

# Principal Color and Its Application to Color Image Segmentation

Arash Abadpour<sup>1</sup> and Shohreh Kasaei<sup>2</sup>

<sup>1</sup> Mathematics Science Department, Sharif Univ. of Tech., Tehran, Iran, email: abadpour@math.sharif.edu

<sup>2</sup> Computer Engineering Department, Sharif Univ. of Tech., Tehran, Iran, email: skasaei@sharif.edu

## Abstract

Color image segmentation is a primitive operation in many image processing and computer vision applications. Accordingly, there exists numerous segmentation approaches in the literature, which might be misleading for a researcher who is looking for a practical algorithm. While many researchers are still using the tools which belong to the old color space paradigm, there are evidences in the research established in eighties that proper descriptor of color vectors should act locally in the color domain. In this paper, we use these results to propose a new color image segmentation method. The proposed method searches for the principal colors, defined as the intersections of the cylindrical representations of homogeneous blocks of the given image. As such, rather than using the noisy individual pixels, which may contain many outliers, the proposed method uses the linear representation of homogeneous blocks of the image. The paper includes comprehensive mathematical discussion of the proposed method and experimental results to show the efficiency of the proposed algorithm.

## I. INTRODUCTION

In 1988, *Klinker, Shafer, and Kanade* presented a novel approach to measure the highlights in color images [1]. In that work, they developed a proper model for the reflected light from an arbitrary point of a dielectric object. In 1990, they applied their approach to color image understanding [2]. However, more than a decade passed since the idea was successfully incorporated into a practical algorithm. In 2003, without paying too much attention to the theoretical aspects, *Cheng and Hsia* used the *principal component analysis* (PCA) for color image processing [3]. Then, in 2004, *Nikolaev and Nikolayev* started the work again from the theory and proved that the PCA is a proper tool for color image processing [4]. The next necessary step had been introduced in 1991, when *Turk and Pentland* proposed their eigenface method [5], in a completely different context. There, they developed a novel idea which connected the eigenproblems in the color domain and the spatial domain together. Although, there is this rich theoretical background for the linear local models of color, it is quite common to see research procedures which are based on the old color space paradigm, even published in 2005. For a more comprehensive discussion of this topic refer to [6].

In different works, the authors have applied the PCA, and then the fuzzy PCA (FPCA), to many different color image processing tasks. The essential idea of those works, which was borrowed from the above-mentioned literature, is that homogeneous swatches in color images of nature produce thin cylinders in the RGB space. In this framework, fuzzy clustering is a proper tool for finding these structures [7]. This theory opens plenty of opportunities for efficient color image processing tasks. Interested readers can refer to [8], [9] for two applications of this theory. Also, see [6] for more examples.

In the PCA framework, each color swatch is represented by a line, which passes through its expectation vector and is parallel to its principal direction. Although, it is proved that the lines representing similar materials lie on a unique plane [1], the theory neglects the (possibly virtual) intersection of these lines. A virtual intersection of a set of lines is a point very close to which almost all of them pass. In this paper, we empirically show that the lines representing similar materials do intersect and show that this intersection leads to a new technique for color image segmentation. We call this intersection point the principal color of the underlying material.

A review of the color image processing literature shows the huge amount of research dedicated to the basic primitives such as color image segmentation. This results in a vast variety of approaches with different motivations and tools. The reader is referred to [10], [11] and the references therein for more details. Although, there are many available segmentation algorithms, no numerical criterion is standardized to give a reasonable comparison between the results of different approaches. In fact, the only globally-accepted, and referred to, factors are the elapsed time and the visual properness of the results. Furthermore, the utilization of many techniques includes many parameters which must be tuned by an expert. Failing to do the required tuning, may result in total or partial collapse of the method (see [12] as an example). In this paper, we compare the performance of the proposed method with that of an available approach which is based on the same theory [7] and show that the new method outperforms the previous one. For the comparison of that method with the available literature refer to [7].

The rest of this paper is organized as follows. Section II introduces the concept of principal colors and gives experimental cues for their existence. Then, it describes the proposed method to use principal colors to perform color image segmentation. Section III holds the experimental results and discussions and finally Section IV concludes the paper.

## II. METHOD

Here, we propose an investigation, through which the existence of the principal colors could be empirically perceived. We emphasize that this empirical result has to be confirmed by theoretical investigation. We believe that the increasing number of research projects devoted to processing color images will pave the way to the formal approval of this theory, something which is outside the scope of this paper.

Figure 1 shows three sets of color swatches representing *skin*, *sky*, and *leaf* materials. Each set contains 30 images, each a  $64 \times 64$  JPEG file with quality of 100. The skin samples are extracted from erotic images downloaded from free resources on the Internet. The sky and leaf samples are captured by a *Cannon A60* digital camera in different times of the day and in different locations. For each category, the lines representing each single swatch are drawn in a common *RGB* space. Figure 2 shows these three sets of lines. Note that for each material, all the lines pass through a small region in the color space. We call this (virtual) intersection of the lines, the principal color of that material. Figure 3 shows the principal colors of skin, sky, and leaf extracted by the proposed method. Section II-A gives the mathematical details about the process which finds this point. Section II-B develops the idea further to include a set of fuzzy (or weighted) lines. Section II-D incorporates the principal color theory with the general clustering algorithm described in Section II-C to propose a new color image segmentation method.

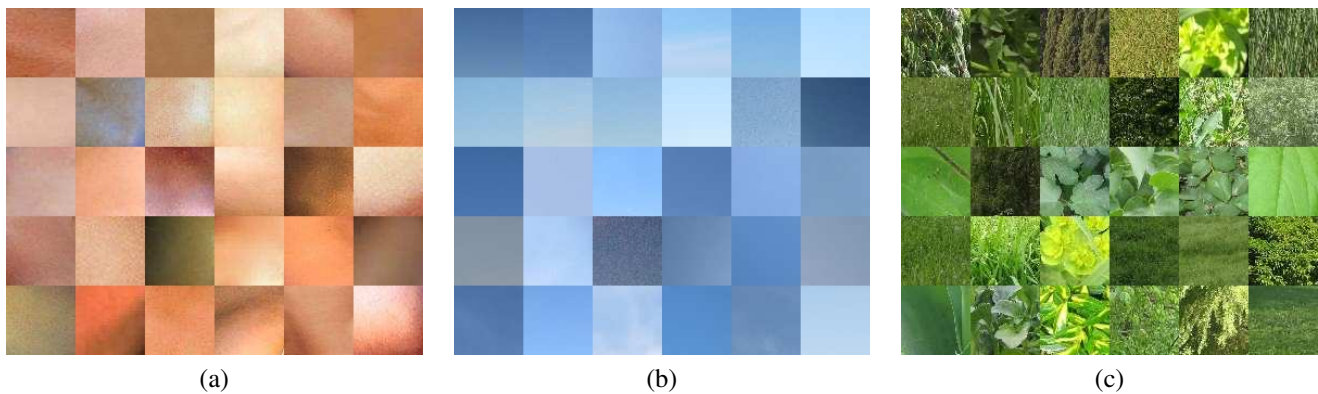


Fig. 1. Randomly selected samples representing three different materials. (a) Skin. (b) Sky. (c) Leaf.

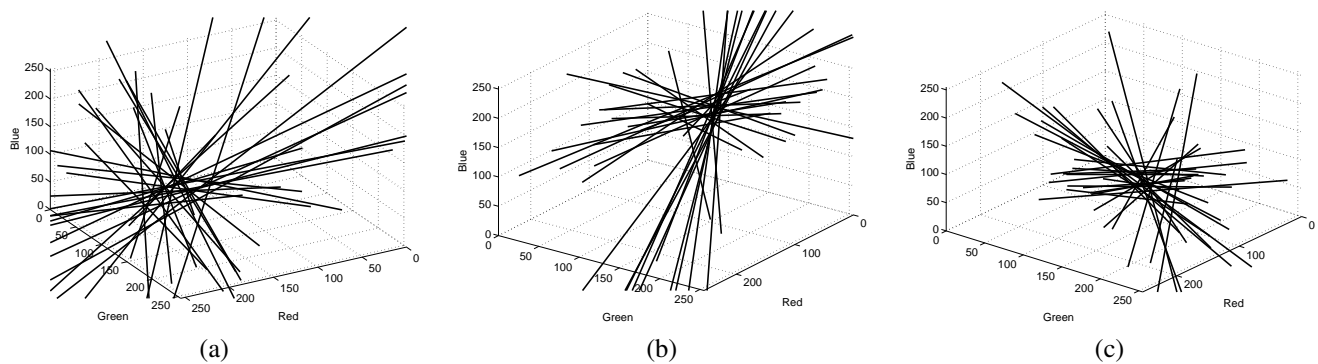


Fig. 2. Linear representation of the samples shown in Figure 1. (a) Skin. (b) Sky. (c) Leaf.

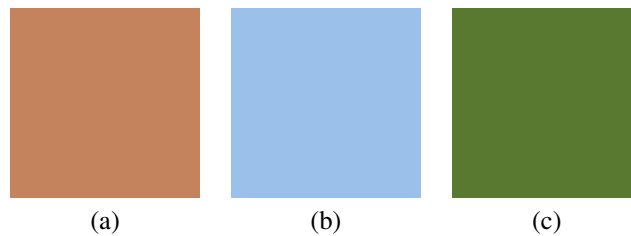


Fig. 3. Principal color of the samples shown in Figure 1 extracted by the proposed method. (a) Skin. (b) Sky. (c) Leaf.

### A. Finding the Closest Point to a Set of Lines

Assume that  $n$  lines of  $l_1, \dots, l_n$  in  $\mathbb{R}^m$  are given. Also, assume that  $l_i$  is indicated by the two vectors  $\vec{\eta}_i$  and  $\vec{v}_i$ . As such,  $\vec{\eta}_i$  is a vector starting from the origin (with its head on  $l_i$ ) and  $\vec{v}_i$  is the direction of  $l_i$ , satisfying  $\|\vec{v}_i\| = 1$ . Hence, the points on  $l_i$  occupy the set  $\mathbf{L}_i \subset \mathbb{R}^m$ , which is defined as,

$$\mathbf{L}_i = \{\vec{\eta}_i + \lambda \vec{v}_i | \lambda \in \mathbb{R}\}. \quad (1)$$

We assume that  $\vec{\eta}_i$  is perpendicular to  $\vec{v}_i$  ( $\vec{\eta}_i^T \vec{v}_i = 0$ ). If this condition is not met,  $\vec{\eta}_i$  is replaced with,

$$\vec{\eta}_i^* = \vec{\eta}_i - (\vec{v}_i^T \vec{\eta}_i) \vec{v}_i. \quad (2)$$

The closest point to the  $n$  lines,  $l_1, \dots, l_n$ , is the point  $\vec{x} \in \mathbb{R}^m$  which minimizes the following objective function,

$$\Delta = \sum_{i=1}^n d(\vec{x}, l_i)^2, \quad (3)$$

where,  $d(\vec{x}, l_i)$  is the distance from the point  $\vec{x}$  to the line  $l_i$ . Linear algebra shows that,

$$d(\vec{x}, l_i)^2 = \|(\vec{x} - \vec{\eta}_i) - [\vec{v}_i^T (\vec{x} - \vec{\eta}_i)] \vec{v}_i\|^2, \quad (4)$$

which is simplified to,

$$d(\vec{x}, l_i)^2 = (\vec{x} - \vec{\eta}_i)^T [\mathbf{I} - \vec{v}_i \vec{v}_i^T] (\vec{x} - \vec{\eta}_i). \quad (5)$$

Substituting (5) in (3) gives,

$$\Delta = \sum_{i=1}^n (\vec{x} - \vec{\eta}_i)^T [\mathbf{I} - \vec{v}_i \vec{v}_i^T] (\vec{x} - \vec{\eta}_i). \quad (6)$$

From calculus we know that [13],

$$\frac{\partial (\vec{x}^T \mathbf{A} \vec{x})}{\partial \vec{x}} = \mathbf{A} \vec{x} + \mathbf{A}^T \vec{x}. \quad (7)$$

Hence,

$$\frac{\partial \Delta}{\partial \vec{x}} = 2 \sum_{i=1}^n [\mathbf{I} - \vec{v}_i \vec{v}_i^T] (\vec{x} - \vec{\eta}_i). \quad (8)$$

Setting  $\frac{\partial \Delta}{\partial \vec{x}} = 0$  gives,

$$\left[ \mathbf{I} - \frac{1}{n} \sum_{i=1}^n \vec{v}_i \vec{v}_i^T \right] \vec{x} = \frac{1}{n} \sum_{i=1}^n \vec{\eta}_i. \quad (9)$$

Now, we will prove that the  $n \times n$  matrix on the left hand side of (9), which we call  $\mathbf{U}$ , is always non-singular, given that not all  $\vec{v}_i$ s are parallel.

A theory in linear algebra states that, “if  $\mathbf{I} + \mathbf{E}$  is singular then  $\|\mathbf{E}\| \geq 1$ , where  $\|\cdot\|$  is a subordinate matrix norm” [14]. We will first prove this theorem. Assume that  $\mathbf{I} + \mathbf{E}$  is singular. Hence, there exists a non-zero vector  $\vec{x}$  satisfying  $(\mathbf{I} + \mathbf{E})\vec{x} = \vec{0}$ . Thus,  $\vec{x} = -\mathbf{E}\vec{x}$  or equivalently,

$$\|\mathbf{E}\| \geq \frac{\|\mathbf{E}\vec{x}\|}{\|\vec{x}\|} = 1. \quad (10)$$

Thus, to prove that  $\mathbf{U}$  in (9) is non-singular, it suffices to show that,

$$\left\| \sum_{i=1}^n \vec{v}_i \vec{v}_i^T \right\| < n. \quad (11)$$

We prove (11) by showing that for any vector  $\vec{y}$  satisfying  $\|\vec{y}\| = 1$ , we have,

$$\left\| \sum_{i=1}^n \vec{v}_i \vec{v}_i^T \vec{y} \right\| < n, \quad (12)$$

First, we know that for any two arbitrary  $m \times 1$  vectors  $\vec{u}$  and  $\vec{v}$ ,

$$|\vec{u}^T \vec{v}| \leq \sqrt{\|\vec{u}\|^2 \|\vec{v}\|^2}. \quad (13)$$

Note that equality occurs for  $\vec{u} \parallel \vec{v}$ . Thus, in the case discussed here we have  $|\vec{v}_i^T \vec{y}| \leq 1$ , which results in,

$$\left\| \sum_{i=1}^n \vec{v}_i \vec{v}_i^T \vec{y} \right\| \leq \sum_{i=1}^n |\vec{v}_i^T \vec{y}| \|\vec{v}_i\| \leq \sum_{i=1}^n \|\vec{v}_i\| = n. \quad (14)$$

Here, the equality happens only if all  $\vec{v}_i$ s are parallel. Thus, ignoring this very especial case, (12) is proved, proving that the solution to (9) exists and is,

$$\vec{x} = \left[ n\mathbf{I} - \sum_{i=1}^n \vec{v}_i \vec{v}_i^T \right]^{-1} \sum_{i=1}^n \vec{\eta}_i. \quad (15)$$

Here,  $\vec{x}$  is the closest point to the  $n$  lines  $l_1, \dots, l_n$ .

### B. Finding the Closest Point to a Set of Fuzzy Lines

Assume that we are looking for the closest point to the set of  $n$  lines  $l_1, \dots, l_n$  in  $\mathbb{R}^m$ . Also, assume that the importance of  $l_i$  is  $p_i \geq 0$ . This means that when composing the objective function, we will weigh satisfaction of  $l_i$  by  $p_i$ . Hence, as  $p_i$  grows, the importance of  $l_i$  in decisions grows linearly, and the vice versa. As such, we are looking for  $\vec{x} \in \mathbb{R}^m$  which minimizes,

$$\tilde{\Delta} = \sum_{i=1}^n p_i d(\vec{x}, l_i)^2. \quad (16)$$

Note that setting  $\forall i, p_i = 1$  converts (16) into (3). An analysis similar to the one performed in Section II-A reveals that  $\vec{x}$  is computed as,

$$\vec{x} = \left[ \sum_{i=1}^n p_i \mathbf{I} - \sum_{i=1}^n p_i \vec{v}_i \vec{v}_i^T \right]^{-1} \sum_{i=1}^n p_i \vec{\eta}_i. \quad (17)$$

Again note that assuming  $\forall i, p_i = 1$  converts (17) into (15), as expected.

### C. General Clustering Algorithm

Assume that the fuzzy set of  $n$  vectors  $\vec{x}_1, \dots, \vec{x}_n$  in  $\mathbb{R}^m$  is given as  $\tilde{\mathbf{X}} = \{(\vec{x}_i; p_i) \mid i = 1, \dots, n\}$ . Also, assume that a human observer has a perception of homogeneity for the subspaces of this space. For example, for  $m = 3$  every human being is able to think and argue about the homogeneity of a set of color vectors. Also, when considering lines in  $\mathbb{R}^3$ , which are members of  $\mathbb{R}^6$ , the meaning of homogeneity used here is the existence of a point which is very close to almost all of them. In many occasions, a homogeneous set may be parameterizable. For example, research shows that a homogeneous set of color vectors occupies a thin cylinder which will be described by two vectors. Similarly, a set of (virtually) intersecting lines may be parameterized by a single point. Thus, we assume that there is a function  $\Upsilon$  that extracts this parametrization for the homogeneous fuzzy set  $\tilde{\mathbf{X}}$ . As an example, if a homogeneous set in  $\mathbb{R}^m$  is defined as a sphere, the  $\Upsilon$  function will be a fuzzy expectation operator. Also, for the cylindrical homogeneity,  $\Upsilon$  will be the FPCA [7]. Having defined a description of a homogeneous set as  $\phi$ , there might exist a function  $\Psi$  which measures the distance from an arbitrary element to  $\phi$ . For the spherical example, the  $\Psi$  function is the Euclidean distance. Also, the  $\Psi$  function of the cylindrical homogeneity, is the *linear partial reconstruction error* (LPRE) distance proposed in [6].

Having proper dual functions  $\Upsilon$  and  $\Psi$ , and a fuzzy set  $\tilde{\mathbf{X}}$ , an important problem is how to separate  $\tilde{\mathbf{X}}$  into  $c$  homogenous clusters, where  $c$  is known. This problem is called the general clustering problem [7]. The special cases of this problem are the *fuzzy C-means* (FCM) [15] (in which the clusters are spherical) and *fuzzy C-varieties* (FCV) [16] (in which the clusters are subspaces). Other examples are *Gath—Geva* [17], fuzzy ellipotypes [18], *Gustafson—Kessel* [19], and FPCA-based clustering (FPCAC) [7] algorithms. Figure 4 shows the pseudo-code of an iterative solution to the general clustering problem. The interested reader is referred to [8] for the related mathematics and the proof of the convergence.

As shown in Figure 4, the algorithm first randomly initializes the clusters. Then, it iterates between computing the distance of each member to each cluster and then renewing each cluster according to the membership value of the realizations to it. The algorithm contains an important parameter of fuzziness [20],  $m$ , which is always more than unity. The general clustering algorithm enables the clustering of any kind of data, given that a proper distance function and its dual are available. In Section II-D we will give a distance function which searches a set of lines for lines passing through or getting very close to a common point. The distance function is then integrated into a novel color image segmentation procedure.

### D. Color Image Segmentation using Principal Colors

Assume that the  $H \times W$  color image  $I$  is given. Also, assume that the integer  $c$  is given as the number of homogeneous segments that  $I$  should be split into. For now, assume that both  $H$  and  $W$  are divisible by  $2^n$ , where  $n$  is an integer larger than 2. It is clear that, using zero-padding or resampling, the same operation is applicable to any image. We propose to cut  $I$  into  $2^{-2n}WH$  non-overlapping rectangular blocks of  $2^n \times 2^n$  pixels.

In [6], the authors proposed a novel homogeneity criterion for color swatches. In that work, the LPRE criterion was compared to the Euclidean and Mahalanobis distances and its superiority was proven. As such, the non-homogeneity of the swatch  $\mathbf{i}$  is defined as [6],

$$\tau(\mathbf{i})^2 = E_{\vec{c} \in \mathbf{i}} \left\{ \left\| (\vec{c} - \vec{\eta}_i) - \vec{v}_i^T (\vec{c} - \vec{\eta}_i) \vec{v}_i \right\|^2 \right\}. \quad (18)$$

- **Aim:** Clustering data according to the given model.
- **Inputs:**
  - Appropriate distance function ( $\Psi$  and its dual  $\Upsilon$ ).
  - Set of realizations ( $\mathbf{X} = \{\vec{\mathbf{x}}_1, \dots, \vec{\mathbf{x}}_n\}$ ).
  - Number of clusters ( $c$ ).
  - Fuzzyness ( $m$ ).
  - Halting threshold ( $\delta$ ).
- **Outputs:**
  - Fuzzy membership values ( $p_{ij}$ ).
  - $c$  cluster descriptors ( $\phi_1, \dots, \phi_c$ ).
- **Method:**
  - 1  $k = 0$  and randomize  $\phi_1, \dots, \phi_c$ .
  - 2  $k = k + 1$ .
  - 3  $D_{ij} = \Psi(\vec{\mathbf{x}}_i, \phi_j)$ .
  - 4  $p_{ij} = D_{ij}^{-1/(m-1)} / \sum_{k=1}^c D_{ik}^{-1/(m-1)}$ .
  - 5  $F_{kij} = p_{ij}$ .
  - 6  $\phi_j = \Upsilon\{(\vec{\mathbf{x}}_1; p_{1j}^m), \dots, (\vec{\mathbf{x}}_n; p_{nj}^m)\}$ .
  - 7 if  $k = 1$ , then goto 2.
  - 8  $\delta_k^2 = \min_{1 \leq p \leq l-1} E_{i,j} \left\{ (F_{pij} - F_{kij})^2 \right\}$ .
  - 9 if  $\delta_k > \delta$ , then goto 2, else return.

Fig. 4. Pseudo-code of the general clustering algorithm [7].

Here,  $\vec{\eta}_i$  is the expectation vector of  $\mathbf{i}$  and  $\vec{\mathbf{v}}_i$  is the principal direction of  $\mathbf{i}$ . Also,  $\|\vec{\mathbf{x}}\|$  is the Euclidean length of the vector  $\vec{\mathbf{x}}$ . We use  $\tau(\mathbf{i}) < \theta$  to decide whether a swatch is homogeneous or not, where  $\theta$  is a constant threshold. Experiments show that  $\theta = 10$  is a proper choice [6].

In the proposed segmentation method, from the  $2^{-2n}WH$  blocks, those which are not homogeneous are deleted. For the remaining blocks, the PCA representations are saved as the  $m$  lines of  $l_1, \dots, l_m$ , all in  $\mathbb{R}^3$ . Now, the problem is to find  $c$  proper intersections of these lines. Using the notations of the general clustering algorithm (see Section II-C) each realization is a line in  $\mathbb{R}^3$  and the cluster model is a 3-D point. As such, the distance between the line  $l$  (denoted by  $\vec{\eta}$  and  $\vec{\mathbf{v}}$ ) and the point  $\vec{\mathbf{x}}$  is defined as,

$$\Psi(l, \vec{\mathbf{x}}) = \|(\vec{\mathbf{x}} - \vec{\eta}) - \vec{\mathbf{v}}^T (\vec{\mathbf{x}} - \vec{\eta}) \vec{\mathbf{v}}\|^2. \quad (19)$$

In this framework, the dual function  $\Upsilon$  is given by (17). Here, the fuzzy set is  $\tilde{\mathbf{L}} = \{(l_i; p_i) | i = 1 \dots, m\}$ , where  $l_i$  is represented by  $\vec{\eta}_i$  and  $\vec{\mathbf{v}}_i$ .

The general clustering algorithm results in a set of  $c$  points in  $\mathbb{R}^3$ , where each point is the principal color of the respective cluster. Note that at the beginning of this process, the non-homogeneous blocks were removed. That was done because the principal direction of those blocks are not meaningful. Therefore, at this point, the process should assign each block to a cluster, even those that are not homogeneous enough. Also, note that the results of the general clustering algorithm is a set of fuzzy membership values for each block and each cluster. Thus, using a maximum-likelihood operator, each block is deterministically assigned to one of the clusters. By finding the distance of each omitted block to all clusters, the process assigns them to one of the computed clusters. The next step is to interpolate the assignment for all pixels. As each block is a  $2^n \times 2^n$  swatch, performing bilinear double-sampling on the resulting index map, for  $n$  times, yields the desired segmentation. In practice, blocks of size  $2^{n_1} \times 2^{n_1}$  are used to find the clusters and then  $2^{n_2} \times 2^{n_2}$  blocks are examined to compute the index-map.

### III. EXPERIMENTAL RESULTS

The proposed algorithm is developed in *MATLAB* 6.5, on a PIV 2600 MHz personal computer with 256 MB of RAM. All sample images used in this paper are  $512 \times 512$  color images in JPEG format with quality of 100 (see Figure 5).

Figure 6 shows the results of three independent runs of the proposed method on the image *Peppers*. Here,  $m = 2$ ,  $c = 2$ ,  $n_1 = 4$ , and  $n_2 = 3$  are used. This figure shows that while the proposed method always results acceptably, its output is never identical in different runs, even with the identical values of the parameters, similar to the entire category of fuzzy clustering algorithms [21]. It should be emphasized, though, that in contrast with the conventional pixel-based fuzzy clustering approaches, in which local minimum is a common unsolved problem [21], the proposed method shows less tendency to falling into local minima. A local minimum in a fuzzy clustering algorithm can be identified by a solution which does not satisfy the merits which the problem defines but still cannot be improved by the local approach utilized in the general clustering algorithm [21]. We argue that the relative immunity of the proposed method against getting trapped in local minimums is the

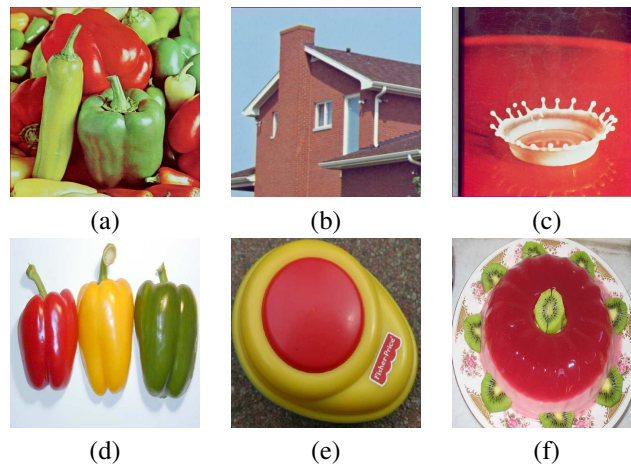


Fig. 5. Some sample images. a) *Peppers*. b) *House*. c) *Splash*. a,b,c) Courtesy of USC-SIPI, Signal and Image Processing Institute at the University of Southern California. d) *Food Peppers*, courtesy of *freemages.co.uk*. e) *Toy*, courtesy of Computational Vision at Caltech, <http://www.vision.caltech.edu>. f) *Dessert*.

result of two innovations in it. First, rather than working on the original realizations, which are noisy and may contain many outliers, the proposed method works on more robust statistical parameters of blocks of the image. Secondly, the proposed distance function models the physical phenomenon more appropriately, compared to the pixel-level operators used before. It is worth to emphasize that the proposed method is more repeatable, compared to the FPCAC [7], which uses a similar distance measure in the pixels level. In the experiments shown in Figure 6, the algorithm has converged in 9, 6, and 5 iterations, elapsing 16s, 14s, and 14s, respectively.

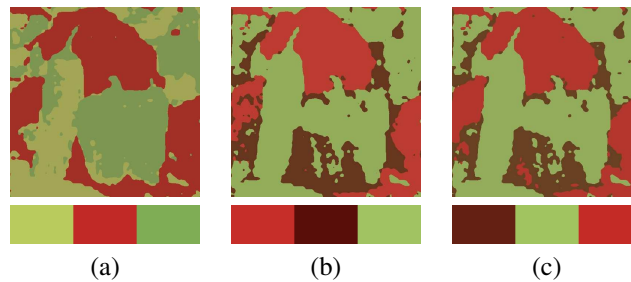


Fig. 6. Results of three independent runs of the proposed method on the image *Peppers*. Top) Results of segmentation. Bottom) Principal colors.

Figure 7 shows the results of the proposed method applied on the image *House*, while different values of  $m$  are used. Here, with  $c = 3$  and  $m \in \{1\frac{1}{3}, 1\frac{2}{3}, 2\}$ , the algorithm has converged in 4, 10, and 9 iterations, elapsing 13s, 16s, and 16s, respectively. More experiments show that the actual value of  $m$  has minimal effect on the results of the proposed method. Hence, the value of  $m = 2$  is used as the default value. Similarly, the values of  $n_1$  and  $n_2$  only affect the spatial precision of the results and the elapsed time. As a compromise, we always select  $n_1 = 4$  and  $n_2 = 3$ .

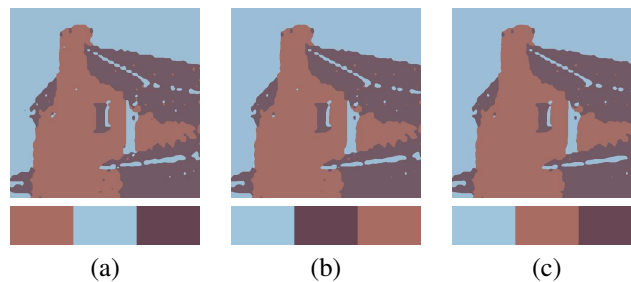


Fig. 7. Effects of  $m$  on the results of the proposed method on the image *House*. Top) Results of segmentation. Bottom) Principal colors.

In a proper segmentation process, it is expected that increasing  $c$  should result in an increasingly better depiction of different regions in the image. Using values of  $c = 2$ ,  $c = 3$ , and  $c = 4$ , this aspect of the proposed method is investigated in Figure 8. As expected, increasing the number of clusters,  $c$  results in more detailed segmentation results. In these experiments, the algorithm

has converged in 9, 5, and 8 iterations, elapsing 12s, 14s, and 18s, respectively. It should be emphasized that the elapsed time of the proposed algorithm depends almost linearly on the number of clusters.

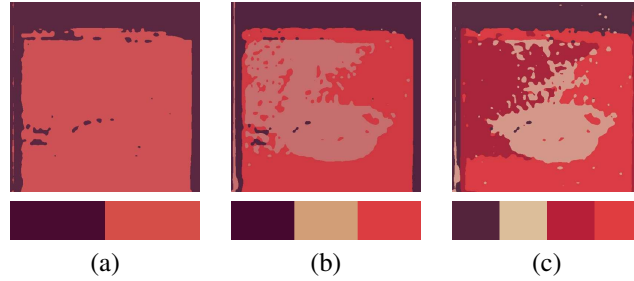


Fig. 8. Effects of  $c$  on the results of the proposed method on the image *Splash*. Top) Results of segmentation. Bottom) Principal colors.

Figure 9 compares the performance of the proposed method with the FPCAC [7]. This comparison is important because both methods search for cylindrical structures in the color space. Note that while the FPCAC works on individual pixels, the proposed method utilizes more robust statistically computed features. In this experiment, the FPCAC is utilized with its own default value of  $m = 1.05$  [7]. Table I lists the elapsed time and the number of iterations of each method when working on each image. Also, the value of  $c = 4$  is used for both methods in all cases.

As mentioned in Section I, there is no standard method for comparing the performance of different segmentation methods. Thus, here, we compare the results of the proposed method and the FPCAC, as shown in Figure 9, based on a heuristic approach. Comparing Figure 9–a with Figure 9–d, a very common problem of many segmentation methods can be seen. This way, in Figure 9–a, FPCAC has mistakenly included parts of the gray background in the yellow pepper. Also, the shadow of the red pepper and other parts of the background are included in the red and green peppers, respectively. However, looking at Figure 9–d, we see a proper segmentation of the image into four distinguished parts of red, yellow, and green peppers, and the background by the proposed method. Figures 9–b and e show an example where the two methods perform similarly, except for the fact that FPCAC has resulted in an over-segmentation. This way, as seen in Figures 9–b, FPCAC has segmented the text on the object into two different classes (because of being pixel-based). The proposed method, on the other hand, has given a smoother segmentation result. This effect is also visible in Figure 9–c, where FPCAC has again resulted in many tiny segments. In contrary, the proposed method has again produced a smooth segmentation result. It is worth to mention that the proposed method is also faster than the FPCAC. The reader is referred to [6] for more examples.

In analyzing the performance of the proposed method, it is worth to mention that as the proposed algorithm is primarily concerned with spectral accuracy, some details in the spatial domain might be ignored. For example, looking at Figure 9–e, we observe that the proposed method has ignored the details in the text on the object. This is a direct result of looking at the image at lower resolutions, as done using  $n_1$  and  $n_2$ . While this is useful in applications which look for the general geometry of the objects, in other applications which demand more precision (including coding) a pixel-based approach would yield more detailed results. Thus, we argue that the proposed method produces desired results when one aims at finding homogeneous regions in an image with more emphasis on color information rather than on spatial details.

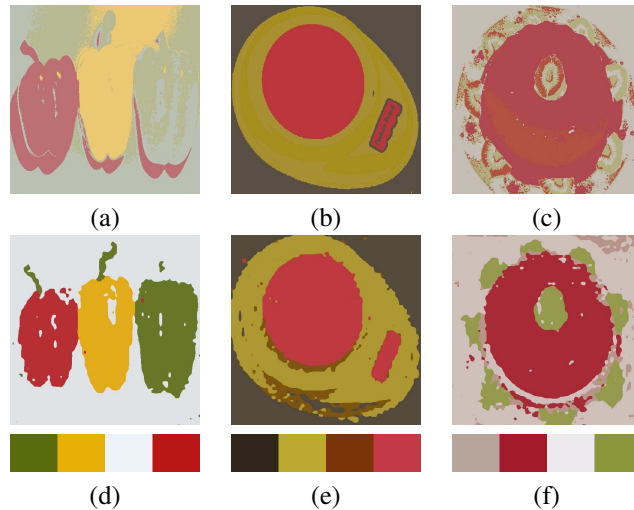


Fig. 9. Comparison of the proposed method with the FPCAC [7]. (a), (b), and (c) Results of the FPCAC. (d), (e), and (f) Results of the proposed method.

TABLE I  
PERFORMANCE COMPARISON OF THE PROPOSED METHOD AND THE FPCAC [7]. [ $t$ : ELAPSED TIME.  $i$ : NUMBER OF ITERATIONS BEFORE CONVERGENCE.]

Image	Proposed Method		FPCAC [7]	
	$t(s)$	$i$	$t(s)$	$i$
Figure 5-d	20	10	26	40
Figure 5-e	18	6	8	19
Figure 5-f	21	11	24	38

#### IV. CONCLUSIONS

In this paper, a new color image segmentation method is proposed which utilizes the general clustering algorithm with an innovative distance function. The mathematics of the proposed method is discussed comprehensively and experimental results are presented. Comparison of the performance of the proposed method with an available clustering method, which searches for similar cylindrical structures in the pixel domain, shows that the proposed method is more stable and faster. We argue that this is mainly because of applying the clustering task on a more robust statistical feature. It is also observed that the proposed method decreases the probability of local minimum entrapment. The repeatability of the proposed segmentation method is also more than the available methods. Furthermore, while the proposed method gives a more perceptually satisfactory segmentation results, it demands less processing resources.

#### ACKNOWLEDGEMENT

The first author wishes to thank Ms. *Azadeh Yadollahi* for her encouragement and invaluable discussions. We also appreciate the respected anonymous referees for their constructive suggestions.

#### REFERENCES

- [1] G. J. Klinker, S. A. Shafer, and T. Kanade, "The measurement of highlights in color images," *International Journal of Computer Vision*, vol. 2, pp. 7–32, 1988.
- [2] —, "A physical approach to color image understanding," *International Journal of Computer Vision*, vol. 4, pp. 7–38, 1990.
- [3] S.-C. Cheng and S.-C. Hsia, "Fast algorithm's for color image processing by principal component analysis," *Journal of Visual Communication and Image Representation*, vol. 14, pp. 184–203, 2003.
- [4] D. O. Nikolaev and P. O. Nikolayev, "Linear color segmentation and its implementation," *Computer Vision and Image Understanding*, vol. 94, pp. 115–139, 2004.
- [5] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3(1), pp. 71–86, 1991.
- [6] A. Abadpour, "Color image processing using principal component analysis," Master's thesis, Sharif University of Technology, Mathematics Science Department, Tehran, Iran, June 2005.
- [7] A. Abadpour and S. Kasaei, "A new FPCA-based fast segmentation method for color images," in *The 4th IEEE International Symposium on Signal Processing and Information Technology (ISSPIT 2004)*, Rome, Italy, 2004.
- [8] —, "Unsupervised, fast and efficient color image copy protection," *IEE Proceedings Communications*, vol. 152 (5), pp. 605–616, 2005.
- [9] —, "An efficient PCA-based color transfer," *Visual Communication & Image Representation*, vol. 18 (1), pp. 15–34, 2007.
- [10] L. Lucchese and S. Mitra, "Color image segmentation: A state-of-the-art survey (invited paper)," *Image Processing, Vision, and Pattern Recognition, Proc. of the Indian National Science Academy (INSA-A)*, vol. 67, A, No. 2, pp. pp. 207–221, 2001.
- [11] W. Skarbek and A. Koschan, "Colour image segmentation—a survey," Institute for Technical Informatics, Technical University of Berlin, Tech. Rep., 1994.
- [12] T. Q. Chen and Y. Lu, "Color image segmentation – an innovative approach," *Pattern Recognition*, vol. 35, pp. 395–405, 2002.
- [13] A. Hyvarinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. John Wiley & Sons, Inc., 2001.
- [14] P. E. Gill, W. Murray, and M. H. Wright, *Numerical Linear Algebra and Optimization*. Redwood City, California: Addison-Wesley Publishing Company, 1991.
- [15] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum Press, 1981.
- [16] K. Honda, N. Sugiura, and H. Ichihashi, "Robust local principal component analyzer with fuzzy clustering," in *IJCNN 2003 Conference Proceedings*, 2003, pp. 732–737.
- [17] I. Gath and A. Geva, "Unsupervised optimal fuzzy clustering," *IEEE Transaction on Pattern Analysis Machine Intelligence*, vol. 11(7), pp. 773–781, 1989.
- [18] J. M. Leski, "Fuzzy c-varieties/elliptotypes clustering in reproducing kernel hilbert space," *Fuzzy Sets and Systems*, vol. 141, pp. 259–280, 2004.
- [19] D. E. Gustafson and 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.
- [20] J. M. Leski, "Generalized weighted conditional fuzzy clustering," *IEEE Transactions on Fuzzy Systems*, vol. 11 (6), pp. 709–715, 2003.
- [21] J. Bezdek, J. Keller, R. Krisnapuram, and N. R. Pal, *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*. Boston: Kluwer Academic Publishers, 1999.