

# Image Segmentation Based on Representative Colors Detection and Region Merging

Giuliana Ramella and Gabriella Sanniti di Baja

Istituto di Cibernetica "E.Caianello", CNR  
Via Campi Flegrei 34, 80078 Pozzuoli, Naples, Italy  
{g.ramella,g.sannitidibaja}@cib.na.cnr.it

**Abstract.** We present a color image segmentation algorithm, RCRM, based on the detection of Representative Colors and on Region Merging. The 3D color histogram of the RGB input image is built. Colors are processed in decreasing frequency order and a grouping process is accomplished to gather in the same cluster all colors that are close enough to the current color. Colormapping is done to originate a preliminary image segmentation. Segmentation regions having small size undergo a merging process. Merging is actually accomplished only for adjacent regions whose colors do not significantly differ. The parameters involved by the algorithm are set automatically by taking into account color distribution in the input image and geometrical features of the regions into which the image is partitioned. The algorithm has been tested on a large number of RGB color images originating satisfactory results.

**Keywords:** RGB color images, 3D histogram, color quantization, image segmentation.

## 1 Introduction

Segmentation is one of the most important processes when dealing with computer vision applications and has a crucial role since the quality of the obtained result is strongly conditioned by the quality of segmentation. It consists in a suitable partition of an image into non-overlapping, connected and homogeneous regions, each of which represents (part of) an object in the scene. All pixels in the same region are rather similar to each other in terms of a specific property (for example, color or texture). In turn, pixels belonging to adjacent regions significantly differ from each other as regards the same property.

While segmentation is a quite easy task for human beings, it is definitely hard for machines and a large number of different segmentation schemes can be found in the literature. Low level image features, such as color and texture, as well as higher level features, such as geometric models for the objects or Gestalt principle of perceptual organization, have been used. Some surveys covering most of the approaches suggested in the literature can be found for example in [1-3].

When color is the main feature used to achieve segmentation, segmentation methods can be framed as following two possible approaches, respectively involving

edge detection and region reconstruction [1]. Edge detection schemes are based on the assumption that local discontinuities of colors exist at the boundaries of adjacent regions representing different objects. In turn, region reconstruction schemes are based on the assumption that a region is a subset of connected pixels sharing similar color properties. Region reconstruction methods can be furthermore divided into two categories, depending on whether color distribution is analyzed in the image plane or in the color space. Spatial analysis is based on region growing that, starting from a set of initial seeds, sequentially groups in the same regions neighboring pixels with similar colors as far as the aggregating conditions are satisfied. In turn, when the analysis of color distribution is accomplished in the color space, where each pixel of the input image is associated with its color point in the three-dimensional color space, regions that are homogeneous in the image plane originate clusters of color points in the color space. Image pixels with similar color properties are grouped in the same cluster. Clusters can be identified by analyzing the color histogram or by means of a cluster analysis procedure, and are successively mapped back to the original image plane to obtain the segmented image. Independently of whether the image plane or the color space are analyzed, the result is likely to be affected by the presence of a number of small size regions scattered through the image. Thus, a merging process is also generally taken into account to reduce over-segmentation [2].

The segmentation algorithm proposed in this paper can be classified as a region reconstruction method, where color distribution is analyzed in the color space. The algorithm, named RCRM, includes two steps, respectively aiming at the detection of Representative Colors in the 3D color space, and at Region Merging in the image plane. The first step originates a preliminary over-segmented partition of the input image. The second step reduces over-segmentation. RCRM is the follow up of a previous method [4]. The main novelties introduced by RCRM are the automatic setting of the parameters and a clever merging process. RCRM has been tested on a large dataset of images taken from publicly available repositories, producing satisfactory results from both the qualitative and the quantitative points of view.

The paper is organized as follows: basic notions are given in Section 2; the algorithm is described in Section 3; experimental results are discussed in Section 4; finally, concluding remarks are given in Section 5.

## 2 Basic Notions

Let  $I$  be a 2D RGB color image. We interpret colors of the pixels of  $I$  as 3D vectors, where each vector element has an 8-bit dynamic range. For each color in the 2D image, the coordinates of the corresponding color point along the three Cartesian axes are integer numbers in the range  $[0, 255]$ . Thus, colors present in the 2D image correspond to color points within a 3D cube. Of course, the same color is likely to characterize more than one pixel in  $I$ . By assigning to each color point in the 3D cube a value counting the number of pixels of  $I$  characterized by the color corresponding to that point, the 3D histogram of colors  $H$  becomes available.

Though in principle 16 millions of different colors are possible, the number of colors in a digital image is generally remarkably smaller. This is due both to the

obvious limitation given by the size of the images (for example, for images with size  $1024 \times 1024$ , about one million different colors are possible), and to the fact that the same color is likely to appear more than once in the image. Thus, sparse points having value different from zero generally exist in the 3D histogram  $H$ . Since the three components and the frequency of a color are expressed by integer numbers,  $H$  can be interpreted as a discrete 3D voxel image.

Let  $C_i$  be a connected component of voxels of  $H$ , where each voxel is characterized by the same frequency  $f_i$ .  $C_i$  has locally maximal frequency if each voxel of  $H$  that does not belong to  $C_i$  but has at least a neighbor in  $C_i$  has frequency smaller than  $f_i$ . For any component with locally maximal frequency,  $MC_i$ , we identify the centroid  $c_i$ . Obviously, if  $MC_i$  consists of a single voxel, the centroid coincides with the voxel itself and the coordinates of  $c_i$  are certainly expressed by integer numbers. When  $MC_i$  includes more than one voxel, we use the ceiling function to express the coordinates of the centroid  $c_i$  by integer numbers, i.e., we map the real numbers expressing the coordinates of  $c_i$  to the largest following integer numbers. The color corresponding to the centroid  $c_i$  is taken as the color for all voxels of  $MC_i$ .

For the sake of completeness, we remark that synthetical images can be built where  $MC_i$  is likely to include voxels so far from each other to correspond to very different colors. However, this does not generally happen for natural images. Thus, we can assume that any  $MC_i$  found in  $H$  includes only voxels corresponding to colors sufficiently similar to each other.

### 3 Segmentation

The segmentation algorithm RCRM consists of two steps. The first step involves the analysis of the 3D histogram  $H$  to identify the representative colors and colormapping to originate a preliminary partition of  $I$ . The connected components of colors having locally maximal frequency are identified in  $H$ . The color corresponding to the centroid of any such a component is taken as the color of the component itself. Then, a grouping process is done to select a subset of the centroids so as to detect the representative colors. The centroids are processed in decreasing frequency order, starting from the centroid with the largest frequency. The currently examined centroid  $c_i$  is taken as a representative color. A grouping process builds the cluster associated to  $c_i$  by including in the same cluster all colors present in the 3D histogram that satisfy the following conditions: i) have smaller frequency with respect to  $c_i$ , ii) are sufficiently close to  $c_i$ , and iii) have not been assigned to any other cluster already built. Since the cluster of  $c_i$  may include also colors with locally maximal frequency, the number of representative colors will generally result to be significantly smaller than the initial number of centroids. When all centroids have been considered, the color points corresponding to the colors of the input image result to be grouped into clusters, each of which obtained in correspondence with a centroid taken as representative color. Colormapping is then accomplished to complete the first step of the process and obtain the preliminary partition of the 2D input image. To this aim, the color of each pixel of the image is replaced by the representative color of the

corresponding cluster. The second step is done to reduce over-segmentation. In fact, since the histogram of colors does not take into account spatial information, the image obtained at the end of the first step is likely to be over-segmented. During the second step, a merging process is done, based on the size of the partition regions and on the distance of colors of adjacent regions.

We point out that our representative colors detection technique can be classified as a color clustering technique. With respect to other color clustering techniques, such as K-means [5] and the fuzzy C-means [6], an advantage of our method is that we do not need to fix a priori the number of clusters and the distribution of the centroids. In fact, the clusters are as many as the centroids that survive the grouping process, and are found in correspondence with the connected components of colors with locally maximal frequency in the histogram, i.e., with the colors that appear more often than their neighboring colors in the image. Since we work directly on the 3D histogram, no information contained in the dependence among the three color components is lost.

RCRM involves four parameters, whose values are automatically computed during the process in terms of features characterizing the image, as it will be described in the following. Two of these parameters,  $\pi$  and  $\delta$ , are employed during Step 1 and are respectively concerned with the minimal value that the frequency of a centroid should have in order that centroid is considered as a possible representative color, and with the maximal distance from a centroid that a color should have to be assigned to the cluster associated to that centroid. The remaining two parameters,  $\alpha$  and  $\beta$ , are used during Step 2. The parameter  $\alpha$  is concerned with the maximal size of regions that undergo the merging process. The parameter  $\beta$  is concerned with the distance of the color of a small size region from the colors of the adjacent regions.

### 3.1 Step 1- Selection of Representative Colors and Colormapping

Preliminarily, the centroid  $c_H$  of the color distribution for the input image  $I$  is detected and the arithmetic mean  $\lambda$  of the distances of all colors in  $H$  from  $c_H$  is computed. All  $MC_i$ , i.e., the connected components of voxels of  $H$  with locally maximal frequency, are identified and the arithmetic mean  $\mu$  of their frequencies is computed. The centroid  $c_i$  of any  $MC_i$  is identified. If all centroids are taken as possible representative colors, a large number of representative colors is likely to be obtained, which are not all equally meaningful. Moreover, a longer computation time would be necessary to complete Step 1. Thus, we introduce the parameter  $\pi$  and consider as possible representative colors only the centroids with frequency larger than  $\pi$ . Of course, the same value of  $\pi$  would not produce equally good results for images with different color distribution. Accordingly, we relate the value of  $\pi$  to the arithmetic mean  $\mu$  of the frequencies of the centroids, so as to have a dependence of  $\pi$  on color distribution. To avoid a too drastic reduction of the number of possible representative colors, we should keep the parameter  $\pi$  rather small. Thus, we tentatively set  $\pi=0,01\mu$  and use the ceiling function to get an integer approximation for  $\pi$ . The final value for  $\pi$  is computed by taking into account also the number of pixels of  $I$  with colors that would not be taken as representative colors due to their small frequency. To this purpose, we compute the sum  $S$  of the frequencies of the centroids that, according to the tentative

value of  $\pi$ , would not be regarded as possible representative colors, and increase or decrease by one the tentative value of  $\pi$  depending on the sum  $S$ . In practice, we compare  $S$  with the total number  $N$  of pixels of  $I$ . If  $S$  is less than a given percentage of  $N$  (set to 40% in our work), the tentative value of  $\pi$  is increased, otherwise the value is decreased. The sum  $S$  is newly computed and  $\pi$  is again suitably increased/decreased. The final value of  $\pi$  is the largest possible value for which  $S$  remains less than the selected percentage of  $N$ . We have experimentally verified that, in the average, satisfactory results are achieved by computing the value of  $\pi$  according to the above strategy and by using the two suggested percentages ( $1\%N$  and  $40\%N$ ). Such a value is in fact large enough to reduce the number of possible representative colors so as to limit the computation time, and at the same time is small enough to guarantee that only for a small number of pixels of  $I$  a representative color is not identified. These are the pixels of  $I$  whose colors have frequency smaller than  $\pi$  (and, hence, are not taken as representative colors themselves) and largely differ from the colors of detected representative colors (and, hence, are not grouped with any detected representative color).

All centroids with frequency not smaller than  $\pi$  are examined in decreasing frequency order, starting from the centroid with the largest frequency. The rationale for checking centroids in decreasing frequency order is that we regard the relevance of a color as proportionally related to the number of times that the color appears in the image. The current centroid  $c_i$  is taken as a representative color. The Euclidean distance is computed between  $c_i$  and each color in  $H$  that has not yet been assigned to clusters built in correspondence with already selected representative colors and has smaller frequency with respect to  $c_i$ . Colors at distance from  $c_i$  smaller than the value of the parameter  $\delta$  are grouped with the centroid  $c_i$ . The value of  $\delta$  should be fixed by taking into account the maximal distance between two colors that a user would not distinguish from each other. The larger is  $\delta$  the smaller are the numbers of representative colors and of segmentation regions, but the quality of the segmentation becomes rougher. To relate  $\delta$  to the distribution of colors in  $I$ , we set its value to a percentage of the arithmetic mean  $\lambda$  of the distances of all colors in  $H$  from the centroid  $c_H$  of the color distribution for the input image  $I$ . We have experimentally verified that satisfactory results are generally obtained by setting  $\delta=0,5\lambda$ . The above process is likely to group with the current centroid also some centroids with smaller frequency placed at distance from  $c_i$  smaller than  $\delta$ . Thus, not all centroids with frequency larger than  $\pi$  will actually be selected as representative colors.

Once the representative colors have been selected, the preliminary segmentation  $I'$  is obtained by colormapping. Any pixel that in the input image  $I$  has a given color is set in  $I'$  to the corresponding representative color. As already remarked, some colors present in  $I$  may not be associated to any representative color and, hence, remain temporarily without any assigned color in  $I'$ . The two parameters  $\pi$  and  $\delta$  play an important role in this respect and their values have to be carefully selected to guarantee at the same time a limited number of representative colors and a small number of pixels without any assigned color in  $I'$ . Large values for  $\pi$  favor reduction of the number of centroids, i.e., of possible representative colors, but may originate large portions of pixels without any assigned color. Analogously, large values of  $\delta$

definitely favor color grouping, which reduces the number of pixels without any assigned color, but can produce an excessive color fusion that would result in unnatural colors. The criteria to compute the values for  $\pi$  and  $\delta$  described in this paper have experimentally resulted adequate to have only tiny regions of  $I'$  consisting of pixels without any assigned color. These pixels are assigned the color of their neighbors. When more than one color can be assigned, we follow a majority rule and assign to the pixel the more frequent color among the colors of its neighbors. While coloring the pixels of  $I'$  that were not assigned a color during colormapping, also noise cleaning is performed by using the same majority rule: pixels of  $I'$  grouped into components having very small size (set to at most four pixels in this paper) are assigned the color of their neighbors. We remark that pixels of  $I'$  that are colored with the same representative color do not necessarily belong to the same connected component. In fact, the histogram of colors counts the number of times that a color appears in the image and the grouping process measures the distance among points in the histogram, but spatial information is not taken into account. As an example of the performance of Step 1 refer to Fig. 1. For the input image 118035 consisting of  $321 \times 481$  pixels with 23786 different colors (Fig. 1 left), 13134 centroids were detected in the 3D histogram characterized by  $\mu=348$ . Only 14 representative colors were selected by using the automatically computed parameter values  $\pi=3$  and  $\delta=46$ . The resulting image (Fig. 1 middle) is over-segmented, since it includes 348 regions.



**Fig. 1.** The input image 118035, left, preliminary segmentation, middle, final segmentation, right

### 3.2 Step 2 - Region Merging

Step 2 is aimed at region merging to reduce over-segmentation. Preliminarily, connected component labeling is accomplished on  $I'$  to distinguish all regions of the partition, some of which are possibly characterized by the same representative color. At the same time, the area of each connected component is also computed as well as the arithmetic mean  $A$  of the areas of all partition regions.

During one inspection of  $I'$ , regions with area smaller than an a priori fixed value  $\alpha$  are considered for merging. The parameter  $\alpha$  is related to a percentage of  $A$ , so that different values of  $\alpha$  will be used for images characterized by preliminary segmentations that differ from each other for the number of partition regions or for their geometric properties. We have used different values for  $\alpha$  to process the images of our dataset. Our experiments suggested to set  $\alpha=0,2A$  as default value. Let  $R_k$  be a region with size smaller than  $\alpha$ . Merging of  $R_k$  is done only if the color of  $R_k$  does not

significantly differ from the colors of the adjacent regions that will absorb the pixels of  $R_k$ . Practically, for  $R_k$  with identity label  $l_k$  and color  $c_j$ , merging is accomplished by ascribing to the pixels of  $R_k$  the identity label  $l_n$  of the adjacent region whose color  $c_m$  is the closest to  $c_j$ , provided that the distance of  $c_m$  from  $c_j$  is smaller than a parameter  $\beta$ . Since the distance between colors of adjacent regions in  $I'$  is not smaller than  $\delta$ , a value larger than  $\delta$  should be used for  $\beta$ . We suggest  $\beta=\lambda$  as default value, by taking into account that we set  $\delta=0,5\lambda$ , where  $\lambda$  is the arithmetic mean of the distances of all colors in  $H$  from the centroid  $c_H$  of the color distribution for the input image  $I$ .

We point out that merging may reduce the number of final representative colors with respect to the number of those selected during Step 1. This happens whenever all pixels that were assigned a given representative color  $c_j$  are grouped into regions satisfying the merging conditions. We also point out that the number of regions is generally larger than the number of final representative colors.

In Fig. 1 right, the final segmentation for the input image 118035 is shown.

The arithmetic mean of the area of the 348 regions detected in  $I'$  before merging is  $A=439$ , which implies  $\alpha=88$ . Out of 312 regions with size smaller than  $\alpha$ , 270 regions are actually merged to adjacent regions. Only 78 regions remain after merging and the number of final colors is 11.

## 4 Experimental Results

RCRM has been tested on about 100 images with different size and color distribution, taken from databases publicly available. A subset of the above collection, including 10 test images taken from Berkeley Segmentation Dataset and Benchmarking (<http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/>) is shown in Fig. 2.



Fig. 2. Test images

As concerns performance evaluation, we remark that this can be accomplished by following either a supervised approach or an unsupervised approach. In the former case, a ground truth reference is necessary for the evaluation. However, the ground truth is often manually generated and different users may originate a different ground truth, so that evaluation is somehow subjective. In the latter case, a quality score based only on the segmented image is used. Several goodness measures are available in the literature, but unfortunately a unique completely satisfactory measure has not yet been found and the combination of different measures to obtain a single evaluation function is subjective. Thus, we prefer to resort to an unsupervised approach and instead of combining different measures we simply count the number of measures for which the performance of a given segmentation method is better with respect to that of other methods. For each segmented image we compute the number

of representative colors NC, the number of regions NR, and three goodness measures (the Liu and Yang function F, the Borsotti et al. function Q and the entropy E [7]).

We have compared the performance of RCRM with that of the Color Structure Code CSC [8]), the Edge Detection and Image Segmentation EDISON [9], and the Recursive Hierarchical Image Segmentation RHSEG [10]. The results are summarized in Table 1. To qualitatively compare the performances of RCRM, CSC, EDISON and RHSEG, refer to Fig. 3, where the results obtained for the test image 46076 are shown.

**Table 1.** Performance evaluation

Image	Methods	NC	NR	F	Q	E
118035	CSC	347	662	3.681	0.732	10.947
	EDISON	30	86	0.155	0.082	<b>10.588</b>
	RHSEG	3	375	0.961	0.238	10.680
	RCRM	11	<b>78</b>	<b>0.100</b>	<b>0.036</b>	10.650
143090	CSC	68	83	0.123	0.135	13.746
	EDISON	40	<b>71</b>	0.162	0.170	13.848
	RHSEG	4	1011	0.810	0.703	13.863
	RCRM	13	102	<b>0.114</b>	<b>0.046</b>	<b>13.382</b>
196973	CSC	339	736	1.785	0.574	9.323
	EDISON	56	163	0.093	0.053	9.324
	RHSEG	2	633	0.219	0.654	<b>9.290</b>
	RCRM	7	<b>60</b>	<b>0.059</b>	<b>0.030</b>	9.353
2092	CSC	203	246	0.285	0.136	12.230
	EDISON	34	<b>78</b>	<b>0.112</b>	0.089	12.213
	RHSEG	4	1327	1.297	2.478	<b>12.183</b>
	RCRM	15	164	0.255	<b>0.060</b>	12.271
22090	CSC	758	1093	3.873	0.631	12.654
	EDISON	144	242	<b>0.322</b>	0.108	<b>12.595</b>
	RHSEG	5	1432	1.044	1.503	12.532
	RCRM	13	<b>224</b>	0.381	<b>0.097</b>	12.599
253036	CSC	192	378	1.666	0.390	12.340
	EDISON	27	<b>31</b>	0.100	0.152	12.355
	RHSEG	2	46	<b>0.059</b>	0.061	<b>12.306</b>
	RCRM	6	43	0.073	<b>0.042</b>	12.418
24004	CSC	2027	3585	52.316	45.661	14.593
	EDISON	214	<b>621</b>	3.159	0.740	14.598
	RHSEG	6	8064	52.731	1455.036	14.631
	RCRM	17	655	<b>3.116</b>	<b>0.532</b>	<b>14.567</b>
368078	CSC	723	1071	5.948	1.562	14.527
	EDISON	233	<b>403</b>	<b>1.848</b>	0.808	14.882
	RHSEG	<b>5</b>	3383	7.888	49.315	<b>14.500</b>
	RCRM	21	597	2.943	<b>0.600</b>	14.621
46076	CSC	167	235	0.444	0.151	12.954
	EDISON	34	<b>82</b>	<b>0.227</b>	0.208	<b>12.814</b>
	RHSEG	3	1437	2.328	3.748	12.852
	RCRM	18	223	0.440	<b>0.109</b>	12.913
60079	CSC	134	241	0.445	0.091	<b>9.251</b>
	EDISON	14	39	0.081	0.109	9.249
	RHSEG	3	1622	0.672	5.475	9.572
	RCRM	10	<b>33</b>	<b>0.019</b>	<b>0.010</b>	9.399

The smallest values of F, Q and E (in bold in Table 1) denote the best performance with respect to that goodness measure. As regards NR, note that a too small (large) value could denote under-segmentation (over-segmentation). It is then difficult to say whether the smallest (largest) value of NR denotes the best performance. To facilitate



reading of Table 1, we have set in bold the smallest value for NR. As for NC, we just give it as an indication of the performance by taking into account the complexity of the input image, but don't consider it as an evaluation measure. In the majority of cases, RCRM has a better performance for a larger number of goodness measures. However, we remark that counting the number of goodness measures for which the performance of a given segmentation method is better with respect to that of other methods is not enough to claim that a given method outperforms with respect to the others. The results should also be at least qualitatively evaluated.



Fig. 3. From left to right, segmentation results with RCRM, CSC, EDISON and RHSEG

## 5 Concluding Remarks

We have presented the algorithm for image segmentation RCRM based on the detection of representative colors in the 3D histogram and on region merging. Colors present in the input image  $I$  are considered as possible representative colors on the basis of their frequency in the histogram and are processed in decreasing frequency order. The currently processed color is taken as a representative color and a grouping process is accomplished to gather with it all colors with smaller frequency, provided that they have not yet been grouped with any previous representative color and are such that their distance from the currently processed representative color is smaller than an a priori fixed value, set depending on color distribution. Once representative colors have been selected, the preliminary segmented image  $I'$  is built by colormapping. Since  $I'$  is likely to result over-segmented, a merging process is accomplished. Only regions with size smaller than an a threshold, whose value is related to the arithmetic mean of the areas of all regions of  $I'$ , undergo to merging. Regions are actually merged only provided that their colors do not significantly differ from the colors of the adjacent regions that will absorb their pixels.

The values for the four parameters involved by the process are automatically computed by taking into account properties evaluated on the image at hand. RCRM has been tested on about 100 RGB color images with different size and color distribution, and the obtained results can be regarded as generally satisfactory.

## References

1. Cheng, H.D., Jiang, X.H., Sun, Y., Wang, J.: Color image segmentation: advances and prospects. *Pattern Recognition* 34, 2259–2281 (2001)
2. Busin, L., Vandenbroucke, N., Macaire, L.: Color spaces and image segmentation. In: *Advances in Imaging and Electron Physics*, ch. 2, JCR Science Edition, Orlando, FL, USA, vol. 1, pp. 65–168 (2008)
3. Bhattacharyya, S.: A Brief Survey of Color Image Preprocessing and Segmentation Techniques. *J. of Pattern Recognition Research* 1, 120–129 (2011)
4. Ramella, G., Sanniti di Baja, G.: Color histogram-based image segmentation. In: Real, P., Diaz-Pernil, D., Molina-Abril, H., Berciano, A., Kropatsch, W. (eds.) *CAIP 2011, Part I. LNCS*, vol. 6854, pp. 76–83. Springer, Heidelberg (2011)
5. Lloyd, S.P.: Least squares quantization in PCM. *IEEE Trans. Information Theory* 28(2), 129–136 (1982)
6. Berkhin, P.: Survey of clustering data mining techniques. *Accrue Software*, San Jose, CA. Technical Report (2002)
7. Zhang, H., Fritts, J., Goldman, S.: An entropy-based objective evaluation method for image segmentation. In: *Proceedings of SPIE 5307 Storage and Retrieval Methods and Applications for Multimedia*, pp. 38–49 (2004)
8. Rehrmann, V., Priebe, L.: Fast and robust segmentation of natural color scenes. In: Chin, R., Pong, T.-C. (eds.) *ACCV 1998. LNCS*, vol. 1351, pp. 598–606. Springer, Heidelberg (1997)
9. Comaniciu, D., Meer, P.: Mean Shift: A Robust Approach Toward Feature Space Analysis. *IEEE Trans. Pattern Anal. Machine Intell.* 24(5), 603–619 (2002)
10. Tilton, J.: D-dimensional formulation and implementation of recursive hierarchical segmentation, *Disclosure of Invention and New Technology: NASA Case No. GSC 15199-1* (2006)