Unsupervised, Fast and Efficient Color Image Copy Protection

Arash Abadpour [a] and Shohreh Kasaei [b,*]

[a] *Mathematics Science Department, Sharif University of Technology, Tehran, Iran*

[b] *Computer Engineering Department, Sharif University of Technology, Tehran, Iran*

Abstract

The ubiquity of broadband digital communications and mass storage in modern society has stimulated the widespread acceptance of digital media. However, easy access to royalty-free digital media has also resulted in a reduced perception in society of the intellectual value of digital media and has promoted unauthorized duplication practices. To detect and discourage the unauthorized duplication of media, researchers have investigated watermarking methods to embed ownership data into media. However, some authorities have expressed doubt over the efficacy of watermarking methods to protect digital media.

This paper introduces a novel method to discourage unauthorized duplication of digital images by introducing reversible, deliberate distortions to the original image. The resultant image preserves the image size and essential content with distortions in edge and color appearance. The proposed method also provides an efficient reconstruction process using a compact key to reconstruct the original image

from the distorted image. Experimental results indicate that the proposed method can achieve effective robustness towards attacks, while its computational cost and quality of results are completely practical.

Key words: Copy Protection, Color Image Processing, Principal Component Analysis, Recoloring, Watermarking.

1 Introduction

The Internet provides an entertainment and informational resource, rich with audio, images and video. While much of the resources on the Internet are recognized as "free", the modern Internet is also a market for businesses supplying and distributing their digital media. With recent developments and price positioning in digital media equipment, the Internet provides an ideal medium for the distribution of digital media. However, the same developments in broadband communications and mass storage hardware also provide opportunities for the unauthorized duplication and distribution of digital media. The reasons which make the Internet so persuasive as a distribution medium are also the concern when protecting intellectual property.

Over the last decade, much research has concentrated on the protection of digital media against unauthorized duplication and distribution [1]. The most rec-

* Corresponding author: Computer Engineering Department, Sharif University of Technology, Azadi St., Tehran, Iran, P.O. Box 11365-9517, Telephone: (+98) 21 616 4631.

Email addresses: abadpour@math.sharif.edu (Arash Abadpour), skasaei@sharif.edu (Shohreh Kasaei).

ognized method in the field is watermarking [2]. In the watermarking method, authorization data is added to the original data to help identifying the owner. Over the last decade, many interesting and original methods have been proposed for adding watermarks to digital media [3–15]. Unfortunately, equally interesting and original methods have been proposed for circumventing the protection offered by these watermarking methods. In fact, the equal development of cheating methods makes it undoubtable that for any watermarking method there are generally affordable methods of attack to corrupt the watermark or embed another watermark [16]. The robustness of different watermarking methods against attack has been considered by many researchers [17–20]. Table 1 compares a few typical watermarking methods in terms of attack resistance. A survey on security scenarios in music watermarking and their failure has been published [21]. The competing sides of watermarking technologies has prompted NASA experts to publish guidelines for image protection methods with a statement advising to "*never expose an image in its large size*", and suggesting to use "*visual watermarks*" and "*programming shields*" [22].

Recently, the efficacy of watermarking methods has been criticized by senior members of the image processing community [23]. After *Herley*'s controversial note on watermarking entitled"*Why Watermarking is Nonsense*" [23], researchers emphasized that this immature field of signal processing is "*oversold*" and that no method has yet been able to claim "*the ability to protect from all possible future attacks*" [24,25]. *M. Barni* comments,"*Why should we hide the information within the data, when we could more easily use headers*

or other means to reach the same goal?" [26].

In response to public criticism, watermark researchers currently emphasize applications with less-restrictive requirements and forward to emerging techniques to solve the fundamental problems [25]. In a very different approach, a few researchers have worked on direct copy detection [27], but it is also not yet a stable practical tool.

Of all digital media, digital image hardware has seen the greatest technological developments over the recent years. The cost of consumer and professional digital camera equipment and duplication hardware has improved to the point where digital images are considered comparable with traditional film. The distribution and protection of commercial digital images presents additional requirements to copy protection methods. This paper considers methods for protecting digital images.

The traditional practice for the distribution of commercial images involves sending down-sampled, cropped or visually-watermarked images for customer evaluation. When the customer chooses to purchase the images, the original, high-quality images are provided to replace the evaluation images. In this approach, the discarded information available in the evaluation images and the bandwidth consumed in acquiring these images is wasted. This paper proposes a method which permits the original images to be regenerated using a small-sized key, and ultimately results in improved bandwidth efficiency.

The evaluation images are intentionally manipulated to maintain the visual content, while containing many fake edges and color alterations to make it inappropriate for unauthorized professional distribution. The original images are registered with the application of a small-sized key. The primary benefit of the proposed method is that the original image is efficiently reconstructed in a lossless framework.

The proposed method manipulates regions on homogeneous image texture for protection against unauthorized duplication and distribution and extends the method of *principal component analysis* (PCA) [28]. Processing methods derived from the PCA technique have been shown to exhibit improved classification in color images compared to well–known techniques such as the *fuzzy C–means* (FCM) [29]. Previous work has also developed a fast PCA–based color transfer method to recolor an image using the color information extracted from a reference image [30].

The proposed method assumes color images as vector geometries and applies vectorial tools on them. This approach contrasts with the general approach of considering color images as a set of parallel gray–scale images or using standard color spaces [31–33]. A technique for describing homogeneous textures within an image based on PCA called *linear partial reconstruction error* (LPRE) has been proven to be a powerful representation [34]. [Add a sentence explaining this why.] The LPRE algorithm has been used as the basis of an algorithm called *fuzzy principal component analysis–based clustering* (FP-

CAC), for the classification of image texture and has proved its superiority over the commonly used *Euclidean* and *Mahalanobis* likelihood distance–based approaches [35,36].

The rest of this paper is organized as follows: Section 2 introduces the proposed copy protection method. It discusses the proper likelihood measurement function for color fields and briefly reviews the fuzzy companion of the PCA (called FPCA). FPCA algorithm is used in a clustering method called FPCAC and introduces a PCA–based recoloring method and incorporates the different approaches in the proposed color image copy protection algorithm. Section 3 holds the experimental results, and Section 4 concludes the paper.

2 Proposed Method

In this section, we first review some preliminary concepts of PCA-based analysis of color images, the classification of homogenous color regions and the efficient representation of homogeneous textures.

2.1 PCA–Based Color Image Processing

PCA considers a color image as vector geometries and applies vectorial tools on them.

When attempting to analyze random variables of dimensions more than unity, there are several choices for the vector–to–set metric. A careless selection in

this stage can decline the performance of the subsequent stages. The classic distance functions, such as Euclidean and Mahalanobis, are frequently used in available approaches. But recent research has shown that neither of them can be considered as the optimal choice for all physical phenomena.

It is mathematically proven [37] and empirically experimented [34,36] that the best available distance function for the sets of 3–D color vectors in the RGB color space is the *linear partial reconstruction error* (LPRE) defined as,

$$\tau(\vec{\mathbf{c}}, C) = \|(\vec{\mathbf{c}} - \vec{\eta}_C) - \vec{\mathbf{v}}_C^T(\vec{\mathbf{c}} - \vec{\eta}_C)\vec{\mathbf{v}}_C\|^2. \tag{1}$$

in which C is the set of color vectors and $\vec{\mathbf{c}}$ is a single color realization. Also, $\vec{\eta}_C$ and $\vec{\mathbf{v}}_C$ are the expectation vector and the first principal direction of C, respectively.

Here, we give an illustrative example for comparing the performance of the LPRE–based query extraction in color images with the results of the same operation applied when using the Euclidean and Mahalanobis–based distances (see Figure 1). Query region extraction is the process of finding the region similar to a given query. Note the superiority of the LPRE–based approach compared with the two other distances in Figure 1. Interested reader can find more examples of LPRE–based query region extraction in [38]. Also, see [36,35] for the performance comparison of the LPRE with the Euclidean and Mahalanobis distances for other image processing applications.

2.2 Fuzzy Principal Component Analysis (FPCA)

Consider the problem of finding the principal components of the fuzzy set $\tilde{X} = \{(\vec{\mathbf{x}}_i; p_i) | i = 1, \cdots, n\}$. It is proved in [35] that computing the fuzzy expectation vector as,

$$\vec{\eta}_{\tilde{X}} = \frac{1}{\sum_{i=1}^{n} p_i} \sum_{i=1}^{n} p_i \vec{\mathbf{x}}_i, \tag{2}$$

and the fuzzy covariance matrix as,

$$C_{\tilde{X}} = \frac{1}{\sum_{i=1}^{n} p_i} \sum_{i=1}^{n} p_i (\vec{\mathbf{x}}_i - \vec{\eta}_{\tilde{X}})^T (\vec{\mathbf{x}}_i - \vec{\eta}_{\tilde{X}}), \tag{3}$$

yields a perfect dual of PCA for fuzzy sets. In this way, the fuzzy principal directions of $\tilde{X}$ are the eigenvectors of $C_{\tilde{X}}$ sorted by the corresponding eigenvalues in an ascending fashion. Also, the PCA matrix of $\tilde{X}$ come from the *singular value decomposition* (SVD) of $C_{\tilde{X}}$.

2.3 Fuzzy Principle Component–Based Clustering (FPCAC) of Color Vectors

Consider the general clustering problem stated as minimizing the objective function defined as,

$$J(X, \Phi) = \sum_{i=1}^{n} \sum_{j=1}^{C} p_{ij}^m D_{ij}. \tag{4}$$

This problem describes the best choice for clustering the members of the set $X = \{\vec{\mathbf{x}}_1, \cdots, \vec{\mathbf{x}}_n\}$ into C clusters described as $\Phi = \{\phi_1, \cdots, \phi_C\}$. Here, p_{ij} is the fuzzy membership of $\vec{\mathbf{x}}_i$ to the j–th cluster and D_{ij} is the distance from

$\vec{\mathbf{x}}_i$ to the j–th cluster. Here, $m > 1$ is the fuzziness parameter. Assume that $D_{ij} = \Psi(\vec{\mathbf{x}}_i, \phi_j)$ is the *appropriate* distance function for the vector geometry under investigation. As an example, as discussed in Section 2.1, the LPRE is the proper distance for color vectors. Here, ϕ_j is the defining parameters of the j–th cluster, according to the general cluster model. Again, as an example in the LPRE methodology, $\phi_j = [\vec{\eta}_j, \vec{\mathbf{v}}_j]$.

Assume that the function Υ tunes the cluster model ϕ_j to best fit a fuzzy set of vectors. This means that for the fuzzy set of vectors $\tilde{X} = \{(\vec{\mathbf{x}}_i; p_i) | i = 1, \cdots, n\}$,

$$\Upsilon(\tilde{X}) = \arg_\phi \min \left\{ \sum_{i=1}^{n} p_i \Psi(\vec{\mathbf{x}}_i, \phi) \right\}. \tag{5}$$

Therefore, the function $\Upsilon(\tilde{X}) \equiv \tilde{E}\{\tilde{X}\}$ is the companion for $\Psi(\vec{\mathbf{x}}, \phi) = \|\vec{\mathbf{x}} - \vec{\eta}\|^2$. Here, $\tilde{E}\{\tilde{X}\}$ stands for the fuzzy expectation of a fuzzy set. Also, the FPCA introduced in Section 2.2 gives the $\Upsilon(\tilde{X})$ function corresponding to the Ψ function defined as the LPRE [35].

Back to the main problem of minimizing $J(X, \Phi)$, assume that we have the dual functions $\Psi(\cdot)$ and $\Upsilon(\cdot)$. In [35], the authors proposed an algorithm that converges to a minimal point of $J(X, \Phi)$, if at least one exists. The Pseudo–code for this algorithm is given below,

- ***Aim****: Clustering data according to the given model.*

- ***Inputs**:*

  · *Appropriate distance function* Ψ *and its dual* Υ.

  · *Set of realizations* $X = \{\vec{\mathbf{x}}_1, \cdots, \vec{\mathbf{x}}_n\}$.

  · *Number of clusters* C.

  · *Fuzzyness* m.

  · *Halting threshold* δ.

- ***Output**:*

  · *Fuzzy membership values* $p_{ij}, 1 \leq i \leq n, 1 \leq j \leq C$.

  · C *cluster descriptors* $\phi_1, \cdots, \phi_C$.

- ***Method**:*

  1 $l = 0$.

  2 Randomize $\phi_1, \cdots, \phi_C$.

  3 $l = l + 1$.

  4 Compute distances as $D_{ij} = \Psi(\vec{\mathbf{x}}_i, \phi_j)$.

  5 Compute fuzzy membership values as (6).

  6 Store the fuzzification scheme as $p_{ij} = F_{lji}$.

  7 Renew clusters as (7).

  8 if $l > 1$ *compute* δ_l *as (9).*

  9 if $l = 1$ *then goto* 3*, elseif* $\delta_l > \delta$ *then goto* 3 *else return.*

As shown at the above, the first step is to randomize a set of initial cluster descriptors $\phi_1, \cdots, \phi_C$. Then, for all $i = 1, \cdots, n$ and $j = 1, \cdots, C$ the distance from $\vec{\mathbf{x}}_i$ to the j–th cluster is computed as $(D_{ij} = \Psi(\vec{\mathbf{x}}_i, \phi_j))$. As proved in [35], the p_{ij} should guide $J(X, \Phi)$ to reach its minimum point, by:

$$p_{ij} = \frac{D_{ij}^{-\frac{1}{m-1}}}{\sum_{k=1}^{c} D_{ik}^{-\frac{1}{m-1}}}. \tag{6}$$

Then, each cluster should be rearranged to best fit the renewed fuzzy set.

$$\phi_j = \Upsilon\left(\left\{(\vec{\mathbf{x}}_i, p_{ij}^m) | i = 1, \cdots, n\right\}\right). \tag{7}$$

This process iterates between finding D_{ij}, computing p_{ij}, and rearranging the clusters, until its convergence.

The halting condition of the algorithm is the stationarity of the fuzzy membership maps. Assume that the algorithm has passed through 1 iteration. Also, assume that F_{ijk} is the membership of $\vec{\mathbf{x}}_k$ to the j–th cluster in the i–th iteration. Now, the difference between the result of the i–th iteration and the j–th iteration is defined as,

$$\delta_{i,j} = \sqrt{\frac{1}{WHc} \sum_{k=1}^{n} \sum_{l=1}^{C} (F_{ilk} - F_{jlk})^2}. \tag{8}$$

Also, the repeatedness of the result of the l–th iteration is defined as,

$$\delta_l = \min\left\{\delta_{l,1}, \cdots, \delta_{l,l-1}\right\}. \tag{9}$$

Now, the condition for ending the iterations is $\delta_i < \delta$, where δ is a predefined threshold. In this paper, we use $\delta = 0.05$; which means that the algorithm will stop iterating when the fuzzy membership values repeat with less than 5% variation.

Along with the FCM [29], other clustering approaches like *Gustafson–Kessel*

(GK) [39], *fuzzy elliptotypes clustering* (FEC) [40], and *fuzzy C–varieties* (FCV) [41], are also special cases of the proposed general clustering method. See [35] for further details and mathematical discussions. Figure 2 illustrates a sample run on an artificial 2–D data using the LPRE distance.

Using the clustering procedure proposed above with the LPRE methodology, a color clustering tool is produced [35]. It must be emphasized that this formulation results in a special case of the FCV ($r = 1$) [40]. We call this method *fuzzy principal component analysis–based clustering* (FPCAC).

When feeding an image $\mathbf{I}$ into the FPCAC process (with preselected values for C and m), the result is a set of C fuzzy maps $\mathbf{J}_1, \cdots, \mathbf{J}_C$. Here, $\mathbf{J}_{ixy}$ denotes the level of membership of $\vec{\mathbf{I}}_{xy}$ to the i–th cluster. Note that,

$$\forall x, y, \sum_{i=1}^{C} \mathbf{J}_{ixy} = 1. \tag{10}$$

In this paper, we change the fuzzy maps $\mathbf{J}_1, \cdots, \mathbf{J}_C$ into binary (crisp) maps $\bar{\mathbf{J}}_1, \cdots, \bar{\mathbf{J}}_C$ using maximum likelihood. As such,

$$\bar{\mathbf{J}}_{ixy} = \begin{cases} 1 & \forall j \neq i, \mathbf{J}_{jxy} < \mathbf{J}_{ixy} \\ 1 & \forall j < i, \mathbf{J}_{jxy} < \mathbf{J}_{ixy} \ \ \& \ \ \forall j > i, \mathbf{J}_{jxy} \leq \mathbf{J}_{ixy} \\ 0 & \text{otherwise.} \end{cases}. \tag{11}$$

The complexity in definition of (11) is to compensate for the cases that for a single pair of x and y, there are distinct values of i and j, for which $\mathbf{J}_{ixy} = \mathbf{J}_{jxy}$. As in this paper the binary FPCAC is utilized, we assume that the membership maps are binary ones, without emphasizing that a maximum likelihood has passed over them. Figure 3 shows some examples for applying the FPCAC

method on some sample images. These results are produced using $m = 1.2$ with values of $C = 3, 4, 3, 2$. The algorithm has converged in 14, 11, 12, and 9 successive iterations, elapsing $5.4s$, $5.5s$, $4.2s$, and $1.9s$, respectively.

2.4 Color Image Recoloring

Assume that we have two homogenous swatches of S_1 and S_2. Also, assume that $\vec{\eta}_1$, $\vec{\eta}_2$, $\mathbf{V}_1$, and $\mathbf{V}_2$ denote the expectation vectors and PCA (or FPCA) matrices of S_1 and S_2, respectively. Now, compute the vector $\vec{c}_2$ for the arbitrary vector $\vec{c}_1$ in S_1, as,

$$\vec{\mathbf{c}}_2 = \mathbf{V}_2 \mathbf{V}_1^T (\vec{\mathbf{c}}_1 - \vec{\eta}_1) + \vec{\eta}_2. \tag{12}$$

The vector $\vec{c}_2$ is the recolorized version of $\vec{c}_1$ according to the reference swatch of C_2. Having $\vec{c}_2$, the vector $\vec{c}_1$ is reconstructed as,

$$\vec{\mathbf{c}}_1 = \mathbf{V}_1 \mathbf{V}_2^T (\vec{\mathbf{c}}_2 - \vec{\eta}_2) + \vec{\eta}_1. \tag{13}$$

The above scheme is a PCA–based single–swatch recoloring process introduced in [30].

Figure 4 illustrates a sample result of the above method. Figures 4–a and 4–b show the source and the reference images, respectively. Both images contain two homogenous swatches which are given by the user to guide the recoloring process. Figures 4–c and 4–d illustrate the fuzzy membership maps of the source image according to these two swatches. Figures 4–e and 4–f show the

single–swatch recoloring results and Figure 4–h shows the final result of the recoloring process. The total time elapsed on producing this result is $1.8s$. Note the high quality of the resulting image along with its low computational cost. Also, note the naturalistic contact of the sky and grasses at the horizon. In this paper, we only use the single–swatch recoloring results shown in Figures 4–e and 4–f.

2.5 Polarization and Depolarization

There is a manipulated form of the common *Euler* angles that relates any right–rotating orthonormal matrix (such as V_{ij}) with three angles, in a one–to–one revertible transform [42].

A right–rotating orthonormal matrix is an orthonormal matrix which satisfies, $(\vec{\mathbf{v}}_1 \times \vec{\mathbf{v}}_2) \cdot \vec{\mathbf{v}}_3 > 0$, where, $\times$ and $\cdot$ represent outer and inner products, respectively. Here, $\mathbf{v}_i$ is the i–th column of $\mathbf{V}$. As such, for the right–rotating orthonormal matrix $\mathbf{V}$ we write $\mathbf{V} \sim (\theta, \phi, \psi)$ when,

$$\theta = \angle(\vec{\mathbf{v}}_1^{xy}, [1,0]^T), \tag{14}$$

$$\phi = \angle\Big((\mathbf{R}_\theta^{xy}\vec{\mathbf{v}}_1)^{xz}, [1,0]^T\Big), \tag{15}$$

$$\psi = \angle\Big((\mathbf{R}_\phi^{xz}\mathbf{R}_\theta^{xy}\vec{\mathbf{v}}_2)^{yz}, [1,0]^T\Big). \tag{16}$$

Here, $\angle(\vec{\mathbf{v}}, \vec{\mathbf{u}})$ denotes the angle between two vectors $\vec{\mathbf{v}}, \vec{\mathbf{u}} \in \mathbf{R}^2$, computed as:

$$\angle(\vec{\mathbf{v}}, \vec{\mathbf{u}}) = sgn((\vec{\mathbf{v}} \times \vec{\mathbf{u}}) \cdot \vec{j}) \cos^{-1} \frac{\vec{\mathbf{v}} \cdot \vec{\mathbf{u}}}{\|\vec{\mathbf{v}}\| \|\vec{\mathbf{u}}\|}, \tag{17}$$

where $sgn(x)$ is the *signum* function, defined as:

$$sgn(x) = \begin{cases} 1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0 \end{cases}. \tag{18}$$

In (14), (15), and (16), $\vec{\mathbf{v}}_p$ denotes the project of the vector $\mathbf{v}$ on the plane p. Also, $\mathbf{R}_\alpha^p$ is the 3×3 matrix of α radians counter–clock–wise rotated in the p plane:

$$\mathbf{R}_\theta^{xy} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{19}$$

$$\mathbf{R}_\phi^{xz} = \begin{pmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{pmatrix}, \tag{20}$$

$$\mathbf{R}_\psi^{yz} = \begin{pmatrix} 0 & \cos\psi & -\sin\psi \\ 1 & 0 & 0 \\ 0 & \sin\psi & \cos\psi \end{pmatrix}. \tag{21}$$

One can prove that,

$$\mathbf{R}_\psi^{yz} \mathbf{R}_\phi^{xz} \mathbf{R}_\theta^{xy} \mathbf{V} = \mathbf{I}. \tag{22}$$

Hence, to produce $\mathbf{V}$ out of the triple (θ, ϕ, ψ) one may use,

$$\mathbf{V} = \mathbf{R}_{-\theta}^{xy} \mathbf{R}_{-\phi}^{xz} \mathbf{R}_{-\psi}^{yz}. \tag{23}$$

Note that $(\mathbf{R}_\alpha^p)^{-1} = \mathbf{R}_{-\alpha}^p$.

Then, while equations (14), (15), and (16) compute the three angles θ, ϕ, and ψ out of $\mathbf{V}$, equation (23) reconstructs $\mathbf{V}$ from θ, ϕ, and ψ. These methods are called *polarization* and *depolarization* of a right–rotating orthonormal matrix, respectively [42].

2.6 Color Image Copy Protection

Figure 5 shows the flowchart of the proposed visual encryption/decryption method. Here, for the given image $\mathbf{I}_\circ$, the encryption block produces the encrypted demo image $\tilde{\mathbf{I}}$. It is done in a manner that $\tilde{\mathbf{I}}$ visualizes the contents of $\mathbf{I}_\circ$ while containing deliberately embedded distortions that prevent an unauthorized consumer to use it for professional applications. When the authorization is performed, the key $\mathbf{K}$ is given to the user. Using $\mathbf{K}$ and $\tilde{\mathbf{I}}$, the image $\mathbf{I}$ is reconstructed so that it perfectly duplicates the original image $\mathbf{I}_\circ$. The main contributions of the proposed method are its robust protection against unauthorized duplication and distribution, the low elapsed time for encryption/decryption, high quality of the reconstructed image, and the low redundancy of $\mathbf{K}$ over $\mathbf{I}_\circ$.

Assume that the original $H \times W$ image $\mathbf{I}_\circ$ is to be encrypted by the proposed method. As shown in Figure 5, the FPCA clustering method is performed on $\mathbf{I}_\circ$. The results are a set of C color classes $\phi_i = [\vec{\eta}_i, \mathbf{V}_i]$ for $i = 1, \cdots, C$ along with C, $H \times W$ binary membership maps $\mathbf{J}_1, \cdots, \mathbf{J}_C$. For each value of i, the vector $\vec{\eta}_i$ is the fuzzy expectation vector of the color vectors in the i–th

cluster [35]. Also, the $\mathbf{V}_i$ is the FPCA matrix of the i–th cluster [35].

The FPCAC process depends on the fuzziness parameter (m) with default value of 1.05 for natural color images [35]. As here we do not expect a satisfactory classification (in fact, we prefer some levels of degradation), the m value can be freely selected ($m > 1$). We propose to select m randomly using,

$$m = 1 + \Big|N(0,1)\Big|, \tag{24}$$

where $N(0,1)$ is a zero–mean, unit–variance Gaussian random variable. In fact, m enables the proposed method to give any number of encrypted version of a single image that is needed. Each encrypted image needs its own key to be unleashed. We will show this ability in Section 3.

Here, our main concern is to recolor a given image in an unsupervised manner such that the image content remains understandable, while its quality degrades considerably. Thus, we propose a disturbing multi–swatch recoloring algorithm that is reversible. Assume producing the $H \times W$ index map $\mathbf{J}$ with its elements being members of $\{1, \cdots, C\}$. As such, $\mathbf{J}_{xy} = i$ iff $\mathbf{J}_{ixy} = 1$. As described in Section 2.3, there is one and only one choice for $\mathbf{J}$. Using the single–swatch recoloring method discussed in Section 2.4, we construct the encrypted image as,

$$\vec{\tilde{\mathbf{I}}}_{xy} = \mathbf{U}_i V_i^T (\vec{\mathbf{I}}_{\circ xy} - \vec{\eta}_i) + \vec{\rho}_i, i = \mathbf{J}_{xy}, \tag{25}$$

where $[\vec{\rho}_i, \mathbf{U}_i]$ for $i = 1, \cdots, C$ are a set of color descriptors and x and y are image coordinates. We will return to the above color descriptors later. For

now, it suffices to know that $\vec{\rho}_i$ is a 3×1 vector and $\mathbf{U}_i$ is a 3×3 unitary matrix.

In fact, in (25) each color vector is transferred from the color descriptor to which it mostly belongs to a second color descriptor. Having $\mathbf{J}$ and $[\vec{\rho}_i, \mathbf{U}_i]$ for $i = 1, \cdots, C$, the original image is reconstructed from $\tilde{\mathbf{I}}$. Hence, $\mathbf{J}$ and $\{[\vec{\rho}_i, \mathbf{U}_i] | i = 1, \cdots, C\}$ produce the key ($\mathbf{K}$). The $\mathbf{J}$ is a large image and occupies memory, but due to its content, conventional lossless binary compression methods are highly efficient for compressing it. Furthermore, we propose to apply a convolution kernel of radius r on the membership map of the FPCAC, prior to the maximum likelihood process. This is done to produce an index map with more smoothed edges. This will result in an index map with a few holes and less isolated pixels (that declines the performance of the run–length coding).

Also, we propose to compute the redundancy of the key as its volume in compressed format over the volume of the whole image. Thus, the redundancy of the key $\mathbf{K}$ over the $H \times W$ image $\mathbf{I}_\circ$ is denoted by ε and is computed as,

$$\varepsilon = \frac{\|\mathbf{K}\|}{\|\mathbf{I}\|} = \frac{\|\mathbf{K}\|}{3HW}, \tag{26}$$

where $\|\mathbf{X}\|$ denotes the volume of $\mathbf{X}$ in bytes and $\mathbf{K}$ contains $[\vec{\rho}_i, \mathbf{V}_i]$s and $\mathbf{J}$.

One problem that arises here is that the $\tilde{\mathbf{I}}$ computed in (25) may contain values out of the range of $[0, 255]$. As the result of encryption is saved in one of the standard color image formats (1 byte per pixel), these truncations

produce artifacts in the results of decryption. Thus, we propose to apply a linear mapping on $\tilde{\mathbf{I}}$ and to send the parameters of the mapping, which are a bias and a scale, to the decryption stage.

The decryption process is performed in an straightly inverse fashion. Note that, theoretically, the process is completely reversible. We show (by examples) that except for numerical errors, no information is lost during the encryption–decryption process.

Now, lets go back to the reference color descriptors ($[\vec{\rho}_i, \mathbf{U}_i]$ for $i = 1, \cdots, C$). In fact, as the proposed visual encryption method is making fake edges, it is very important to alter the color appearance of the image deliberately. We propose to use a bank of homogenous swatches for this purpose (see Figure 6). Assume that the bank contains k swatches with the color descriptors of $[\vec{\rho}_i, \mathbf{U}_i]$ for $i = 1, \cdots, k$. The correspondence table in Figure 5 holds an index to this bank for each cluster. The elements of this table are produced by a proposed matching process which compares two color descriptors. The comparison is performed using a weighted Euclidean distance incorporating the expectation vectors and the polarized version of the PCA matrices discussed in Section 2.5.

Doing as such, the distance between a cluster of the image (with expectation vector of $\vec{\eta}$ and FPCA matrix of $\mathbf{V}$) and a homogenous swatch in the bank (with expectation vector of $\vec{\rho}$ and PCA matrix of $\mathbf{U}$) is defined as,

$$\delta^2 = \frac{1}{4\pi^2}\Big[(\theta_{\mathbf{V}} - \theta_{\mathbf{U}})^2 + (\phi_{\mathbf{V}} - \phi_{\mathbf{U}})^2 + (\psi_{\mathbf{V}} - \psi_{\mathbf{U}})^2\Big] + \frac{1}{255^2}\|\vec{\eta} - \vec{\eta}\|^2. \quad (27)$$

The $1/4\pi^2$ and $1/255^2$ coefficients are added to map both angles and color values to the range $[0, 1]$. So, for each cluster, a proper homogenous swatch is selected and its descriptor is used to encrypt the image and is also sent to the decryption process to decrypt the true color vectors. Section 3 holds experimental evaluation of the proposed method.

3 Experimental Results

The proposed algorithm is developed in *MATLAB 6.5*, on a PIV 2600 MHz personal computer with 512 MB of RAM. In practice, *WinZip 7.0 SR-1 (1285)* by *Nico Mark Computing, Inc.* is used as the compression process for the key.

Figure 6 shows the homogenous swatch bank used in this work. The bank contains 49 swatches manually extracted from some color images captured by an *A60 Canon* digital camera. The images are acquired from the natural and artificial objects in daylight or with flash. No other specific constraints are fulfilled. The color descriptors of the swatches of the bank are saved in a *.mat* file occupying $6,184$ bytes of memory (in non–compressed format). After this stage, the swatches are not used and the extracted color information are used.

Figure 7 shows some sample images used in this study. All these images are adopted from color transfer literature. In different experiments, it is proved that the actual number of clusters of the FPCAC is not affecting the performance of the algorithm. In this paper, we illustrate samples relating to the case of $C = 3$, but similar results are obtained for other values of C. In fact,

no user supervision is needed for determining the actual value of C. Note that the actual value of C is a secured parameter, which blocks cheating attempts. The radius of the convolution kernel in all experiments is set to 5. There is no other parameters in the proposed method to be selected by the user. Hence, the method is a fully unsupervised approach.

The proposed encryption/decryption process is performed on images shown in Figure 7. The *peak signal-to-noise ratio* (PSNR) is computed between the original images and the decrypted images. Also, the redundancy of the key is computed. Two elapsed times are measured, t_1 is the time needed to encrypt an image and t_2 is the time needed to decrypt it. Numerical information are listed in Table 2. In this experiment, we have fixed the value of m equal to 1.05. It is observed in this experiment that in all cases, the PSNR is above $45dB$. Note that, as indicated recently, PSNR values of above $38dB$ are visually satisfactory even for professionals [43]. Figure 8 and Figure 9 show the encrypted and decrypted images, respectively. Note the distorted pattern of edges in the pavement, sky, and grass regions in Figures 8–a, 8–b and 8–c. Also, note that in all encrypted images while the contents of the image is readable, it is made unrealistic and unsatisfactory for unauthorized distribution. Also, note the perfect quality of the reconstructed images in Figure 9. None of the intentionally produced artifacts are visible in reconstructed images. Also, the redundancy of the key is less than 2% in all cases and the elapsed time for a set of encryption and decryption stages is $15s$ averagely (always less than $20s$). The interesting feature is that, averagely, 75% of the computational load

is for the supplier, which can have specialized acceleration tools to boost the performance.

Figure 10 illustrates the sequential results of the proposed method. In this figure, Figure 10–a shows the original image, Figure 10–b shows the results of FPCA clustering, and Figure 10–c shows the encrypted image. Then, Figure 10–d shows the results of decryption. In this example, $m = 1.29$ and we have $t_1 = 7.7s$ and $t_2 = 1.7s$. The resulting PSNR is $45dB$ and the redundancy of the key is less than 0.9%.

The important question here is the security of the proposed method. It is not enough to know m to infer the code of an image. This is based on an interesting feature of the FPCA clustering method. In fact, FPCAC is not a repeatable operation. This is theoretically trivial because FPCAC starts from an entirely random set of clusters. Figure 11 shows the results of 9 runs of FPCAC on a single image with a unique set of parameters ($C = 3, m = 1.2$). Table 3 lists the mean square distance between different clustering results. This table shows that there is averagely 16% difference between the clustering results of the same image, compensated for switched clusters. Another interesting feature is that only three pairs are more than 95% alike (a–e, b–i, and f–g). As such, in 9 trials, 6 distinct clustering results are encountered. We emphasize that here we have assumed that the attacker knows C and m, lacking knowledge of which makes cracking the code ultimately harder. Figure 12 shows 9 different encryptions of a single image with different values of m.

Here, we examine a scenario for attacking the proposed encryption method. Assume that the attacker has an encrypted image and a key to a different encryption of that image, and wants to use the key in place. In the worst situation, assume that the two encryptions share the same values of C and m. Figure 13 illustrates this situation. With the desperate results of this experiment (switched code resulted in $PSNR < 12dB$), the high security of the proposed method is clear. The reader should be aware that the difference between two distinct encryption of a single image mainly relies on the different results of the FPCAC. Hence, in a practical system, having the history of encryption will help to avoid producing a similarly–repeated encryption. This will eventually result in maximum security of the proposed method.

We have not discussed the cracking attempt which depends on cheating the index map and the color descriptors manually. We believe that according to the complexity of producing an image–sized indexed map, this is a dead end. Also, note that as the edges of the index map are completely merged with the edges of the original image, it is impractical to even re–engineer the index map by manual efforts.

4 Conclusion

With the recent developments in digital communications and the ease of access to multimedia resources, the need for protection against unauthorized duplication and distribution of digital media has arisen. In this paper, a novel and

efficient unsupervised color image copy protection method is proposed and analyzed. In contrast with the commonly used watermarking approaches that embed the owner authentication data into the original data, the proposed method applies reversible deliberate distortions in the original image. This method permits the efficient reconstruction of the original from and evaluation image which is not appropriate for professional distribution. Furthermore, while almost all of the available watermarking methods are not resistant to obvious attack, the proposed method prevents unauthorized distribution and gives a higher confidence to true ownership of digital media. The method is also fast and reliable.

Acknowledgement

This work was in part supported by a grant from ITRC. The authors wish to thank Dr. Gregory J. McGarry for his editorial comments. Also, the first author wishes to thank Ms. *Azadeh Yadollahi* for her encouragement and invaluable ideas. We also appreciate the respected anonymous referees for their constructive suggestions.

References

[1] M. Swanson, M. Kobayashi, A. Tewfik, Multimedia data–embedding and watermarking technologies, in: Proceedings of the IEEE, Vol. 86(6), 1998, pp. 1064–1087.

[2] J. Ruanaidh, W. Dowling, F. Boland, Watermarking digital images for copyright protection, IEE Proceedings on Vision, Signal and Image Processing 143(4) (1996) 250–256.

[3] M. Kutter, F. Jordan, F. Bossen, Digital watermarking of color images using amplitude modulation, Electronic Imaging 7(2) (1998) 326–332.

[4] P.-T. Yu, H.-H. Tasi, J.-S. Lin, Digital watermarking based on neural networks for color images, Signal Processing 81 (2001) 663–671.

[5] C.-H. Chou, T.-L. Wu, Embedding color watermarks in color images, EURASIP Journal on Applied Signal Processing 1 (2001) 327–332.

[6] P. Tsai, Y.-C. Hu, C.-C. Chang, A color image watermarking scheme based on color quantization, Signal Processing 84 (2004) 95–106.

[7] T.-Y. Chung, M.-S. Hong, Y.-N. Oh, D.-H. Shin, S.-H. Park, Digital watermarking for copyright protection of mpeg2 compressed video, IEEE Transactions on Consumer Electronics 44(3) (1998) 895–901.

[8] C. Busch, W. Funk, S. Wolthusen, Digital watermarking: From concepts to real–time video applications, IEEE Computer Graphics and Applications (1999) 25–35.

[9] J. A. Bloom, I. J. Cox, T. Kalker, J.-P. M. G. Linnartz, M. L. Miller, C. B. S. Traw, Copy protection for dvd video, Proceedings of the IEEE 87(7) (1999) 1267–1276.

[10] M. Maes, T. Kalker, J.-P. M. Linnartz, J. Talstra, G. F. G. Depovere, J. Haitsma, Digital watermarking for dvd copy protection, what issues play a role in designing and effective system, IEEE Signal Processing Magazine (2000) 47–57.

[11] G. C. Langelaar, I. Setyawan, R. L. Lagendijk, Watermarking digital image and video data, a state–of–the–art overview, IEEE Signal Processing Magazine (2000) 20–46.

[12] C. W. Wu, On the design of content-based multimedia authentication systems, IEEE Transactions on Multimedia 4 (3) (2002) 385–393.

[13] N. J. Mathai, D. Kundur, , A. Sheikholeslami, Hardware implementation perspectives of digital video watermarking algorithms, IEEE Transactions on Signal Processing 51(4) (2003) 625–638.

[14] G. Doerr, J.-L. Dugelay, A guide tour of video watermarking, Signal Processing: Image Coomunication 18 (2003) 263–282.

[15] D. Kundur, K. Marthik, Video fingerprinting and encryption principles for digital rights management, Proceedings of the IEEE 92 (6) (2004) 918–932.

[16] D. S. Wallach, Copy protection technology is doomed, IEEE Computer (2001) 48–49.

[17] M. Kutter, F. A. P. Petitcolas, A fair benchmark for image watermarking systems, in: Electronic Imaging'99, Security and Watermarking of Multimedia Content, San Jose, Ca, USA, 1990.

[18] F. A. P. Petitcolas, R. J. Anderson, M. G. Kuhn, Attacks on copyright marking systems, in: D. Aucsmith (Ed.), Information Hiding, Second International Workshop, IH98, Proceedings, LNCS 1525, Springer–Verlag, Portland, Oregon, USA, 1998, pp. 219–239.

[19] F. A. P. Petitcolas, Watermarking schemes evaluation, IEEE Signal Processing 17 (5) (2000) 58–64.

[20] F. A. P. Petitcolas, Stirmark benchmark 4.0, http://www.petitcolas.net/fabien/ watermarking/ evaluation/ index.html (Visited 2004).

[21] S. A. Craver, M. Wu, B. Liu, What can we reasonably expect from watermarks?, in: 2001 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, New Paltz, New York, 2001, pp. 223–226.

[22] B. Grisworld, Stopping webway rubbery: On–line image protection in the digital age, bgriswol@pop200.gsfc.nasa.goc (Visited 2004).

[23] C. Herley, Why watermarking is nonsense, IEEE Signal Processing Magazine (2002) 10–11.

[24] P. Moulin, Comments on "why watermarking is nonsense", IEEE Signal Processing Magazine (2003) 57–59.

[25] What is the future for watermarking (part i), IEEE Signal Processing Magazine (2003) 55–59.

[26] What is the future for watermarking (part ii), IEEE Signal Processing Magazine (2003) 53–57.

[27] C. Kim, Content–based image copy detection, Signal Processing: Image Communication 18 (2003) 169–184.

[28] S.-C. Cheng, S.-C. Hsia, Fast algorithm's for color image processing by principal component analysis, Journal of Visual Communication and Image Representation 14 (2003) 184–203.

[29] J. C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York, 1981.

[30] A. Abadpour, S. Kasaei, A new fast fuzzy color transfer method, in: The 4th IEEE International Symposium on Signal Processing and Information Technology (ISSPIT 2004), Rome, Italy, 2004, pp. 72–75.

[31] L. Lucchese, S. Mitra, Colour segmentation based on separate anisotropic diffusion of chromatic and achromatic channels, Vision, Image, and Signal Processing 148(3) (2001) 141–150.

[32] H. Cheng, J. Li, Fuzzy homogeneity and scale–space approach to color image segmentation, Pattern Recognition 36 (2003) 1545–1562.

[33] T. Chaira, A. Ray, Fuzzy approach for color region extraction, Pattern Recognition 24 (2003) 1943–1950.

[34] A. Abadpour, S. Kasaei, A new parametric linear adaptive color space and its pca–based implementation, in: The 9th Annual CSI Computer Conference, CSICC, Tehran, Iran, 2004, pp. 125–132.

[35] A. Abadpour, S. Kasaei, A Fast and Efficient Fuzzy Color Transfer Method, in: The 4^{th} IEEE International Symposium on Signal Processing and Information Technology, Isspit, Rome, Italy, 2004, pp. 491–494.

[36] A. Abadpour, S. Kasaei, Performance analysis of three homogeneity criteria for color image processing, in: IPM Workshop on Computer Vision, Tehran, Iran, 2004.

[37] D. O. Nikolaev, P. O. Nikolayev, Linear color segmentation and its implementation, Computer Vision and Image Understanding 94 (2004) 115–139.

[38] A. Abadpour, S. Kasaei, New fast fuzzy method for query region extraction in color images, in: 10th Annual CSI Computer Conference (CSICC2005), Tehran, Iran, 2005, pp. 53–59.

[39] D. E. Gustafson, W. C. Kessel, Fuzzy clustering with a fuzzy covariance matrix, in: Proceedings of the IEEE CDC, Vol. 2, San Diego, CA, 1979, pp. 761–766.

[40] J. M. Leski, Fuzzy c–varieties/elliptotypes clustering in reproducing kernel hilbert space, Fuzzy Sets and Systems 141 (2004) 259–280.

[41] K. Honda, N. Sugiura, H. Ichihashi, Robust local principal component analyzer with fuzzy clustering, in: IJCNN 2003 Conference Proceedings, 2003, pp. 732–737.

[42] A. Abadpour, S. Kasaei, A new pca–based robust color image watermarking method, in: the 2nd IEEE Conference on Advancing Technology in the GCC: Challenges, and Solutions, Manama, Bahrain, 2004, pp. 326–331.

[43] S. Katzenbeisser, A. Petitcolas, Information Hiding Techniques for Steganography and Digital Watermarking, Artech House Inc., 2000.

[44] E. Reinhard, M. Ashikhmin, B. Gooch, P. Shirley, Color transfer between images, IEEE Computer Graphics and Applications September/October.

[45] Y. Chang, S. Saito, M. Nakajima, A framework for transfer colors based on the basic color categories, in: Proceedings of the Computer Graphics International (CGI'03), IEEE, 2003.

[46] A. Nikoladis, I. Pitas, Robust watermarking of facial images based on salient geometric pattern matching, IEEE Transaction on Multimedia 2(3) (2000) 172–184.

[47] M.-S. Hsieh, D.-C. Tseng, Y.-H. Huang, Hiding digital watermarks using multiresolution wavelet transform, IEEE Transaction on Industrial Electronics 48(5) (2001) 875–882.

[48] M. Barni, F. Bartolini, A. Piva, Multichannel watermarking of color images, IEEE Transaction on Circuits and Systems for Video Technology 12(3) (2002) 142–156.

[49] M. Kutter, S. Winkler, A vision–based masking model for spread–spectrum image watermarking, IEEE Transaction on Image Processing 11(1) (2002) 16–25.

[50] C.-W. Tang, H.-M. Hang, A feature–based robust digital image watermarking scheme, IEEE Transaction on Signal Processing 51(4) (2003) 950–959.

[51] X. Kang, J. Huang, Y. Q. Shi, Y. Lin, A dwt–dft composite watermarking scheme robust to both affine transform and jpeg compression, IEEE Transactions on Circuits and Systems for Video Technology 13(8) (2003) 776–786.

[52] Shih-Hao, Y.-P. Lin, Wavelet tree quantization for copyright protection watermarking, IEEE Transactions on Image Processing 13(2) (2004) 154–165.

List of Figures

1 Query region extraction using different distance functions. a) Original image. b) Euclidean. c) Mahalanobis. d) LPRE. 35

2 Results of applying the proposed clustering method on an artificial 2–D data cloud using the LPRE distance. (a) Input data. (b) Resulted clusters and the path of clusters for convergence. 36

3 Results of the FPCAC method on some sample images. Top) Original images. Bottom) Results of FPCAC. (a) and (b) Courtesy of Kodak *PhotoCD* (PCD0992). (c) Courtesy of Ohio State University. (d) *Lena*, Courtesy of Signal and Image Processing Institute at University of Southern California. 36

4 Illustrative sample for PCA–based recoloring. (a) Source image. (b) Reference image. (c) and (d) Fuzzy membership maps. (e) and (f) Results of single–swatch recoloring. (g) Final result. 37

5 Flowchart of the proposed copy protection algorithm. 38

6 Homogenous swatch bank. 39

7 Some sample images. (a) *Vincent van Gogh*s *Cafe Terrace on the Place du Forum, Arles, at Night*, *Arles*, September 1888, oil on canvas, image from the Vincent van Gogh Gallery, *http://www.vangoghgallery.com*. (b) and (c) Courtesy of *Oliver Deussen*. (a), (b), and (c) Adopted from [44]. (d) "*Mc. Cormic Creck State Park, Indiana* " by *Mike Briner*, *mbphoto@spraynet.com*, *www.mikebrinerphoto.com*. (e) "*Hanging Lake*" by *Brent Reed*, *brent@reedservices.com*. (f) Adopted from [45]. 40

8 Encrypted versions of the images shown in Figure 7, using proposed algorithm. 41

9 Decrypted versions of the images shown in Figure 8, using proposed method. 42

10 Sequential results of proposed algorithm. (a) Original image, courtesy of Kodak *PhotoCD* (PCD0992). (b) Results of FPCA clustering. (c) Encrypted image. (d) Decrypted image. 43

11 Results of applying FPCAC with a unique set of parameters on a single image for 9 different runs. Original image, courtesy of Signal and Image Processing Institute at University of Southern California. 44

12 Different encryptions of a single image. 45

13 Decryption an image with an invalid key when C and m are known. (a) Original image. (b) and (c) Results of FPCAC. (d) and (e) Decryption with correct keys. (f) and (g) Decryption with invalid (switched) keys. 46

List of Tables

1 Attack resistance performance of some typical watermarking methods (adopted from [42]). –: Not Resistant. ∼: Marginally Resistant. √: Completely Resistant. [Abbreviations: LG: Linear Geometrical Transformation, NLG: Nonlinear Geometrical Transformation, LPO: Linear Point Operations, NLPO:Nonlinear Point Operations, SO: Spatial Domain Operations, EO: Editing Operations, CMP: JPEG Compression]. 33

2 Numerical results of proposed algorithm performed on images shown in Figure 7. [m: Fuzziness. t_1 Encryption time. t_2 Decryption time. PSNR after reconstructing the image. ε Redundancy of key.] 33

3 Mean square distance between different clustering results of a single image with a unique set of parameters. Values less than 0.05 are written in bold. Images are shown in Figure 11. 34

Table 1
Attack resistance performance of some typical watermarking methods (adopted from [42]). –: Not Resistant. ∼: Marginally Resistant. √: Completely Resistant. [Abbreviations: LG: Linear Geometrical Transformation, NLG: Nonlinear Geometrical Transformation, LPO: Linear Point Operations, NLPO:Nonlinear Point Operations, SO: Spatial Domain Operations, EO: Editing Operations, CMP: JPEG Compression].

Method		[2]	[46]	[47]	[5]	[4]	[48]	[49]	[50]	[51]	[52]	[6]
Attack Type	LG	∼	√	∼	∼	∼	√	–	√	√	∼	∼
	NLG	–	–	–	–	–	–	–	–	–	–	–
	LPO	–	–	–	∼	–	–	–	–	–	–	–
	NLPO	–	∼	∼	∼	∼	∼	–	∼	–	–	∼
	SO	–	∼	∼	∼	∼	∼	–	∼	∼	∼	∼
	EO	–	–	–	–	–	–	–	–	–	–	–
	CMP	√	√	√	√	√	√	∼	√	√	√	√

Table 2
Numerical results of proposed algorithm performed on images shown in Figure 7. [m: Fuzziness. t_1 Encryption time. t_2 Decryption time. PSNR after reconstructing the image. ε Redundancy of key.]

Image	**Size**	m	t_1	t_2	PSNR	ε
7–a	482×683	1.05	$8.9s$	$2.8s$	$45.3dB$	0.81%
7–b	896×602	1.05	$14.9s$	$5.2s$	$46.6dB$	0.60%
7–c	896×602	1.05	$16.4s$	$5.1s$	$46.9dB$	0.73%
7–d	600×450	1.05	$8.2s$	$2.6s$	$45.7dB$	1.69%
7–e	600×450	1.05	$10.8s$	$2.7s$	$45.0dB$	1.30%
7–f	680×448	1.05	$8.8s$	$3.0s$	$45.2dB$	0.73%

Table 3
Mean square distance between different clustering results of a single image with a unique set of parameters. Values less than 0.05 are written in bold. Images are shown in Figure 11.

Trial	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)
(a)	0	0.23	0.091	0.17	**0.0042**	0.14	0.12	0.21	0.23
(b)	0.23	0	0.23	0.17	0.23	0.21	0.23	0.059	**0.00063**
(c)	0.091	0.23	0	0.12	0.093	0.086	0.088	0.25	0.23
(d)	0.17	0.17	0.12	0	0.17	0.19	0.2	0.17	0.17
(e)	**0.0042**	0.23	0.093	0.17	0	0.14	0.13	0.21	0.23
(f)	0.14	0.21	0.086	0.19	0.14	0	**0.024**	0.22	0.21
(g)	0.12	0.23	0.088	0.2	0.13	**0.024**	0	0.23	0.23
(h)	0.21	0.059	0.25	0.17	0.21	0.22	0.23	0	0.059
(i)	0.23	**0.00063**	0.23	0.17	0.23	0.21	0.23	0.059	0

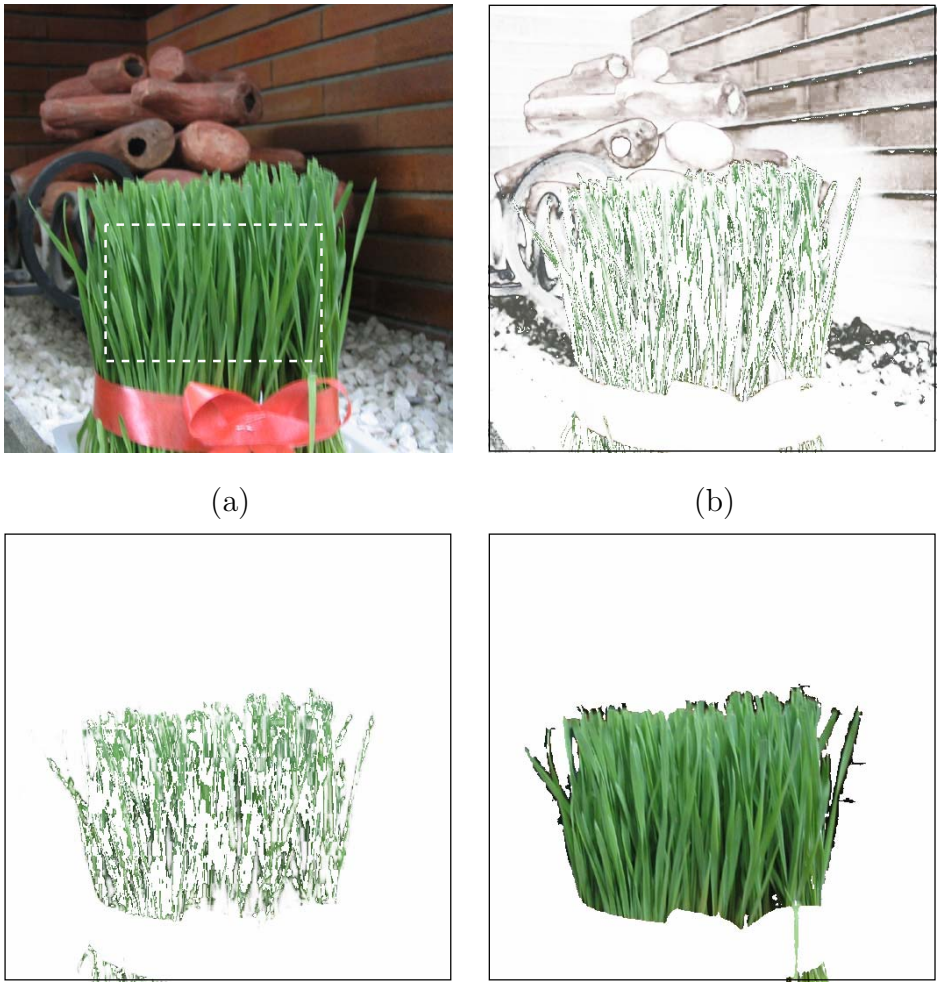
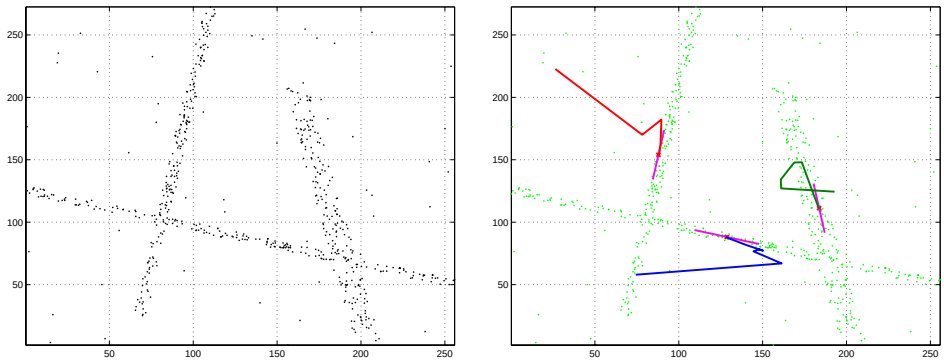
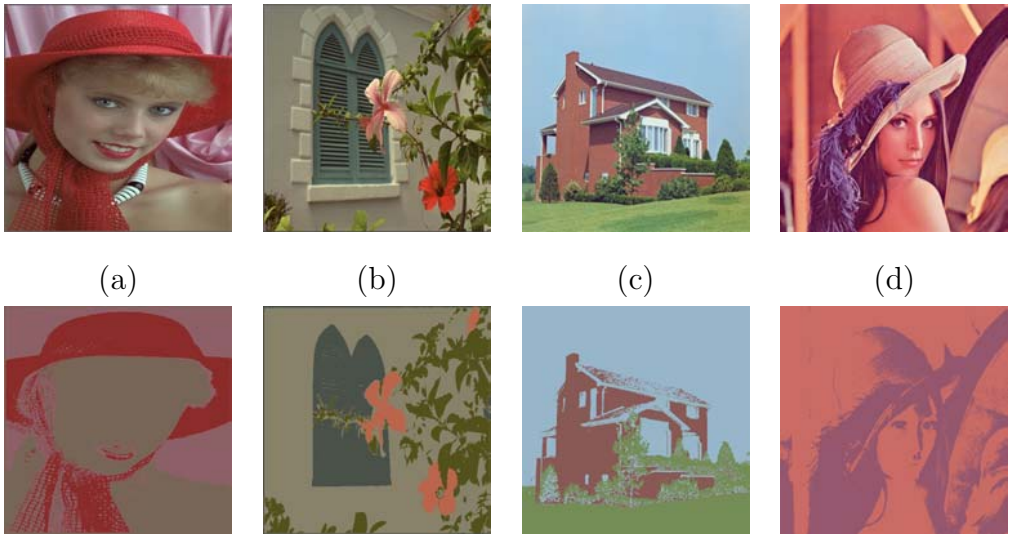


Fig. 1. Query region extraction using different distance functions. a) Original image. b) Euclidean. c) Mahalanobis. d) LPRE.



(a) (b)

Fig. 2. Results of applying the proposed clustering method on an artificial 2–D data cloud using the LPRE distance. (a) Input data. (b) Resulted clusters and the path of clusters for convergence.



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 3. Results of the FPCAC method on some sample images. Top) Original images. Bottom) Results of FPCAC. (a) and (b) Courtesy of Kodak *PhotoCD* (PCD0992). (c) Courtesy of Ohio State University. (d) *Lena*, Courtesy of Signal and Image Processing Institute at University of Southern California.

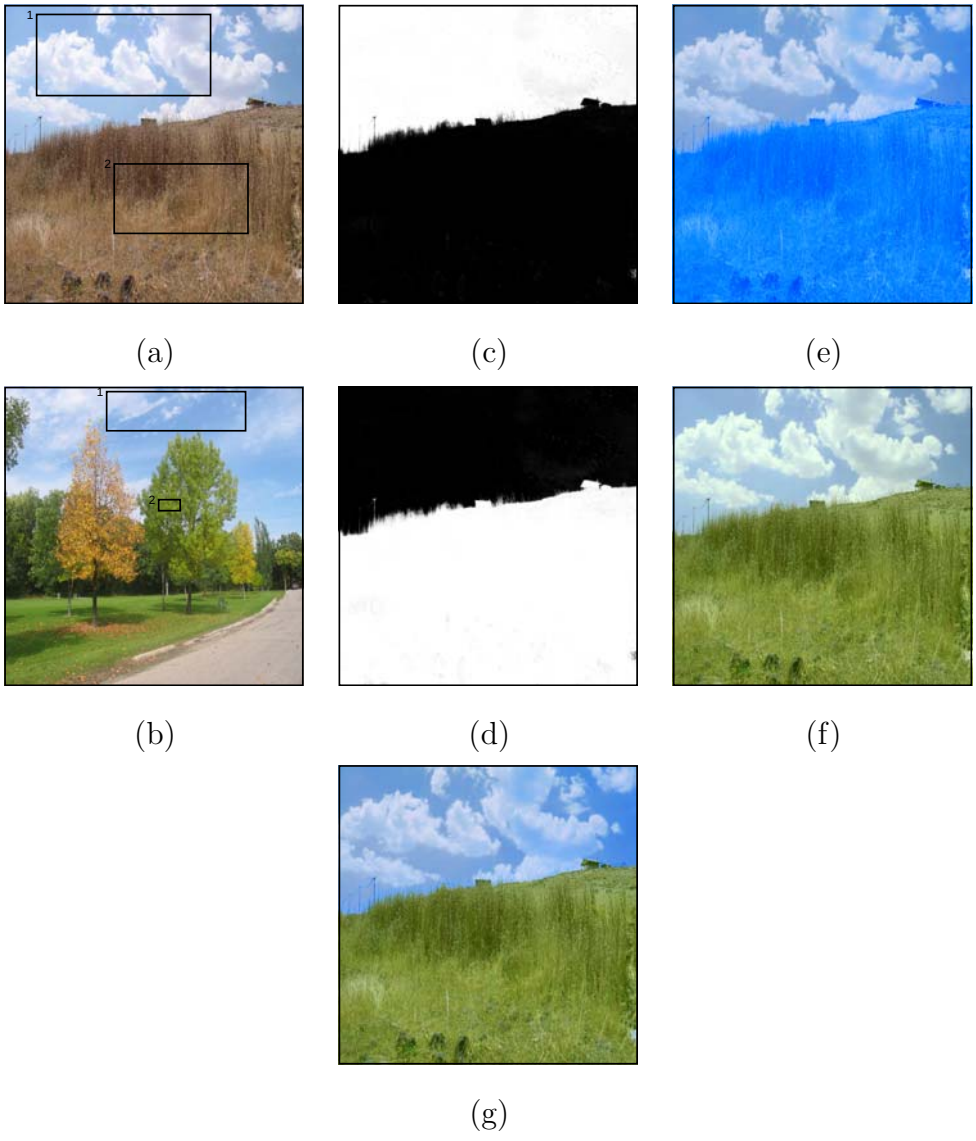


Fig. 4. Illustrative sample for PCA–based recoloring. (a) Source image. (b) Reference image. (c) and (d) Fuzzy membership maps. (e) and (f) Results of single–swatch recoloring. (g) Final result.

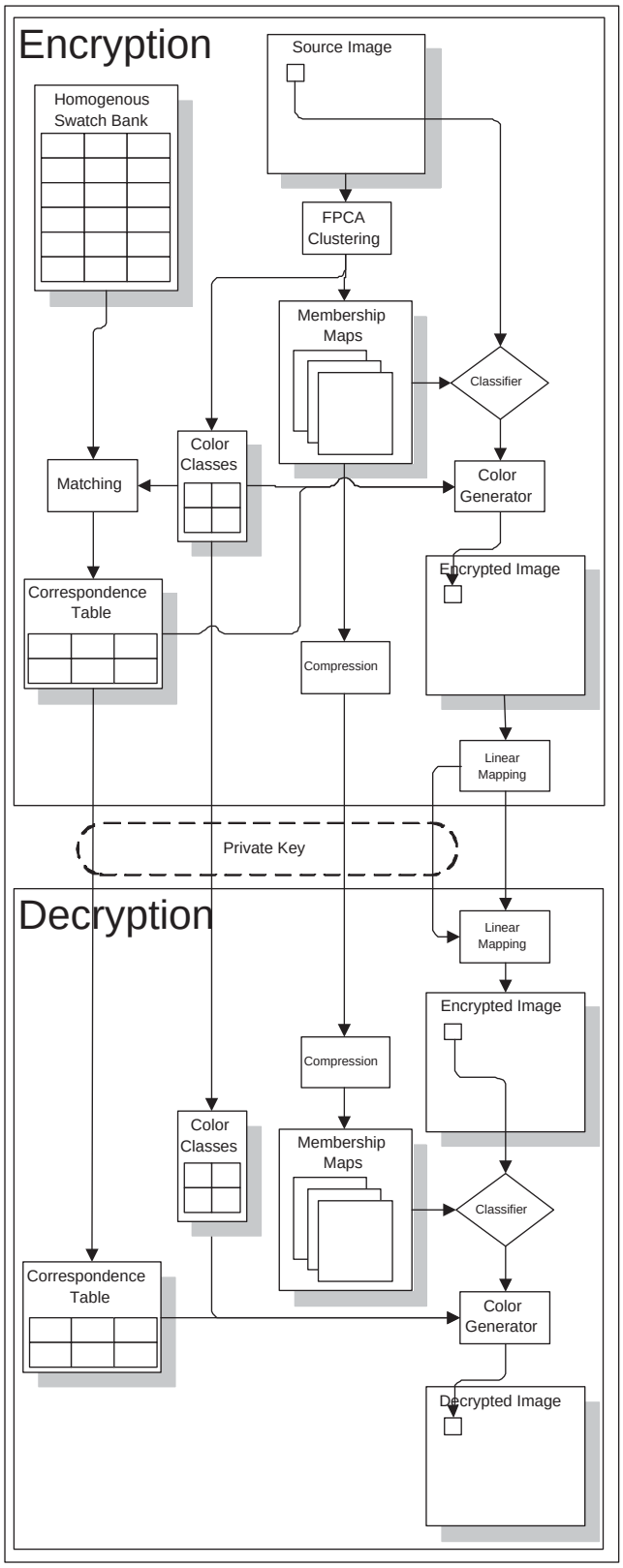


Fig. 5. Flowchart of the proposed copy protection algorithm.

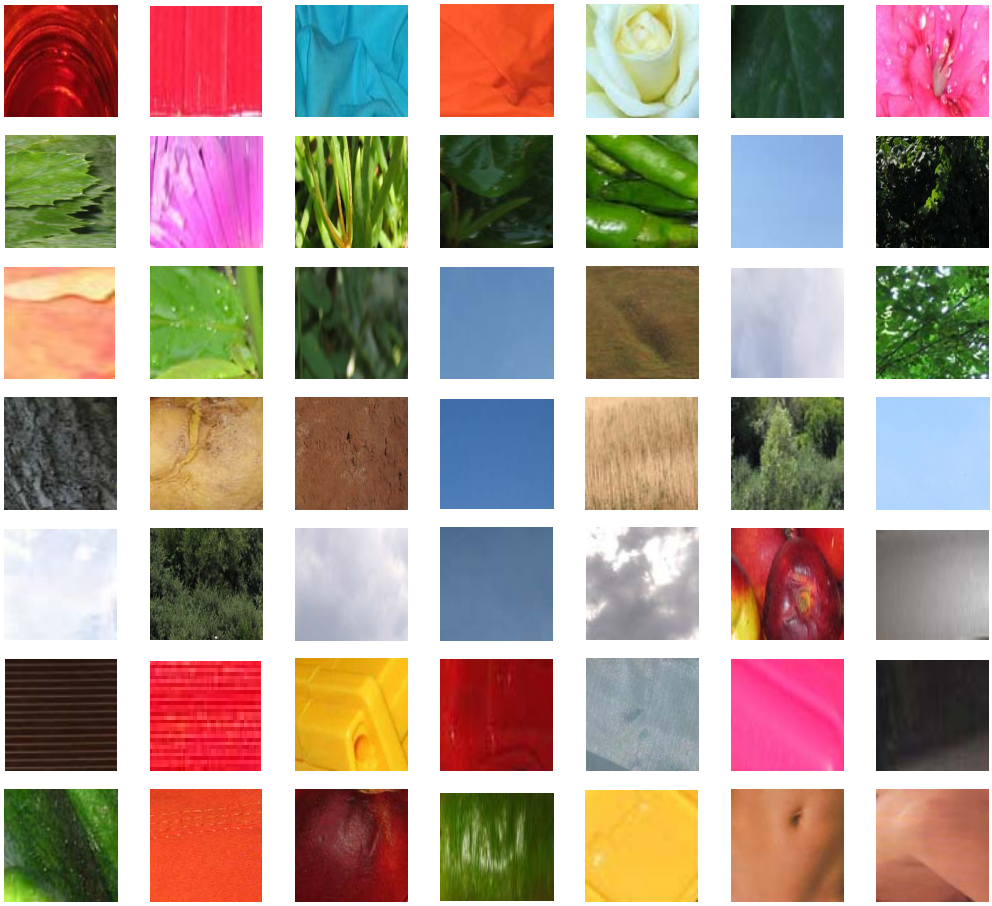


Fig. 6. Homogenous swatch bank.

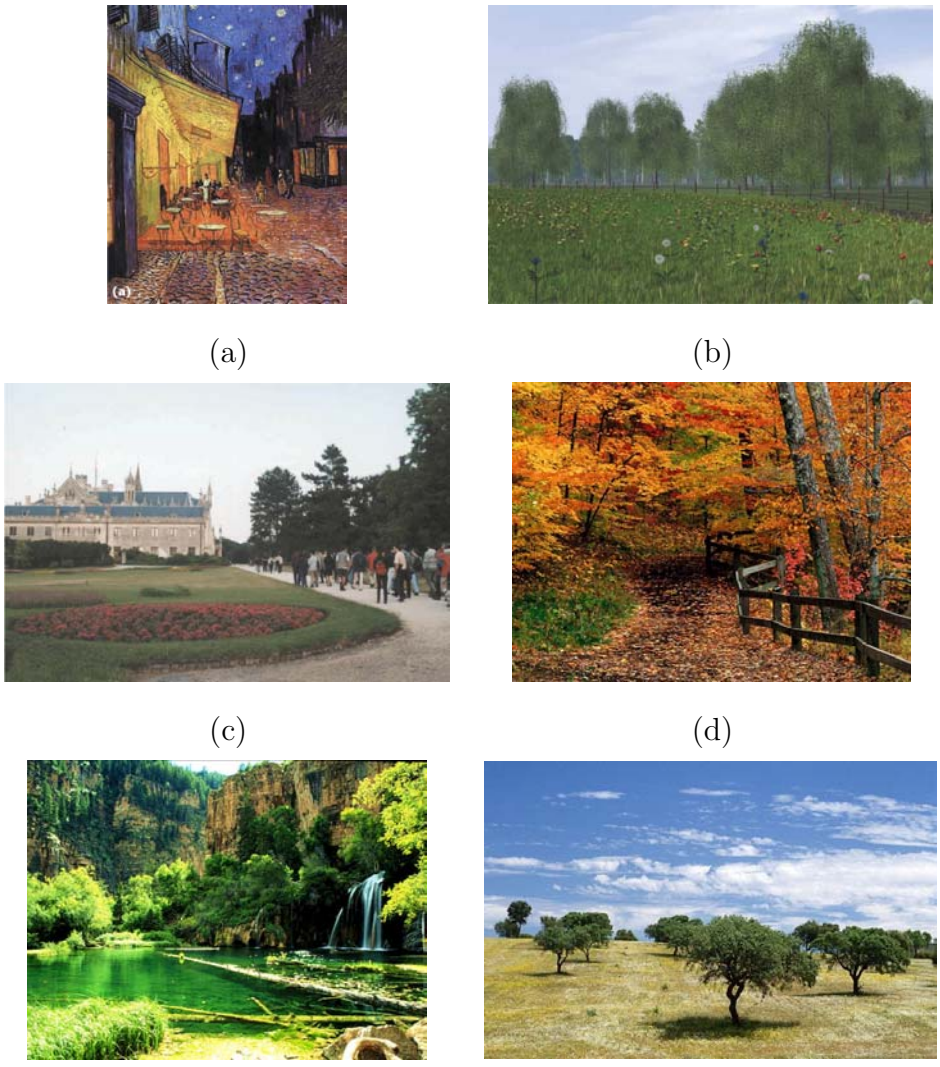


Fig. 7. Some sample images. (a) *Vincent van Goghs Cafe Terrace on the Place du Forum, Arles, at Night*, *Arles*, September 1888, oil on canvas, image from the Vincent van Gogh Gallery, *http://www.vangoghgallery.com*. (b) and (c) Courtesy of *Oliver Deussen*. (a), (b), and (c) Adopted from [44]. (d) "*Mc. Cormic Creck State Park, Indiana* " by *Mike Briner*, *mbphoto@spraynet.com*, *www.mikebrinerphoto.com*. (e) "*Hanging Lake*" by *Brent Reed*, *brent@reedservices.com*. (f) Adopted from [45].

Fig. 8. Encrypted versions of the images shown in Figure 7, using proposed algorithm.

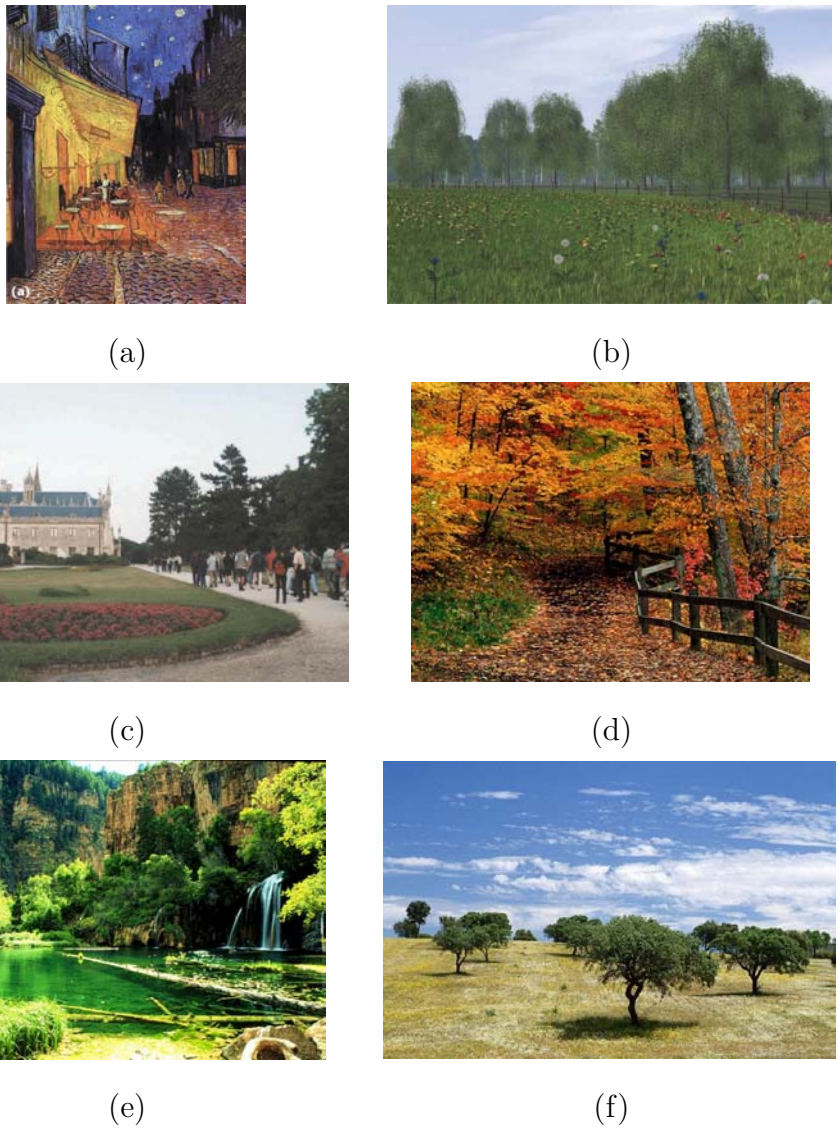


Fig. 9. Decrypted versions of the images shown in Figure 8, using proposed method.

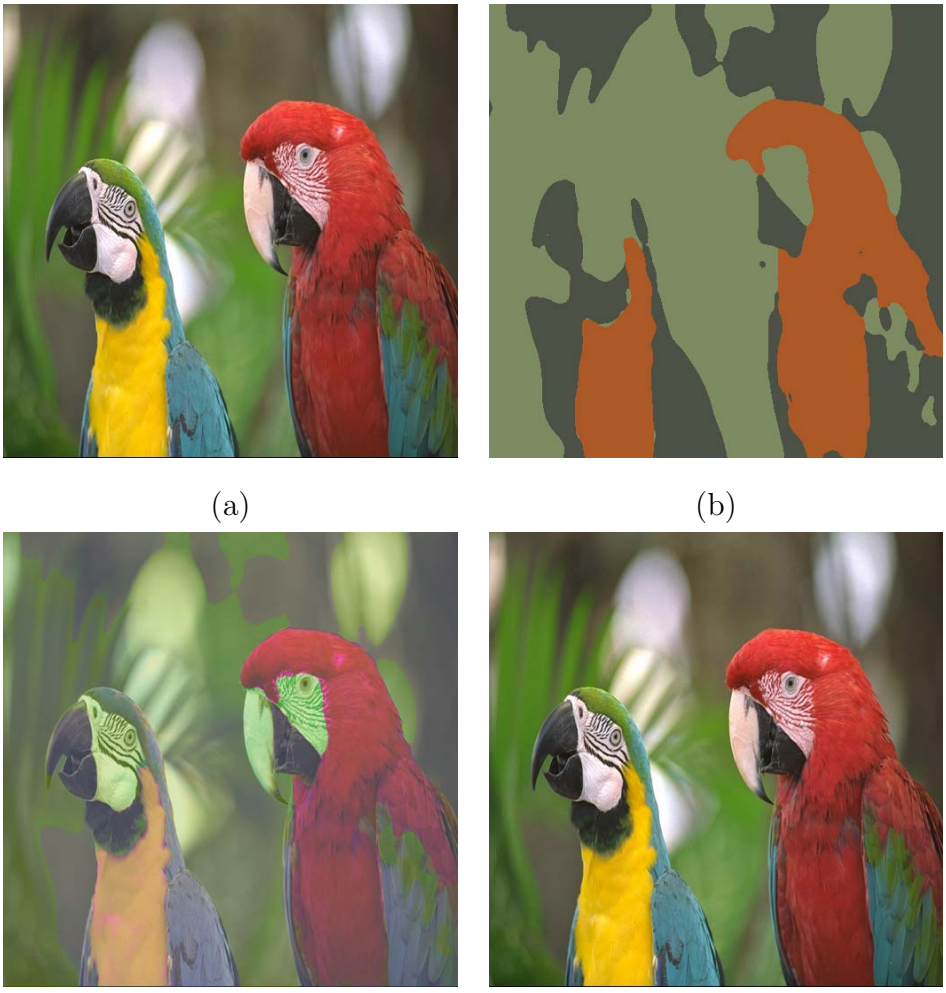


Fig. 10. Sequential results of proposed algorithm. (a) Original image, courtesy of Kodak *PhotoCD* (PCD0992). (b) Results of FPCA clustering. (c) Encrypted image. (d) Decrypted image.

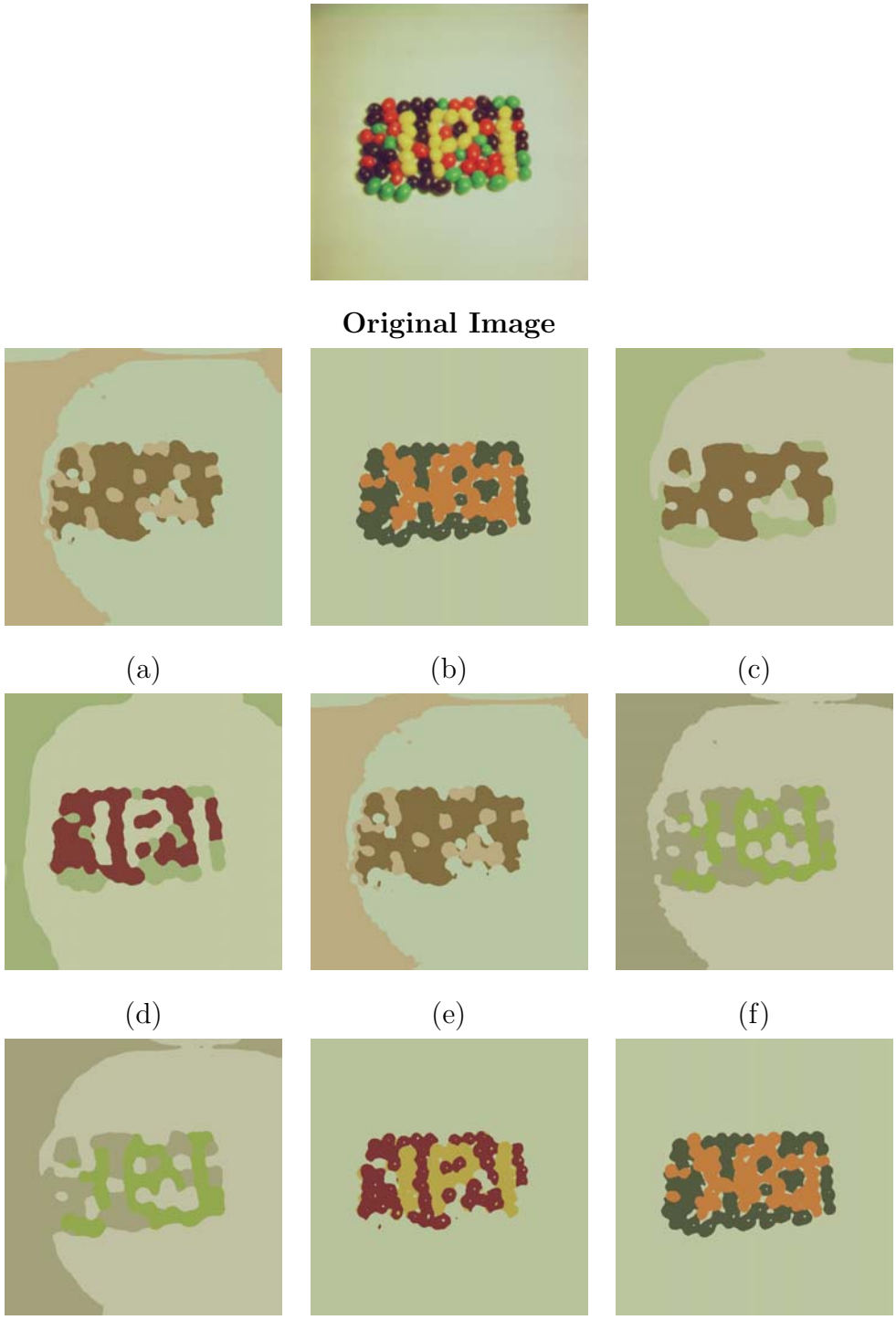


Fig. 11. Results of applying FPCAC with a unique set of parameters on a single image for 9 different runs. Original image, courtesy of Signal and Image Processing Institute at University of Southern California.

(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig. 12. Different encryptions of a single image.

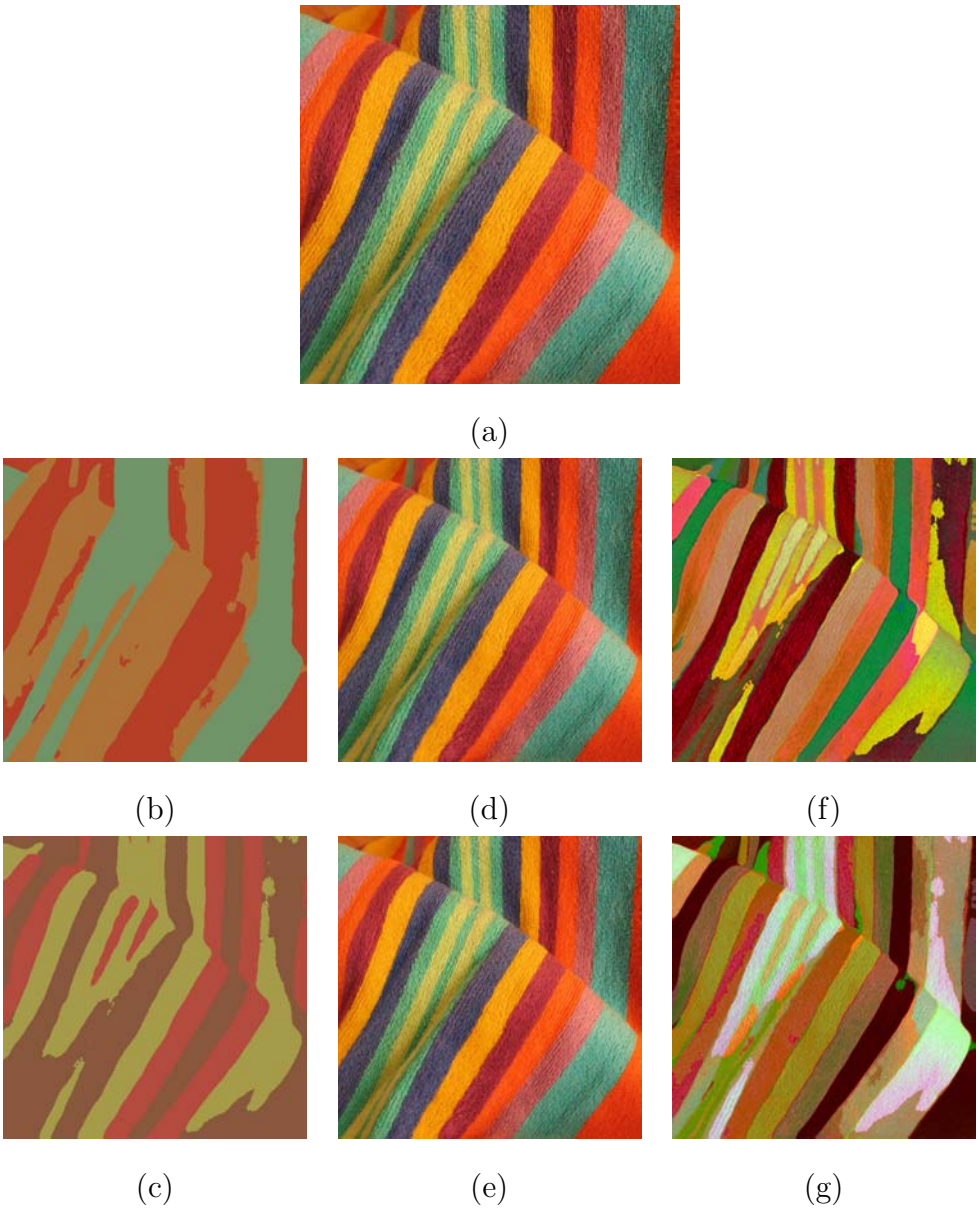


Fig. 13. Decryption an image with an invalid key when C and m are known. (a) Original image. (b) and (c) Results of FPCAC. (d) and (e) Decryption with correct keys. (f) and (g) Decryption with invalid (switched) keys.