

# **A Multimodal Particle Swarm Optimization-based Approach for Image Segmentation**

,

Computer Engineering Department, Altinbas University, Istanbul, Turkey.

taymaz.farshi@altinbas.edu.tr, +90 534 219 3521

ORCID: 0000-0003-4070-1058

co-author: John H. Drake

School of Informatics, University of Leicester, University Road, Leicester, LE1 7RH, United Kingdom

drjohndrake@gmail.com

co-author: Ender Özcan

School of Computer Science, University of Nottingham, NG8 1BB, Nottingham, UK

ender.ozcan@nottingham.ac.uk

## **ACKNOWLEDGMENTS**

The authors declare that they have no competing interests.

# A Multimodal Particle Swarm Optimization-based Approach for Image Segmentation

Taymaz Rahkar Farshi<sup>1,\*</sup>, John H. Drake<sup>2</sup>, Ender Özcan<sup>3</sup>

<sup>1</sup>Department of Software Engineering, Altınbaş University, Istanbul. 34217, Turkey

<sup>2</sup>School of Informatics, University of Leicester, University Road, Leicester, LE1 7RH, UK

<sup>3</sup>School of Computer Science, University of Nottingham, Nottingham, NG8 1BB, UK

## Abstract

Color image segmentation is a fundamental challenge in the field of image analysis and pattern recognition. In this paper, a novel automated pixel clustering and color image segmentation algorithm is presented. The proposed method operates in three successive stages. In the first stage, a three-dimensional histogram of pixel colors based on the RGB model is smoothened using a Gaussian filter. This process helps to eliminate unreliable and non-dominating peaks that are too close to one another in the histogram. In the next stage, the peaks representing different clusters in the histogram are identified using a multimodal particle swarm optimization algorithm. Finally, pixels are assigned to the most appropriate cluster based on Euclidean distance. Determining the number of clusters to be used is often a manual process left for a user and represents a challenge for various segmentation algorithms. The proposed method is designed to determine an appropriate number of clusters, in addition to the actual peaks, automatically. Experiments confirm that the proposed approach yields desirable results, demonstrating that it can find an appropriate set of clusters for a set of well-known benchmark images.

**Keywords:** *Color image segmentation, Clustering, Particle Swarm Optimisation, Multimodal optimisation*

## 1. Introduction

Image segmentation is the first step in image analysis and refers to the grouping of pixels in an image into several meaningful homogeneous regions (Kurugollu, Sankur, & Harmanci, 2001). There are a wide range of existing methods for image segmentation, which can be categorized into threshold-based, clustering-based, region-based, edge-based, and physics-based segmentation methodologies. Additionally, there are other hybrid image segmentation techniques that use a combination of multiple approaches (Hettiarachchi & Peters, 2017). Approaches to segmentation can be further decomposed into bi-level segmentation methods, which split images into two segments, and multi-level segmentation methods which split images into multiple segments (Pare, Kumar, Bajaj, & Singh, 2016; Sarkar & Das, 2013). Although some segmentation algorithms, such as thresholding methods (e.g., (Otsu, 1979; Kapur, Sahoo, & Wong, 1985)), are developed for bi-level segmentation, they can also be extended to deal with multi-level segmentation (Aziz, Ewees, & Hassanien, 2017; Horng & Liou, 2011; Khairuzzaman & Chaudhury, 2017; Raja, Rajinikanth, & Latha, 2014; V Rajinikanth, Aashiha, & Atchaya, 2014; Sathya &

Kayalvizhi, 2011). Multi-level segmentation is generally a more complex and computationally expensive problem than bi-level segmentation. Upon increasing the desired number of segments, the computational complexity of the problem increases exponentially, making the use of exact methods to exhaustively search all possible solutions impractical. As a result, heuristic algorithms are often preferred, and have proven successful in solving such problems in the literature previously.

The segmentation of color images (RGB) is extremely challenging, due to the variety of possible color intensities and the presence of three color channels, unlike gray images which have only a single color channel (Kumar, Pant, Kumar, & Dutt, 2015). According to Cheng et al. (2001), the segmentation of color images has attracted increasing research attention due to the larger quantity of information contained within color images, and the computational power required to handle the processing of such images is now less expensive than it was previously.

The *k*-means and *c*-means algorithms are two of the most well-known clustering approaches used in color image segmentation, often providing very good results. However, one of the limitations is that the number of clusters is a parameter that must be defined *a priori*, and deciding this value is not trivial. Computational time is also a major concern while solving the problem, as it is dependent on the number of clusters required, as well as the size of the image. Threshold-based methods using histograms are commonly adopted in image segmentation. Unlike region-based methods which require a high volume of computation to calculate spatial pixel similarity, threshold-based approaches use information contained in histograms. Threshold-based techniques are also considered to be relatively quick, since they generally only need to process the pixels in an image once (Shapiro & Stockman 2001), however most are applied to gray-level images using one-dimensional histograms. Historically, few studies applying such methods to color images have appeared in the literature, due to the higher dimensionality involved, and the complexity associated with each color component in each dimension being independent. However, in recent years there has been increased research attention given to color image segmentation based on two- and three-dimensional histograms. The main difficulty faced by existing approaches is determining the number of segments to split an image into, a user-defined parameter (Yang & Huang, 2012).

Due to the nature of the three-dimensional data structures used to represent color images as RGB values, the analysis of color images for global threshold selection to be used in segmentation is a demanding task. There are studies in the literature presenting transformation techniques that map the representation of an image into one or two dimensions, before performing segmentation, i.e., (Tenenbaum, Garvey, Weyl, & Wolf, 1974; Underwood & Aggarwal, 1977). Among others, Sarabi & Aggarwal (1981) and Schacter, Davis, & Rosenfeld (1976) convert the three-dimensional histogram into a binary tree form, where each node is an indicator of a band in the RGB range. As a result, the performance of these algorithms is sensitive to the number of RGB points which quantify the nodal values in the transformed binary tree structure.

Kurugollu et al. (2001) proposed a color image segmentation algorithm that contained two main steps: multi-thresholding and fusion. Firstly, two-dimensional histograms are formed by combining pair-wise color bands

(RG, GB, and BR). The histogram of each band-pair was used to find existing peaks that corresponded to cluster centers. Based on the peaks obtained, the fusion phase aligns the cluster labels in each histogram before applying a spatial-chromatic majority filter to combine the two-dimensional histograms into a final segmentation map. Tan and Isa (2011) introduced a hybrid method based on histogram thresholding and fuzzy *c*-means (FCM). This method used histogram thresholding to attempt to overcome the issue that fuzzy *c*-means is sensitive to the number of clusters and initial assignment of cluster centroids. Their histogram thresholding technique was used to obtain all possible uniform regions of color images, before the FCM algorithm was used to improve the compactness of the regions formed by the clusters.

Panagiotakis et. al. (2011) proposed an image segmentation method using a growing-merging in spatial domain based on tree equipartition and Bayesian flooding processes for feature extraction. Rajinikanth and Couceiro (2015) introduced an approach for color image segmentation based on RGB histograms. The “firefly” optimization algorithm and modified variants were applied to optimize Otsu's between-class variance function for each color component. The RGB histogram of an image was taken into account for bi-level and multi-level segmentation. Lifang and Songwei (2017) introduced a color image segmentation method using a modified firefly algorithm to optimize multi-level Kapur's entropy, minimum cross entropy and between-class variance objective functions. All three functions were applied to all three color components. Syu et. al. (2017) proposed a method which was built on hierarchical image segmentation based on iterative contraction and merging. In their work, finding the optimum number of similar region pairs among neighbouring regions was considered as an optimization problem. Deep Learning was used for semantic image segmentation by Chen et. al. (2018).

As discussed above, the choice of the number of segments to split an image into is critical to the performance of an image segmentation method, and usually requires human expert input. In this paper, we will introduce a novel image segmentation approach that aims to automatically determine both the number of clusters that exist within that image and the pixels that are contained within each cluster. The center of each cluster can be determined by finding the peaks within a three-dimensional histogram of a color image, derived using the RGB values of the pixels in the image and smoothened via the application of a Gaussian filter. Here we use a multimodal variant of particle swarm optimization (PSO) with a local search strategy, to locate all of the global and local peaks within a histogram, and hence determine the centre points for each cluster. The number of peaks discovered by PSO provides the number of clusters contained within the image automatically. Based on the peaks discovered, individual pixels are then assigned to the closest cluster by Euclidean distance, providing the final segmented image.

The paper is structured as follows. Section 2 presents the concepts of multimodal optimization and discovery of peaks in a given RGB histogram. Section 3 provides a description of the proposed method. Section 4 analyzes and compares the results obtained for the proposed approach and *c*-means to a set of well-known benchmark problems. Finally, some concluding remarks are given in Section 5.

## 2. Multimodal optimization and Particle Swarm Optimisation

Unimodal optimization approaches usually search for a single global optimum when solving a given problem. On the other hand, multimodal optimization approaches explore the search space with the goal of detecting global and local optima simultaneously. Multimodal optimization algorithms are attractive in many real-world problems, particularly where multiple solutions of differing quality are required by the end users. Particle Swarm Optimization (PSO) is a well-known optimization algorithm introduced by Eberhart and Kennedy (1995). Although this algorithm was initially proposed as a unimodal approach, it has been extended to multimodal form a number of times in the literature, exploiting the mechanisms for particles' motion to detect both global and local optima (Parsopoulos and Vrahatis, 2001; Brits et al., 2007).

In traditional PSO, each particle uses two vectors: *position* ( $\mathbf{x}$ ) and *velocity* ( $\mathbf{v}$ ). The position vector encodes the location of a particle and the velocity vector shows the amount of change in position and direction of a particle. PSO is an iterative algorithm. The search process starts by assigning random values (locations) to each particle in the solution space. The position components are then updated based on the particles' velocity components at each iteration  $i$ . From each individual particle's experience previously gained during the search process, the swarm's overall experience and an element of stochasticity, the new velocity vector of a particle can be calculated by Equation (1).

$$v_i(t+1) = w \times v_i(t) + R_1 \times C_1 (p_i^{best} - x_i) + R_2 \times C_2 (g^{best} - x_i) \quad (1)$$

$$x_i(t+1) = x_i(t) + v_i(t+1)$$

where  $v_i(t)$  and  $x_i(t)$  represent the velocity and position of the  $i^{th}$  particle at iteration  $t$ ,  $w$  is the inertia weight,

$p_i^{best}$  and  $g^{best}$  represent the position of the best solution found so far by the  $i^{th}$  particle and its neighbors, respectively.  $R_1$  and  $R_2$  are two randomly generated numbers uniformly distributed in the range  $[0,1]$ .  $C_1$  and  $C_2$  are the confidence of a given particle in itself and its neighbors respectively. The mechanism for particle motion in traditional PSO can easily be extended to deal with multimodal problems. In the unimodal form of PSO, all particles in the population converge towards the same point ( $g^{best}$ ) in the search space. However, unlike the unimodal form, multimodal PSO seeks multiple  $g^{best}$ s across the search space (Wang, Moon, Yang, & Wang, 2012).

Inspired by electrostatic interactions between particles, Barrera and Coello Coello (2009) presented a modified PSO variant to tackle multimodal problems. To reach multiple optima, individual particles move from their current position towards the particle with greatest electrostatic conduction calculated based on current fitness value. These

interactions are mathematically calculated per  $F_{i,j} = Q_i Q_j / (4\pi r^2 \epsilon_0)$ , where  $Q_{i,j}$ ,  $r \neq 0$ , and  $\epsilon_0$  are the electrical charges of the interacting particles, the distance between them, and the vacuum permittivity respectively. To put these concepts in the context of an optimization framework, the electric charge of the particles represents the value of the fitness function, which is weighted by the Euclidean distance, i.e.,  $F_{i,j} = \alpha f(p_i) f(p_j) / \|p_i - p_j\|^2$ . Here  $4\pi\epsilon_0$  as constant scalar is replaced by  $\alpha$  which is calculated

following Li (2007). For a constant index  $j$ ,  $index_i = \arg \max_{j=1:M} F_{i,j}$  is used to replace the value of  $g_{best}$  in Eq. (1)

$$v_t = w.v_{t-1} + R_1 \times C_1(p_i^{best} - x_i) + R_2 \times C_2(p_{index_i} - x_i) \quad (2)$$

$$x_i = x_i + v_i$$

This modified variant of PSO for multimodal problems is used in the experimentation performed within this paper.

### 3. Proposed Segmentation Method (3DHP)

In this section we will describe our proposed approach, referred to as 3DHP herein. As discussed in the introduction, due to the difficulty in processing three-dimensional histograms, many segmentation methods based on histograms only deal with one-dimensional gray images. For color images using the RGB model, the color of a pixel is a combination of the three independent color channels red, green and blue. Each pixel can be represented by a three-dimensional feature vector that contains three colors of an image pixel. Accordingly, a histogram based on these three color components can be formed (Navon, Miller, & Averbuch, 2005).

The existence of peaks in a histogram indicates that there are different segments in the image, with each peak representing a different segment. Because of the nature of the data, the histograms obtained are usually very noisy (Kurugollu, et al., 2001). Consequently, three-dimensional histograms are often smoothed by a three-dimensional Gaussian filter to reduce the effect of this noise. This procedure also removes small non-significant local peaks from the histogram. The three-dimensional histogram, original color distribution and color distribution after the smoothening process for the Lenna image are illustrated in Figure 1.

Next, we use the multimodal variant of PSO introduced by Barrera and Coello Coello (2009) and discussed in Section 2 above to locate all of the peaks within the image, using the smoothed histogram. It is well-known that the fine search aspect of multimodal algorithms is a challenging task, as the algorithm may converge close to the global/local optima without reaching the desired goal. Qu et al. (2012) proposed an additional step to several

existing multimodal PSO algorithms, aimed at enhancing the effectiveness of local search, which increases the likelihood of finding optima as well as reducing the number of function evaluations required for convergence.

$$\begin{cases} f(bestNearest_i) \geq f(pbest_i) \rightarrow temp = \sum_{d=1}^D p_{d,i}^{best} + C_1 \cdot rand. (p_{d,i}^{best\_nearest} - p_{d,i}^{best}) \\ f(bestNearest_i) < f(pbest_i) \rightarrow temp = \sum_{d=1}^D p_{d,i}^{best} + C_1 \cdot rand. (p_{d,i}^{best} - p_{d,i}^{best\_nearest}) \end{cases} \quad (3)$$

$$f(temp) > f(pbest_i) \rightarrow pbest_i = temp \quad (4)$$

In the proposed method, we employ this additional local search step, in order to increase the performance level of our approach. After locating the best  $K$  dominant peaks,  $K$  sets of peak intensity level in each RGB component are automatically obtained.

Then  $P_1^{rgb} = (r_1, g_1, b_1)$ ,  $P_2^{rgb} = (r_2, g_2, b_2)$ ,  $P_3^{rgb} = (r_3, g_3, b_3)$

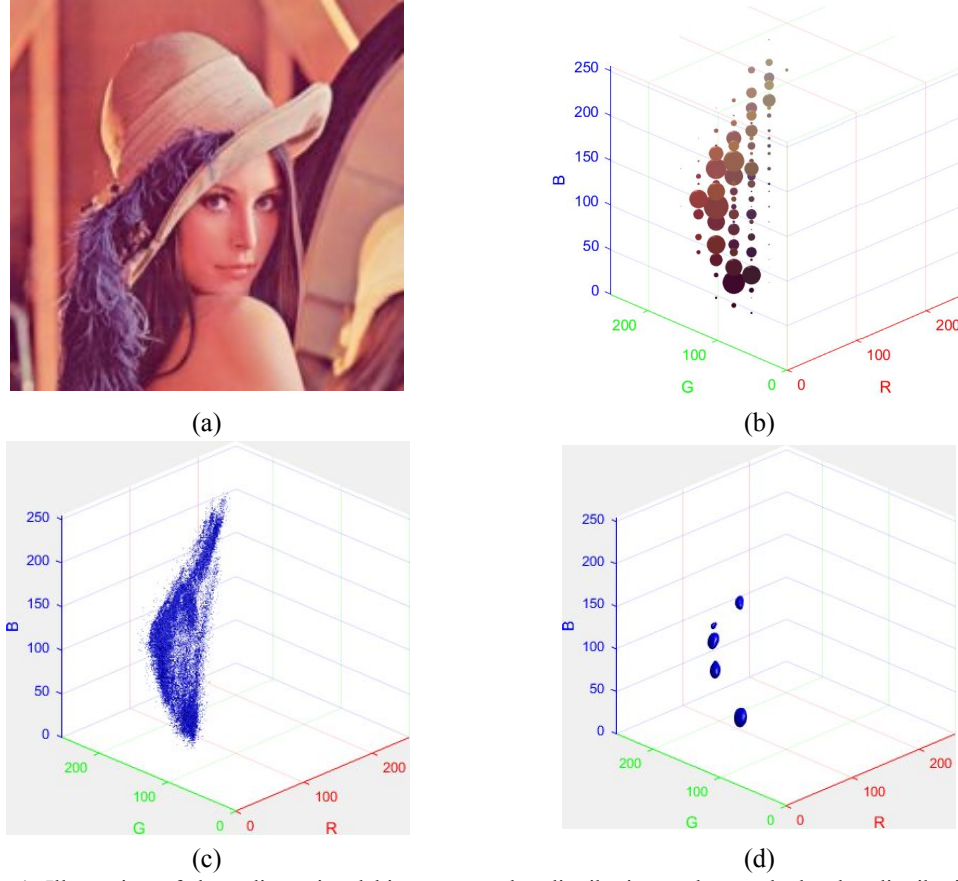
$\dots P_K^{rgb} = (r_K, g_K, b_K)$  are the sets of peaks that are considered as cluster centers. In addition, in order to eliminate non-dominant clusters, it is advantageous to limit the distance between two peaks. Based on a given distance limit parameter, dominating peaks eliminate non-dominating peaks within that radius. It is important to note that this procedure is optional and could be omitted. In our experiments, this parameter is set to 80 pixels. The number of peaks discovered represents the number of clusters and each peak is considered as the cluster head.

Eventually, each pixel is assigned to the closest peak in terms of Euclidean distance. The Euclidean distance between  $k_{th}$  peak and  $(i,j)_{th}$  pixel is calculated as follows:

$$\|P_k^{rgb} - I_{i,j}^{rgb}\| = \sqrt{(P_k^r - I_{i,j}^r)^2 + (P_k^g - I_{i,j}^g)^2 + (P_k^b - I_{i,j}^b)^2} \quad (5)$$

The proposed algorithm is summarized by the following three steps:

- Compute (Figure 1(c)) and smoothen (Figure 1(d)) the three-dimensional histogram
- Apply multimodal PSO to find the dominant peaks within the histogram, representing the clusters within the image
- Assign each pixel to the closest peak (cluster) in order to segment the image



**Figure 1.** Illustration of three-dimensional histogram, color distribution and smoothed color distribution of Lenna. (a) original Lenna image, (b) three-dimensional histogram of Lenna, (c) and (d) show the normal and smoothed RGB representation of Lenna.

#### 4. Experimental results and performance evaluation

Our experiments were implemented using Matlab R2014 on a Core i7-3632qm 2.20GHz CPU, 8 GB RAM running Windows 10. The proposed approach has been tested over the well-known Lenna image and the standard publicly accessible Berkeley segmentation dataset (Martin, Fowlkes, Tal, & Malik, 2001). In this paper, 20 images from this dataset have been selected to demonstrate the capability of the proposed method. The size and variance of the Gaussian filter used to smoothen the are empirically set to 11 and 7. The segmentation results of the proposed scheme depend on the quality of the clusters. In order to evaluate the quality of the proposed method, we compare to the fuzzy *c*-means (FCM) (Sutton, Bezdek, & Cahoon, 2000) and recently proposed SFFCM (Lei, et al., 2018) methods from the literature, using six quantitative performance assessment metrics and computation time (T).

As the test images are somewhat heterogeneous, visual judgment is difficult and may not be sufficient for analysis purposes. Therefore quantitative evaluation criteria is required to measure the performance of segmentation (Chang, Zhao, Liu, & Zheng, 2016). Dividing one region of the reference image into two or more regions



(over-segmentation), and conversely, representing two or more regions of the reference image by a single region (under-segmentation) are both undesirable. It is obvious that by increasing the number of segments, the homogeneity of pixels in each segment will also increase. On the other hand, a segmented image formed by a large number of small segments may not be satisfactory. Hence the number of segments and their homogeneity plays an important role in a successful segmentation (Hettiarachchi & Peters, 2017).

There are multiple quantitative assessment functions that can be used to evaluate the image segmentation results. Three of the most fundamental functions used for numerical evaluation of image segmentation results are as follows:

$F(I)$  proposed by Liu and Yang (1994) which penalizes over-segmentation:

$$F = \frac{1}{1000(M \times N)} \sqrt{R} \sum_{i=1}^R \frac{e_i^2}{\sqrt{A_i}} \quad (6)$$

$F'(I)$  proposed by Borsotti et al. (1998) which is robust for noisy images:

$$F'(I) = \frac{1}{10000(N \times M)} \sqrt{\sum_{A=1}^{Max} [R(A)]^{1+\frac{1}{A}}} \times \sum_{i=1}^R \frac{e_i^2}{\sqrt{A_i}} \quad (7)$$

and  $Q(I)$  further refined from  $F(I)$  by Borsotti et al. (1998), which penalizes non-homogeneous regions:

$$Q(I) = \frac{1}{10000(M \times N)} \sqrt{R} \sum_{i=1}^R \left[ \frac{e_i^2}{1 + \log A_i} + \left( \frac{R(A_i)}{A_i} \right)^2 \right] \quad (8)$$

For the three formulae above,  $I$  is image,  $M \times N$  is the image size (number of pixels),  $R$  is the number of regions identified,  $A_i$  is the number of pixels present in the  $i^{th}$  region.  $e_i$  represents the color error in region  $i$ , which is defined as the sum of the Euclidean distances between (RGB) pixels of region  $i$  in the original color image and the attributed (RGB) pixel values in region  $i$  in the archived segmented image.  $\sqrt{R}$  is a penalizing term that discourages over-segmentation (non-homogeneous regions). A small value of  $F$  and  $F(I)$  is desirable.  $R(A)$  represents the number of regions that have an area of exactly  $A$ , and  $Max$  represents the largest region in the segmented image.

Moreover, three other common evaluation criteria are used for quantitative comparison. The Probabilistic Rand Index (PRI) (Martin, et al., 2001) counts the pairs of pixels that not only have consistent labels in the segmented image, but also have consistent labels in the ground truth image. Variation of Information (VoI) or shared information distance (Meila, 2002) measure the correctness of segmentation by calculating the distance between two segmentations. The Global Consistency Error (GCE) (Martin, et al., 2001) evaluates the extent to which one segmentation can be a refinement of another. In this way, the associated segmentations are consistent because they represent the same image segmented at different scales.

The visual qualitative analysis of all images is shown in Figure 2 and Figure 3. In Figure 3, the segmentation results for each method are shown using the mean average color value for all pixels in that cluster, and also using a distinct color set to the original image to clearly show the clusters found. The three-dimensional histogram peak locations and cluster centroids for each cluster identified by 3DHP, FCM and SFFCM are provided in Table 1. Likewise, Table 2 and Table 3 indicate the numerical qualitative analysis of the results obtained using each of the three methods tested. If the ideal number of clusters was known in advance, FCM could yield robust segmentation results. In our experiments, the number of clusters for FCM is determined based on the number of peaks identified by 3DHP.


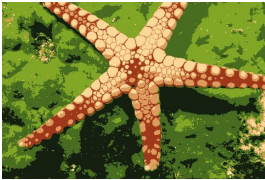
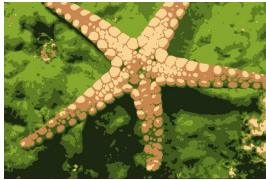

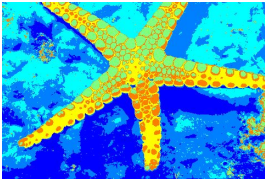
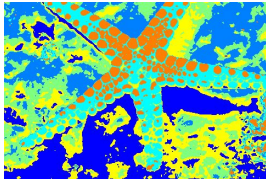
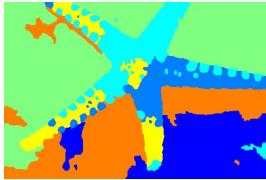

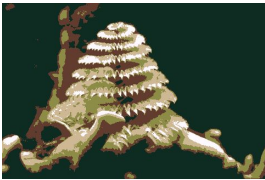
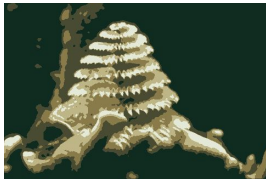

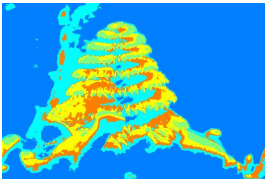
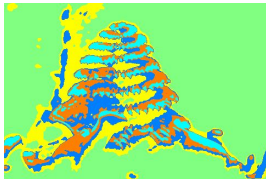





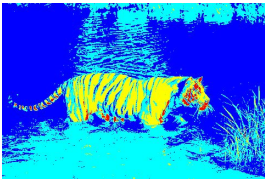

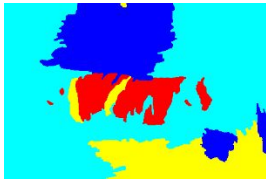




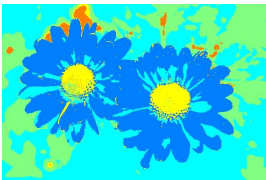

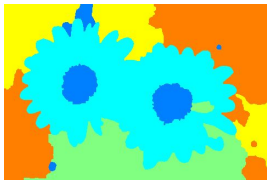
It is clear from Figure 2 and Figure 3 that the proposed scheme is capable to achieve viable segmentation with well-preserved edges. Table 2 shows that for all of the test images, 3DHP, FCM and SFFCM all produce favorable and reliable results. The main difference is that 3DHP does not require the number of segments to be determined in advance. Table 2 demonstrates that the actual computation time of the proposed technique is significantly lower than FCM. The computational complexity of FCM and SFFCM increases exponentially as the image size and number of clusters increases, whereas the computational effort required to execute 3DHP is independent of the size of the image.



**Figure 2.** (a) RGB distribution and peak locations (b) segmented image by 3DHP,  $m=4$ , (c) segmented image by FCM,  $m=4$ , (d) segmented image by SFFCM,  $m=4$

Figure 2(a) shows the cluster centroids located in the Lenna image by 3DHP while Figure 2(b), Figure 2(c) and Figure 2(d) show the segmented image obtained by 3DHP, FCM and SFFCM, respectively. By observing the results

shown in Figure 3 for '135069' and '238011', it seems that 3DHP is more effective at segmenting large homogenous regions, such as the background region in these two images. For '135069', the sky is divided into multiple segments using FCM and SFFCM, whereas with 3DHP, except for the top-left corner, the sky is well distinguished. For the '238011' image, the moon in the sky disappears entirely when using FCM and SFFCM. For the '232038' image, 3DHP and FCM show better segmentation results than SFFCM as in the case of SFFCM, pixels representing the subject's eyes are mistakenly assigned to the face. For the '124084' image, with SFFCM all pieces of the flower and background are clearly distinguished, however this is not the case with two other algorithms. Additionally, for image '71046', using 3DHP the sky is segmented correctly, whereas FCM over-segments the sky, dividing it into two separate regions.

Name	Original Image	m	Segmented by 3DHP	Segmented by FCM	Segmented by SFFCM
1200 3		m =			
		6			
1207 4		m =			
		5			
1080 73		m =			
		4			
1240 84		m =			
		5			

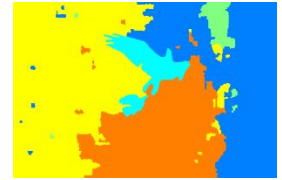
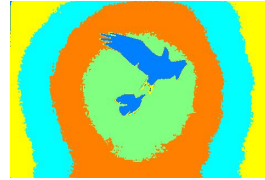
**Figure 3.** Original benchmark image and segmented results



1350  
69



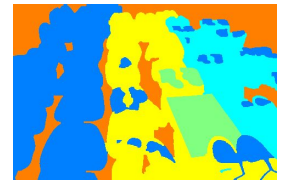
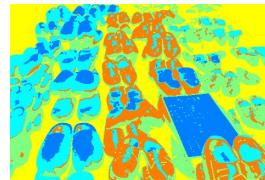
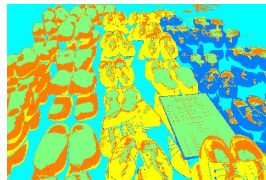
m  
=  
5



1400  
75



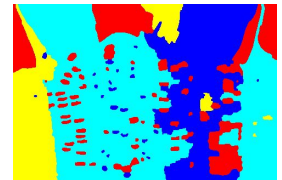
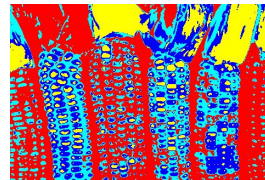
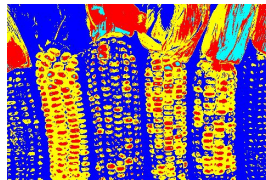
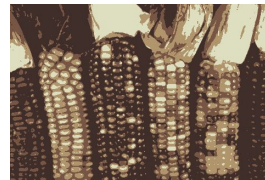
m  
=  
5



1690  
12



m  
=  
4



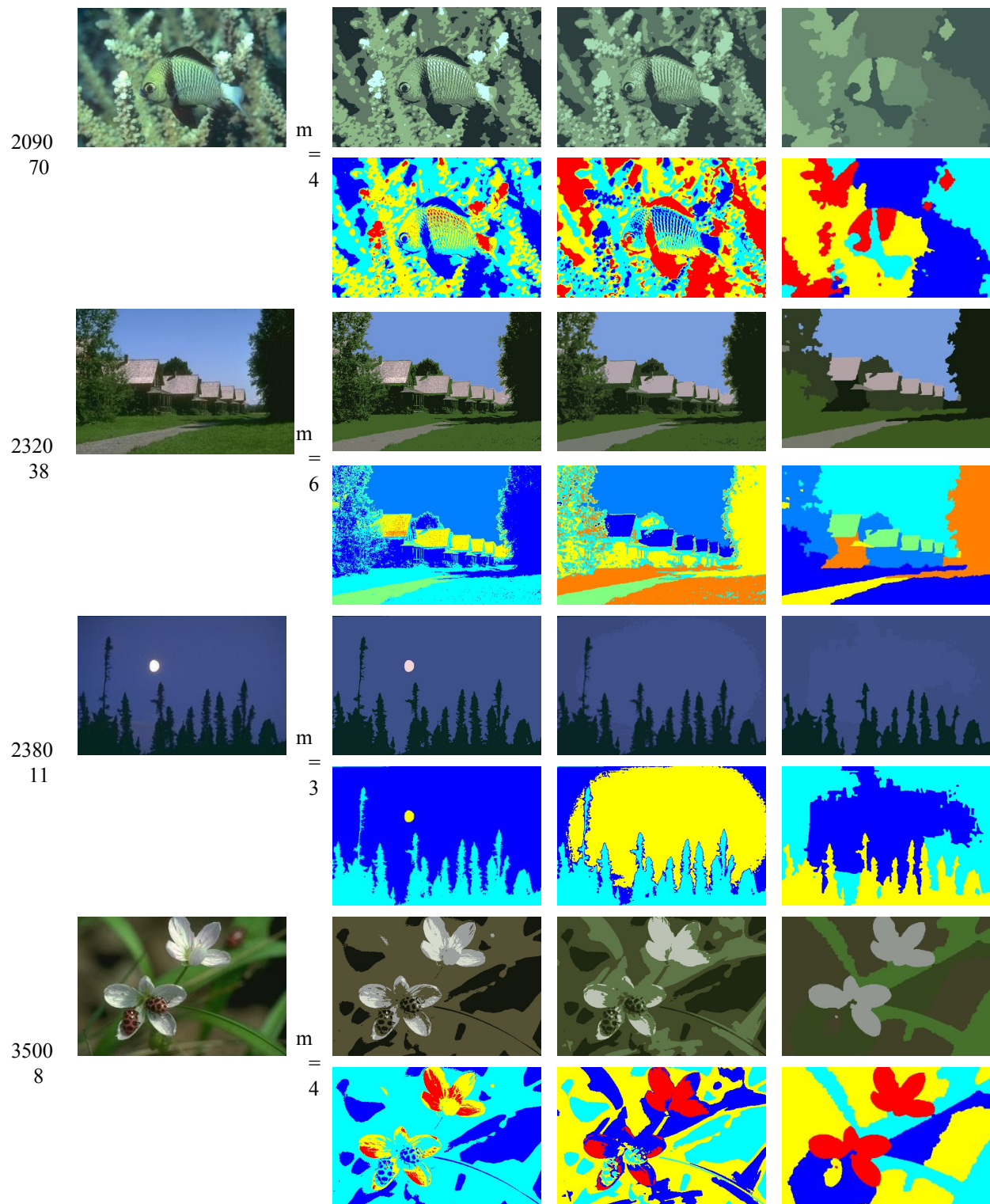
1890  
03



m  
=  
8



Figure 3. Continued



**Figure 3.** Continued



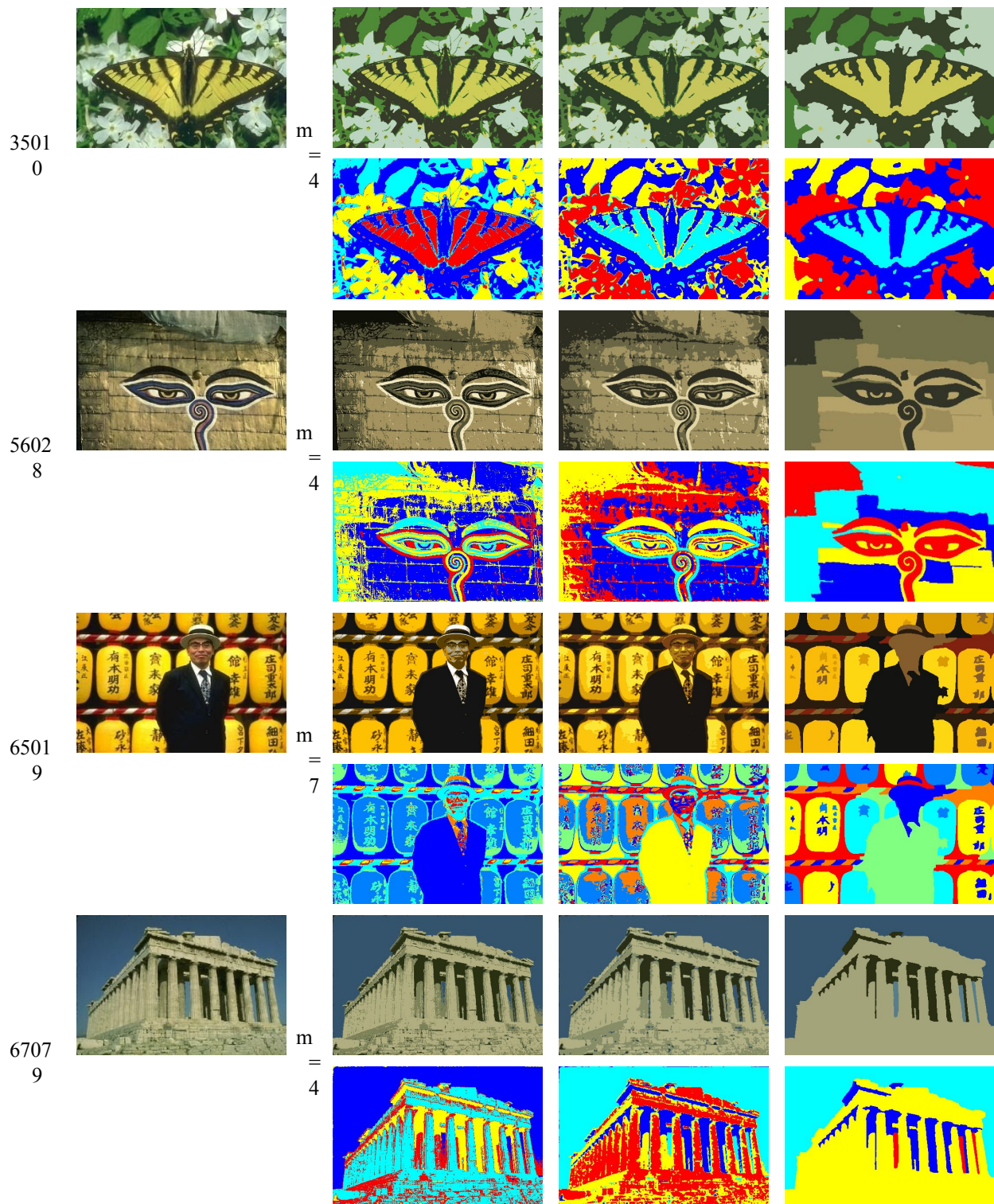
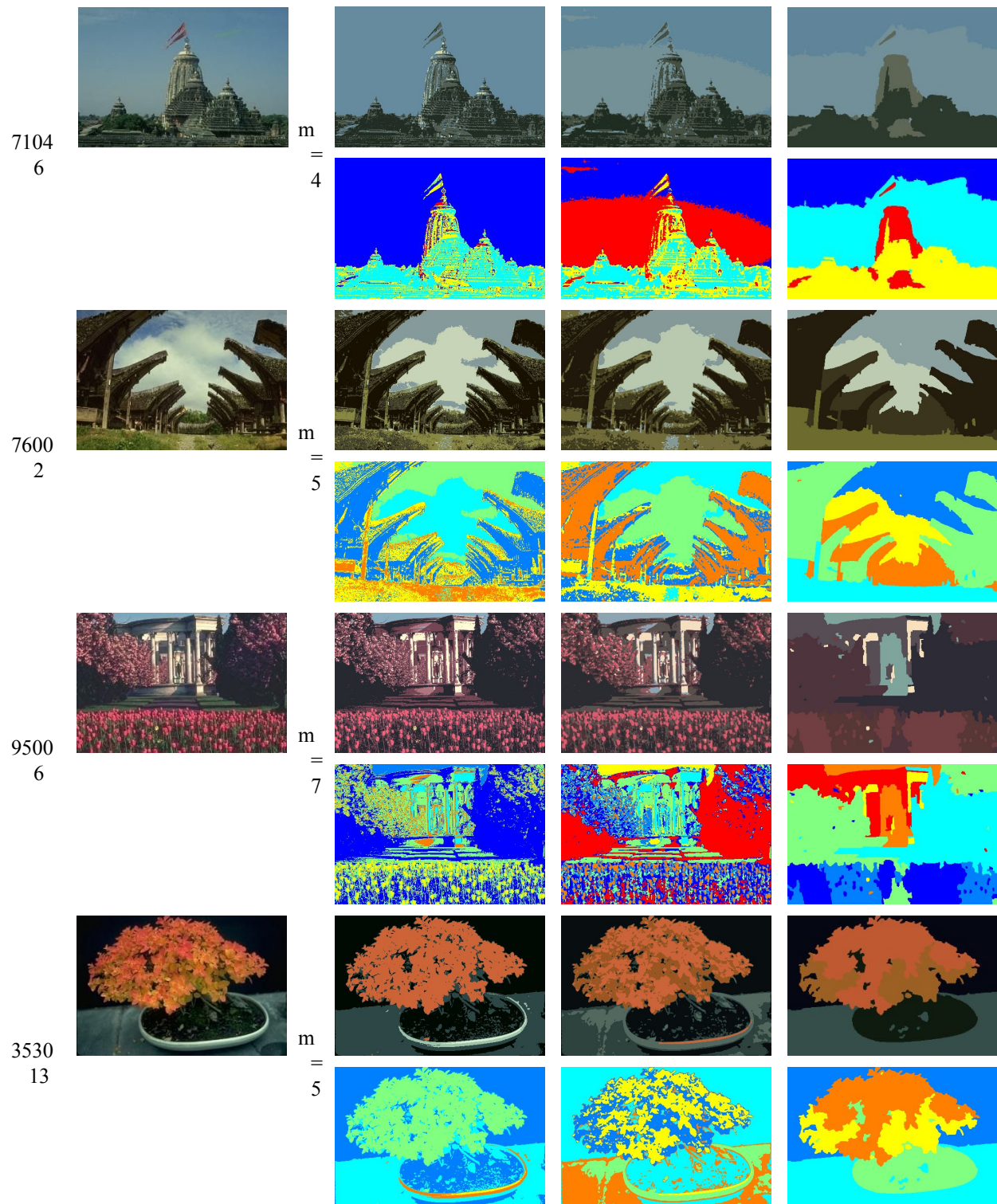


Figure 3. Continued



**Figure 3.** Continued



Table 2 shows that 3DHP required almost the same execution time for all images, while FCM took much longer to process large images such as ‘12003’, ‘140075’ and ‘189003’. With the exception of the images with four or less clusters, among all test images, the computational time of 3DHP is lower than FCM. However, the computational time of SFFCM for all images is lower than both 3DHP and FCM. The values achieved for the three evaluation functions  $F(I)$ ,  $F'(I)$ , and  $Q(I)$  suggest that all three methods yield consistent quantitative performance on the same image. However, the difference in these values is not substantial and in all cases they approach zero. The segmentation regions produced by the 3DHP method are more homogenous when inspected visually. The FCM method shows effective performance by producing good values for the three statistical measures  $F(I)$ ,  $F'(I)$ , and  $Q(I)$ . In most cases 3DHP provides better performance than SFFCM respect to  $F(I)$ ,  $F'(I)$ , and  $Q(I)$ . The success of FCM and SFFCM on certain images is a result of an appropriate number of clusters being chosen by the 3DHP method.

Table 3 shows that the results obtained by all methods are competitive for at least some images, as they outperformed each other in many cases. Due to a large number of test images in the Berkeley dataset, providing tables for all PRI, VoI and GCE values is impractical. Hence the average of whole dataset results has been presented in Table 4.

**Table 1.** Cluster centroids and peaks

Name	Num of clusters		Peak locations: 3DHP	Cluster centroid: FCM	Cluster centroid: SFFCM
Lenna	m=4	r	94 176 208 228	102 176 211 228	59 32 125 242
		g	24 69 137 195	32 77 124 158	99 34 140 200
		b	64 78 125 179	70 86 113 162	95 32 126 170
12003	m=6	r	20 74 132 256 171 214	29 134 202 98 68 249	67 201 252 110 176 34
		g	31 109 168 222 75 150	40 168 136 129 91 214	98 134 201 145 83 46
		b	19 31 41 149 31 94	19 44 78 37 30 147	31 77 135 37 40 21
12074	m=5	r	16 81 253 201 166	138 229 16 74 191	171 178 40 14 209
		g	45 94 256 183 128	123 234 44 75 182	149 181 53 45 205
		b	33 58 250 144 75	75 216 32 51 135	99 99 36 33 182
108073	m=4	r	27 93 249 253	111 237 30 62	49 41 99 237
		g	45 90 167 251	100 172 46 75	75 55 91 158
		b	36 54 102 256	63 113 35 49	48 40 53 99
124084	m=5	r	166 15 85 243 256	62 103 23 154 227	223 149 18 39 72
		g	1 17 94 191 254	56 107 21 11 173	168 10 15 43 80
		b	10 10 55 1 158	19 68 8 9 24	7 8 8 10 41

135069	m=5	<div><div>r</div><div>g</div><div>b</div></div> <table><tr><td>72</td><td>29</td><td>93</td><td>192</td><td>238</td></tr><tr><td>135</td><td>52</td><td>99</td><td>183</td><td>256</td></tr><tr><td>165</td><td>50</td><td>84</td><td>172</td><td>256</td></tr></table> <table><tr><td>29</td><td>62</td><td>79</td><td>55</td><td>70</td></tr><tr><td>43</td><td>122</td><td>144</td><td>110</td><td>134</td></tr><tr><td>47</td><td>151</td><td>177</td><td>136</td><td>164</td></tr></table> <table><tr><td>61</td><td>29</td><td>63</td><td>65</td><td>75</td></tr><tr><td>120</td><td>43</td><td>123</td><td>127</td><td>139</td></tr><tr><td>148</td><td>46</td><td>153</td><td>157</td><td>170</td></tr></table>	72	29	93	192	238	135	52	99	183	256	165	50	84	172	256	29	62	79	55	70	43	122	144	110	134	47	151	177	136	164	61	29	63	65	75	120	43	123	127	139	148	46	153	157	170																											
72	29	93	192	238																																																																						
135	52	99	183	256																																																																						
165	50	84	172	256																																																																						
29	62	79	55	70																																																																						
43	122	144	110	134																																																																						
47	151	177	136	164																																																																						
61	29	63	65	75																																																																						
120	43	123	127	139																																																																						
148	46	153	157	170																																																																						
140075	m=5	<div><div>r</div><div>g</div><div>b</div></div> <table><tr><td>102</td><td>11</td><td>169</td><td>115</td><td>115</td></tr><tr><td>1</td><td>12</td><td>161</td><td>84</td><td>107</td></tr><tr><td>3</td><td>14</td><td>132</td><td>1</td><td>86</td></tr></table> <table><tr><td>187</td><td>146</td><td>89</td><td>32</td><td>138</td></tr><tr><td>182</td><td>131</td><td>51</td><td>20</td><td>104</td></tr><tr><td>156</td><td>101</td><td>23</td><td>13</td><td>27</td></tr></table> <table><tr><td>136</td><td>92</td><td>194</td><td>133</td><td>26</td></tr><tr><td>123</td><td>6</td><td>195</td><td>99</td><td>22</td></tr><tr><td>95</td><td>9</td><td>179</td><td>14</td><td>13</td></tr></table>	102	11	169	115	115	1	12	161	84	107	3	14	132	1	86	187	146	89	32	138	182	131	51	20	104	156	101	23	13	27	136	92	194	133	26	123	6	195	99	22	95	9	179	14	13																											
102	11	169	115	115																																																																						
1	12	161	84	107																																																																						
3	14	132	1	86																																																																						
187	146	89	32	138																																																																						
182	131	51	20	104																																																																						
156	101	23	13	27																																																																						
136	92	194	133	26																																																																						
123	6	195	99	22																																																																						
95	9	179	14	13																																																																						
169012	m=4	<div><div>r</div><div>g</div><div>b</div></div> <table><tr><td>50</td><td>244</td><td>132</td><td>216</td></tr><tr><td>42</td><td>254</td><td>124</td><td>218</td></tr><tr><td>34</td><td>256</td><td>88</td><td>183</td></tr></table> <table><tr><td>177</td><td>119</td><td>222</td><td>63</td></tr><tr><td>149</td><td>86</td><td>223</td><td>46</td></tr><tr><td>109</td><td>69</td><td>192</td><td>42</td></tr></table> <table><tr><td>142</td><td>77</td><td>90</td><td>210</td></tr><tr><td>115</td><td>58</td><td>44</td><td>202</td></tr><tr><td>90</td><td>52</td><td>41</td><td>154</td></tr></table>	50	244	132	216	42	254	124	218	34	256	88	183	177	119	222	63	149	86	223	46	109	69	192	42	142	77	90	210	115	58	44	202	90	52	41	154																																				
50	244	132	216																																																																							
42	254	124	218																																																																							
34	256	88	183																																																																							
177	119	222	63																																																																							
149	86	223	46																																																																							
109	69	192	42																																																																							
142	77	90	210																																																																							
115	58	44	202																																																																							
90	52	41	154																																																																							
189003	m=8	<div><div>r</div><div>g</div><div>b</div></div> <table><tr><td>15</td><td>256</td><td>209</td><td>145</td><td>68</td><td>196</td><td>142</td><td>60</td></tr><tr><td>14</td><td>253</td><td>159</td><td>1</td><td>67</td><td>38</td><td>112</td><td>88</td></tr><tr><td>12</td><td>232</td><td>141</td><td>34</td><td>65</td><td>88</td><td>96</td><td>154</td></tr></table> <table><tr><td>170</td><td>251</td><td>153</td><td>191</td><td>95</td><td>19</td><td>52</td><td>227</td></tr><tr><td>32</td><td>243</td><td>120</td><td>155</td><td>78</td><td>16</td><td>46</td><td>192</td></tr><tr><td>68</td><td>222</td><td>108</td><td>141</td><td>78</td><td>14</td><td>47</td><td>177</td></tr></table> <table><tr><td>177</td><td>43</td><td>183</td><td>234</td><td>26</td><td>63</td><td>144</td><td>201</td></tr><tr><td>158</td><td>39</td><td>36</td><td>222</td><td>18</td><td>73</td><td>32</td><td>156</td></tr><tr><td>151</td><td>38</td><td>78</td><td>209</td><td>16</td><td>125</td><td>36</td><td>137</td></tr></table>	15	256	209	145	68	196	142	60	14	253	159	1	67	38	112	88	12	232	141	34	65	88	96	154	170	251	153	191	95	19	52	227	32	243	120	155	78	16	46	192	68	222	108	141	78	14	47	177	177	43	183	234	26	63	144	201	158	39	36	222	18	73	32	156	151	38	78	209	16	125	36	137
15	256	209	145	68	196	142	60																																																																			
14	253	159	1	67	38	112	88																																																																			
12	232	141	34	65	88	96	154																																																																			
170	251	153	191	95	19	52	227																																																																			
32	243	120	155	78	16	46	192																																																																			
68	222	108	141	78	14	47	177																																																																			
177	43	183	234	26	63	144	201																																																																			
158	39	36	222	18	73	32	156																																																																			
151	38	78	209	16	125	36	137																																																																			
209070	m=4	<div><div>r</div><div>g</div><div>b</div></div> <table><tr><td>31</td><td>96</td><td>141</td><td>200</td></tr><tr><td>47</td><td>122</td><td>185</td><td>256</td></tr><tr><td>49</td><td>103</td><td>137</td><td>256</td></tr></table> <table><tr><td>165</td><td>128</td><td>91</td><td>45</td></tr><tr><td>220</td><td>168</td><td>119</td><td>63</td></tr><tr><td>181</td><td>131</td><td>101</td><td>63</td></tr></table> <table><tr><td>92</td><td>65</td><td>105</td><td>137</td></tr><tr><td>118</td><td>90</td><td>139</td><td>180</td></tr><tr><td>99</td><td>83</td><td>111</td><td>134</td></tr></table>	31	96	141	200	47	122	185	256	49	103	137	256	165	128	91	45	220	168	119	63	181	131	101	63	92	65	105	137	118	90	139	180	99	83	111	134																																				
31	96	141	200																																																																							
47	122	185	256																																																																							
49	103	137	256																																																																							
165	128	91	45																																																																							
220	168	119	63																																																																							
181	131	101	63																																																																							
92	65	105	137																																																																							
118	90	139	180																																																																							
99	83	111	134																																																																							
232038	m=6	<div><div>r</div><div>g</div><div>b</div></div> <table><tr><td>19</td><td>110</td><td>65</td><td>122</td><td>179</td><td>243</td></tr><tr><td>29</td><td>147</td><td>98</td><td>121</td><td>160</td><td>221</td></tr><tr><td>15</td><td>215</td><td>38</td><td>111</td><td>162</td><td>214</td></tr></table> <table><tr><td>177</td><td>120</td><td>46</td><td>114</td><td>20</td><td>63</td></tr><tr><td>165</td><td>156</td><td>62</td><td>119</td><td>32</td><td>92</td></tr><tr><td>171</td><td>220</td><td>34</td><td>110</td><td>15</td><td>36</td></tr></table> <table><tr><td>60</td><td>48</td><td>119</td><td>178</td><td>118</td><td>18</td></tr><tr><td>90</td><td>61</td><td>156</td><td>162</td><td>119</td><td>29</td></tr><tr><td>32</td><td>36</td><td>221</td><td>165</td><td>107</td><td>14</td></tr></table>	19	110	65	122	179	243	29	147	98	121	160	221	15	215	38	111	162	214	177	120	46	114	20	63	165	156	62	119	32	92	171	220	34	110	15	36	60	48	119	178	118	18	90	61	156	162	119	29	32	36	221	165	107	14																		
19	110	65	122	179	243																																																																					
29	147	98	121	160	221																																																																					
15	215	38	111	162	214																																																																					
177	120	46	114	20	63																																																																					
165	156	62	119	32	92																																																																					
171	220	34	110	15	36																																																																					
60	48	119	178	118	18																																																																					
90	61	156	162	119	29																																																																					
32	36	221	165	107	14																																																																					
238011	m=3	<div><div>r</div><div>g</div><div>b</div></div> <table><tr><td>62</td><td>8</td><td>244</td></tr><tr><td>80</td><td>38</td><td>213</td></tr><tr><td>137</td><td>36</td><td>218</td></tr></table> <table><tr><td>54</td><td>9</td><td>62</td></tr><tr><td>70</td><td>37</td><td>79</td></tr><tr><td>121</td><td>39</td><td>136</td></tr></table> <table><tr><td>63</td><td>59</td><td>7</td></tr><tr><td>80</td><td>76</td><td>37</td></tr><tr><td>138</td><td>131</td><td>34</td></tr></table>	62	8	244	80	38	213	137	36	218	54	9	62	70	37	79	121	39	136	63	59	7	80	76	37	138	131	34																																													
62	8	244																																																																								
80	38	213																																																																								
137	36	218																																																																								
54	9	62																																																																								
70	37	79																																																																								
121	39	136																																																																								
63	59	7																																																																								
80	76	37																																																																								
138	131	34																																																																								
35008	m=4	<div><div>r</div><div>g</div><div>b</div></div> <table><tr><td>20</td><td>84</td><td>156</td><td>216</td></tr><tr><td>23</td><td>81</td><td>160</td><td>220</td></tr><tr><td>14</td><td>53</td><td>158</td><td>205</td></tr></table> <table><tr><td>94</td><td>30</td><td>67</td><td>188</td></tr><tr><td>117</td><td>40</td><td>75</td><td>195</td></tr><tr><td>72</td><td>17</td><td>40</td><td>181</td></tr></table> <table><tr><td>54</td><td>70</td><td>64</td><td>145</td></tr><tr><td>70</td><td>113</td><td>65</td><td>150</td></tr><tr><td>30</td><td>45</td><td>38</td><td>144</td></tr></table>	20	84	156	216	23	81	160	220	14	53	158	205	94	30	67	188	117	40	75	195	72	17	40	181	54	70	64	145	70	113	65	150	30	45	38	144																																				
20	84	156	216																																																																							
23	81	160	220																																																																							
14	53	158	205																																																																							
94	30	67	188																																																																							
117	40	75	195																																																																							
72	17	40	181																																																																							
54	70	64	145																																																																							
70	113	65	150																																																																							
30	45	38	144																																																																							
35010	m=4	<div><div>r</div><div>g</div><div>b</div></div> <table><tr><td>54</td><td>79</td><td>188</td><td>206</td></tr><tr><td>60</td><td>138</td><td>213</td><td>205</td></tr><tr><td>41</td><td>55</td><td>193</td><td>88</td></tr></table> <table><tr><td>46</td><td>199</td><td>83</td><td>193</td></tr><tr><td>63</td><td>198</td><td>123</td><td>216</td></tr><tr><td>41</td><td>92</td><td>61</td><td>190</td></tr></table> <table><tr><td>54</td><td>203</td><td>72</td><td>184</td></tr><tr><td>66</td><td>200</td><td>129</td><td>209</td></tr><tr><td>42</td><td>85</td><td>54</td><td>183</td></tr></table>	54	79	188	206	60	138	213	205	41	55	193	88	46	199	83	193	63	198	123	216	41	92	61	190	54	203	72	184	66	200	129	209	42	85	54	183																																				
54	79	188	206																																																																							
60	138	213	205																																																																							
41	55	193	88																																																																							
46	199	83	193																																																																							
63	198	123	216																																																																							
41	92	61	190																																																																							
54	203	72	184																																																																							
66	200	129	209																																																																							
42	85	54	183																																																																							
56028	m=4	<div><div>r</div><div>g</div><div>b</div></div> <table><tr><td>166</td><td>15</td><td>86</td><td>243</td></tr><tr><td>148</td><td>16</td><td>85</td><td>233</td></tr><tr><td>96</td><td>7</td><td>55</td><td>194</td></tr></table> <table><tr><td>162</td><td>226</td><td>45</td><td>107</td></tr><tr><td>148</td><td>214</td><td>46</td><td>103</td></tr><tr><td>99</td><td>166</td><td>33</td><td>70</td></tr></table> <table><tr><td>157</td><td>115</td><td>182</td><td>49</td></tr><tr><td>141</td><td>113</td><td>163</td><td>52</td></tr><tr><td>91</td><td>74</td><td>105</td><td>39</td></tr></table>	166	15	86	243	148	16	85	233	96	7	55	194	162	226	45	107	148	214	46	103	99	166	33	70	157	115	182	49	141	113	163	52	91	74	105	39																																				
166	15	86	243																																																																							
148	16	85	233																																																																							
96	7	55	194																																																																							
162	226	45	107																																																																							
148	214	46	103																																																																							
99	166	33	70																																																																							
157	115	182	49																																																																							
141	113	163	52																																																																							
91	74	105	39																																																																							

65019	m=7	r	21	254	91	196	214	255	166	169	251	65	222	24	252	125	133	219	254	19	250	129	44
		g	20	193	64	132	210	251	159	124	198	43	162	21	231	66	93	155	215	20	187	33	28
		b	21	2	1	1	187	256	132	23	11	11	10	19	55	22	45	2	19	18	9	16	14
67079	m=4	r	52	177	41	125				42	56	136	177				42	53	163	88			
		g	83	175	49	130				47	87	142	176				47	85	164	125			
		b	109	130	18	89				22	110	103	132				19	111	121	139			

**Table 1.** Continued.

Name	Num of clusters		Peak locations: 3DHP								Cluster centroid: FCM								Cluster centroid: SFFCM							
71046	m=4	r	102	33	83	212					96	31	70	116				96	113	44	94					
		g	139	45	95	219					134	44	81	145				133	144	57	102					
		b	158	37	83	169					154	34	68	151				155	154	45	85					
76002	m=5	r	24	199	133	71	138			64	132	185	115	32				140	109	36	189	59				
		g	16	212	154	70	137			60	153	201	109	24				162	107	28	204	53				
		b	1	183	155	47	67			31	152	175	60	11				161	46	12	176	29				
95006	m=7	r	39	136	256	117	255	176	192	81	142	245	90	141	225	43	113	87	45	78	249	127	87			
		g	41	171	252	70	97	118	156	53	88	223	82	165	90	43	63	56	43	53	225	164	78			
		b	49	189	213	84	124	129	59	58	92	192	79	178	112	50	66	55	54	61	190	155	85			
35301 3	m=5	r	1	45	205	215	161			154	8	83	204	47			5	53	15	157	190					
		g	10	71	100	240	195			87	13	104	100	67			8	78	27	96	89					
		b	2	70	56	210	170			36	16	94	57	62			18	74	15	36	49					

**Table 2.** Quantitative evaluation of results (F, F', Q and T).

Name	Quantitative evaluation	Quantitative evaluation	Quantitative evaluation
	3DHP	FCM	SFFCM
Lenn a	F = 1.3700e-06	F = 1.2400e-06	F = 2.8797e-06
	F' = 1.4000e-07	F' = 1.2000e-07	F' = 2.8797e-07
	Q = 2.6600e-06	Q = 2.3800e-06	F = 5.5049e-06
	T = 6.0102	T = 1.2037	T = 1.0731
1200 3	F = 5.7133e-07	F = 2.234e-07	F = 2.1131e-06
	F' = 5.7133e-08	F' = 2.234e-08	F' = 2.1131e-07
	Q = 0.00000159	Q = 6.1514e-07	Q = 5.1966e-06
	T = 6.0912	T = 10.8642	T = 2.0153
1207 4	F = 6.8033e-07	F = 4.1665e-07	F = 3.9226e-06
	F' = 6.8033e-08	F' = 4.1665e-08	F' = 3.9226e-07
	Q = 1.477e-06	Q = 9.3783e-07	Q = 8.0792e-06
	T = 6.2889	T = 6.2983	T = 1.9179
1080 73	F = 1.2326e-06	F = 2.5369e-07	F = 8.5821e-07
	F' = 1.2326e-07	F' = 2.5369e-08	F' = 8.5821e-08
	Q = 1.56e-06	Q = 6.245e-07	Q = 2.203e-06
	T = 6.1383	T = 3.2681	T = 1.9291
1240 84	F = 1.8806e-06	F = 6.1935e-07	F = 1.4316e-06
	F' = 1.8806e-07	F' = 6.1935e-08	F' = 1.4316e-07
	Q = 2.9142e-06	Q = 1.4125e-06	Q = 3.5087e-06
	T = 5.8947	T = 6.1178	T = 1.9511
1350 69	F = 5.2978e-06	F = 1.4337e-08	F = 1.3683e-07
	F' = 5.2978e-07	F' = 1.4337e-09	F' = 1.3683e-08
	Q = 2.2316e-06	Q = 3.3285e-08	Q = 3.1574e-07
	T = 6.3629	T = 7.2662	T = 2.4723
1400 75	F = 4.6692e-07	F = 3.6162e-07	F = 8.0344e-07
	F' = 4.6692e-08	F' = 3.6162e-08	F' = 8.0344e-08
	Q = 1.4902e-06	Q = 1.1465e-06	Q = 2.4868e-06
	T = 6.5012	T = 10.2341	T = 2.4577
1690 12	F = 4.6059e-07	F = 3.9575e-07	F = 1.3975e-06
	F' = 4.6059e-08	F' = 3.9575e-08	F' = 1.3975e-07
	Q = 1.4619e-06	Q = 1.2622e-06	Q = 4.6398e-06
	T = 6.1061	T = 5.9679	T = 2.6330
1890 03	F = 1.2906e-06	F = 7.4229e-07	F = 4.9384e-06
	F' = 1.2906e-07	F' = 7.4229e-08	F' = 4.9384e-07
	Q = 2.7658e-06	Q = 1.8435e-06	Q = 1.132e-05
	T = 6.2119	T = 14.7445	T = 2.3438
2090 70	F = 5.4227e-07	F = 2.4162e-07	F = 8.6215e-07
	F' = 5.4227e-08	F' = 2.4162e-08	F' = 8.6215e-08
	Q = 1.1735e-06	Q = 7.1333e-07	Q = 3e-06
	T = 6.2352	T = 6.6647	T = 2.1026
23203 8	F = 9.4895e-07	F = 2.1004e-07	F = 5.6393e-07
	F' = 9.4895e-08	F' = 2.1004e-08	F' = 5.6393e-08
	Q = 1.2228e-06	Q = 5.1624e-07	Q = 1.4172e-06
	T = 6.2871	T = 4.7355	T = 2.0963
2380 11	F = 1.457e-06	F = 1.4087e-08	F = 2.059e-08
	F' = 1.457e-07	F' = 1.4087e-09	F' = 2.059e-09
	Q = 9.0549e-07	Q = 4.6698e-08	Q = 8.14e-08
	T = 6.0678	T = 2.9922	T = 1.9768

**Table 2.** Continued

3500 8	F = 4.1193e-07	F = 2.8003e-07	F = 6.7645e-07
	F' = 4.1193e-08	F' = 2.8003e-08	F' = 6.7645e-08
	Q = 1.0089e-06	Q = 7.4904e-07	Q = 2.0802e-06
	T = 6.4512	T = 4.0833	T = 1.9318
3501 0	F = 2.4355e-07	F = 1.0827e-07	F = 4.6721e-07
	F' = 2.4355e-08	F' = 1.0827e-08	F' = 4.6721e-08
	Q = 8.2145e-07	Q = 3.643e-07	Q = 1.6177e-06
	T = 6.3137	T = 4.0775	T = 2.3095
5602 8	F = 3.5229e-07	F = 2.3402e-07	F = 7.001e-07
	F = 3.5229e-08	F' = 2.3402e-08	F' = 7.001e-08
	Q = 1.0031e-06	Q = 6.7532e-07	Q = 2.4089e-06
	T = 6.2556	T = 5.6569	T = 2.0573
6501 9	F = 2.0482e-06	F = 8.2575e-07	F = 3.3699e-06
	F = 2.0482e-07	F' = 8.2575e-08	F' = 3.3699e-07
	Q = 3.3029e-06	Q = 2.0284e-06	Q = 8.0541e-06
	T = 6.1657	T = 13.7039	T = 2.5992
6707 9	F = 1.4427e-07	F = 5.7994e-08	F = 3.4167e-07
	F = 1.4427e-08	F' = 5.7994e-09	F' = 3.4167e-08
	Q = 4.6106e-07	Q = 1.8602e-07	Q = 8.7742e-07
	T = 6.2369	T = 4.0155	T = 1.9546
7104 6	F = 9.764e-07	F = 5.975e-08	F = 5.0245e-07
	F = 9.764e-08	F' = 5.975e-09	F' = 5.0245e-08
	Q = 9.7541e-07	Q = 1.8072e-07	Q = 1.2051e-06
	T = 6.2244	T = 4.0063	T = 2.1358
7600 2	F = 3.2006e-07	F = 2.1645e-07	F = 6.8293e-07
	F = 3.2006e-08	F' = 2.1645e-08	F' = 6.8293e-08
	Q = 9.3846e-07	Q = 6.4376e-07	Q = 2.0449e-06
	T = 6.1982	T = 6.4097	T = 1.9069
9500 6	F = 7.8068e-06	F = 9.6367e-07	F = 4.0633e-06
	F = 7.8068e-07	F' = 9.6367e-08	F' = 4.0633e-07
	Q = 4.0658e-06	Q = 1.9532e-06	Q = 9.1928e-06
	T = 6.3001	T = 11.3612	T = 2.5159
3530 13	F = 1.1838e-06	F = 2.2597e-07	F = 6.0339e-07
	F = 1.1838e-07	F' = 2.2597e-08	F' = 6.0339e-08
	Q = 1.9317e-06	Q = 6.2344e-07	Q = 1.8535e-06
	T = 6.2308	T = 10.1865	T = 2.1671

**Table 3.** Quantitative evaluation of results (PRI, VoI and GCE).

Name	Quantitative evaluation	Quantitative evaluation	Quantitative evaluation
	3DHP	FCM	SFFCM
1200 3	PRI = 0.702839	PRI = 0.699288	PRI = 0.706441
	VOI = 3.031937	VOI = 3.365230	VOI = 2.409350
	GCE = 0.392419	GCE = 0.432079	GCE = 0.308196
1207 4	PRI = 0.646981	PRI = 0.657461	PRI = 0.756473
	VOI = 2.431389	VOI = 2.475026	VOI = 1.574813
	GCE = 0.371896	GCE = 0.381657	GCE = 0.196605

1080 73	PRI = 0.591311 VOI = 2.212177 GCE = 0.301846	PRI = 0.575904 VOI = 2.552988 GCE = 0.318638	PRI = 0.594306 VOI = 2.265117 GCE = 0.291351
1240 84	PRI = 0.715632 VOI = 2.458209 GCE = 0.337242	PRI = 0.705431 VOI = 2.718757 GCE = 0.370913	PRI = 0.719107 VOI = 2.163888 GCE = 0.272354
1350 69	PRI = 0.985861 VOI = 0.147977 GCE = 0.016432	PRI = 0.335102 VOI = 1.994055 GCE = 0.025972	PRI = 0.396392 VOI = 1.720451 GCE = 0.025217
1400 75	PRI = 0.749074 VOI = 3.508455 GCE = 0.498595	PRI = 0.737769 VOI = 3.716510 GCE = 0.539043	PRI = 0.836686 VOI = 2.073448 GCE = 0.216793
1690 12	PRI = 0.626588 VOI = 4.269046 GCE = 0.501943	PRI = 0.674412 VOI = 4.418008 GCE = 0.556223	PRI = 0.700653 VOI = 3.275286 GCE = 0.334645
1890 03	PRI = 0.669132 VOI = 4.147374 GCE = 0.574200	PRI = 0.683224 VOI = 4.443397 GCE = 0.607502	PRI = 0.689975 VOI = 3.404865 GCE = 0.470955
2090 70	PRI = 0.635576 VOI = 4.409536 GCE = 0.501888	PRI = 0.663834 VOI = 4.569162 GCE = 0.539298	PRI = 0.696154 VOI = 3.573111 GCE = 0.360163
23203 8	PRI = 0.838627 VOI = 2.528876 GCE = 0.300290	PRI = 0.878349 VOI = 2.503639 GCE = 0.335550	PRI = 0.899851 VOI = 1.771766 GCE = 0.217234
2380 11	PRI = 0.930953 VOI = 0.473332 GCE = 0.055855	PRI = 0.804132 VOI = 0.957979 GCE = 0.104131	PRI = 0.669144 VOI = 1.407235 GCE = 0.145632
3500 8	PRI = 0.600557 VOI = 2.863001 GCE = 0.260084	PRI = 0.625892 VOI = 3.237172 GCE = 0.355314	PRI = 0.658769 VOI = 2.601046 GCE = 0.222847
3501 0	PRI = 0.728838 VOI = 3.515352 GCE = 0.419921	PRI = 0.733839 VOI = 3.552653 GCE = 0.432048	PRI = 0.719058 VOI = 3.148263 GCE = 0.345505
5602 8	PRI = 0.592725 VOI = 3.714739 GCE = 0.438474	PRI = 0.603916 VOI = 3.765604 GCE = 0.442325	PRI = 0.625383 VOI = 3.040313 GCE = 0.316286
6501 9	PRI = 0.764709 VOI = 4.731903 GCE = 0.422061	PRI = 0.838701 VOI = 5.246779 GCE = 0.581516	PRI = 0.867411 VOI = 3.410182 GCE = 0.258204
6707 9	PRI = 0.752014 VOI = 2.840880 GCE = 0.327411	PRI = 0.750624 VOI = 2.917691 GCE = 0.342623	PRI = 0.716400 VOI = 2.143727 GCE = 0.153132
7104 6	PRI = 0.902722 VOI = 1.547911 GCE = 0.183469	PRI = 0.708500 VOI = 2.201056 GCE = 0.297266	PRI = 0.722012 VOI = 1.825423 GCE = 0.266854

7600 2	PRI = 0.766120 VOI = 3.483626 GCE = 0.521048	PRI = 0.779309 VOI = 3.440196 GCE = 0.513478	PRI = 0.799879 VOI = 2.458226 GCE = 0.306349
9500 6	PRI = 0.617331 VOI = 3.350518 GCE = 0.543149	PRI = 0.687417 VOI = 3.671150 GCE = 0.583522	PRI = 0.770048 VOI = 2.420249 GCE = 0.359053
3530 13	PRI = 0.751297 VOI = 2.025594 GCE = 0.286181	PRI = 0.724700 VOI = 2.350025 GCE = 0.415924	PRI = 0.825604 VOI = 1.402835 GCE = 0.240564

**Table 4.** Mean values of PRI, Voi and GCE over the Berkeley dataset.

	PRI	VoI	GCE
3DHP	0.685857	2.765545	0.360208
FCM	0.688451	2.979884	0.413387
SFFCM	0.739651	2.130512	0.258597

Based on these results, we conclude that the 3DHP, FCM and SFFCM techniques can all show high quality performance in the segmentation process for at least some images. As it is clear from both visual and numerical results, the proposed 3DHP technique yields promising segmentation results. This is supported by the capability of the method to produce the number of clusters and cluster centroids automatically.

## 5. Conclusion

In this paper, we have introduced a new automated pixel clustering and color image segmentation algorithm. The proposed approach (3DHP) can automatically determine an appropriate number of clusters as well as the cluster centroids, demonstrating the advantage of peak detection using a multimodal optimization algorithm. Since the best number of clusters is often not known a priori in many practical applications, 3DHP can be utilized more widely in practice than existing approaches. The majority of images with differing numbers of clusters from a well-known benchmark data set have been demonstrated to be handled effectively by the proposed approach. The computational experiments have illustrated that the proposed algorithm can automatically discover all known cluster centroids. More importantly, the time required for clustering is not dependent on the size of the image to be segmented. Our approach uses relatively less time to find the cluster centroids compared to FCM, making it a viable algorithm for image segmentation. Furthermore, both the proposed method and FCM and SFFCM yield desirable results in terms of the quantitative evaluation function. The difference in these values is not significant and, for all three techniques, these values approach zero. Finally, experimental results confirm that the proposed 3DHP method can obtain robust and promising segmentation results.

## References

- Aziz, M. A. E., Ewees, A. A., & Hassanien, A. E. (2017). Whale Optimization Algorithm and Moth-Flame Optimization for multilevel thresholding image segmentation. *Expert Systems with Applications*, 83, 242-256.
- Barrera, J., & Coello Coello, C. A. (2009). A Particle Swarm Optimization Method for Multimodal Optimization Based on Electrostatic Interaction. In A. H. Aguirre, R. M. Borja & C. A. R. García (Eds.), *MICA 2009: Advances in Artificial Intelligence: 8th Mexican International Conference on Artificial Intelligence, Guanajuato, México, November 9-13, 2009. Proceedings* (pp. 622-632). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Borsotti, M., Campadelli, P., & Schettini, R. (1998). Quantitative evaluation of color image segmentation results. *Pattern recognition letters*, 19, 741-747.
- Brits, R., Engelbrecht, A. P., & van den Bergh, F. (2007). Locating multiple optima using particle swarm optimization. *Applied Mathematics and Computation*, 189(2), 1859-1883.
- Chang, D., Zhao, Y., Liu, L., & Zheng, C. (2016). A dynamic niching clustering algorithm based on individual-connectedness and its application to color image segmentation. *Pattern recognition*, 60, 334-347.
- Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2018). DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40, 834-848.
- Cheng, H.-D., Jiang, X. H., Sun, Y., & Wang, J. (2001). Color image segmentation: advances and prospects. *Pattern recognition*, 34, 2259-2281.
- Eberhart, R., & Kennedy, J. (1995). A new optimizer using particle swarm theory. In *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on* (pp. 39-43): IEEE.
- He, L., & Huang, S. (2017). Modified firefly algorithm based multilevel thresholding for color image segmentation. *Neurocomputing*, 240, 152-174.
- Hettiarachchi, R., & Peters, J. F. (2017). Voronoï region-based adaptive unsupervised color image segmentation. *Pattern recognition*, 65, 119-135.
- Horng, M.-H., & Liou, R.-J. (2011). Multilevel minimum cross entropy threshold selection based on the firefly algorithm. *Expert Systems with Applications*, 38, 14805-14811.
- Kapur, J., Sahoo, P. K., & Wong, A. K. (1985). A new method for gray-level picture thresholding using the entropy of the histogram. *Computer vision, graphics, and image processing*, 29, 273-285.
- Khairuzzaman, A. K. M., & Chaudhury, S. (2017). Multilevel thresholding using grey wolf optimizer for image segmentation. *Expert Systems with Applications*, 86, 64-76.
- Kumar, S., Pant, M., Kumar, M., & Dutt, A. (2015). Colour image segmentation with histogram and homogeneity histogram difference using evolutionary algorithms. *International Journal of Machine Learning and Cybernetics*, 1-21.
- Kurugollu, F., Sankur, B., & Harmanci, A. E. (2001). Color image segmentation using histogram multithresholding and fusion. *Image and vision computing*, 19, 915-928.
- Lei, T., Jia, X., Zhang, Y., Liu, S., Meng, H., & Nandi, A. K. (2018). Superpixel-based Fast Fuzzy C-Means Clustering for Color Image Segmentation. *IEEE Transactions on Fuzzy Systems*, 1-1.
- Li, X. (2007). A multimodal particle swarm optimizer based on fitness Euclidean-distance ratio. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation* (pp. 78-85): ACM.
- Liu, J., & Yang, Y.-H. (1994). Multiresolution color image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16, 689-700.
- Martin, D., Fowlkes, C., Tal, D., & Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001* (Vol. 2, pp. 416-423 vol.412).
- Meila, M. (2002). Comparing clusterings.
- Navon, E., Miller, O., & Averbuch, A. (2005). Color image segmentation based on adaptive local thresholds. *Image and vision computing*, 23, 69-85.
- Otsu, N. (1979). An automatic threshold selection method based on discriminate and least squares criteria. *Denshi Tsushin Gakkai Ronbunshi*, 63, 349-356.
- Panagiotakis, C., Grinias, I., & Tziritas, G. (2011). Natural Image Segmentation Based on Tree Equipartition, Bayesian Flooding and Region Merging. *IEEE Transactions on Image Processing*, 20, 2276-2287.
- Pare, S., Kumar, A., Bajaj, V., & Singh, G. K. (2016). A multilevel color image segmentation technique based on cuckoo search algorithm and energy curve. *Applied Soft Computing*, 47, 76-102.
- Parsopoulos, K. E., & Vrahatis, M. N. (2001). Modification of the particle swarm optimizer for locating all the global minima. In *Artificial Neural Nets and Genetic Algorithms* (pp. 324-327). Springer, Vienna.
- Qu, B.-Y., Liang, J. J., & Suganthan, P. N. (2012). Niching particle swarm optimization with local search for multi-modal optimization. *Information Sciences*, 197, 131-143.
- Raja, N., Rajinikanth, V., & Latha, K. (2014). Otsu based optimal multilevel image thresholding using firefly algorithm. *Modelling and Simulation in Engineering*, 2014, 37.
- Rajinikanth, V., Aashiha, J., & Atchaya, A. (2014). Gray-level histogram based multilevel threshold selection with bat algorithm. *International Journal of Computer Applications*, 93.
- Rajinikanth, V., & Couceiro, M. S. (2015). RGB Histogram Based Color Image Segmentation Using Firefly Algorithm. *Procedia Computer Science*, 46, 1449-1457.
- Sarabi, A., & Aggarwal, J. K. (1981). Segmentation of chromatic images. *Pattern recognition*, 13, 417-427.
- Sarkar, S., & Das, S. (2013). Multilevel image thresholding based on 2D histogram and maximum Tsallis entropy—a differential evolution approach. *IEEE Transactions on Image Processing*, 22, 4788-4797.
- Sathya, P. D., & Kayalvizhi, R. (2011). Optimal multilevel thresholding using bacterial foraging algorithm. *Expert Systems with Applications*, 38, 15549-15564.



- Schacter, B. J., Davis, L. S., & Rosenfeld, A. (1976). Scene segmentation by cluster detection in color spaces. *ACM SIGART Bulletin*, 16-17.
- Shapiro, L. G., & Stockman, G. C. Computer Vision, 2001, 279-325. In: New Jersey, Prentice-Hall, ISBN 0-13-030796-3.
- Siang Tan, K., & Mat Isa, N. A. (2011). Color image segmentation using histogram thresholding – Fuzzy C-means hybrid approach. *Pattern recognition*, 44, 1-15.
- Sutton, M. A., Bezdek, J. C., & Cahoon, T. C. (2000). Image segmentation by fuzzy clustering: methods and issues. In *Handbook of medical imaging* (pp. 87-106): Academic Press, Inc.
- Syu, J., Wang, S., & Wang, L. (2017). Hierarchical Image Segmentation Based on Iterative Contraction and Merging. *IEEE Transactions on Image Processing*, 26, 2246-2260.
- Tenenbaum, J. M., Garvey, T. D., Weyl, S., & Wolf, H. C. (1974). An Interactive Facility for Scene Analysis Research. In: STANFORD RESEARCH INST MENLO PARK CALIF.
- Underwood, S., & Aggarwal, J. (1977). Interactive computer analysis of aerial color infrared photographs. *Computer Graphics and Image Processing*, 6, 1-24.
- Wang, H., Moon, I., Yang, S., & Wang, D. (2012). A memetic particle swarm optimization algorithm for multimodal optimization problems. *Information Sciences*, 197, 38-52.
- Yang, Y., & Huang, S. (2012). Image segmentation by fuzzy C-means clustering algorithm with a novel penalty term. *Computing and Informatics*, 26, 17-31.