INVERTIBLE COLOR-TO-GRAYSCALE CONVERSION BY USING CLUSTERING AND REVERSIBLE WATERMARKING

Qiaoyi Liang, Runwen Hu and Shijun Xiang[*]

School of Information Science and Technology/School of Cyber Security,
Jinan University, Guangzhou, China
{qiaoyi_liang, runwen_hu, shijun_xiang}@qq.com.

2021 IEEE International Conference on Multimedia and Expo (ICME) | 978-1-6654-3864-3/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICME51207.2021.9428464

ABSTRACT

Invertible color-to-grayscale conversion is a method that embeds the color information into the corresponding grayscale image and extracts the color information to reconstruct the color image when necessary. In this paper, we propose an efficient method by using K-means clustering to generate a color palette and its corresponding grayscale image, and further making use of a reversible watermarking technique to embed the color palette into the grayscale image. For purpose of integrality authentication, the grayscale image is hashed as part of the embedded information before the embedding. In the process of reconstructing the color image, the color palette can be extracted correctly and the grayscale image can be recovered without any loss. Experimental results have shown that the proposed method can provide satisfactory performance.

***Index Terms*—** Invertible color-to-grayscale conversion, K-means clustering, Reversible watermarking

1. INTRODUCTION

Invertible conversion from color image to grayscale image is a new technology that ordinary users can only access to grayscale images while privileged users can obtain high-quality visualization color images with a secret key. Previous works about invertible conversion from color image to grayscale image are either based on subband embedding (SE) technique [1–5] or vector quantization (VQ) technique [6–12]. The SE based methods adopt the subband transform techniques and replace the high-frequency subbands with the down-sampled chrominance planes. Due to the loss of high-frequency information in the replacing process, the grayscale image and reconstructed color image are usually blurred. The method in [1] proposed a color quantization method by carrying a wavelet transform and replacing high-frequency subbands with chrominance signal. A more robust method [2] was proposed by using a large amount of redundant representation of the chrominance and embedding them into multiple subbands of the cosine transform. To compensate for the color saturation, Ko *et al.* [3, 4] proposed a color saturation embedding strategy based on wavelet packet transform. To preserve the chrominance and spatial resolution, Horiuchi *et al.* [5] proposed an accurate reversible color-to-gray mapping algorithm that achieved good performance by distributing color information effectively on the one-level wavelet subbands. In the VQ based methods, the color image was decomposed into the index image and the color palette. Chaumont *et al.* proposed a type of color reordering algorithms after performing a color quantization operation [6]-[9]. The type of approaches for the color quantization process were mainly based on the fuzzy c-means algorithm. After the color quantization, a color reordering algorithm was applied to get an index image where each grey-level is not too far from the luminance of the original color image and a color palette where two consecutive colors are close. Then, the color palette can be embedded in the index image by using a least significant bit (LSB) substitution operation. A new invertible color-to-monochrome transform method was proposed in [10], which clusters colors by K-means-based algorithm, and the output monochrome image was obtained by using the clustering result. A scheme that emphasizes the quality of both the color-embedded grayscale image and the reconstructed color image simultaneously was proposed in [11,12]. Especially, a halftone is utilized to get better visual quality. The method in [12] designed the color palette based on a convex hull generation algorithm and inserted the color palette by using the LSB substitution.

In practice, the quality of the reconstructed color image depends on the design of the color palette and the corresponding grayscale image. For transmission, the color palette should be embedded into the grayscale image. In the literature, all the hiding processes are based on LSB substitution. Such a way will distort the grayscale image more or less, especially when the color palette information is large. The distortion due to the LSB substitution will degrade the reconstruction quality of the color image due to the loss of the LSB information of the grayscale image. To this problem, a solution based on the reversible watermarking technique is ap-

[*]Corresponding author: Shijun Xiang (e_mail: shijun_xiang@qq.com).
This work was supported by National Natural Science Foundation of China (NSFC) under Grant 61772234.

978-1-6654-3864-3/21/$31.00 ©2021 IEEE

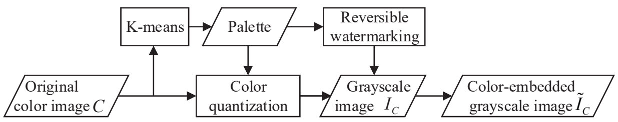


Fig. 1. Coding process

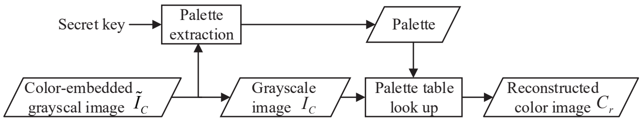


Fig. 2. Decoding process

plied, which can recover the grayscale image without any loss after extracting the color palette. Further, in our scheme, a strategy-based K-means clustering is adopted to generate the color palette. Compared with the two recent excellent works, the proposed method can achieve better performance.

The rest of this paper is organized as follows. Section 2 presents our proposed method in detail. Section 3 provides some simulation results for performance evaluation and a conclusion is given in Section 4.

2. PROPOSED METHOD

There are four main aspects of invertible conversion from color image to grayscale image: generation of the color palette, the embedding of the color palette, generation of the grayscale image, and reconstruction of the color image. The proposed scheme is similar to that in [12]. Differently, in the process of designing the color palette, a K-means clustering-based strategy is exploited instead of the convex hull generation algorithm for the reduction of the square error. Furthermore, the embedding process based on the LSB substitution in [12] is improved by using the reversible watermarking technique, which not only can extract the color palette correctly but can recover the grayscale image without any loss. Fig. 1 plots the proposed method on how to generate the color palette, which aims to design a color palette that is good at representing the original color image C with less distortion and generating a color-embedded grayscale image $\tilde{I}_c$ that is similar to the luminance plane of the color image. After obtaining the grayscale image and the color palette, the color palette will be embedded into the grayscale image reversibly. Fig. 2 depicts the process of how to reconstruct the color image, where the color palette is extracted from the image $\tilde{I}_c$. Finally, the color image C_r is reconstructed by using the grayscale I_c and the color palette.

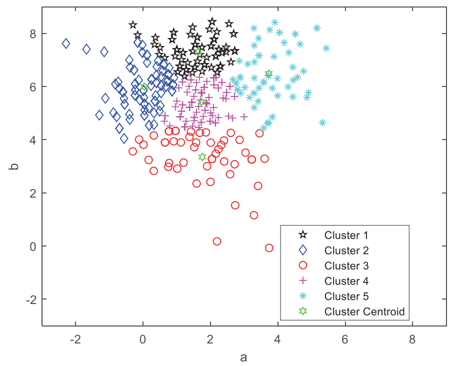


Fig. 3. The result of clustering using K-means in a layer

2.1. Generation of the Color Palette

The color palette is generated in the CIElab space since this color system is based on physiological characteristics. Specifically, the magnitude of numerical change is consistent with the change perceived by the human eye in the CIElab space. When a color image is represented in RGB format, a conversion between the RGB space and the CIElab space (with D65 white point) is required. After the conversion, all pixels are represented in the form of vector $(L, \mathrm{a}, \mathrm{b})$, where L is the luminance, a and b are the chrominance components, respectively. Hereafter, the chrominance and luminance mentioned in this paper refer to the CIElab space. It is worth noting that the range of luminance in CIElab space is $[0, 100]$ while that of a and b are in $[-100, 100]$.

Noting that an 8-bit grayscale image only has 256 grayscale, the size of the color palette must be 256. Besides, the pixel values of the grayscale image to the color palette is better to be a one-one mapping so that the unique color information can be found for reconstructing the color image. Consequently, the goal is to find 256 different colors to represent all colors of the original color image, which means that the luminance plane, a plane, and b plane must be quantified into a 256×3 vector. One common idea is to divide the CIElab space into 256 intervals along the luminance direction as shown in (1):

$$L_k = \{c | \tfrac{100k}{255} \leq luminance\ \ of\ \ color\ \ c < \tfrac{100k}{255} + \tfrac{100}{255}\} \quad \text{for } 0 \leq \mathrm{k} \leq 255. \tag{1}$$

The subspace is divided into intervals and the middle luminance value of the interval is assigned to match the index value of a palette color to its luminance value by

$$L_{(k)} = (L_{k,UB} + L_{k,LB})/2, \tag{2}$$

where $L_{k,UB}$ and $L_{k,LB}$ are the maximum and the minimum luminance values in the same interval, respectively. As shown in Fig. 2(a) of [12], the variation of visible colors in the middle intervals is larger than that of the top or bottom intervals.

Therefore, more colors in the palette should be assigned to the middle intervals so that the average chrominance distortion can be reduced. Knowing that palette colors are distributed to every layer with the constraint $\sum_{0\leq k<N_L} N_k = 256$, we divide the subspace into 54 layers (i.e., N_L=54) and N_k is allocated according to (3) inspired by the framework proposed in [12].

$$N_k = \begin{cases} 1, \text{ if } \mathrm{k} = 0 \text{ or } 53 \\ 4, \text{ if } \mathrm{k} = 1, 2, 3\ , 50, 51 \text{ or } \ 52 \\ 5, \text{ if } 3 < \mathrm{k} < 50. \end{cases} \tag{3}$$

Consequently, Equation (1) is adjusted to accommodate the uneven allocation and the thickness of each layer is proportional to N_k instead of being a constant for all k. This can be expressed as

$$L_k = \Big\{ c \Big| \tfrac{100}{255} \sum_{n=0}^{k} N_{n-1} \leq luminance \ \ of \ \ color \ \ c \\ < \tfrac{100}{255} \sum_{n=0}^{k} N_n \Big\} \qquad \text{for } \ 0 \leq \mathrm{k} \leq \mathrm{N_L}, \mathrm{N_{-1}} = 0. \tag{4}$$

Suppose C is the color image and $C(i, j)$ is the color of one pixel. All pixels in the image C are sorted into 54 sets according to the luminance as shown in (5):

$$\Omega_k = \{C(i, j) | C(i, j) \in L_k, \forall (i, j)\} \\ \text{for } \ 0 \leq \mathrm{k} \leq 53. \tag{5}$$

For the pixels belong to the same luminance layer Ω_k, the luminance can be neglected and dimension reduction can be applied according to (6) in order to reduce the computational complexity:

$$f(L, a, b) = (a, b). \tag{6}$$

In (6), L is the luminance, a and b are the chrominance components of pixel (i, j), respectively. As a result, the mapping pixels belong to each layer in (3) can be clustered into different quantity cluster centers to generate the color palette.

Clustering means to find K groups from the representation N by using the measure of similarity, the level of which is an important basis to determine whether the targets belong to the same group or not [13]. In this scenario, the clustering technique is used to identify the similarity among chrominance information. Since the computational complexity of K-means is low, K-means is adopted in this paper for clustering. K-means partitions the pixels belong to Ω_k into K (i.e., $K = N_k$) exhaustive and mutually exclusive clusters $S = \{S_1\ , S_{2,...,}\ S_K\}$, where each cluster satisfies

$$\begin{cases} \cup_{K=1}^{K} S_K = \Omega_k \\ S_i \cap S_j = \emptyset \end{cases}, \ \text{for } \ 1 \leq i \neq j \leq K. \tag{7}$$

The purpose of adopting K-means is to minimize the squared error between the empirical mean of a cluster center and the points in the cluster. Supposed $C_{i,j}$ is a pixel in Ω_k and the squared error is defined as

$$J(S_K) = \sum_{C_{i,j} \in S_K} ||f(C_{i,j}) - \mu_K||^2 \tag{8}$$

where μ_K and $||.||^2$ are the mean of cluster S_K and the Euclidean distance, respectively.

In particular, one palette color is assigned for layers L_0 and L_{53} in (3) since such two palette colors are the absolute black and absolute white in CIElab format respectively (i.e., (0,0,0) and (100,0,0)). For layers $L_k, 3 \leq k \leq 50$, 5 palette colors are assigned according to (3). In addition, 4 palette colors are assigned for layers L_1, L_2, L_3, L_{50}, L_{51} and L_{52}. An illustration of clustering is shown in Fig. 3.

Considering the processes above, the generation of palette can be completed as follows:

$$\vec{P}_{f_0(k,m)} \begin{cases} (0, 0, 0) \ \text{ for } \ (\mathrm{k}, \mathrm{m}) = (0, 1) \\ (\frac{1}{2}(L_{k,UB} + L_{k,LB}), a * (\mu_{k,m}), b * (\mu_{k,m})) \\ \qquad \text{for } \ 0 < \mathrm{k} < 53, 1 \leq \mathrm{m} \leq \mathrm{N_k} \\ (100, 0, 0) \ \text{ for } \ (\mathrm{k}, \mathrm{m}) = (53, 1) \end{cases}, \tag{9}$$

where $\vec{P}_n$ is the n^{th} palette color of the 256-color palette, k is the k^{th} layer, m is the m^{th} palette color of the k^{th} layer, $a * \mu_{k,m}$ and $b * \mu_{k,m}$ are the a and b components of $\mu_{k,m}$ respectively. $f_0(\cdot)$ is a bijective function defined as follows:

$$f_0(k, m) = \textstyle\sum_{s=0}^{k-1} N_s + m - 1 \in \{0, 1... \ 255\} \\ \text{for } \ 0 \leq \mathrm{k} \leq 53 \ \text{ and } \ 1 \leq \mathrm{m} \leq \mathrm{N_k}. \tag{10}$$

2.2. Generation of Grayscale Image

After building the color palette, the grayscale image can be generated by using the index of the color palette since all the pixels in image C have been quantized to the values of the generalized color palette. Suppose one pixel is $C_{i,j}$ and the luminance component is $L_{i,j}$, the layer of the quantized luminance component is determined as

$$\kappa \in \{k | \ L_{k,UB} > L_{i,j} > L_{k,LB} \ \ and \ \ 53 \geq k \geq 0\}. \tag{11}$$

Obtaining the luminance layer to which $C_{i,j}$ belongs, the chrominance of $C_{i,j}$ will be quantized to the cluster center of the current layer:

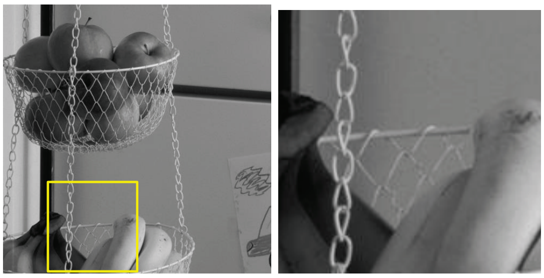
$$\lambda = \arg \min_{\lambda=1,2...N_k} ||C_{i,j} - f(\vec{p}_{f_0(\kappa,\lambda)})||^2. \tag{12}$$

Consequently, the quantized value of $C_{i,j}$ (i.e., $C_{quan}(i, j)$) can be expressed as follows:

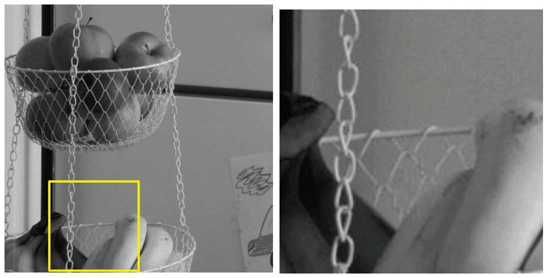
$$C_{quan}(i, j) = \vec{P}_{f_0(\kappa,\lambda)}. \tag{13}$$

Table 1. Comparison of the embedding methods between LSB substitution and reversible watermarking

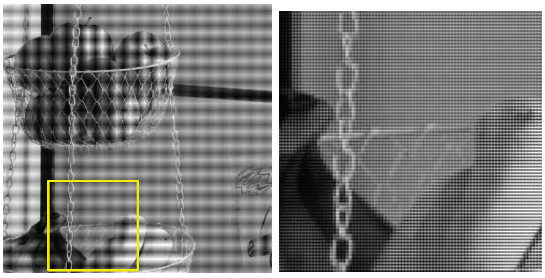
KODAK IMAGE DATASET						MCMASTER IMAGE DATASET				
	Color-embedded Grayscale Image			Restroed Color Image		Color-embedded Grayscale Image			Restroed Color Image	
Method	PSNR (dB)	SSIM	GMSD	PSNR (dB)	FSIM	PSNR (dB)	SSIM	GMSD	PSNR (dB)	FSIM
The proposed method	40.15	0.9908	0.0085	44.90	0.9965	41.07	0.9918	0.0067	42.29	0.9935
Method based on K-means and LSB embedding	41.74	0.9924	0.0071	44.71	0.9964	42.01	0.9925	0.0056	42.11	0.9929



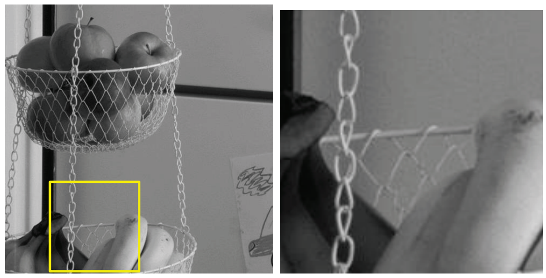
(a) Ground truth



(b) The proposed method



(c) Queiroz *et al.*'s [1]



(d) Chan *et al.*'s [12]

Fig. 4. Comparison of grayscale images and their enlarge portion.

According to color-to-grayscale conversion shown in Fig. 1, the index value of $C_{quan}(i,j)$ is

$$I_c(i,j) = f_0(\kappa, \lambda). \tag{14}$$

After the conversion, the index image acts as a 256-level grayscale image corresponds to the original image C.

2.3. Reversible Embedding of the Color Palette

After generating the color palette and the grayscale image, the color palette should be embedded into the grayscale image for gray-to-color conversation. In the literature, LSB substitution is widely used for embedding color palette, such as [6–12]. The basic idea of LSB substitution is to substitute the least significant bit of the gray level with the bitstream of the color palette. However, the loss of the replaced information will distort the gray image to some extent and further lead to the distortion of the reconstructed color image. Consequently, an ideal solution to this problem is to use a technique that can recover the grayscale image without any loss and extract the color palette from the color-embedded gray image in the process of reconstructing the color image.

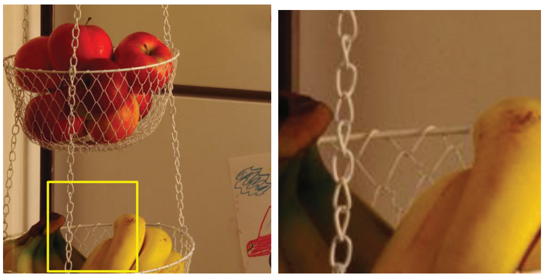
Noting that the reversible watermarking technique can meet the requirement of recovering the cover image and the hiding information in a lossless way [14–16], in this paper we adopt the reversible watermarking method proposed in [14] to embed the color palette. Aiming to achieve the requirement of verifying the content integrity in transmission, a practical and convenient approach is to hash the grayscale image and embed the hash value. In this paper, SHA256 is used to hash the grayscale image, producing 256 bits for integrity authentication. For hiding the color palette and the hash value, the payload is $254 \times 2 \times 8 + 256 = 4,320$ bits since each chrominance component is encoded with 8 bits.

2.4. Grayscale-to-Color Reconstruction

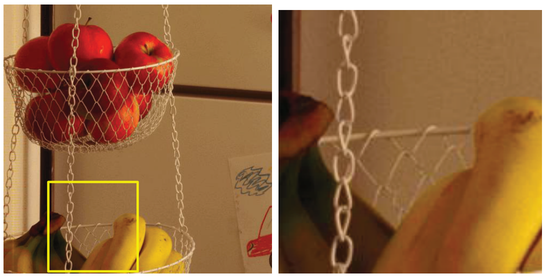
Fig. 2 shows the flow chart of reconstructing the color image by using the recovered grayscale image and the extracted color palette. The private key is applied to extract the col-

Table 2. Comparison of the simulation result between different methods

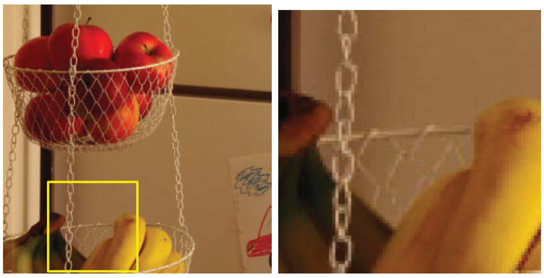
	KODAK IMAGE DATASET					MCMASTER IMAGE DATASET				
	Color-embedded Grayscale Image			Restroed Color Image		Color-embedded Grayscale Image			Restroed Color Image	
Method	PSNR (dB)	SSIM	GMSD	PSNR (dB)	FSIM	PSNR (dB)	SSIM	GMSD	PSNR (dB)	FSIM
Queiroz et al. [1]	31.64	0.9760	0.0019	33.14	0.9978	28.57	0.9533	0.0017	29.43	0.9964
Chan et al. [12]	41.40	0.9915	0.0025	38.37	0.9985	41.48	0.9959	0.0023	33.85	0.9955
The proposed method	40.15	0.9908	0.0085	44.90	0.9965	41.07	0.9918	0.0067	42.29	0.9935



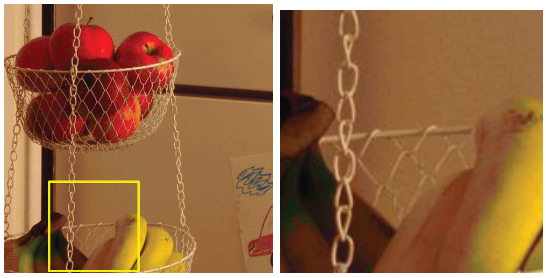
(a) Ground truth



(b) The proposed method



(c) Queiroz *et al.*'s [1]



(d) Chan *et al.*'s [12]

Fig. 5. Comparison of reconstructed color images their enlarge portion.

or palette and recover the grayscale image I_c correctly. After that, the chrominance components of each pixel are reconstructed by indexing the pixel values to the color palette. Based on the table-lookup process, the reconstructed color image C_r can be generated.

3. SIMULATION RESULTS

Aiming to evaluate the performance of the proposed method, several experiments have been carried out in this section by comparing with two state of the art works [1, 12]. One of the datasets is the Kodak dataset including 24 color images of size 768×512 or 512×768, and the other is the McMaster dataset which contains 18 color images of size 500×500. To better evaluate the performance of the proposed method, the peak signal-to-noise ratio (PSNR), structural similarity (SSIM) [17], gradient magnitude similarity deviation (GMSD) [18] and feature similarity(FSIM) [19] have been used to measure the visual quality of the color-embedding grayscale images and the reconstructed images.

As shown in Table 1, for the proposed method, the performance (i.e., PSNR, SSIM, and GMSD) of the color-embedding grayscale images by using reversible watermarking to hide the color palette is somewhat lower than that by using the LSB substitution, but the visual quality is still high with the average PSNR value of 40.15 dB for Kodak image dataset when that of using LSB substitution is 41.74 dB. As a bonus, in the proposed reversible watermarking based strategy the reconstructed color images is better due to the index image can be recovered correctly. For the Kodak image dataset, the average PSNR value of the reconstructed color images is 44.90 dB when its counterpart is 44.74 dB. On average, the proposed method outperforms the LSB substitution based strategy by 0.18 dB for the reconstructed color images. Overall, we consider that the reversible watermarking based strategy is more promising since the reconstructed color image is better when the other conditions keep unchanged.

Furthermore, we have compared the proposed reversible watermarking based strategy against two existing excellent works [1] and [12]. As shown in Table 2, the proposed method performs extremely well in referring to PSNR for the reconstructed color images no matter using reversible watermarking or LSB substitution. For the Kodak image dataset, the average PSNR for the reconstructed color images of the proposed method is 11.76dB and 5.53dB higher than method in [1] and method in [12], respectively. Notice that, the performance (i.e., FSIM) of the two state of the art work [1] [12] outperform the proposed method, but it still achieves a good result. These results show that the use of K-means clustering is efficient.

Fig. 4 and Fig. 5 show the experimental results of the proposed method against two state of the art works in visual quality. From these two figures, we can find that there exists blurring for the method in [1] while a color-shifting problem has been caused in [12]. For the proposed method, the visual quality of the grayscale image and the reconstructed color image are better than the other two methods.

4. CONCLUSION

In this paper, we propose an effectively invertible color-to-grayscale conversion algorithm based on clustering and reversible watermarking. For generating the color palette, all pixels are classified into various sets according to the luminance component, and the K-means clustering approach is applied to generate specific quantified points. The grayscale image is generated by indexing the color palette. In particular, the strategy that the color palette and the hash of the index image are reversible embedded into the index image. Such a way can recover the index image without any loss. Experimental results have shown that the proposed scheme with both clustering and reversible watermarking techniques can achieve better performance.

5. REFERENCES

[1] Ricardo L de Queiroz and Karen M Braun, "Color to gray and back: color embedding into textured gray images," *IEEE Trans. Image Process.*, vol. 15, no. 6, pp. 1464–1470, 2006.

[2] Ricardo L de Queiroz, "Reversible color-to-gray mapping using subband domain texturization," *Pattern Recognit. Lett.*, vol. 31, no. 4, pp. 269–276, 2010.

[3] Kyung-Woo Ko, Oh-Seol Kwon, Chang-Hwan Son, and Yeong-Ho Ha, "Color embedding and recovery based on wavelet packet transform," *J. Imaging Sci. Technol.*, vol. 52, no. 1, pp. 10501–1, 2008.

[4] Kyung-Woo Ko, Dae-Chul Kim, Wang-Jun Kyung, and Yeong-Ho Ha, "Color embedding and recovery using wavelet packet transform with pseudorandomized saturation code," *J. Imaging Sci. Technol.*, vol. 55, no. 3, pp. 30501–1, 2011.

[5] Takahiko Horiuchi, Fuminori Nohara, and Shoji Tominaga, "Accurate reversible color-to-gray mapping algorithm without distortion conditions," *Pattern Recognit. Lett.*, vol. 31, no. 15, pp. 2405–2414, 2010.

[6] Marc Chaumont and William Puech, "A grey-level image embedding its color palette," in *2007 IEEE Int. Conf. on Image Process.* IEEE, 2007, vol. 1, pp. I–389.

[7] Marc Chaumont and William Puech, "A fast and efficient method to protect color images," in *Vis. Commun. and Image Process. 2007*. International Society for Optics and Photonics, 2007, vol. 6508, p. 65081T.

[8] Marc Chaumont and William Puech, "Protecting the color information by hiding it," *Recent Adv. in Signal Process.*, vol. 22, 2009.

[9] Marc Chaumont, William Puech, and Christian Lahanier, "Securing color information of an image by concealing the color palette," *J. Syst. Softw.*, vol. 86, no. 3, pp. 809–825, 2013.

[10] Go Tanaka, Noriaki Suetake, and Eiji Uchino, "Invertible color-to-monochrome transformation based on clustering with lightness constraint," in *2010 IEEE Int. Conf. on Syst., Man and Cybern.* IEEE, 2010, pp. 2151–2154.

[11] Zi-Xin Xu and Yuk-Hee Chan, "Improving reversible color-to-grayscale conversion with halftoning," *Signal Process.-Image Commun.*, vol. 52, pp. 111–123, 2017.

[12] Yuk-Hee Chan, Zi-Xin Xu, and Daniel Pak-Kong Lun, "A framework of reversible color-to-grayscale conversion with watermarking feature," *IEEE Trans. Image Process.*, vol. 29, pp. 859–870, 2019.

[13] Stuart Lloyd, "Least squares quantization in pcm," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[14] Vasiliy Sachnev, Hyoung Joong Kim, Jeho Nam, Sundaram Suresh, and Yun Qing Shi, "Reversible watermarking algorithm using sorting and prediction," *IEEE Trans. Circuits Syst. Video Technol*, vol. 19, no. 7, pp. 989–999, 2009.

[15] Xingyuan Liang and Shijun Xiang, "Robust reversible audio watermarking based on high-order difference statistics," *Signal Process.*, vol. 173, pp. 107584, 2020.

[16] Runwen Hu and Shijun Xiang, "Cnn prediction based reversible data hiding," *IEEE Signal Process. Lett.*, vol. 28, pp. 464–468, 2021.

[17] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.

[18] Wufeng Xue, Lei Zhang, Xuanqin Mou, and Alan C Bovik, "Gradient magnitude similarity deviation: A highly efficient perceptual image quality index," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 684–695, 2013.

[19] Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang, "Fsim: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, 2011.