed colors are merged to a palette using the Weighted Rough K-means

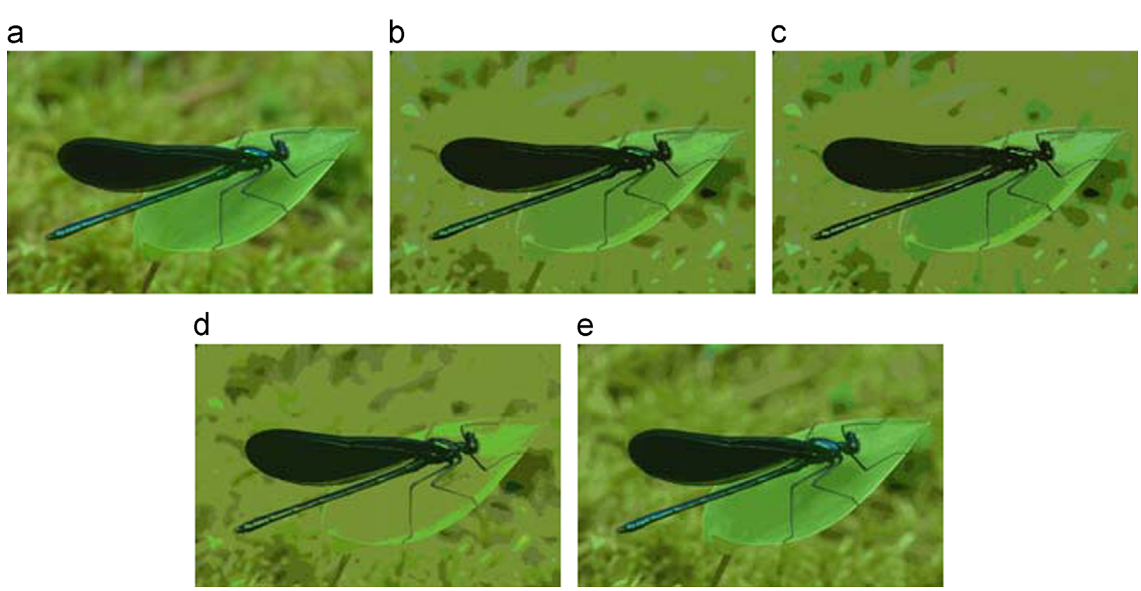

Fig. 1. (a) Image "Dragonfly", (b–d) color quantization based on histogram, histon and basic roughness measure, (e) quantization based on K-means clustering. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

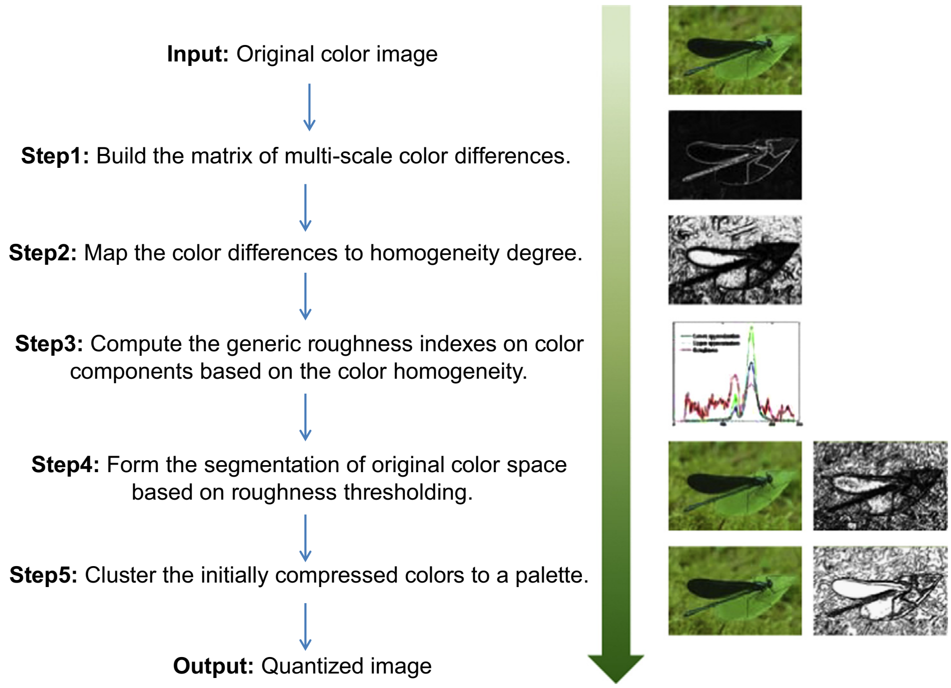

Fig. 2. The workflow of the quantization based on generic roughness measure.

clustering. The weights are determined by the numbers of pixels belonging to the same color segments.

4. Generic roughness measure

Roughness measure is obtained according to the boundary between lower approximation and upper approximation. Histogram represents certain pixels distribution in terms of intensity values of color component. From the view of rough sets, it is considered as the lower approximation of color component. By measuring the spatial color homogeneity, histogram of every color component is extended to form the upper approximation. Upper approximations of all color components represent the color distribution with uncertainty.

4.1. Homogeneity of color distribution

In generic roughness measure, to form the upper approximation, the spatial color homogeneity is obtained based on the multi-scale color differences and homogeneity mapping.

4.2. Multi-scale color differences

To tolerate the interference caused by noisy points and trivial details in homogeneity detection, we utilize the linear scale-space kernel (Gaussian kernel) to carry out the smoothing process. This process forms a multi-scale representation of local color consistency [17].

Definition 1. Given an RGB image F of size $M \times N$, for a pixel $P: F(i,j)$, $1 \le i \le M$, $1 \le j \le N$, the color difference of its neighborhood is defined as

$$D(i,j) = d_P^{3\times3} = \sum_{Q \in NB_P^{3\times3}} d(P,Q) \tag{3}$$

$$d(P,Q) = \sqrt{(R_P - R_Q)^2 + (G_P - G_Q)^2 + (B_P - B_Q)^2} \tag{4}$$

where $d(P,Q)$ is the Euclidean distance between pixel P and Q in RGB space. $NB_P^{3\times3}$ is the set of neighboring pixels of P. The matrix of neighborhood difference of image F is

$$D_F = \{D(i,j) | 1 \le i \le M, 1 \le j \le N\} \tag{5}$$

The above neighborhood difference can be extended to multi-scale color difference through filtering D_F with the linear scale-space kernel.

Definition 2. For $M \times N$ neighborhood difference matrix D, given a scale parameter t and $R \times R$ template that denotes the local area, $3 < R \ll \max\{M,N\}$, the difference matrix under the scale t is generated from the convolution of D with t-scale Gaussian kernel

$$D^t = D * g^t = \{D^t(i,j)\} = \{D(i,j) * g_{i,j}^t\}, \quad 1 \le i \le M, \; 1 \le j \le N \tag{6}$$

where '$*$' is the convolution operator, $g_{i,j}^t$ is the Gaussian kernel covering the $R \times R$ template of center (i,j), and each element of the kernel is computed as $g^t(x,y) = (1/2\pi t)e^{-(x^2+y^2)/2t}$.

The matrix of color difference under multiple scales actually provides the multilevel observation of color homogeneity. When scale $t=0$, D^t degrades to the matrix D of neighborhood difference.

4.3. Color homogeneity measurement

The multi-scale color differences reflect the homogeneity of color distribution. In this paper, homogeneity function is developed to map color difference into homogeneity degree.

Definition 3. Let D^t be the matrix of color difference under scale t, for a pixel $F(m,n)$, the homogeneity degree of the local area centering on $F(m,n)$ is decided by the following homogeneity function:

$$h^t(m,n) = \begin{cases} 1, & D^t(m,n) \le r \\ \dfrac{1}{1 + [0.25 \times (D^t(m,n)/r - 1)]^\beta}, & r \le D^t(m,n) \le kr \\ 0, & kr \le D^t(m,n) \end{cases} \tag{7}$$

Referring to the fuzzy set theory, h^t is a typical membership function to measure the degree of an image area belonging to the concept

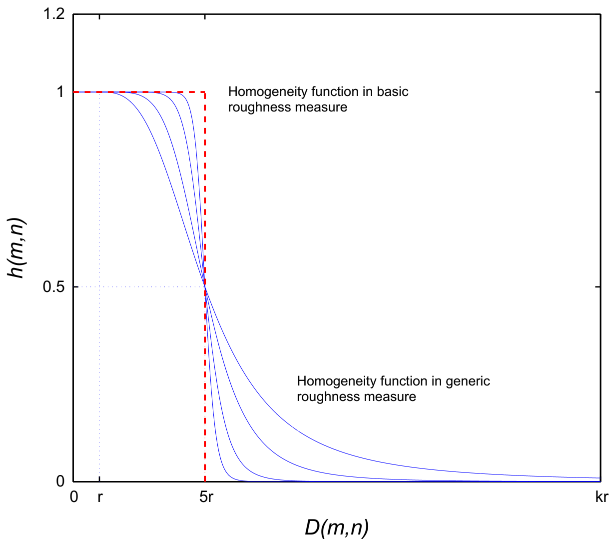

Fig. 3. Homogeneity function in roughness measure.

'homogeneity'. Parameter r denotes the threshold of indistinguishable color difference and β is the function order.

It can be seen that when color difference D^t increases, homogeneity h^t decreases. When $D^t(m,n)$ exceeds the predefined kr, $F(m,n)$ is heterogeneous. Considering the diversity of color distributions, r is assigned as one-fifth of the median value of all distinct color differences, and k is set big enough to precisely quantify the homogeneity. Fig. 3 illustrates the homogeneity function with different orders. When β is big enough, h^t becomes the discrete step function in the basic roughness measure. In this paper, we set $\beta=3$ to obtain the smoothed homogeneity mapping.

4.4. Roughness of color components

Based on the representation of color homogeneity, we can produce the upper approximation of color components and obtain generic roughness measure accordingly.

Definition 4. Given an image F and its homogeneity representation under a scale t, the upper approximation of color component i, $i \in \{R,G,B\}$ is defined as

$$\overline{H}_i^t(l) = \sum_{m=1}^{M} \sum_{n=1}^{N} (1+h^t(m,n)) \cdot \delta(F(m,n,i)=l), \quad 0 \le l \le L \tag{8}$$

where $h^t(m,n)$ is the homogeneity around the position $F(m,n)$, L is the number of intensity level on a given color component.

$\overline{H}_i^t(l)$ at all intensities form the upper approximation of the ith color component. Considering the lower approximation $\underline{H}_i : i \in \{R,G,B\}$, which consists of histograms as mentioned above, it is obvious that $\overline{H}_i^t(l) \ge \underline{H}_i(l)$. The boundary between these two approximations represents the uncertainty of color distribution. According to the approximation boundary, the generic roughness measure under the scale t is generated. When a pixel and its neighboring pixels have a similar color, the gap between upper approximation and lower approximation is large. Thus the colors of the intensity l are considered having the property of roughness. Consequently, on a color image, the roughness index will be small on the heterogeneous regions and large on the homogeneous regions.

Definition 5. Given an RGB image F and a scale t, the generic roughness of color components is defined as

$$r_i^t(l) = 2*(1 - |\underline{H}_i(l)|/|\overline{H}_i^t(l)|), \quad 0 \le l \le L, \ i \in \{R,G,B\} \tag{9}$$

where L is the number of intensity level, $\underline{H}_i(l)$ and $\overline{H}_i^t(l)$ are the lower approximation and the t-scale upper approximation of the lth intensity at color component i respectively. The constant '2' is determined to assure the roughness value staying in interval [0,1].

Compared with the basic single-level roughness measure [24], the proposed measure consists of multilevel approximation and roughness of color components. Thus generic roughness measure is more precise and flexible to represent the intrinsic homogeneity of color regions. This will be further validated in the experiment section. It is also found that when scale $t=0$ and β is big enough (see Eq. (7)), the generic roughness measure will degrade to the basic roughness measure. Moreover, the optimal homogeneity representation relies on the scale of Gaussian kernel. Bigger scales may ruin the significant homogeneous region while smaller scales may not be effective to remove the noise. Thus a reasonable strategy should be designed to determine the proper scales for generic roughness measurement.

4.5. Selection of scale

Based on the scale-space filtering, image contents can be represented on multiple levels. It is natural to investigate the properties of these representations in terms of information measurement [16,21,40,41]. Sporring and Weickert [40,41] proved the monotony and smoothness properties of generalized entropies in linear scale-spaces. Such properties were adopted for scale selection in texture image analysis. A similar strategy is developed in this paper to select the optimal scale. The generalized entropies of multi-scale difference matrix are defined as follows.

Definition 6. By discretizing D^t into L equal width intervals, we have $(p_1^t, p_2^t, \ldots, p_L^t)$ as the probabilistic representation of the color differences under scale t, in which p_i^t is the probability of the ith difference level and $p_i^t > 0$. The generalized entropies of the difference matrix D^t are defined as

$$S_\alpha^t = \frac{1}{1-\alpha} \log \sum_{i=1}^{L} (p_i^t)^\alpha \tag{10}$$

The parameter α is information order. When $\alpha=1$, it is the Shannon entropy

$$S_1^t = -\sum_{i=1}^{L} p_i^t \log p_i^t \tag{11}$$

According to [40,41], we can directly obtain two important mathematical properties of the generalized entropies in scale-spaces. These properties are helpful to analyze the information contained in multi-scale color differences.

Theorem 1. *Given a fixed scale t, the generalized entropies S_α^t decrease with information order α.*

Theorem 2. *Given an information order α, the generalized entropies S_α^t increase with t for $\alpha > 0$, keep constant $\log L$ for $\alpha=0$, and converge to constant $\log L$ for $t \to \infty$.*

Theorems 1 and 2 indicate for any information order $\alpha > 0$, the entropies of multi-scale differences keep rising as the scale t is augmented but increase slow at large scales to converge to the limit. Fig. 4 illustrates the entropies of the color differences of all testing images. When the scale increases within the interval [0, 1], the entropies grow quickly. In this stage, smooth processing removes the trivial details and highlights the dominant color regions, thus leads to fast rising entropies. When the scale is larger than 1, the generalized entropies change smoothly and slowly. Extremely large scales damage

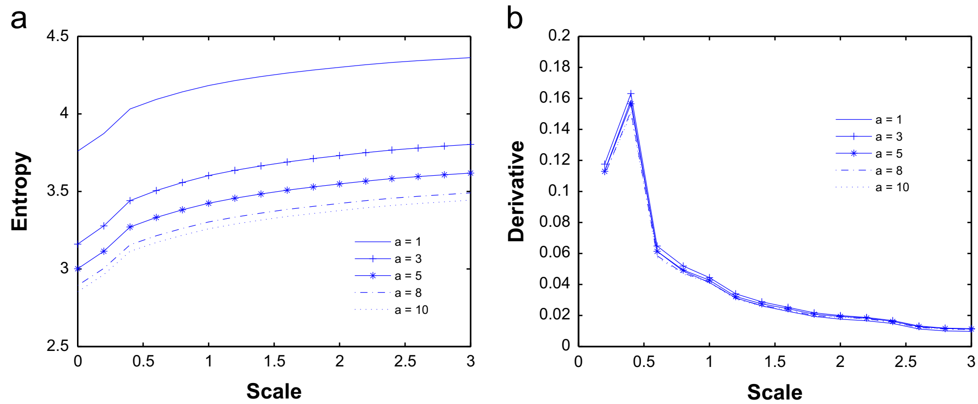

Fig. 4. (a) Average entropies against the varying scales, (b) the first order derivative of the average entropies.

the visual information on both heterogeneous regions and homogeneous regions. In other words, the large scales tend to average color differences and blur image regions. Fast growth of information actually signifies the visual impact aroused by changing distinct regions. Therefore the optimal scales should be selected from the stage where the entropies increase fast (i.e. [0, 1]). In Section 6, the experimental results further indicate the smoothing process under the scales in [0.6, 1] can effectively improve the color quantization quality.

5. Color quantization based on generic roughness measure

5.1. Initial color compression

In the first stage of *GR* quantization framework, the original image color is compressed based on the color space segmentation induced by generic roughness measure.

Algorithm 1. Adaptive roughness thresholding.

Input: roughness indices on each color component $r(l), 0 \le l \le L$
Output: band cuts $\{c_1, c_2, \dots, c_m\}$ on color component
Find all peaks of roughness indices $P = \{P_{l_1}, P_{l_2}, \dots, P_{l_n}\}$;
Compute the threshold of peak heights T_h;
$P_{\max} = \max\{P\}$; $P_{\min} = \min\{P\}$; $\mu = (P_{\max} + P_{\min})/2$;
$\sigma = \sqrt{\sum_{i=1}^{n} (P_{l_i} - \mu)^2 / n}$;
$T_h = \mu - \sigma$;
Select peaks $P_s = \{P_{l_1}, P_{l_2}, \dots, P_{l_s}\} \ge T_h$ from P;
Set T_w=5; merge adjacent peaks in $P_s = \{P_{l_1}, P_{l_2}, \dots, P_{l_s}\}$;
for j=1 **to** $s-1$ **do**
  if $l_{j+1} - l_j <= T_w$ **then**
    if $l_j = \arg\min_l \{P_{l_j}, P_{l_{j+1}}\}$**then** remove P_{l_j} from P_s;
    else remove $P_{l_{j+1}}$ from P_s; set $P_{l_{j+1}} = P_{l_j}$;
  end
end
Form band cuts $\{c_1, c_2, \dots, c_{s+1}\}$, the cut c_k locates between the pair of adjacent peaks
$(P_{l_{k-1}}, P_{l_k}) \in P_s$; set $c_1 = 0$; $c_{s+1} = L$;
for k=1 **to** $s-1$ **do**
  if $c_{k+1} = \arg\min_l \{r(l) | l_k < l < l_{k+1}\}$;
end
Split the over big bands; set $T_s = 2^5$;
for k=2 **to** $s+1$ **do**
  if $c_k - c_{k-1} > T_s$ **then**
    $c = (c_k + c_{k-1})/2$; insert new cut c into $\{c_1, c_2, \dots, c_{s+1}\}$;
  end
end
Output the set of band cuts $\{c_1, c_2, \dots, c_m\}$.

Like the histogram-based color space segmentation, the roughness fluctuation reflects the homogeneity distribution on color components. Thus color components can be divided into bands by separating the relevant roughness indices into segments. Considering the diversity of image color, an adaptive thresholding algorithm is proposed in order to achieve precise color space segmentation. This algorithm utilizes the roughness distribution on intensities to compute the roughness thresholds and locate the band cuts on the color component. The detailed algorithm is illustrated in Algorithm 1.

5.2. Color merging

In the second stage of quantization, the initially compressed color space is further merged to a palette using Weighted Rough *K*-means clustering.

Algorithm 2. Weighted Rough *K*-means clustering.

Input: N segmented colors $\{x_1 \dots x_n \dots x_N\}$ (x_n is an RGB entry), weights of colors $\{q_1 \dots q_n \dots q_N\}$
Output: K clusters of colors and the cluster centers $\{m_1 \dots m_k \dots m_K\}$
Initialize the clustering by Greedy Orthogonal Bi-partitioning;
while *termination criterion is not met* **do**
  Calculate the new means of clusters;

$$m_k = \begin{cases} w_l \sum\limits_{x_n \in \underline{C_k}} q_n x_n \Big/ \sum\limits_{x_n \in \underline{C_k}} q_n + w_b \sum\limits_{x_n \in C_k^B} q_n x_n \Big/ \sum\limits_{x_n \in C_k^B} q_n, & C_k^B \neq \emptyset \\ w_l \sum\limits_{x_n \in \underline{C_k}} q_n x_n \Big/ \sum\limits_{x_n \in \underline{C_k}} q_n, & \textit{otherwise} \end{cases}$$

  $\underline{C_k}$ is the lower approximation of cluster k, $C_k^B = \overline{C_k} - \underline{C_k}$ is the cluster boundary, parameters w_l, w_b define the importance of the lower approximation and boundary;
For each entry x_n, determine its closest mean:

$$h = \arg\min_{1 \le k \le K} \{d(x_n, m_k)\}$$

Assign x_n to the upper approximation of cluster h, $x_n \in \overline{C_h}$ and check other means close to x_n:

$$T = \{t | d(x_n, m_t) - d(x_n, m_h) \le \varepsilon \quad and \quad t \neq h\}$$

if $T \neq \emptyset$ **then**
  assign x_n to the upper approximations of all the clusters in T, $x_n \in \overline{C_t}$, $\forall t \in T$;
else
  assign x_n to the lower approximation of cluster h, $x_n \in \underline{C_h}$;
end

As a soft clustering method, Rough *K*-means [18,33] forms overlap clusters of data. Each cluster has a lower approximation and an upper approximation. Depending on the two layer clustering strategy with both approximations, Rough K-means can produce more stable and precise clustering results than most hard clustering methods. We further modify the Rough K-means to a weighted version for merging color. The weights are computed according to the numbers of pixels belonging to different colors, which can be directly calculated in the first stage. Moreover, the preclustering quantization method Greedy Orthogonal Bi-partitioning [48] is used to initialize the clustering. The output K cluster centers form a palette and the initially compressed color space is quantized based on the newly created palette. That is, the original color values are replaced with the nearest values on the palette.

5.3. Complexity analysis

The quantization process involves color space segmentation and color merging. The complexity of color space segmentation relies on the generic roughness calculation and thresholding. Generating roughness needs to traverse the whole image multiple times. The complexity of constructing the matrixes of color differences and homogeneity is $O((8+R^2)\cdot N+N)$, in which N is the total pixel number and $R^2 \ll N$ denotes the template size. Based on the histograms and homogeneity on intensities, computing roughness indices on all color components needs $O(3L)$ calculations, L is the number of intensity levels. As shown in Algorithm 1, the complexity of thresholding roughness is $O(2L+3t)$, t is the number of candidate peaks. Finally, updating the original image colors according to color bands needs N operations. Therefore the complexity of color space segmentation is $O((10+R^2)\cdot N+5L+3t)$, because $t \ll N$, we have $O((10+R^2)\cdot N+5L)$.

The complexity of color merging relies on the rough clustering. Supposing M the number of colors in the initially compressed color space, in every iteration, constructing the pairwise distance matrix and the approximations of clusters need $O(K\cdot M)$ calculations and $O(K\cdot M)$ comparisons, updating cluster centers needs $O(M+K)$ calculations. Let *iter* be the iteration times, the complexity of clustering the compressed colors using Rough K-means is $O(iter\cdot((2K+1)\cdot M+K))$. Through the roughness-based segmentation, the original huge color space can be reduced to hundreds of colors, in general $M \ll N$. Compared with the complexity $O(iter\cdot K\cdot N)$ of most pixel-level clustering methods, the proposed color space compression not only leads to less computational cost in each iteration but also speeds up clustering convergence. Considering N operations to adjust each pixel's color onto the palette, the complexity of color merging is $O(iter\cdot((2K+1)\cdot M+K)+N)$.

To sum up, the complexity of *GR* quantization framework is approximately $O((11+R^2)\cdot N+iter\cdot(2K+1)\cdot M)$. Because $M \ll N$ and *iter* is a small integer, the computational complexity mainly depends on the color space segmentation based on roughness thresholding rather than the clustering part. Thus this quantization framework can keep the low complexity similar to the histogram-based methods. In the next section, abundant experimental results will further validate the high efficiency of the proposed quantization method.

6. Experimental results

The experiments include the tests of roughness measure, quantization quality and quantization efficiency. In the test of roughness measure, we demonstrate the ability of generic roughness measure to represent color homogeneity. In the tests of quantization quality and efficiency, through comparing with other hybrid quantization methods, we demonstrate that *GR* quantization framework can achieve a good balance between quantization quality and computational complexity. All the testing images are collected from Berkeley color image database.[1]

In the experiments, we use two criteria for the evaluation of quantization quality. First, as the most commonly used evaluation measure, Mean Squared Error (*MSE*) is used to evaluate quantization precision. It is formally defined as

$$MSE(F,\tilde{F}) = \frac{1}{H\times W}\sum_{h=1}^{H}\sum_{w=1}^{W}\|F(h,w)-\tilde{F}(h,w)\|^2 \tag{12}$$

where F and $\tilde{F}$ are respectively the original image and the quantized image in RGB color space, $H\times W$ denotes the image size. *MSE* represents the average color distortion of an image after quantization.

The peak signal-to-noise ratio, often abbreviated *PSNR*, is adopted as the second criterion to evaluate quantization quality. *PSNR* is a popular measure to evaluate the reconstruction quality of image compression codecs and thus is used to evaluate color compression quality. *PSNR* is computed via *MSE*

$$PSNR = 20\times\log_{10}\left(\frac{L}{\sqrt{MSE}}\right) \tag{13}$$

Generally speaking, a higher *PSNR* (or a lower *MSE*) would normally indicate that the color quantization is of higher quality. But it should be noticed that these measures are just the approximation of human perception for evaluating image reconstruction quality. In some cases one reconstruction may appear to be closer to the original than another, even though it has a lower *PSNR* (or a higher *MSE*) [14,46]. This phenomenon was also found in our experiments.

6.1. Evaluation on roughness measure

To demonstrate the superiority of generic roughness measure, we first test the ability of generic roughness measure to represent color homogeneity. Fig. 5 and Table 1 show the color space segmentation results based on histogram, histon [22], basic roughness measure [24] and generic roughness measure (scale $t=0$). Utilizing approximation boundary rather than pixels distribution to represent color homogeneity, both roughness measures can tolerate the interference of imbalanced color distribution and produce more precise segmentation of color space. See Fig. 5 (j) and (n), the roughness measures depict small homogeneous regions well such as the regions of flowerpot edges and rootstalks.

However, limited by the qualitative description of color homogeneity (0 or 1), the basic roughness measure is still not precise enough. The segmentation of sill regions is influenced by the flower color and the sill is labeled over red color. Based on the smoothed homogeneity function and the adaptive thresholding algorithm, generic roughness measure forms the accurate representation of color homogeneous regions. As shown in Fig. 5 and Table 1, compared with other kinds of statistics, generic roughness measure produces delicate initial color space segmentation, which guarantees the quality of final quantization. Fig. 6 indicates the average distortion of color space segmentation of 30 images randomly selected from the testing database. Generic roughness measure owns the highest precision.

Besides the homogeneity representation, we expect to demonstrate the robustness of generic roughness measure. Smoothed with the linear scale-space kernel, the multi-scale color differences weaken the trivial details and thus better reflect salient homogenous regions. Fig. 7 and Table 2 show the quantization results of image 'Old man' under different scales. It is found that the smoothing process of small scales is not sufficient to remove trivial homogeneity. See Fig. 7(b) and (c), influenced by the trivial black color, the collar regions are quantized into over dark color. On the other hand, the smoothing of big scales may ruin distinct homogenous regions and result in

[1] http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds

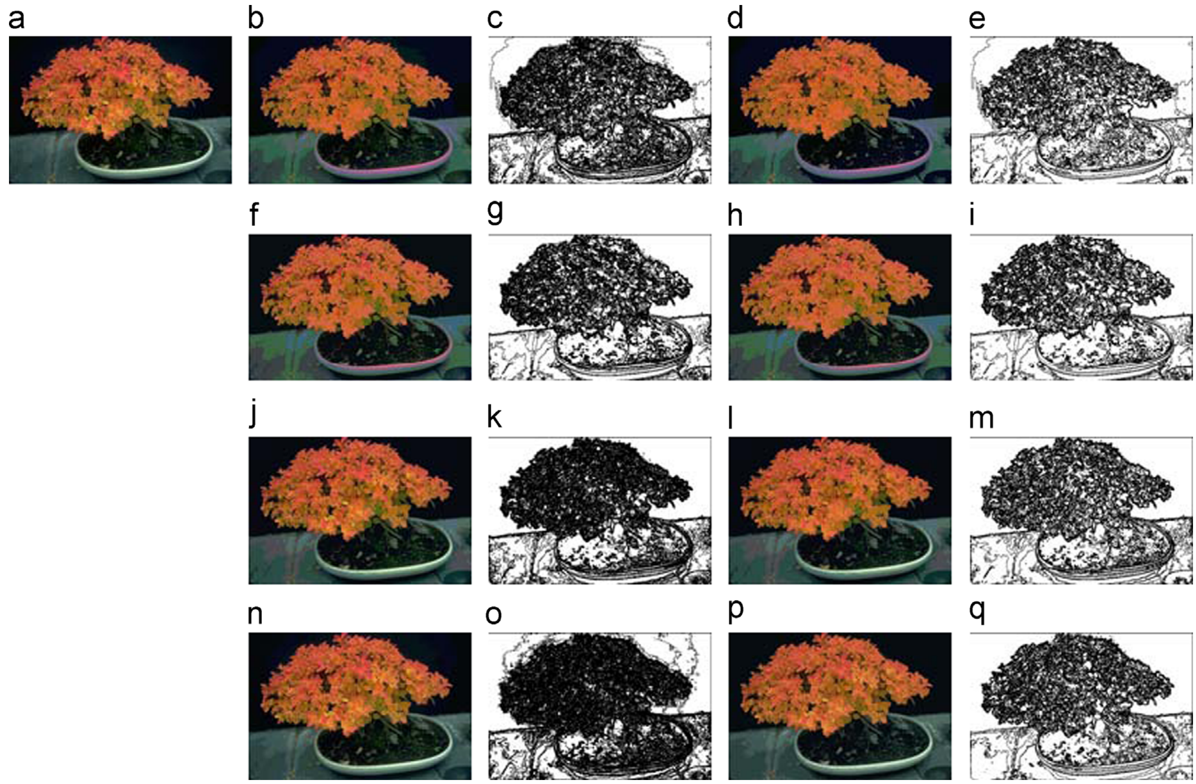

Fig. 5. (a) Image 'Flowers', (b, c) the initial color space segmentation based on histogram, (d, e) color quantization after color merging, (f–i) the results based on histon, (j–m) the results based on basic roughness measure, (n–q) the results based on generic roughness measure.

Table 1
Color space segmentation of image 'Flowers'.

Statistics	Initial color segmentation			Quantization after color merging		
	MSE	PSNR	Color number	MSE	PSNR	Color number
Histogram	482.222	21.298	171	693.22	19.722	30
Histon	586.538	20.448	132	660.22	19.934	33
Roughness	230.785	24.499	307	338.951	22.829	37
Generic roughness	68.005	29.805	836	224.13	24.626	32

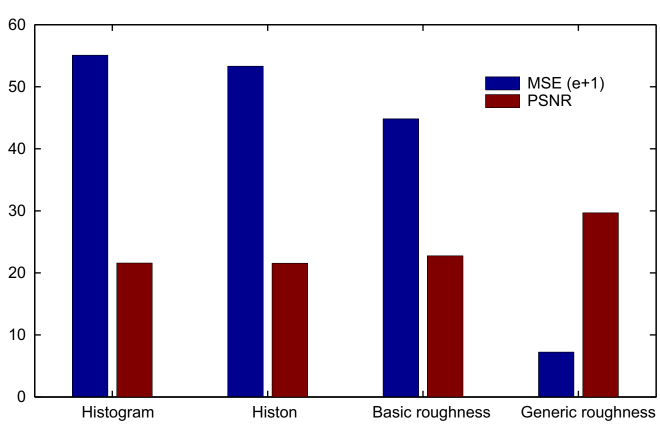

Fig. 6. Distortion of initial color space segmentation.

distorted color quantization. As shown in Fig. 7(d) and (e), some regions of clothes and neck are blurred. Under the scale $t=0.8$, the smoothing process makes color homogeneity more prominent and produces precise quantization results.

The multi-scale smoothing process was also performed on 30 testing color images, the average quantization quality under different scales is illustrated in Fig. 8. Experimental results indicate the smoothing under the scales in [0.6, 1] can enhance the quantization quality of 63% testing images. As shown in Fig. 8, after the initial small scales, quantization quality increases in the scale interval [0.6, 1]. When the scale exceeds 1, the smoothing process erodes region homogeneity and gradually aggravates quantization distortion. Moreover, the smoothing effects are related to image contents. For the images containing primary foreground objects in complex background, the smoothing process is effective to improve quantization quality. In some cases, if the quantization with no smoothing is good enough for human eyes, we can set scale $t=0$ to further speed up the quantization process.

6.2. Evaluation on color quantization

GR quantization framework aims to compress image color precisely and fast. The two-stage framework reduces the computational complexity of quantization but carries a risk of color distortion caused by color space segmentation in the first stage. To evaluate the quantization quality of the proposed method, we compare *GR* framework with different kinds of quantization methods, which include the roughness-based methods i.e. histogram, histon [22], basic roughness measure [24], and two recent hybrid quantization approaches: HTFCM [42] and improved *K*-means (IKM) [3].

Like our method, HTFCM is also a hybrid approach which uses histogram thresholding and fuzzy *C*-means clustering (FCM) to compress image color. The differences are that histogram thresholding is used to initialize the color cluster centers rather than form the initial color space segmentation, and FCM is used to group pixels rather than compressed colors as in *GR* framework. Based on the scheme of clustering initialization and acceleration, IKM implements several fast and exact variants of *K*-means for color quantization. It is concluded

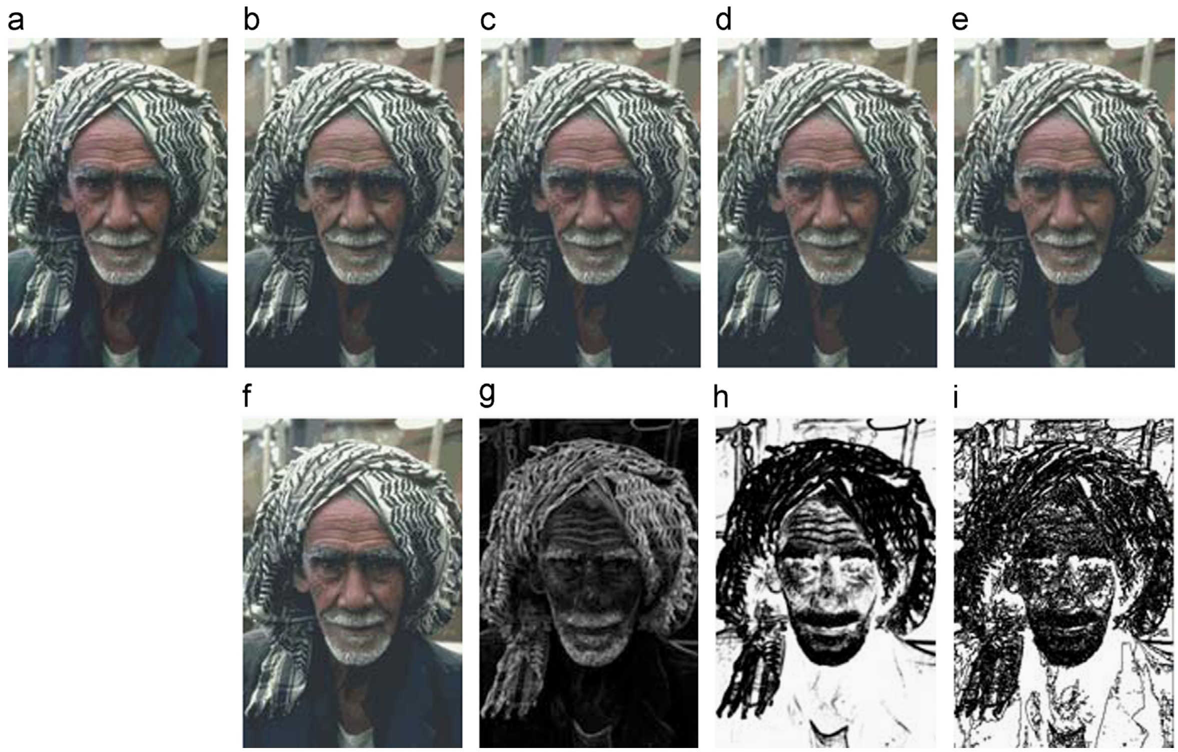

Fig. 7. (a) Image 'Old man', (b–e) the quantization results under the scales t=0, 0.4, 2 and 3, (f) the quantization result under the scale t=0.8, (g–i) the smoothed color differences, homogeneity matrix and homogeneous color regions of (f). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

Table 2
Quantization results of image 'Old man' under different scales.

Scale t	Initial color number	Merged color number	MSE	PSNR
0	324	32	132.029	26.924
0.4	379	32	146.22	26.481
0.8	382	32	126.427	27.112
1	379	32	133.017	26.892
2	374	32	145.536	26.501
3	369	32	160.154	26.085

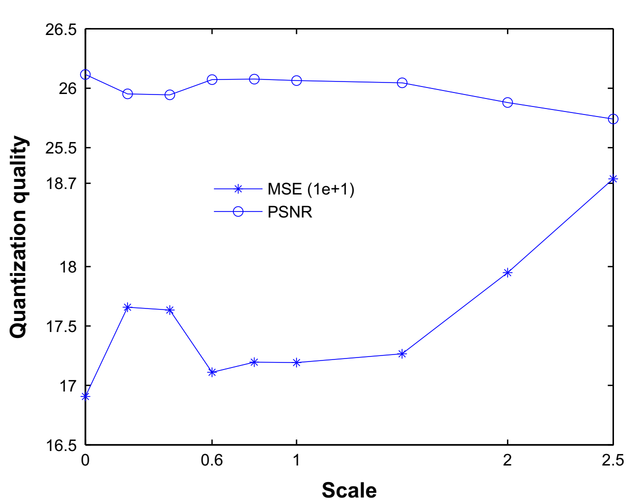

Fig. 8. Average quantization quality under different scales.

that the best one of these variants is the WSM-WU method [3], which uses the strategy of greedy orthogonal bi-partitioning [48] to initialize the clustering centers and accelerates the clustering process with a weighted sort-means algorithm. In the following tests, we adopt WSM-WU as the representative of IKM methods.

Figs. 9, 10 and Table 3 show the color quantization results based on different methods. Compared with histogram and histon, both methods based on roughness measure have better performances. However, the basic roughness measure emphasizes the homogeneity of trivial color regions, and thus leads to the inaccurate color space segmentation. See Fig. 9(d), the leaf regions are labeled greenish yellow. This distortion results from the disturbance of the yellow color hidden in background. Depending on the improved homogeneity representation, generic roughness measure focuses on distinct homogenous regions and leads to more precise quantization than the traditional roughness-based methods.

The experiments indicate that the hybrid quantization methods of HTFCM, IKM and *GR* framework generally perform better than the other methods. As shown in Figs. 9 and 10, these methods can produce the refined quantization results which represent image details well. HTFCM merges color through grouping pixels by distance-based clustering. But this strategy is sensitive to the sizes of pixel clusters. Fixing the cluster number, the cluster centers will shift to large clusters to minimize the sum of distances. This means HTFCM tends to merge a small color region into a bigger one. See Fig. 9(e), although HTFCM can represent the foreground in detail, it quantizes the highlights on dragonfly to grayish green as the background. The similar case is shown in Fig. 10(e). Referring to the introduction of evaluation criteria, even though HTFCM obtains the minimized *MSE*, the visual quality of color quantization results cannot be guaranteed. Moreover, because HTFCM initializes the cluster centers with histograms, if the intensity distribution on color component is very imbalanced, it is difficult to locate the reasonable color clusters. See some results in Table 3, HTFCM initializes too few clusters to keep the original colors in quantization. Compared with HTFCM, *GR* initializes color clusters with homogeneity distribution rather than intensity distribution, and clusters compressed colors rather than pixels for making palette.

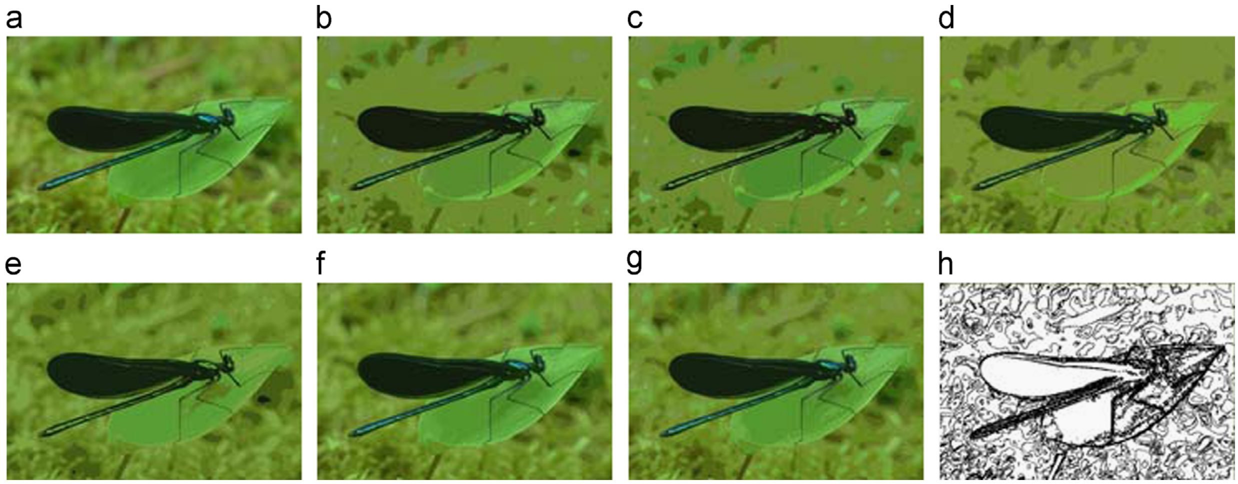

Fig. 9. (a) Image 'Dragonfly', (b–d) quantization based on histogram, histon and basic roughness measure, (e) quantization based on HTFCM, (f) quantization based on IKM, (g) quantization result based on *GR* framework, (h) homogeneous color regions of (g). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

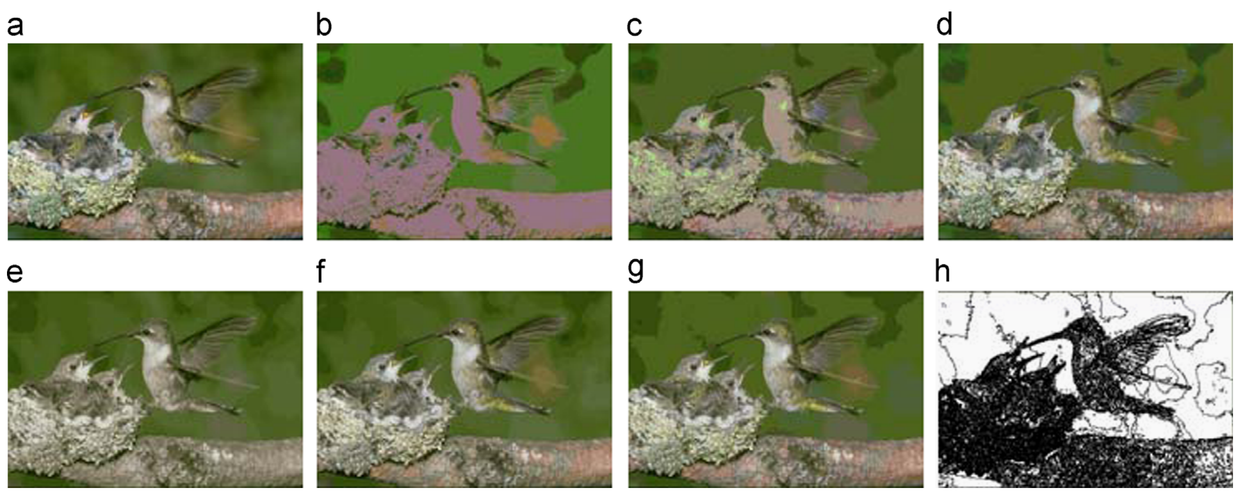

Fig. 10. (a) Image 'Hummers', (b–d) quantization based on histogram, histon and basic roughness measure, (e) quantization based on HTFCM, (f) quantization based on IKM, (g) quantization result based on *GR* framework, (h) homogeneous color regions of (g). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

Table 3
Quantization results based on different methods.

Image	Methods	MSE	PSNR	Color number
Dragonfly	Histogram	408.236	22.022	19
	Histon	504.694	21.101	15
	Roughness	538.57	20.818	13
	HTFCM	327.893	22.973	7
	Improved *K*-means	97.595	28.237	32
	Generic roughness	169.766	25.832	32
Hummers	Histogram	1842.733	15.476	11
	Histon	911.416	18.534	18
	Roughness	355.717	22.62	35
	HTFCM	215.436	24.798	15
	Improved *K*-means	116.203	27.479	32
	Generic roughness	197.782	25.169	32

Thus it can tolerate the disturbance of pixel scales in quantization and produce more precise and stable results.

Using preclustering strategy to generate initial color clusters, the method of IKM overcomes the drawback of clustering-based quantization sensitive to the clustering initialization. Fig. 11 shows the average color quantization quality of all testing images, IKM takes the highest quantization precision. However, even adopting the weighted sort-means algorithm to reduce the computational complexity of clustering, the color quantization based on IKM is still a time-cost task. The objective of *GR* framework is to balance quantization quality and computational complexity. As shown in Fig. 11, *GR* framework achieves the eminent quantization quality close to IKM. Meanwhile, compared with the other hybrid methods involving pixel-level clustering, the computational time of *GR* framework is just linearly dependent on the image size. The efficiency of *GR* framework will be further validated in the following test.

6.3. Evaluation on quantization efficiency

For different color quantization strategies, the histogram-based methods have low computational complexity but usually produce poor results, while the clustering-based methods produce high quality quantization but expend much computational time. Considering the advantages of both strategies, *GR* framework first generates the precise color space segmentation based on roughness thresholding, and then merge the initially compressed colors using Weighted Rough *K*-means. *GR* framework can achieve a good balance between computational efficiency and quantization quality. To demonstrate this, we compare the performance of *GR* framework with the hybrid approaches HTFCM and IKM. All the quantization algorithms are implemented using Matlab 7 in Windows system.

Testing all the images in database (set color number $C=32$), we present the average runtime and clustering iteration times of quantization in Fig. 12 and Table 4. Obviously, *GR* framework has the lowest time cost and iteration times. Compared with the existing postclustering methods, *GR* framework greatly improves the quantization efficiency. Because clustering the limited compressed colors rather than pixels, the computational cost of *GR* framework relies on the initial color space segmentation which just needs to traverse image several times. In general, there are about 60 000 pixels and 20 000–50 000 different colors in each testing image, it is a heavy computational burden to directly cluster these pixels/colors to obtain a precise quantization result. On the other hand, there are just 100–400 different colors in a segmented color space and it is an easy task to merge the compressed colors into a palette. Fig. 12 also presents the variance of runtime and iteration times of quantization. It can be found that the computational costs of HTFCM and IKM have great variation on different images while *GR* framework obtains the stable performance on all testing images. This is because *GR* compresses original color space to a small number of sparse RGB entries through thresholding roughness. The initial color space segmentation makes the clustering converge to stable results fast .

To further validate the efficiency of *GR* framework, we test the quantization methods against multiple palette sizes, i.e. quantized color numbers. Five images are randomly selected to be quantized into 2^k colors respectively, $k=\{3, 4, 5, 6, 7, 8\}$. Fig. 13 and Table 5 present the performances of IKM and *GR* framework. Although sorting cluster centers to reduce the pairwise distance calculations, the computational cost of IKM still rises fast as palette size increases. The increased cluster number requires more distance calculations and comparisons thus slows the clustering convergence. For *GR* framework, its computational cost relies on the initial color space segmentation. Even the augmented color number causes a minor increase of clustering iteration in the color merging stage, *GR* framework can keep runtime stable when palette size changes.

Besides the palette size, we also test the quantization efficiency with image size. We enlarge the selected images multiple times $\alpha^2 \cdot width \cdot height$, $\alpha=\{1.2, 1.4, 1.6, 1.8, 2\}$. The quantization efficiency of the multiple size images are presented in Fig. 14 and Table 6. Because the computational complexity of pixel-level quantization depends on the pixel number N (or the original color number N'), the augmented pixels generally require more distance calculations and iterations for converging clustering.

Table 4
Running time and iteration times of different quantization methods.

Image	HTFCM		Improved *K*-means		Generic roughness	
	Time (s)	Iteration	Time (s)	Iteration	Time (s)	Iteration
Zebra	84.544	92	78.268	114	18.453	9
Bear	40.486	46	77.478	140	18.407	13
Birds	28.147	26	40.019	57	18.593	19
Bridge	70.439	52	186.943	112	19.219	28
Bull	21.159	27	17.675	61	18.438	18
Butterfly	46.037	36	80.739	59	18.578	12
Flowers	64.124	31	303.573	162	18.5	15
Horse	39.7	63	185.68	204	18.344	11
Basket	55.05	35	120.138	107	18.453	13
Lions	19.021	36	27.738	58	18.422	6

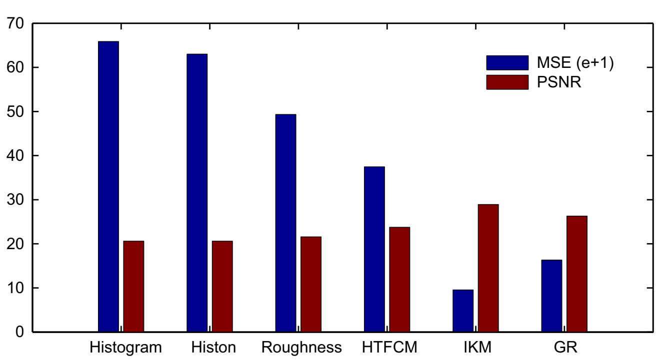

Fig. 11. Average quantization quality based on different methods.

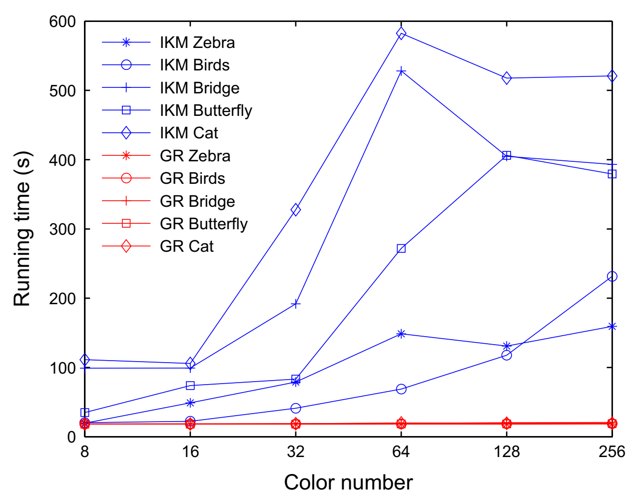

Fig. 13. Running time of quantization methods with different color numbers. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

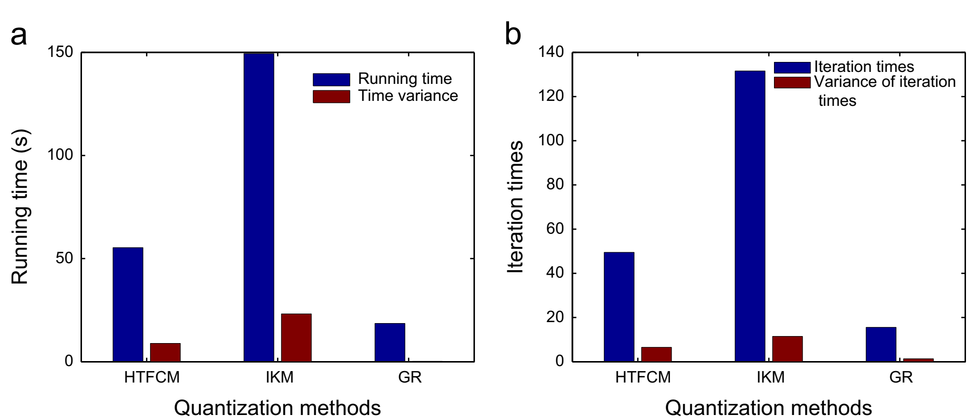

Fig. 12. (a) Running time and time variances of HTFCM, IKM and *GR* framework, (b) clustering iteration times and iteration variances of HTFCM, IKM and *GR* framework.

Table 5
Running time and iteration times with different color numbers.

Image	Color number	Improved *K*-means		Generic roughness	
		Time (s)	Iteration	Time (s)	Iteration
Zebra	8	19.556	60	18.167	8
	16	49.023	101	18.166	11
	32	78.879	114	18.167	9
	64	148.55	144	18.307	9
	128	130.899	89	18.26	8
	256	159.382	78	18.4	6
Birds	8	20.333	51	18.397	12
	16	22.336	44	18.398	10
	32	41.229	57	18.571	19
	64	68.777	70	18.539	12
	128	117.778	90	18.761	11
	256	231.519	131	18.995	9

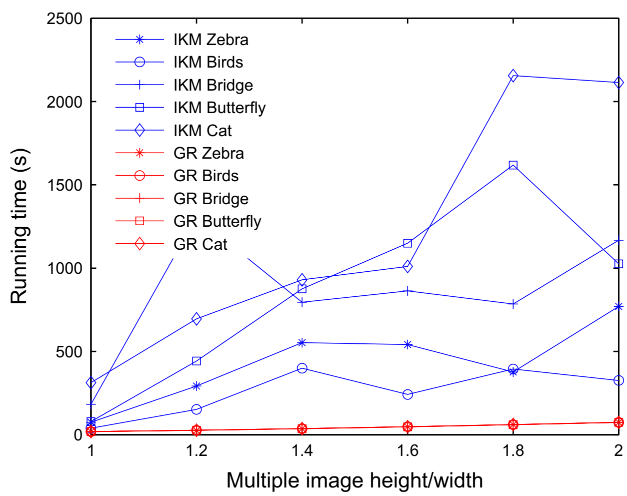

Fig. 14. Running time of quantization methods with multiple image sizes.

Table 6
Running time and iteration times with multiple image sizes.

Image	Image size	Improved *K*-means		Generic roughness	
		Time (s)	Iteration	Time (s)	Iteration
Zebra	h* w	75.203	114	18.574	9
height:321	1.2 h*1.2 w	292.594	187	26.62	7
width:481	1.4 h*1.4 w	552.797	294	36.195	7
	1.6 h*1.6 w	541.766	256	47.443	13
	1.8 h*1.8 w	376.703	158	60.128	14
	2 h*2 w	770.203	286	74.062	11
Birds	h* w	39.125	57	18.699	19
height:481	1.2 h*1.2 w	152.047	112	26.978	17
width:321	1.4 h*1.4 w	399.547	235	36.678	20
	1.6 h*1.6 w	242.328	118	47.863	22
	1.8 h*1.8 w	395.016	164	60.58	29
	2 h*2 w	325.312	120	74.545	32

See Fig. 14 and Table 6, for IKM, the computational cost rises fast as the image size increases. In some cases, when image size is continuously enlarged, it will cost too much time to obtain a stable quantization results. *GR* framework achieves superior performance in this test, the computational cost increases very slow as the image size increases. In the initial color space segmentation stage, the enlarged image needs more calculations for generating roughness indices but the runtime just increases linearly with the pixel number. In the stage of merging color, because the augmented pixel set has been compressed to hundreds of colors, the increased image sizes cause little increment of time cost for color clustering.

To sum up, *GR* quantization framework achieves a good balance between the quantization quality and computational complexity. Abundant experimental results indicate that this framework performs well on most testing images. It produces precise quantization results as superior postclustering methods while keeps low computational cost.

7. Conclusions

Although many postclustering quantization methods can achieve high quality results, they always suffer from heavy computational cost. To improve the efficiency of postclustering quantization, a two-stage color quantization framework based on generic roughness measure (*GR* framework) is investigated in this paper. The basic idea of this novel framework is to synthesize the techniques of roughness-based color space segmentation and clustering-based quantization. In the first stage, through thresholding roughness of color components, the original color space is compressed to a set of representative colors. And in the second stage, the initially compressed colors are merged to a palette using clustering method.

The key of *GR* framework is to guarantee the precision of the initial color space segmentation. Therefore we propose generic roughness measure to generate the delicate segmentation. The superiorities of *GR* quantization framework are summarized as follows. First, depending on the smoothed homogeneity mapping, generic roughness measure can tolerate the disturbance of trivial regions and precisely represent the spatial color homogeneity. Second, compressing the original image color to sparse RGB entries, the initial color space segmentation makes the quantization process more robust to various color distributions. The last but most important advantage is that the computational cost of *GR* framework relies on the initial color space segmentation. This makes *GR* framework own the low computational complexity as histogram-based methods. Extensive experiments on Berkeley color image segmentation database demonstrate the high efficiency of the proposed quantization framework. The extra testing results can be further provided by the authors.

Conflict of interest statement

None declared.

Acknowledgment

This work was supported by National Natural Science Foundation of China (Serial Nos. 61103067, 61273304) and Innovation Foundation of Shanghai University.

References

[1] L. Brun, A. Trémeau, Digital Color Imaging Handbook, CRC Press, Boca Raton, FL, 2002.
[2] M.E. Celebi, An effective color quantization method based on the competitive learning paradigm, in: Proceedings of the 2009 International Conference on Image Processing, Computer Vision, and Pattern Recognition, vol. 2, 2009, pp. 876–880.
[3] M.E. Celebi, Improving the performance of k-means for color quantization, Image Vis. Comput. 29 (2011) 260–271.
[4] C.H. Chang, P.F. Xu, R. Xiao, T. Srikanthan, New adaptive color quantization method based on self-organizing maps, IEEE Trans. Neural Networks 16 (1) (2005) 237–249.
[5] H.D. Cheng, X.H. Jiang, Y. Sun, J.L. Wang, Color image segmentation: advances and prospects, Pattern Recognition 34 (2001) 2259–2281.
[6] H.D. Cheng, X.H. Jiang, J.L. Wang, Color image segmentation based on homogram thresholding and region merging, Pattern Recognition 35 (2002) 373–393.

[7] S. Cheng, C. Yang, Fast and novel technique for color quantization using reduction of color space dimensionality, Pattern Recognition Lett. 22 (8) (2001) 845–856.
[8] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis, IEEE Trans. Pattern Anal. Mach. Intell. 24 (5) (2002) 603–619.
[9] M. Gervautz, W. Purgathofer, A simple method for color quantization: octree quantization, in: New Trends in Computer Graphics, Springer-Verlag, 1988, pp. 219–2318.
[10] A.E. Hassanien, A. Abraham, J.F. Peters, G. Schaefer, C. Henry, Rough sets and near sets in medical imaging: a review, IEEE Trans. Inf. Technol. Biomed. 13 (6) (2009) 955–968.
[11] P. Heckbert, Color image quantization for frame buffer display, ACM SIGGRAPH Comput. Graph. 16 (3) (1982) 297–307.
[12] I.S. Hsieh, K.C. Fan, An adaptive clustering algorithm for color quantization, Pattern Recognition Lett. 21 (4) (2000) 337–346.
[13] Y.C. Hu, B.H. Su, Accelerated k-means clustering algorithm for colour image quantization, Imaging Sci. J. 56 (1) (2008) 29–40.
[14] Q. Huynh-Thu, M. Ghanbari, Scope of validity of PSNR in image/video quality assessment, Electron. Lett. 44 (13) (2008) 800–801.
[15] Y.W. Lim, S.U. Lee, On the color image segmentation algorithm based on the thresholding and the fuzzy C-means techniques, Pattern Recognition 23 (9) (1990) 935–952.
[16] T. Lindeberg, A scale selection principle for estimating image deformations, Image Vis. Comput. 16 (1998) 961–977.
[17] T. Lindeberg, B.M.T.H. Romeny, Linear Scale Space, Kluwer Academic Publishers, Netherlands, 1994.
[18] P. Lingras, C. West, Interval set clustering of web users with rough k-means, J. Intell. Inf. Syst. 23 (2004) 5–16.
[19] K. Lo, Y. Chan, M. Yu, Colour quantization by three-dimensional frequency diffusion, Pattern Recognition Lett. 24 (14) (2003) 2325–2334.
[20] S. Makrogiannis, G. Economou, S. Fotopoulos, N.G. Bourbakis, Segmentation of color images using multiscale clustering and graph theoretic region synthesis, IEEE Trans. Syst. Man Cybern. 35 (2) (2005) 224–238.
[21] T. Masaru, W. Takashi, M. Taketoshi, Tsallis entropy in scale-spaces, in: Proceedings of SPIE Conference on Vision Geometry VIII, vol. 3811, 1999, pp. 273–281.
[22] A. Mohabey, A.K. Ray, Rough set theory based segmentation of color images, in: Proceedings of 19th International Conference of the North American Fuzzy Information Processing Society, 2000, pp. 338–342.
[23] A. Mojsilović, E. Soljanin, Color quantization and processing by Fibonacci lattices, IEEE Trans. Image Process. 10 (11) (2001) 1712–1725.
[24] M.M. Mushrif, A.K. Ray, Color image segmentation: rough-set theoretic approach, Pattern Recognition Lett. 29 (2008) 483–493.
[25] M.G. Omran, A.P. Engelbrecht, A color image quantization algorithm based on particle swarm optimization, Informatica 29 (2005) 261–269.
[26] M.T. Orchard, C.A. Bouman, Color quantization of images, IEEE Trans. Signal Process. 39 (12) (1991) 2677–2690.
[27] D. Özdemir, L. Akarun, Fuzzy algorithm for color quantization of images, Pattern Recognition 35 (8) (2002) 1785–1791.
[28] N.R. Pal, S.K. Pal, A review on image segmentation techniques, Pattern Recognition 26 (9) (1993) 1277–1294.
[29] N. Papamarkos, A. Atsalakis, C. Strouthopoulos, Adaptive color reduction, IEEE Trans. Syst. Man Cybern. Part B 32 (1) (2002) 44–56.
[30] Z. Pawlak, Rough sets, Int. J. Inf. Comput. Sci. 11 (5) (1982) 314–356.
[31] Z. Pawlak, Rough sets: Theoretical Aspects of Reasoning About Data, Kluwer Academic Publishers, Boston, 1991.
[32] Z. Pawlak, Some issues on rough sets, Trans. Rough Sets I 3100 (2004) 1–58.
[33] G. Peters, Some refinements of rough k-means clustering, Pattern Recognition 39 (2006) 1481–1491.
[34] J. Puzicha, M. Held, J. Ketterer, J.M. Buhmann, D.W. Fellner, On spatial quantization of color images, IEEE Trans. Image Process. 9 (4) (2000) 666–682.
[35] G. Schaefer, H. Zhou, Fuzzy clustering for colour reduction in images, Telecommun. Syst. 40 (1–2) (2009) 17–25.
[36] G. Schaefer, H.Y. Zhou, Q.H. Hu, A.E. Hassanien, Rough image colour quantisation, in: Proceedings of the 12th International Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, vol. 5908, 2009, pp. 217–222.
[37] E. Scheunders, A genetic C-means clustering algorithm applied to color image quantization, Pattern Recognition 30 (6) (1997) 859–866.
[38] P. Scheunders, Comparison of clustering algorithms applied to color image quantization, Pattern Recognition Lett. 18 (11–13) (1997) 1379–1384.
[39] Y. Sirisathitkul, S. Auwatanamongkol, B. Uyyanonvara, Color image quantization using distances between adjacent colors along the color axis with highest color variance, Pattern Recognition Lett. 25 (9) (2004) 1025–1043.
[40] J. Sporring, The entropy of scale-space, in: Proceedings of the 1996 International Conference on Pattern Recognition (ICPR 96), 1996, pp. 900–904.
[41] J. Sporring, J. Weichert, Information measures in scale-spaces, IEEE Trans. Inf. Theory 45 (3) (1999) 1051–1058.
[42] K.S. Tan, N.A.M. Isa, Color image segmentation using histogram thresholding—fuzzy C-means hybrid approach, Pattern Recognition 44 (2011) 1–15.
[43] P.E. Trahanias, A.N. Venetsanopoulos, Color edge detection using vector order statistics, IEEE Trans. Image Process. 2 (2) (1993) 259–265.
[44] P.E. Trahanias, A.N. Venetsanopoulos, Vector order statistics operators as color edge detectors, IEEE Trans. Syst. Man Cybern. Part B: Cybern. 26 (1) (1996) 135–143.
[45] T. Uchiyama, M. Arbib, An algorithm for competitive learning in clustering problems, Pattern Recognition 27 (10) (1994) 1415–1421.
[46] Z. Wang, A.C. Bovik, Mean squared error: love it or leave it? A new look at signal fidelity measures, IEEE Signal Process. Mag. 26 (1) (2009) 98–117.
[47] X. Wu, Color quantization by dynamic programming and principal analysis, ACM Trans. Graph. 11 (4) (1992) 348–372.
[48] X. Wu, Efficient statistical computations for optimal color quantization in: Graphics Gems Volume II, Academic Press, 1991, pp. 126–1338.
[49] Z. Xiang, Color image quantization by minimizing the maximum intercluster distance, ACM Trans. Graph. 16 (3) (1997) 260–276.
[50] B. Zhou, J.Y. Shen, Q.K. Peng, An adjustable algorithm for color quantization, Pattern Recognition Lett. 25 (16) (2004) 1787–1797.

Xiaodong Yue is presently a Research Fellow in the Advanced Analytic Institute (AAI) at University of Technology Sydney (UTS), supervised by Prof. Longbing Cao. Before this postdoctoral period, he received the Ph.D. degree from Tongji University in 2010. From October 2008 to April 2009, he was a research assistant in the Department of Computer Science at Hong Kong Baptist University, supervised by Prof. Jiming Liu. His research topics include pattern recognition, soft computing and multimedia.

Duoqian Miao received his Ph.D. degree in Pattern Recognition and Intelligent System at Institute of Automation, Chinese Academy of Sciences in 1997. He is a Professor and Vice dean of the School of Electronics and Information Engineering of Tongji University. Prof. Miao's present research interests include Soft Computing and Data Mining. Prof. Miao has published over 160 scientific articles in international journals, books, and conferences, such as Pattern Recognition, Pattern Recognition Letters and Information Sciences. He is a committee member of International Rough Sets Society, a senior member of the China Computer Federation (CCF), a committee member of the CCF Artificial Intelligence and Pattern Recognition and a vice chair of the CAAI Rough Set and Soft Computing Society.

Longbing Cao is a professor of information technology at the University of Technology Sydney (UTS). He got one Ph.D. in Pattern Recognition and Intelligent Systems and another in Computing Science. He is the Director of the Advanced Analytics Institute at UTS. He is also the Research Leader of the Data Mining Program at the Australian Capital Markets Cooperative Research Centre. He is a Senior Member of IEEE, SMC Society and Computer Society. His primary research interests include data analysis and machine learning. He works/worked with several major organizations on enterprise data mining, such as Australian Commonwealth Government Agency Centrelink, ATO, Westpac Banking Group, Qantas and Shanghai Stock Exchange.

Qiang Wu received the Ph.D. degree in computing science from the University of Technology Sydney, Australia, in 2004. He is currently a Senior Lecturer with the School of Computing and Communications, University of Technology Sydney. His major research interests include computer vision, image processing, pattern recognition and multimedia. He has published more than 70 refereed papers in these areas, including those published in prestigious journals and top international conferences. Dr. Wu has been a Guest Editor of several international journals, such as Pattern Recognition Letters. He has served as a Chair and/or a Program Committee Member for a number of international conferences. He has also served as a Reviewer for several international journals, such as the IEEE Transactions on Systems, Man, and Cybernetics Part B, the IEEE Transactions on Circuits and Systems for Video Technology, Pattern Recognition, Pattern Recognition Letters.

Yufei Chen is presently a senior lecturer in the School of Electronics and Information Engineering of Tongji University. She received Ph.D. degree from Tongji University in 2010. She was also a research fellow in Darmstadt University of Technology in 2009. Her research topics include pattern recognition and medical image analysis.