Color Image Segmentation

Chapter · October 2006

DOI: 10.1201/9781420009781.ch5

1 author:

Henryk Palus
Silesian University of Technology

Chapter 1

Color Image Segmentation: Selected Techniques

HENRYK PALUS

Silesian University of Technology
Department of Automatic Control
Akademicka 16 Str., 44-101 Gliwice, Poland
Email: hpalus@polsl.pl

Invited Chapter to appear in "Color Image Processing: Emerging Applications", CRC Press, Rastislav Lukac and Konstantinos N. Plataniotis, Editors.

1.1 Introduction

In color image processing and its applications the great importance is attached to the techniques used for image segmentation. The quality of segmentation results have a big impact on the next steps of image processing, for example on the object recognition and tracking, the retrieval in image databases etc. The goal of image segmentation is partitioning of the image into homogeneous and connected regions without using an additional knowledge about objects in the image. The homogeneity of regions in color image segmentation involves in natural way colors and sometimes color textures [9]. In the segmented image the regions have, in contrast to single pixels, many interesting features like shape, texture etc. The human being recognizes objects in the environment using his visual system and by the way he or she segments color images.

First ”state of the art” papers in the field of color image segmentation come from 1990s [44, 24, 19, 8]. In the first color image processing handbooks we can find separate chapters devoted to the color image segmentation [39, 38]. Almost all image segmentation techniques developed earlier for grayscale images [32] have been also applied to segmentation of color images. Each such expansion into color images is connected with a choice of some color space. The segmentation techniques use very different mathematical tools, but one method, which is effective for each color image, has not as yet been developed.

Often the segmentation of object from the background requires processing of its color image. An example of such situation is presented in Fig1.1, that shows color and grayscale versions of image *Flowers1*. Both versions of the image have been segmented by the technique of seeded region growing, described in Section 1.3. The goal of example segmentation task is to segment out blue petals of flower, that is placed in the center of the image. The segmentation for both image versions (Fig1.1(a), Fig1.1(b)) starts from a seed located in the same point on the upper petal of flower. All attempts to segment out the blue petals from grayscale image consisting in changing a parameter d (Fig1.1(c)-(e)), have failed. In contrast to grayscale image, the segmentation of color image by the same technique gives good result (Fig1.1(f)) and shows a potential of color image processing.

If an image after segmentation contains many small regions corresponding to homogeneous objects in the original image, then we can use a new term: the oversegmentation. On the other hand, if an image after segmentation contains few large regions and each region corresponds to several objects in the original image, then this case can be named the undersegmentation. Fig1.2 shows the color image *Parrots* and examples of oversegmented and undersegmented image. Pseudocolors have been used for better visualization of oversegmentation effect (Fig1.2(d)). Erroneous image segmentation (e.g. oversegmentation, undersegmentation) is a source of errors in the further image analysis and recognition. However, the oversegmentation is more convenient in further processing, because with help of suitable postprocessing techniques, we can decrease

(a) (b)

(c) (d)

(e) (f)

Figure 1.1: Example segmentation results: (a) color image *Flowers1*, (b) grayscale image *Flowers1*, (c) segmented grayscale image (parameter $d = 30$), (d) segmented grayscale image (parameter $d = 50$), (e) segmented grayscale image (parameter $d = 70$), (f) segmented color image (parameter $d = 100$).

the number of regions in the image. Fig1.2(e) shows for color image *Parrots* relatively good segmentation result (62 regions). The white contours have been superimposed on the original image for presenting the results of segmentation.

Among many existing methods for color image segmentation, we can distinguish four main categories: pixel-based techniques, region-based techniques, contour-based techniques and hybrid techniques. The last category collects methods that integrate two techniques from former categories, for example pixel-based and region-based techniques [6] as well as methods using regions and contours simultaneously [14]. Sometimes

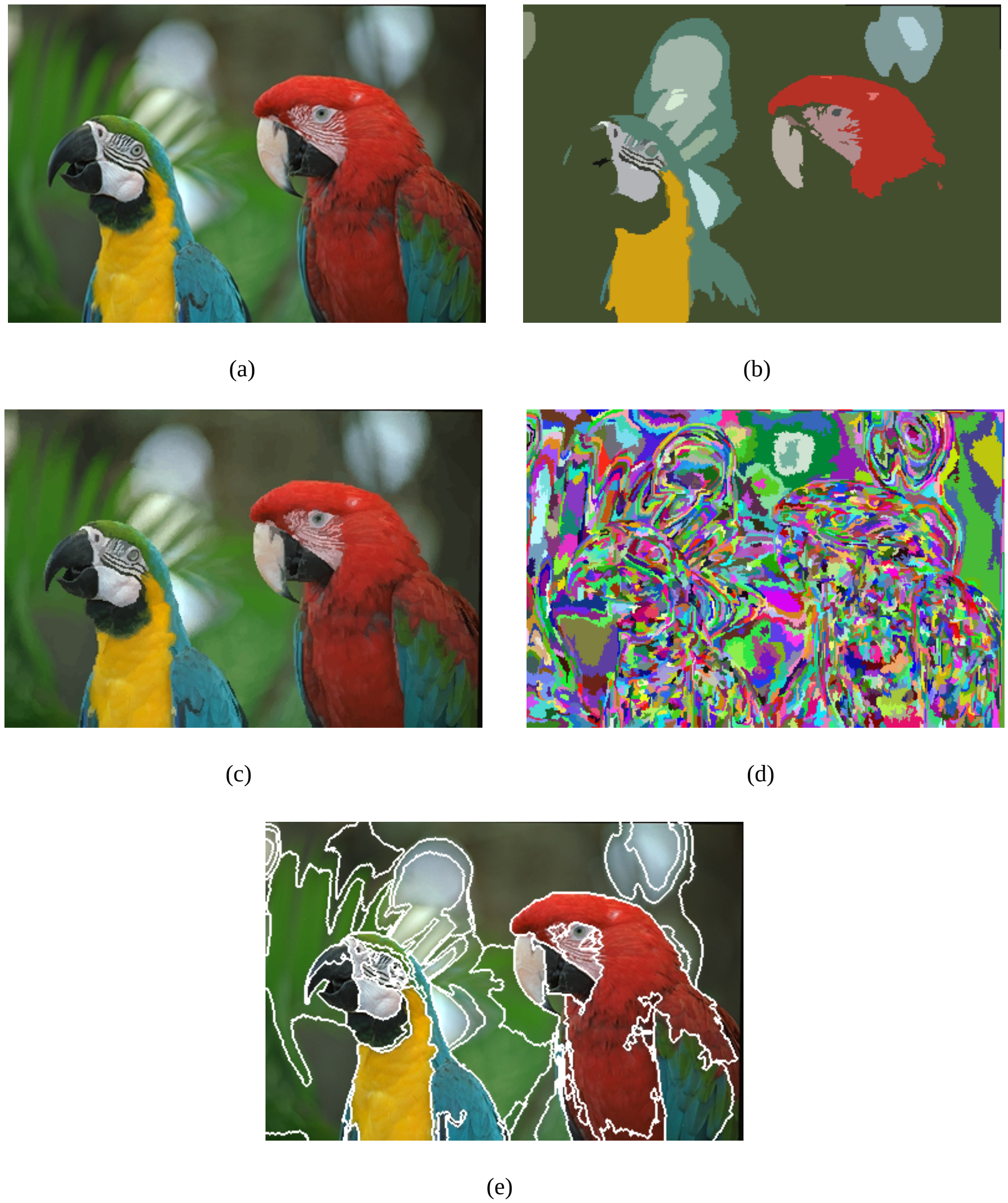


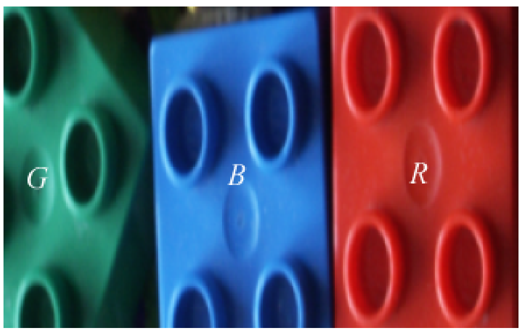
Figure 1.2: Figure 1.2: Parrots image and segmentation results: Fig1.2(a) original image, Fig1.2(b) segmentation into 48 regions (undersegmentation), Fig1.2(c) segmentation into 3000 regions (oversegmentation), Fig1.2(d) segmentation into 3000 regions presented in pseudocolors, Fig1.2(e) segmentation into 62 white bordered regions.

in such taxonomies the separated categories for techniques using special mathematical tools, for example: graph techniques, mathematical morphology, fuzzy techniques [32, 8] or techniques based on artificial neural networks [24, 8] are created.

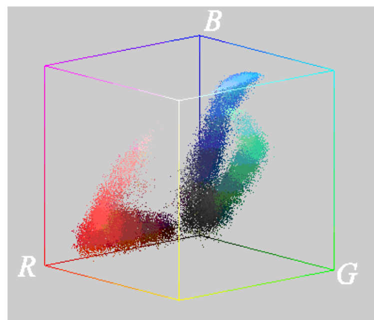
This chapter presents two classical image segmentation techniques: the k-means clustering and the region growing technique in application to the color images.

1.2 Clustering in the color space

Clustering is the process of partitioning a set of objects (pattern vectors) into subsets of similar objects called clusters. Pixel clustering in three-dimensional color space on the basis of their color similarity is one of popular approaches in the field of color image segmentation. Clustering is often seen as an unsupervised classification of pixels. Generally, the a priori knowledge about the image is not used during a clustering process. Colors, dominated in the image, create dense clusters in the color space in natural way. Fig1.3 shows three ”pixel clouds” in the RGB color space that represent clusters. Many different clustering techniques, proposed in the pattern recognition literature [21], can be applied to color image segmentation. One of the most popular and fastest clustering techniques is the k-means technique.



(a)



(b)

Figure 1.3: (a) Color image *Blocks1*, (b) its clusters formed in RGB color space.

The k-means technique has been proposed in the 1960s [25] and has been described in many pattern recognition handbooks e.g. [2]. The first step of this technique needs to determine a number of clusters $k$and to choose initial cluster centers C_i:

$$C_1, C_2, \dots, C_k \quad \text{where} \quad C_i = [R_i, G_i, B_i],\ i = 1, 2, \dots, k \tag{1.1}$$

The necessity of determination of input data is the drawback of k-means technique. During the clustering process each pixel x is allocated to cluster K_j with the closest cluster center using a predefined metric, for example the Euclidean metric, the City Block metric, the Mahalanobis metric etc. For pixel x, the condition of membership to the cluster K_j during the n-th iteration can be formulated following:

$$x \in K_j(n) \Longleftrightarrow \forall i = 1, 2, \dots, j-1, j+1, \dots, k \quad \|x - C_{j(n)}\| < \|x - C_{i(n)}\| \tag{1.2}$$

where: C_j is the center of cluster K_j.

The main idea of k-means is to change the positions of cluster centers so long as the sum of distances between all points of clusters and their centers will be minimal. For cluster K_j the minimization index J

can be defined as follows:

$$J_j = \sum_{x \in K_j(n)} \|x - C_j(n+1)\|^2 \tag{1.3}$$

After each allocation of pixels a new positions of cluster centers are computed as arithmetical means. Starting from (1.3) we can calculate color components of center of cluster K_j formed after $n+1$ iterations as arithmetical means of color components of pixels belonging to this cluster:

$$C_{jR}(n+1) = \frac{1}{N_j(n)} \sum_{x \in K_j(n)} x_R \tag{1.4}$$

$$C_{jG}(n+1) = \frac{1}{N_j(n)} \sum_{x \in K_j(n)} x_G \tag{1.5}$$

$$C_{jB}(n+1) = \frac{1}{N_j(n)} \sum_{x \in K_j(n)} x_B \tag{1.6}$$

where $N_j(n)$ means the number of pixels in cluster K_j after n iterations. Since this kind of averaging based on (1.4)–(1.6) is repeated for all k clusters, the clustering procedure can be named k-means technique.

In the next step a difference between new and old positions of centers is checked. If the difference is larger than some threshold δ, then the next iteration is starting and the distances from pixels to the new centers, pixels membership etc. are calculated. If the difference is smaller than δ, then the clustering process is stopped. The smaller is the value of δ, the larger is the number of iterations. This stop criterion can be calculated according formula:

$$\forall i = 1, 2, \ldots, k \quad \left\|C_{i(n+1)} - C_{i(n)}\right\| < \delta. \tag{1.7}$$

The stop criterion can be also realized by limiting the number of iterations. During the last step of k-means technique the color of each pixel is turned to the color of its cluster center. The number of colors in the segmented image is reduced to k colors. The k-means algorithm is converged, but it finds a local minimum only [41].

The results of segmentation by k-means depend on the position of initial cluster centers. In the case of semiautomated version of k-means the input data can be defined by human operator. In the case of automated version of k-means the initial centers can be choose randomly from all colors of the image. There exist also other possibilities for the choice of centers: k colors from the first pixels in the image, k gray levels from the gray line uniformly partitioned into k segments. Fig1.4 and Fig1.5 depicts the results obtained for color image Objects in individual iterations in image domain as well as in RGB color space. This image has been clustered into eight clusters and eight initial cluster centers have been located on gray level axis i.e. the diagonal of RGB cube.

The above presented results of segmentation have been obtained by k-means technique operated in the RGB color space. An image can be converted from RGB space to a new color space and then clustered in this color space. Other modifications of this clustering technique are realized by increasing the dimension of feature space through introducing additional features e.g. geometrical coordinates of the pixel in the image, gradient of color, texture etc. [42, 13].

The result of segmentation as well as the number of iterations is dependent on initial cluster centers.

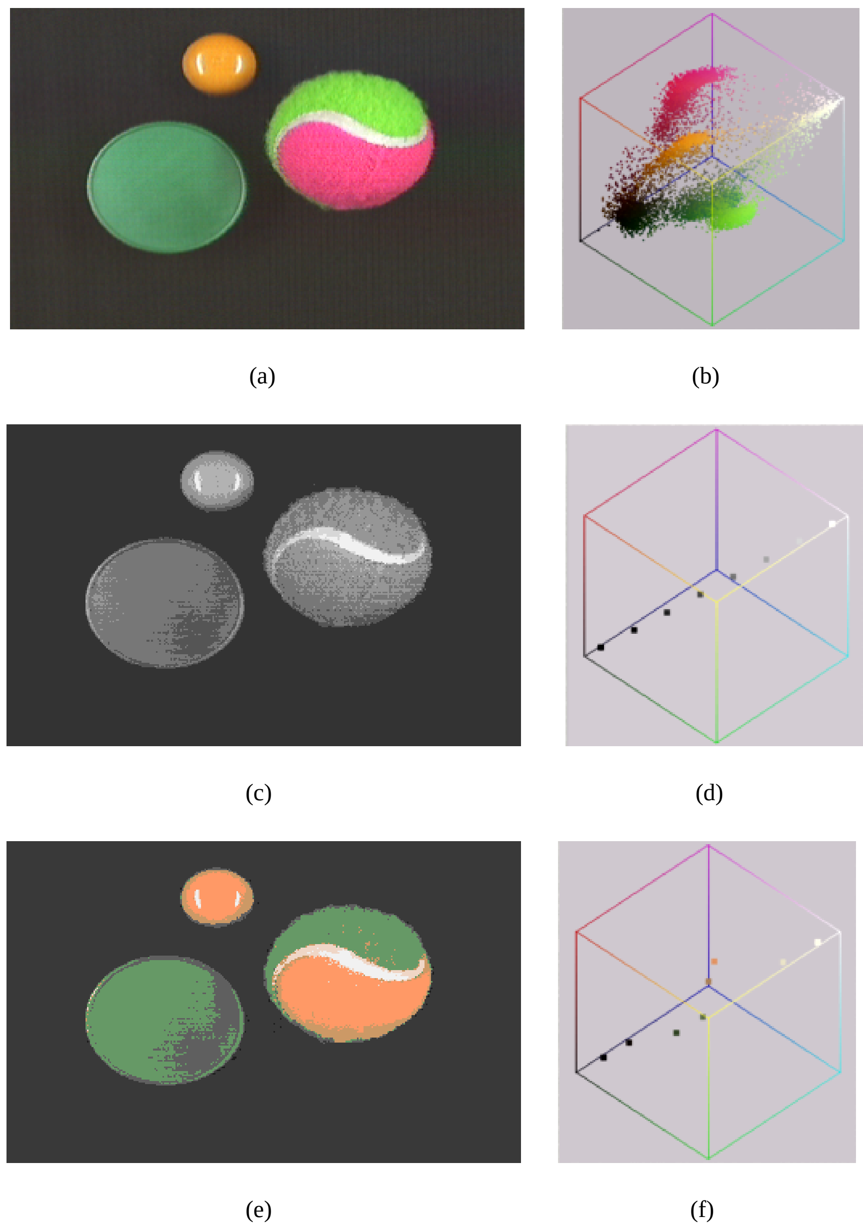
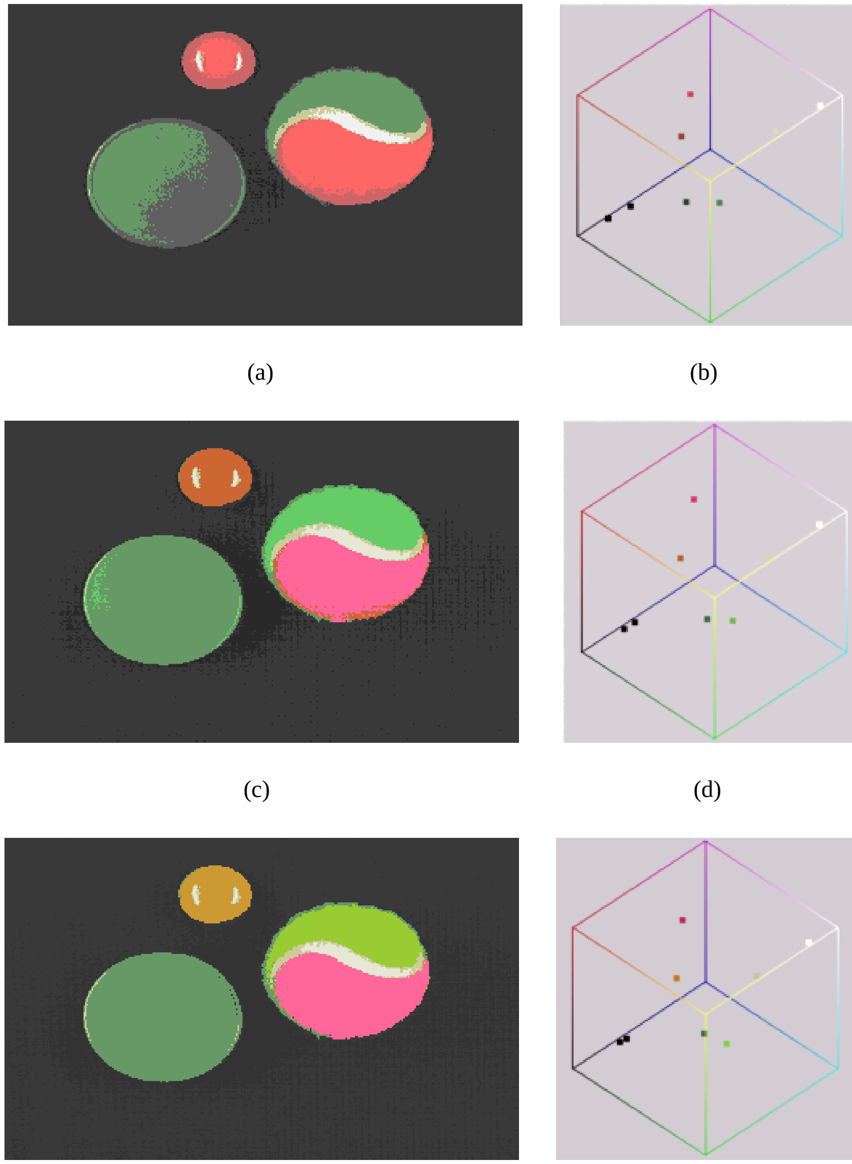


Figure 1.4: Illustrations of the development of the iterative clustering process: (a) color image Objects, (b) distribution of initial cluster centers, (c) segmented image after 1st iteration, (d) distribution of cluster centers after 1st iteration, (e) segmented image after 2nd iteration, (f) distribution of cluster centers after 2nd iteration.



(a) (b)

(c) (d)

(e) (f)

Figure 1.5: Continuation of Fig1.4: (a) segmented image after 3rd iteration, (b) distribution of cluster centers after 3rd iteration, (c) segmented image after 5th iteration, (d) distribution of cluster centers after 5th iteration, (e) segmented image after 7th iteration, (f) distribution of cluster centers after 7th iteration.

This dependence is presented in Fig1.6 Images in Fig1.6(b) and Fig1.6(c) have been obtained in the case of using random choice of 20 initial centers. The program have stopped, dependent on initial centers, after 26 iterations (Fig1.6(b)) and 18 iterations (Fig1.6(c)). The difference between both segmented images is observable: in the first segmented image exist two yellow regions, that do not exist in the second segmented image.

In the segmented image, the pixels that belong to one cluster can belong to many different regions. The larger is the number of clusters k, the image will be segmented into more regions. The processing of pixels

(a) (b) (c)

Figure 1.6: Segmentation of *Flowers1* image into 20 clusters, (a) original *Flowers1* image, (b) example segmented image (26iterations), (c) example segmented image (18 iterations).

without taking into consideration their neighborhoods is inherent to the nature of clustering techniques. It often results in sensitivity to noise and therefore there needs to be postprocessed (section 1.4) for the elimination of oversegmentation.

1.3 Region growing for color image

Region-based techniques group pixels into homogeneous regions. In this family of techniques we can find following techniques: region growing, region splitting, region merging and others. Particularly the region growing technique, proposed for grayscale images so long ago [4], is constantly popular in color image processing. In this section we will examine region growing technique.

The region growing is a typical bottom-up technique. Neighboring pixels are merged into regions, if their attributes, for example colors, are sufficiently similar. This similarity is often represented by a homogeneity criterion. If a pixel satisfied the homogeneity criterion, then the pixel can be included to the region and than the region attributes (a mean color, an area of region etc.) are updated. The region growing process, in its classical version, is starting from chosen pixels called seeds and is continued so long as all pixels will be assigned to regions. Each of these techniques varies in homogeneity criteria and methods of seeds location. The advantages of region growing techniques result from taking into consideration two important elements: the color similarity and the pixel proximity in the image.

If the color in the image is described by RGB components, then the homogeneity criterion based on the Euclidean metric has following form:

$$\sqrt{(R-R^*)^2+(G-G^*)^2+(B-B^*)^2} \leq d \tag{1.8}$$

where: R, G, B are color components of tested pixel, R^*, G^*, B^* are color components of mean color of creating region and d is the parameter, that is very important for segmentation results. The homogeneity criterion in RGB space can be formulated as a conjunction of few inequalities:

$$R^* - T_1^R \leq R \leq R^* - T_2^R \quad \wedge \quad G^* - T_1^G \leq G \leq G^* - T_2^G \quad \wedge \quad B^* - T_1^B \leq B \leq B^* - T_2^B \tag{1.9}$$

where: $T_1^R, \ldots, T_2^B$ — thresholds.

Similar growing process is sometimes realized in other color space, for example in HSI cylindrical color space [33]. This space better represents a human perception of colors than the RGB color space. In this case the formula (1.8) must be more complicated:

$$\sqrt{(I-I^*)^2+S^2+{S^*}^2-2SS^2\cos{(H-H^*)}} \leq d \tag{1.10}$$

where: H, S, I — hue, saturation and intensity of tested pixel, H^*, S^*, I^* — color components of mean color of creating region. The homogeneity criterion can be also based on variances of color components of the creating region. Such approach has been applied to the HSV space in the paper [10]:

$$\gamma = \frac{A_1}{\sigma_H^2} + \frac{A_2}{\sigma_S^2} + \frac{A_3}{\sigma_V^2} \tag{1.11}$$

where: A_1, A_2, A_3 — constants, σ_H^2, σ_S^2, σ_V^2, — the variances of hue, saturation and value in the region after pixel-candidate inclusion. In this case a pixel may be included in the region when the criterion's value γ will increase after this operation.

Sometimes in literature e.g. [15] a need for real color difference between the pixel-candidate and the mean color of creating region is stressed. It means necessity of using a perceptually uniformly color space [33], for example CIE $L^*a^*b^*$ space. In this case the homogeneity criterion has following form:

$$\sqrt{(L-L^*)^2+(a-a^*)^2+(b-b^*)^2} \leq d \tag{1.12}$$

where: L, a, b are color components of tested pixel, L^*, a^*, b^* are color components of mean color of creating region.

If tested pixel fulfills the homogeneity criterion, then the following recurrent formula can be used for updating of mean color of region. Here is the example version for intensity I only:

$$I_n^* = \frac{(n-1)I_{n-1}^* + I_{ij}}{n} \tag{1.13}$$

where: I_{ij} is the intensity of pixel with coordinates ij, I_{n-1}^* is the intensity of region with $(n-1)$-pixels and I_n^* is the intensity of region with n-pixels, after merging of tested pixel. Similar recurrent formulae can be derived for other region descriptors e.g. variance:

$$\sigma_n^2 = \frac{(n-1)(\sigma_{n-1}^2 + {I_{n-1}^*}^2) + I_{ij}^2}{n} - {I_n^*}^2 \tag{1.14}$$

where: σ_{n-1}^2 is the variance of intensity in the region with $n-1$ pixels, σ_n^2 is the variance of intensity in the region with n pixels, after pixel-candidate merging. It is necessary to know, that during the segmentation process the values of described above statistics of region are known only approximately, because not all pixels, members of region, are known.

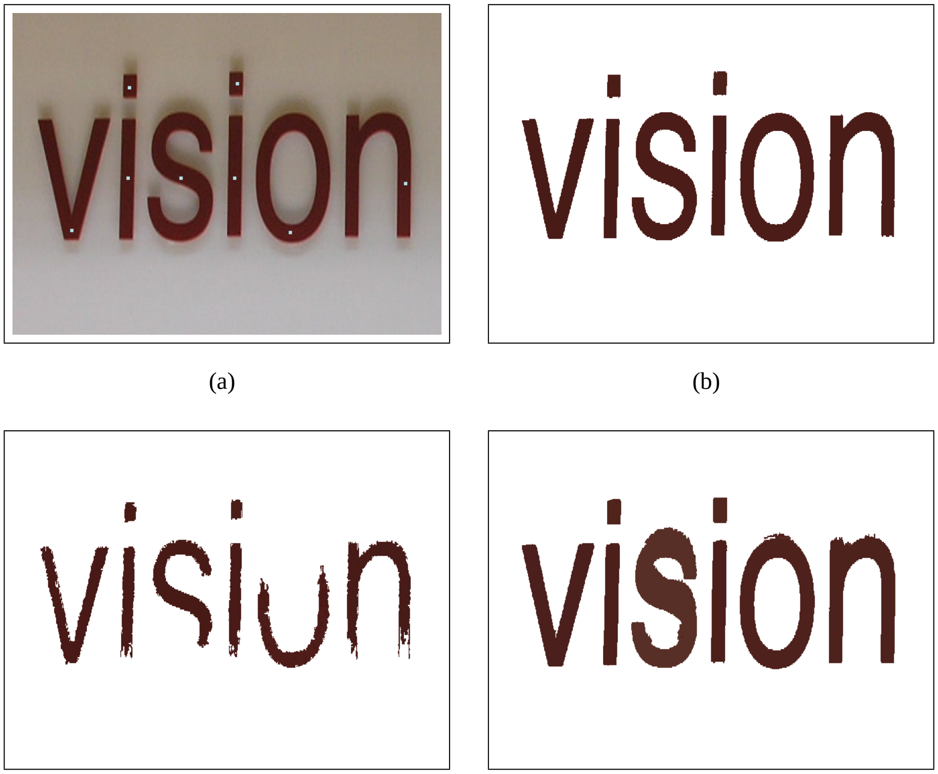
In the region growing process we can use a 4-connectivity or an 8-connectivity concept [42]. A pixel that satisfied requirements of 4-connectivity or 8-connectivity is merged into creating region, when its color fulfills the homogeneity criterion.

1.3.1 Seeded region growing

In 1990s two versions of seeded region growing (SRG) algorithm for grayscale images have been proposed [1, 28]. The seeds in this technique are often chosen in regions of interests (ROI). One good example is an image *Sign* shown in (Fig1.7) that contains six characters but simultaneously eight separate regions. For complete segmentation of the image *Sign* into N regions, one seed in each potential region should be chosen and a proper value of parameter d should be selected.

The distribution of seeds presents (Fig1.7(a)) and the result of segmentation by SRG technique in the RGB color space with color component values from the range $[0, 255]$ and the value of the parameter $d = 50$ is shown in Fig1.7(b). It should be pointed out that for each image a range of values of the parameter d exist, which is necessary for good segmentation. Using out-of-range values of the parameter d results in incomplete segmentation (Fig1.7(c)) or in character deformation (Fig1.7(d)). The background of segmented image has been removed for simplification purposes. The regions that grow around the seeds can be parallel segmented from the rest of the image. The SRG technique with an application of a manual location of seeds is working properly when the image contains a limited number of objects [51].

However, the SRG technique results depend on the position of seeds. In (Fig1.8) the images segmented using seeds placed in two different places are presented. (Fig1.8(d)) demonstrates that the position of seed is essential for segmentation results. A use of two seeds placed on one object i.e. a blue flower results in two



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

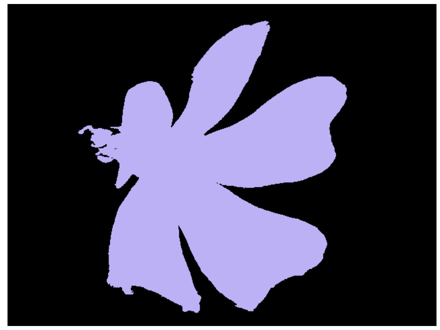
Figure 1.7: Seeded region growing for color image *Sign*: (a) original *Sign* image with the seeds, (b) segmented image — $d = 50$, (c) segmented image — $d = 10$, (d) segmented image — $d = 90$.

regions instead of one region (Fig1.8(e)). The postprocessing step is capable of merging both regions into one region. The influence of size of seed, from range 1 pixel to 20×20 pixels, on the results of segmentation has been tested. It is definitely smaller than the influence of seed position in the image. The difference image between the segmented images, that have been segmented based on the small and the large seeds, contains a few dozen of single pixels by the image resolution 640×480 pixels.

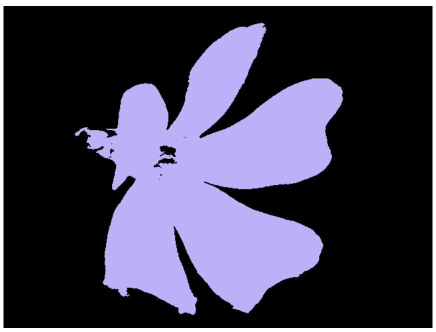
The seeds can be located in the image manually, randomly and automatically. An operator uses his knowledge about the image for location of seeds. Random choice of seeds is particularly risky in the case of noisy image, because a seed can be located on a noisy pixel. The seeds can be also found by the use of color histogram peaks. An additional edge information is sometimes also applied to the location of seeds inside of closed contours. Sinclair has proposed [43] a position of seeds on the peaks in the Voronoi image. His method needs first of all to find the edges in the original image. Binarized edge image is a base for generating the Voronoi image. The gray level in the Voronoi image is a function of a distance from the pixel to the closest edge. The larger is this distance, the brighter is the pixel in Voronoi image. Therefore the seeds are located in the brightest pixels of Voronoi image (Fig1.9).

Ouerhani et al. [31] have applied as natural candidates to fulfill the role of seeds the so-called attention

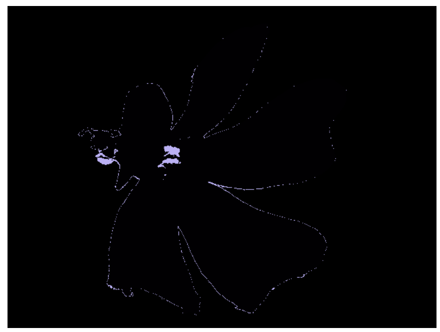
(a)



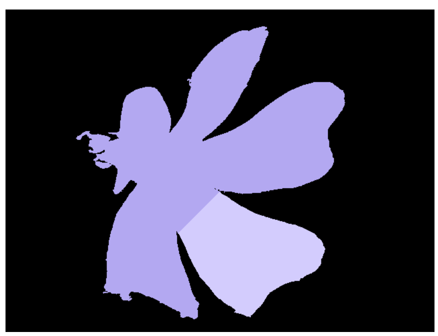
(b)



(c)



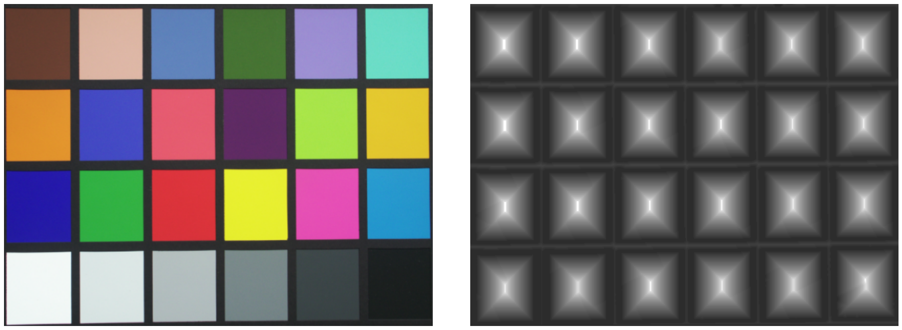
(d)



(e)

Figure 1.8: Seeded region growing for color image *Flowers2*, parameter $d = 150$: (a) original *Flowers2* image with marked 2 locations for one seed, (b) result of segmentation using seed in the middle of flower, (c) results of segmentation using seed on the petal of flower, (d) difference of images (b) and (c), (e) result of segmentation using two seeds located as in the image (a).

points coming from a visual attention model. In the paper of Ikonomakis et al. [20] the seeds for chromatic regions have been determined by checking the variance of hue in symmetrical masks. If this variance is smaller than some threshold, then a seed in the mask is located. In practical applications a knowledge about processed images is used for location of seeds. For example, a mobile robot equipped with a vision system should find all doors in the interior rooms. In images from robot's camera the seeds are automatically placed in some distance from the corners of doors, that have been found with the help of Harris method [5].

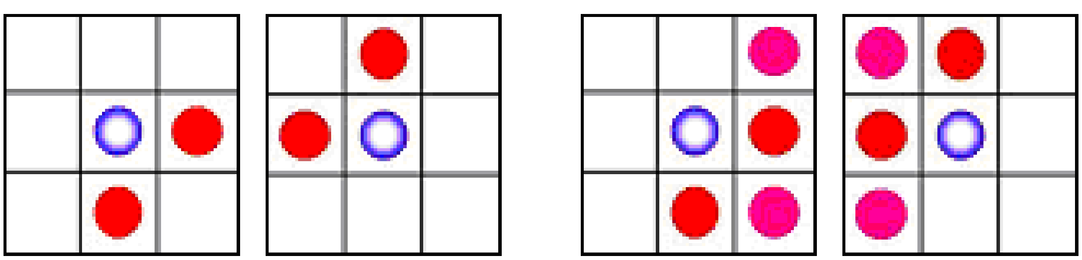


(a) (b)

Figure 1.9: (a) Color image *Chart*, (b) Voronoi image obtained from the image (a).

1.3.2 Unseeded region growing

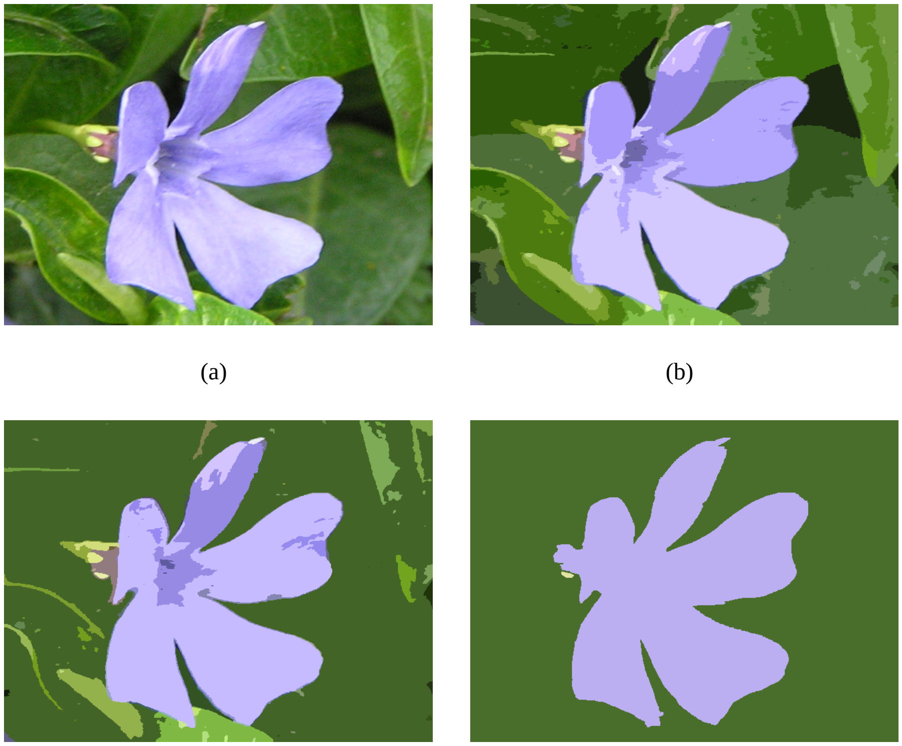
This technique can be also named as a region growing by pixel aggregation. It is based on the concept of region growing without seeds needed to start the segmentation process. At the beginning of the algorithm each pixel has its own label (one-pixel regions). A pixel is included into region, if it is 4-connected or 8-connected to this region and has a color value in the specified range from the mean color of an already constructed region. After each inclusion of pixel to the region, the region's mean color is updated. For this updating the recurrent formulae are used. One or two simple raster scans of the color image are applied: one pass from the left to the right and from the top to the bottom can be followed by an additional reverse pass over the image [34]. In the case of 4-connectivity four neighbors of tested pixel are checked during both scans. In the case of 8-connectivity two pixels are checked additionally during each scan (Fig1.10). The pixel aggregation process results in a set of regions characterized by their mean colors, their sizes and lists of pixels that belong to proper regions. The regions in this process are generated sequentially. The basic version of algorithm works in the RGB color space.



(a) (b)

Figure 1.10: Masks used in the segmentation process: (a) mask for 4-connectivity, (b) mask for 8-connectivity.

During tests, we observed that if the value of parameter d increases, then the number of regions R in the



(a) (b)

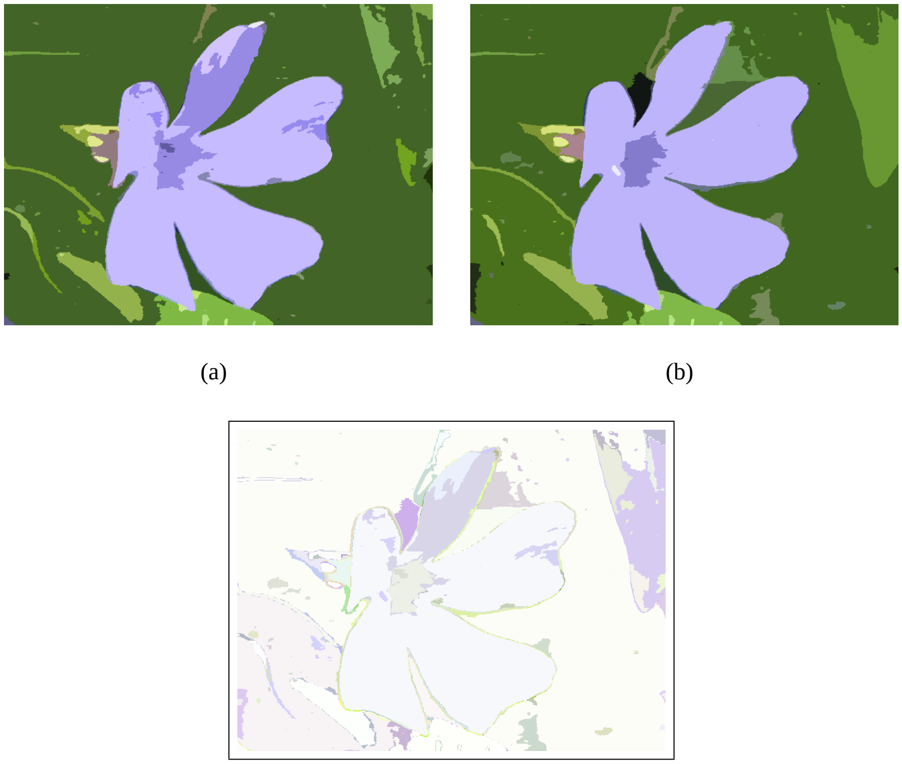
(c) (d)

Figure 1.11: Segmentation results of the color image *Flowers2* — different values of parameter d: (a) original *Flowers2* image, (b) segmented image $d = 40$ (2194 regions), (c) segmented image $d = 70$ (413 regions), (d) segmented image $d = 180$ (3 regions).

segmented image simultaneously decreases. The results for color image *Flowers2* are shown in Fig1.11. Too low value of parameter d is lead to oversegmentation and too high value is a reason of undersegmentation.

The described technique demonstrates a sensitivity to the direction of scanning process. Several images have also been segmented using a reverse direction i.e. from the bottom right pixel to the top left pixel. The numbers of regions obtained for images scanned in both direct and reverse directions differ slightly. For each pair of so segmented images can be generated a difference image. Fig1.12 shows the results of such experiment for the image *Flowers2*. The negative of difference image (Fig1.12(c)) is not a white image, what means that the result of proposed segmentation technique is dependent on the order of merging pixels into regions. The independence of segmentation results from the order of merging pixels needs further complication and parallelization of growing algorithm. The idea of such approach used for the grayscale images is described, as the ISRG algorithm, in the paper [28].

The influence of 4-connectivity and 8-connectvity on the segmentation results has been studied. Using the 8-connectivity, we have observed that the number of regions in the segmented image have decreased about a dozen or so percent, what results in increasing the mean color error. During tests we have found



(a) (b)

(c)

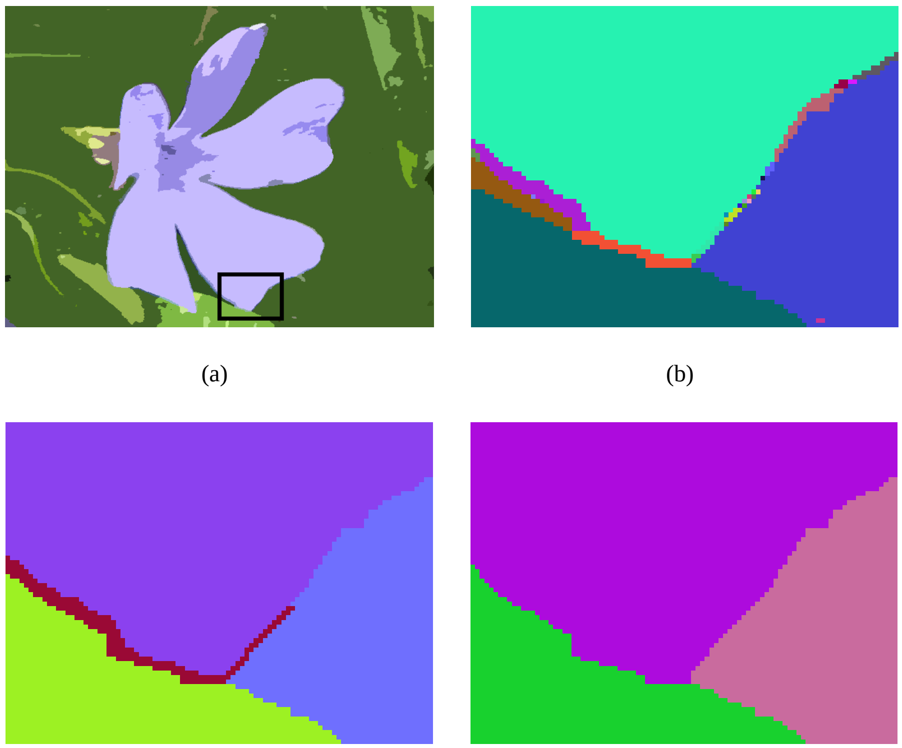
Figure 1.12: Segmentation results of the color image *Flowers2* - different scanning directions, parameter $d = 70$: (a) segmentation based on the direct scanning direction (413 regions), (b) segmentation based on the inverse scanning direction (411 regions), (c) negative image of difference image between (a) and (b).

that the use of 8-connectivity also extends the time of segmentation process about 20-30% in relation to the segmentation based on the 4-connectivity. The reason of this extension is the necessity to check the double number of pixels. Therefore sometimes the 4-connectivity version, for sake of speed, is chosen.

Region techniques are inherently sequential, hence the significance of the used order of pixel and region processing. We can easily notice that the SRG technique is very useful for cases where an image should be segment to small number of objects. On the other hand the unseeded region growing is suitable for application in the complete image segmentation. Information about edges obtained with the use of gradient can help in the control of the region growing process. Such approach has been proposed for grayscale images in the work [18]. Many hybrid techniques, where region and edge information complement each other in the image segmentation process have been developed [12]. The idea of region growing has been also applied in the watershed segmentation that for color images was first time used by Meyer [29].

1.4 Postprocessing

One of the reasons of oversementation can be the presence of noise contained in the image before segmentation. The segmentation of good quality images result also in a large number of small regions on the edges of objects. It is possible to remove these regions from the segmented image by postprocessing. Below is shown an application of postprocessing to the region growing based on pixel aggregation.



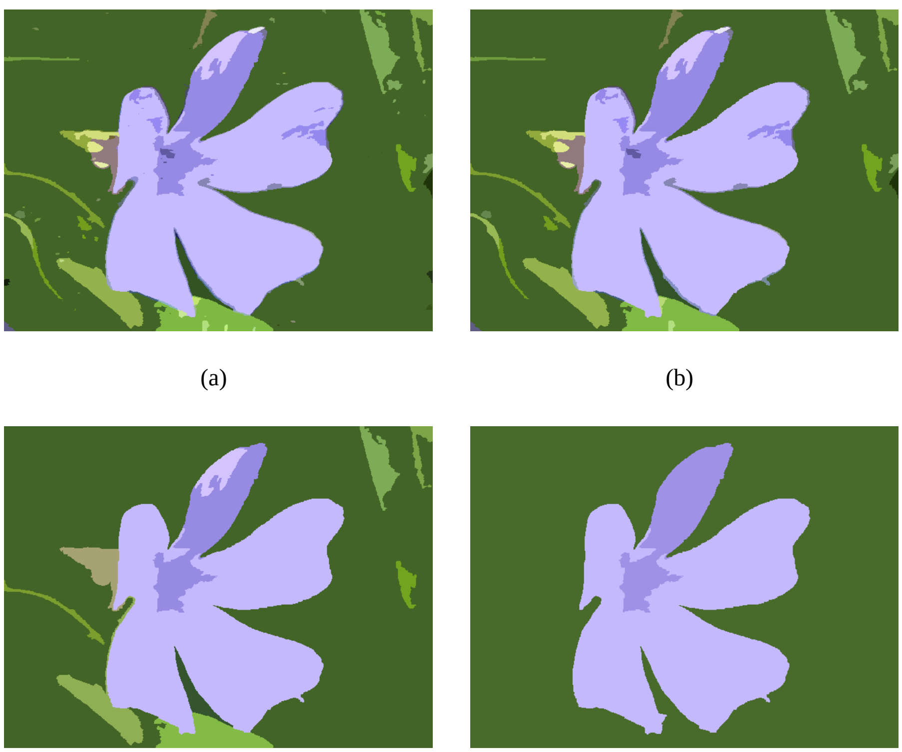
(a) (b) (c) (d)

Figure 1.13: Results of postprocessing step on the segmented image *Flowers2* (parameter $d = 70$): (a) segmented image with marked window, (b) enlarged and pseudocolorized contents of the window (26 regions), (c) result of postprocessing — parameter $A = 100$, (d) result of postprocessing — parameter $A = 1000$.

One of the most effective methods of postprocessing is to remove the small regions from the image and to merge them into the neighboring regions with most similar color. It is not difficult task, because after the region-based segmentation we have at our disposal a list of regions that can be sorted according to their area. A threshold value of the area of small region A depends on the image. In the one image the same threshold A allows to remove unnecessary artifacts e.g. highlights or noises and in the other image it removes necessary details. After merging of pixel the mean color of new region is computed and a label of pixel, which shows that the pixel lies in defined region, is changed. In result of such postprocessing, the number of regions in the segmented image significantly decreases.

In (Fig1.13(b) an enlarged part of the segmented image is presented. We can without problems see small regions on the edges of objects thanks to pseudocolors use. In our case there are regions on the edge of flower petal. (Fig1.13(b) contains 26 small regions. After application of postprocessing, which is based on removing regions with the area smaller than 100 pixels, the number of regions in this part of image has been decreased to 4 regions. Among these regions is a one small region located along the edges (Fig1.13(c). The expansion of definition of small region to the maximal value $A = 1000$ pixels reasons in removing all small regions and in leaving one region of petal and two regions of leaves (Fig1.13(d).

Below are presented the results of postprocessing obtained for the whole tested image *Flowers2* (Fig1.14. We can observe that the postprocessing procedure can be used not only for small regions removing, but after introducing a large value of the parameter A, also for merging of adjacent regions. In (Fig1.14(d) we can see, for the value of parameter $A = 10000$, an effect of merging of regions into one homogeneous region of the background.



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

Figure 1.14: Results of postprocessing step on the segmented image *Flowers2* (parameter $d = 70$): (a) segmented image (413 regions), (b) result of postprocessing — parameter $A = 100$ (47 regions), (c) result of postprocessing — parameter $A = 1000$ (12 regions), (d) result of postprocessing — parameter $A = 10000$ (3 regions).

Merging of small regions into large regions with similar color must not be founded on the defining of the threshold value for the area of small region. Sometimes it is assumed that, for example, 90% of all pixels in the image belong to the essential regions, next all regions are sorted according to the size, the large regions containing 90% pixels are chosen and remaining regions are treated as small regions [23]. Yagi et al. have proposed [48] to compare before merging the colors of regions with the use of distance in the HSV color space and a variable value of threshold that depends inversely proportional on the area of region. Besides the small area of region an additional indicators for the region merging can be proposed e.g. a low value of color variance of region, expressed by the trace of covariance matrix or a location of region near to the image border.

The color similarity of neighboring regions we can evaluate independently of their areas. As a measure of regions similarity is often applied a difference between their mean colors determined with the use of formulae (1.8)-(1.12) or more computationally complicated features:

- color histograms of regions [45] with the evaluation of color similarity through histogram intersection technique [22],
- formula dependent on mean color gradient calculated for pixels included in these regions [30],
- Fisher distance between adjacent regions for one color component [40, 50]:

$$FD_{12} = \frac{\sqrt{(n_1 + n_2)}|\hat{\mu_1} - \hat{\mu_2}|}{\sqrt{n_1 \hat{\sigma_1}^2 + n_2 \hat{\sigma_2}^2}} \tag{1.15}$$

  where: n_1, n_2, $\hat{\mu_1}$, $\hat{\mu_2}$, $\hat{\sigma_1}^2$, $\hat{\sigma_2}^2$ denoted the number of pixels, a sample mean and sample variance of color of the first and the second regions. As a final measure of color similarity between regions can be used a maximal value of Fisher distance chosen among the distances calculated for all three color components.

Since the first publications about the region growing [4] the attention was called to the length of common part of contours of neighboring regions. Often an additional condition for merging of neighboring regions is formulated: some minimal value of length of common part their contours should be exceeded [17]:

$$\frac{B_{ij}}{B_i + B_j} > T \tag{1.16}$$

where: B_i, B_j — the length of contour of regions R_i, R_j, B_{ij} — the length of common part of contours R_i, R_j, T - threshold e.g. 0,05.

Also the graph methods [16, 26], headed by a region adjacency graph (RAG) [46] and defined in many handbooks [37], are applied for determining of the relations between regions and fixing the order of regions in the merging process. The nodes of the RAG represent regions. An arc in the graph links two nodes that represent two adjacent regions. For each arc we can calculate a cost of region merging. The main idea of RAG-based region merging is removing of arcs with lowest cost and connecting the corresponding nodes.

The order of region merging has an influence on the final result of segmentation [36]. Methodical approach to operating on the list of regions with the use of metric from (1.12) as the color difference between regions has been described in the work [7]. Final goal of region merging is to receive the segmented image that contains the as large as possible homogeneous regions.

1.5 Shadows and highlights in the image segmentation process

In practical applications of color image processing, it is often important to have a segmentation technique that is robust to shadows and highlights in the image. This problem is presented below for the case of the region growing by pixel aggregation. The shadows in the image can be so large that during the postprocessing is not possible to remove them without simultaneous removing some other essential regions. We can see such situation in the color image *Blocks2* (Fig1.15(a)). The application of the Euclidean metric, defined in the RGB space, results in the segmented image with large regions of shadow (Fig1.15(b)). The selection of suitable color space and metric for the use in homogeneity criterion together with suitable parameter values (d, A) make possible to remove the shadow regions from segmented image. Such operation needs an early nonlinear transformation from the RGB color space to the HSI or the CIE $L^*a^*b^*$ color spaces. After omitting the luminance component in formulae (1.10) and (1.12), the technique can segment objects from their images (Fig1.15(c)), (Fig1.15(d)). We can avoid time-consuming color space transformations using an angle between color vectors in the RGB space instead of Euclidean metric [11].

Let us assume that a mean color of the creating region and the color of the tested pixel are represented by following vectors:

$$c_1 = [R_1, G_1, B_1], \qquad c_2 = [R_2, G_2, B_2] \tag{1.17}$$

An angle between color vectors c_1 and c_2, denoted by θ, based on the definition of the scalar product of vectors, can be written as:

$$\cos\theta = \frac{c_1^T \circ c_2}{\|c_1\|\|c_2\|} \tag{1.18}$$

Smaller the angle θ, closer is the pixel color to the mean color of region. Hence as a measure of color similarity can be use the sine function and on its base we can formulate following homogeneity criterion:

$$255 \cdot \sqrt{1 - \left(\frac{c_1^T \circ c_2}{\|c_1\|\|c_2\|}\right)^2} \leq d \tag{1.19}$$

The application of formula (1.19) in the region growing process results in robust image segmentation, what is shown in (Fig1.15(e)). The angular data like an angle between color vectors θ and a hue H need, due to its periodicity, a special methods of statistical analysis e.g. directional statistics [27]. The integration of shadows into the image background results in a growth of a bluish color in the background (Fig1.15(c)),(Fig1.15(d)), (Fig1.15(e)).

Highlights, produced by smooth surfaces of the objects in the scene and specular reflections may also impede the image segmentation process (Fig1.16(a)). In general the highlights occur in the image as separate regions with the colors similar to the color of light source e.g. white color (Fig1.16(b)). The highlight regions can be removed by the postprocesing (Fig1.16(c)). It is possible, because the size of the highlight is definitely smaller than the size of the object.

1.6 Quantitative evaluation of segmentation results

Nowadays there exist many color image segmentation techniques, but there are few results in the field of evaluation of segmented images. The reason is a lack of generally accepted criteria and a deficiency of uniform procedures for the evaluation of segmentation results. In the literature are presented many algorithms of segmentation that are tested on the small number of images.

The simplest kind of evaluation of segmented image is a subjective evaluation by the human expert or experts. Some researchers suppose that a human is the best judge in this evaluation process [32]. In some applications of image segmentation e.g. in an object recognition a recognition rate can serve as an indirect assessment of a segmentation algorithm independently of expert opinions about the segmented image. In the classical Zhang paper [49] the quantitative methods of evaluation of segmentation results have been grouped in two categories: analytical and experimental methods. The analytical methods are weakly developed, because does not exist the general image segmentation theory.

Among experimental methods two approaches predominate: empirical goodness approach and empirical discrepancy approach. The first approach does not need a reference image and the evaluation of segmented image is based on the original image. An example of goodness measure can be in the case of region-based segmentation homogeneity of region or a contrast between regions. In the second approach, a discrepancy measure expressed as a difference between segmented and a reference image is computed. The reference image (also called ground truth) is an image that has been manually segmented by the expert. Generation of the reference image is sometimes a difficult problem, because different people create different segmentations for the same image. Often the discrepancy measure is based on a number of mis-segmented pixels, a position of mis-segmented pixels etc.

In the field of data clustering own measures for the evaluation of results have been developed. A simple measure is an intra-cluster distance that can be computed as a sum of distances between the cluster points and the cluster center. This measure evaluates a compactness of clusters. A complementary measure to the intra-cluster distance is an inter-cluster distance that can be computed as a sum of distances between the centers of clusters. The value of inter-cluster distance estimates a separation of clusters.

R. Turi [47] have proposed a validity measure VM defined for clustering technique as a proportion of

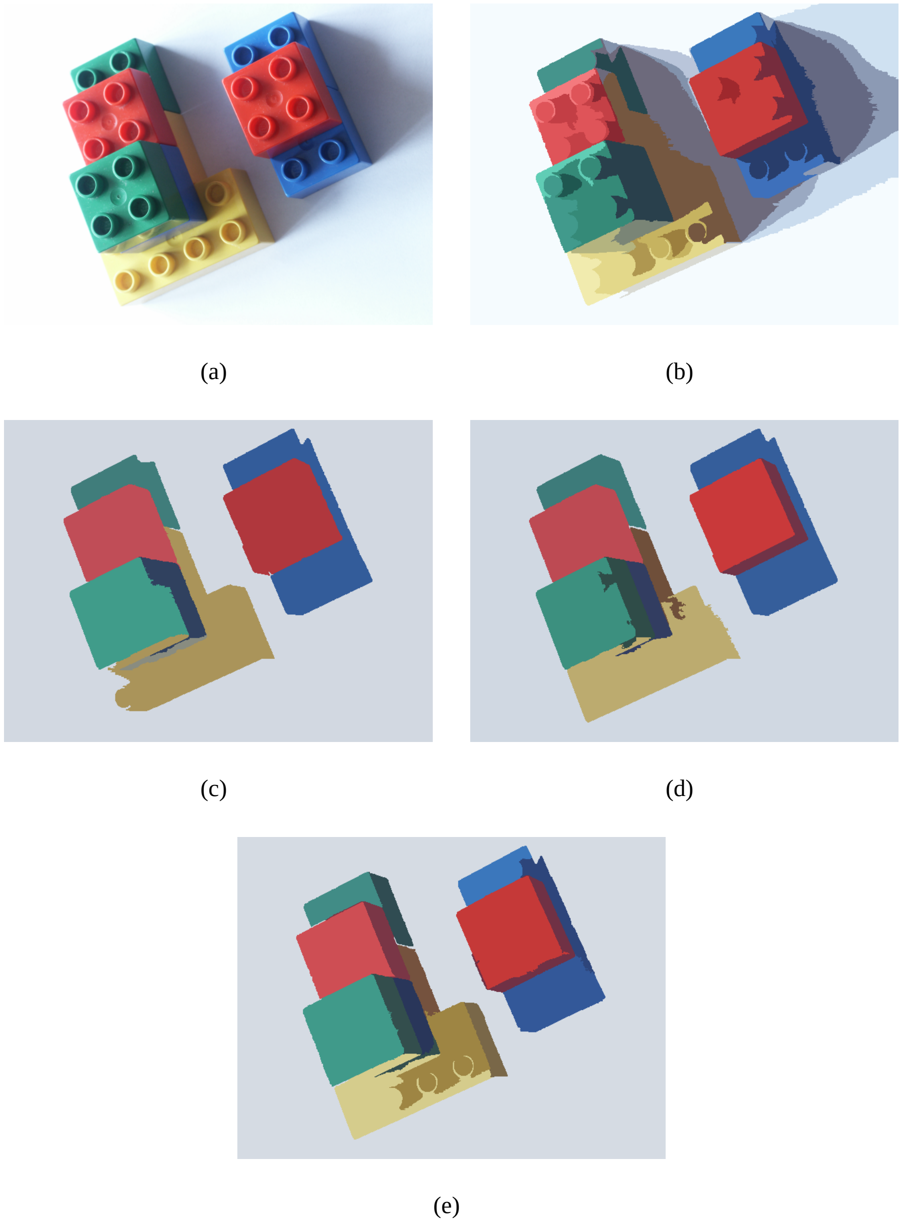
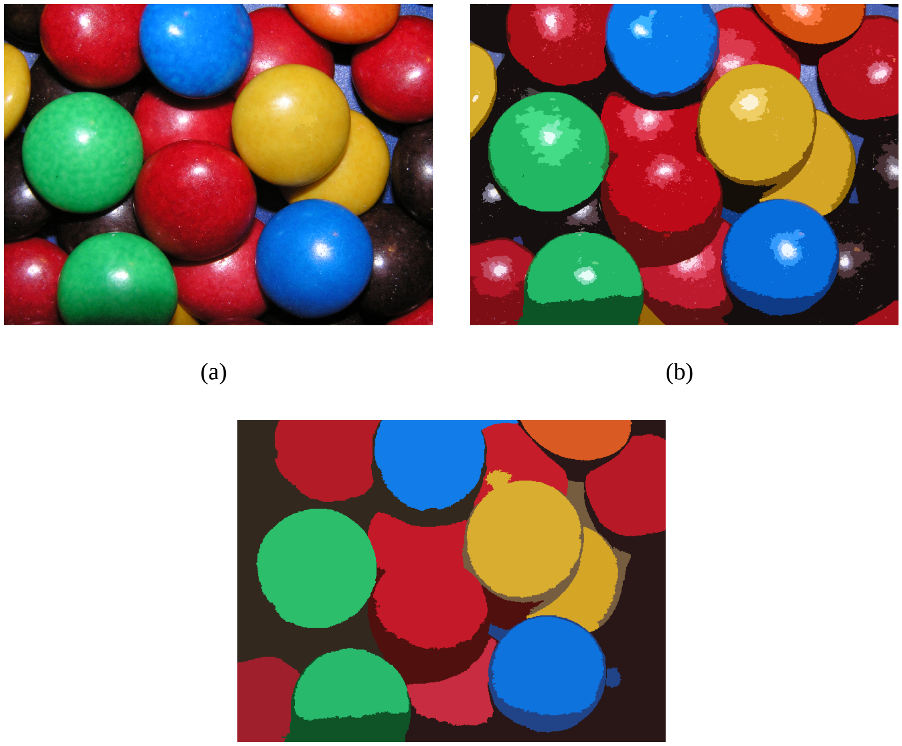


Figure 1.15: Segmentation results for color image *Blocks2*: Fig1.15(a) original *Blocks2* image, Fig1.15(b) result for Euclidean metric in the RGB space and parameters $d = 45$ and $A = 710$, Fig1.15(c) result for Euclidean metric on the HS plane and parameters $d = 50$ and $A = 1000$, Fig1.15(d) result for Euclidean metric on the a^*b^* plane and parameters $d = 12$ and $A = 1800$, Fig1.15(e) result for "angle between vectors" metric in the RGB space and parameters $d = 21$ and $A = 2000$.

two above described measures, that characterizing the clusters:

$$VM = y(k)\frac{intra}{inter} \tag{1.20}$$

where: $intra$ is a measure of cluster compactness, $inter$ is a measure of distance between clusters and



(a) (b)

(c)

Figure 1.16: Segmentation results for color image *Candies*: Fig1.16(a) original *Candies* image, Fig1.16(b) result for Euclidean metric in the RGB space and parameter $d = 60$, Fig1.16(c) result for Euclidean metric in the RGB space and parameters $d = 60$ and $A = 5000$.

$y(k)$ is a continuous function based on the Gauss function and is decreasing when the number of clusters is growing. This last function is equal to 1 for the numbers of clusters larger than 4 and then does not have an influence on the VM measure.

The main goal of the segmentation by clustering is the minimization of VM measure. It means we should minimize the distances between elements of clusters and their centers to create compact clusters. At the same time we should maximize the distances between clusters for the good separation of clusters. We can calculate the value of intra, used in the VM formula (1.20), with the help of following equation:

$$intra = \frac{1}{MN} \sum_{i=1}^{k} \sum_{x \in K_i} \|x - C_i\|^2 \tag{1.21}$$

where: $M \times N$ is the number of pixels in the image, k is the number of clusters, x is a pixel from the cluster K_i, C_i is the color of the center of cluster K_i.

The distance between the centers of clusters is defined as follows:

$$\forall i = 1, 2, \ldots, k-1, \quad \forall j = i+1, \ldots, k, \quad inter = \min(\|C_i - C_j\|^2) \tag{1.22}$$

The VM minimization can be used for the determination of optimal value of the number of clusters k.

In the paper [3] Borsotti et al. have proposed an empirical function $Q(I)$ designed for evaluation of segmentation results and checked for different clustering techniques:

$$Q(I) = \frac{\sqrt{R}}{10000(N \times M)} \cdot \sum_{i=1}^{R} \left[\frac{{e_i}^2}{1 + logA_i} + \left(\frac{R(A_i)}{A_i} \right)^2 \right] \tag{1.23}$$

(1.23) where: I is the segmented image, $M \times N$, the size of the image, R, the number of regions in the segmented image, A_i, the area of pixels of the i-th region, e_i the color error of the region i and $R(A_i)$ is the number of regions with the area equal to A_i. The color error in the RGB space is calculated as a sum of the Euclidean distances between color components of pixels of the region and components of an average color which is an attribute of this region in the segmented image. The color errors calculated in different color spaces are not comparable; hence their values are transformed back to the original RGB space.

The first term of equation (1.23) is a normalization factor; the second term penalizes results with too many regions (oversegmentation), while the third term penalizes the results with non-homogeneous regions. The last term is scaled by the area factor, because the color error is higher for large regions. The main idea of using $Q(I)$ function can be formulated as follows: the lower the value of $Q(I)$, the better the segmentation result. Usefulness of $Q(I)$ function for evaluation of unseeded region growing has been presented in the paper [35].

Both described quality indexes VM and $Q(I)$ allow to choose the values of parameters that enable to avoid the oversegmentation as well as the undersegmentation for some class of images. We have used the $Q(I)$ function for a comparison of segmentation results from clustering method (k-means) and from region-based method (unseeded region growing). During the comparison are used all images presented in this chapter with the exception of two images with shadows (*Sign*,*Blocks2*) that demand the use of another metric. For each tested image the values of segmentation parameters k and d, that minimize the function $Q(I)$, have been found. The below presented Tab1.1 contains the values of segmentation parameters and evaluation indexes $Q(I)$. We can see, that for each image the region-based technique gives smaller value of $Q(I)$, i.e. we have better segmentation results than in the case of k-means method.

1.7 Summary

In this work most of the attention has been concentrated on relatively simple image segmentation techniques (k-means clustering technique, region growing technique), that are a natural fit to incorporate into applications. The influences of location of initial cluster centers and the number of iterations on the segmentation results obtained from k-means technique have been investigated experimentally. Two versions of region growing technique i.e. semiautomated seeded version and automated unseeded version have been described in detail. It can be concluded that exist many possibilities of reasonable choice of seeds for seeded version.

Name	k	Q_{min}	d	Q_{min}
Flowers1	24	3953	34	3675
Parrots	19	2420	29	1019
Blocks1	24	188	25	144
Objects	8	456	81	266
Flowers2	17	3023	26	2747
Chart	25	4191	7	94
Candies	15	12740	40	3799

Table 1.1: Values of parameters and evaluation indexes

The directions used in the pixel aggregation process and an accepted concept of connectivity are important in the case of unseeded region growing. It was shown that a segmentation algorithm can be robust to shadows in the image when a special angular metric have been applied. Postprocessing step eliminates the oversegmentation caused by each of two segmentation techniques, removes the highlights and merges regions with similar colors. The problem of impact of chosen color space on the segmentation results has not been decided. During color image segmentation, a color space should be chosen that gives best results for solving task or images class, because does not exist one ideal color space. The segmentation results obtained using two above mentioned techniques have been compared on the base of the values of quantitative quality index $Q(I)$. This comparison has unambiguously pointed to region-based technique as a better segmentation technique.

All images, presented in this chapter, were good quality images. If the level of noise (impulsive noise, Gaussian etc.) in the image is higher, then the image needs to be filtered before being segmented into regions. Appropriate filter should smooth the image and at the same time preserve its edges. If a noisy unfiltered image is segmented, then the number of regions in the segmented image is significantly increasing, the mean color error is increasing and simultaneously the value of quality function $Q(I)$ is also significantly increasing.

Finally, it should be noted that the universal technique for color image segmentation probably does not exist. The main goals of segmentation clearly depend on the art of solving problem, for which this segmentation process is one step only. The growing computational power causes that increase also the possibilities of applications of more complicated segmentation techniques than the techniques presented in this chapter. It concerns especially the hybrid methods, the methods with use of color texture and other methods, which are important for future development of color image processing.

Bibliography

[1] R. Adams, L. Bischof, ”Seeded region growing”, IEEE Transactions on Pattern Analysis Machine Intelligence., vol.16, no.6, pp.641-647, 1994.

[2] M. Anderberg, ”Cluster Analysis for Applications”, Academic Press, New York, 1973.

[3] M. Borsotti, P. Campadelli, R. Schettini, ”Quantitative evaluation of color image segmentation results”, Pattern Recognition Letters, vol.19, no.8, pp.741-747, 1998.

[4] C.R. Brice, C.L. Fennema, ”Scene analysis using regions”, Artificial Intelligence, vol.1, no.3, pp.205-226, 1970.

[5] J. Chamorro-Martnez, D. Snchez, B. Prados-Suarez, ”A fuzzy color image segmentation applied to robot vision”, In: J.M. Benitez, O. Cordon, F. Hoffman, R. Roy (eds.), Advances in Soft Computing, Engineering, Design and Manufacturing, pp.129-138, Springer-Verlag, Berlin 2003.

[6] T.Q. Chen, Y. Lu, ”Color image segmentation - an innovative approach”, Pattern Recognition, vol. 35, no.2, pp.395-405, 2002.

[7] H.-D. Cheng, ”A hierarchical approach to color image segmentation using homogeneity”, IEEE Transactions on Image Processing, vol.9, no.12, pp.2071-2082, 2000.

[8] H.D. Cheng, X.H. Jiang, Y. Sun, J. Wang, ”Color image segmentation: advances and prospects”, Pattern Recognition, vol.34, no.12, pp.2259-2281, 2001.

[9] Y.N. Deng, B.S. Manjunath, ”Unsupervised segmentation of color-texture regions in images and video”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.23, no.8, pp.800 - 810, 2001.

[10] A. De Rosa, A.M. Bonacchi, V. Cappellini, M. Barni, ”Image segmentation and region filling for virtual restoration of art-works”, Proceedings of IEEE Int. Conference on Image Processing (ICIP’01), vol. 1. pp.562-565, Thessaloniki, Greece, 2001.

[11] R.D. Dony, S. Wesolkowski, ”Edge detection on color images using RGB vector angle”, Proc. IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), pp.687-692, Edmonton, Canada, 1999.

[12] J. Fan, D. Yau, A. Elmagarmid, W. Aref, ”Automatic image segmentation by integrating color-edge extraction and seeded region growing, IEEE Transactions on Image Processing, vol.10, no.10, pp.1454-1466, 2001.

[13] D.A. Forsyth, J. Ponce, ”Computer Vision: A Modern Approach”, Prentice Hall, Upper Saddle River, NJ, 2003.

[14] J. Freixenet, X. Munot, D. Raba, J. Marti, X. Cuti, ”Yet another survey on image segmentation: region and boundary information integration”, In: A. Heyden et al. (Eds.), ECCV 2002, pp.408-422, Springer-Verlag, Berlin, 2002.

[15] H. Gao, W.-C. Siu, C.-H. Hou, ”Improved techniques for automatic image segmentation”, IEEE Transactions on Circuits and Systems for Video Technology, vol.11, no.12, pp.1273-1280, December 2001

[16] C. Garcia, G. Tziritas, ”Face detection using quantized skin color regions merging and wavelet packet analysis”, IEEE Transactions on Multimedia, vol.1, no.3, pp.264-277, 1999.

[17] I. Grinias, Y. Mavrikakis, G. Tziritas, ”Region growing colour image segmentation applied to face detection”, Proc. of International Workshop Low Level Video Coding, Athens, Greece, 2001.

[18] S.A. Hojjatoleslami, J. Kittler, ”Region growing: A new approach”, IEEE Transactions on Image Processing, vol. 7, no.7, pp.1079-1084, 1998.

[19] N. Ikonomakis, K.N. Plataniotis, A.N. Venetsanopoulos, ”Color image segmentation for multimedia applications”, Journal of Intelligent and Robotic Systems, vol.28, vol.1-2, pp.5-20, 2000.

[20] N. Ikonomakis, K.N. Plataniotis, A.N. Venetsanopoulos, ”Unsupervised ”, Proceedings of IEEE Int. Conference on Image Processing (ICIP’00), vol.1, pp.537-540, Vancouver, BC, Canada, 2000.

[21] A.K. Jain, R.C. Dubes, ”Algorithms for Clustering Data”, Prentice Hall, Englewood Cliffs, N.J, 1988.

[22] X. Jie, S. Peng-fei, ”Natural color image segmentation”, Proc. of IEEE Int. Conf. on Image Processing, (ICIP’03), vol.1, pp.973-976, Barcelona 2003.

[23] M. Li, I.K. Sethi, D. Li, N. Dimitrova, ”Region growing using online learning”, Proceedings of International Conference on Imaging Science, Systems, and Technology, (CISST’03), Las Vegas (Nevada), vol.1, pp.73-76, 2003.

[24] L. Lucchese, S.K. Mitra, ”Advances in color image segmentation”, Proceedings of Globecom’99, vol.4, pp.2038-2044, Rio de Janeiro 1999.

[25] J. Mac Queen, ”Some methods for classification and analysis of multivariate observations”, In: L.M. Le Cam, J. Neyman, (Eds.), Proceedings of the Fifth Berkeley Symposium on Mathematics, Statistics, and Probabilities, vol.1, pp.281-297, Berkeley and Los Angeles, University of California Press, 1967.

[26] S. Makrogiannis, G. Economou, S. Fotopoulos, ”A graph theory approach for automatic segmentation of color images”, Proc. of International Workshop Low Level Video Coding, pp.162-166, Athens, Greece, 2001.

[27] K.V. Mardia, P.E. Jupp, ”Directional statistics”, John Wiley & Sons, Chichester, 2000.

[28] A. Mehnert, P. Jackway, ”An improved seeded region growing algorithm”, Pattern Rec. Letters, vol.18, no.10, pp.1065-1071, 1997.

[29] F. Meyer, ”Color image segmentation”, Proc. IEE Int. Conf. Image Processing and its Applications, pp.303-306, The Netherlands, 1992.

[30] E. Navon, O. Miller, A. Averbuch, ”Color image segmentation based on adaptive local thresholds”, Image and Vision Computing, vol.23, no.1, pp.69-85, 2005.

[31] N. Ouerhani, N. Archip, H. Hgli, P.J. Erard, ”Visual attention guided seed selection for color image segmentation”, Proceedings of the 9th International Conference on Computer Analysis of Images and Patterns (CAIP’01), pp.630-637, Warsaw, 2001.

[32] N.R. Pal, S.P.Pal, ”A review on image segmentation techniques”, Pattern Recognition, vol.26, no.9, pp.1277-1293, 1993.

[33] H. Palus, ”Representations of colour images in different colour spaces”, in: Sangwine S.J., Horne R.E.N., /eds/, ”The Colour Image Processing Handbook”, Chapman and Hall, pp.67-90, London 1998.

[34] H. Palus, D. Bereska, ”Region-based colour image segmentation”, Proc. of 5th Workshop ”Farbbildverarbeitung”, pp.67-74, Ilmenau, Germany 1999.

[35] H. Palus, ”Region-based colour image segmentation: control parameters and evaluation functions”, Proceedings of the First European Conference on Color in Graphics, Imaging and Vision, (CGIV’02), pp.259-262, Poitiers, 2002.

[36] H.S. Park, J.B. Ra, ”Homogeneous region merging approach for image segmentation preserving semantic object contours”, Proc. International Workshop on Very Low Bitrate Video Coding, pp.149-152, Chicago, IL,1998.

[37] T. Pavlidis, ”Structural Pattern Recognition”, Springer, New York 1977.

[38] K.N. Plataniotis, A.N. Venetsanopoulos, ”Color Image Processing and Applications”, Springer-Verlag, Berlin 2000.

[39] S.J. Sangwine, R.E.N. Horne, Eds., ”The Colour Image Processing Handbook”, Chapman and Hall, London 1998.

[40] R.Schettini, ”A segmentation algorithm for color images”, Pattern Recognition Letters, vol.14, no.6, pp.499-506, 1993.

[41] S. Z. Selim, M. A. Ismail, ”K-means-type algorithms”, IEEE Transactions on Pattern Analysis and Machine. Intelligence, vol.6, no.1, pp. 81-87, 1984.

[42] L.G. Shapiro, G.C. Stockman, ”Computer Vision”, Prentice Hall, Upper Saddle River, NJ, 2003.

[43] D. Sinclair, ”Voronoi seeded colour image segmentation”, Technical Report TR99-4, AT&T Laboratories, Cambridge 1999.

[44] W. Skarbek, A. Koschan, ”Colour image segmentation - a survey”, Technical Report 94-32, Tech. Univ. of Berlin, October 1994.

[45] M.J. Swain, D.H. Ballard, ”Color indexing”, International Journal of Computer Vision, vol.7, no.1, pp.11-32, 1991.

[46] A. Trmeau, P. Colantoni, ”Region adjacency graph applied to color image segmentation”, IEEE Transactions on Image Processing, vol.9, no.4, pp.735-744, 2000.

[47] R.H. Turi, S. Ray, ”An application of clustering in colour image segmentation”, Proceedings of 6th International Conference on Control, Automation, Robotics and Vision (ICARCV’00), Singapore, 2000 (CD-ROM).

[48] D. Yagi, K. Abe, H. Nakatani, Segmentation of color aerial photographs using HSV color models, Proc. IAPR Workshop on Machine Vision Applications, (MVA’92), pp.367-370, Tokyo, Japan, 1992.

[49] Y.J. Zhang, ”A survey on evaluation methods for image segmentation”, Pattern Recognition, vol.29, no.8, pp.1335-1346, 1996.

[50] S.C. Zhu, A. Yuille, ”Region competition: unifying snakes, region growing and Bayes/MDL for multiband image segmentation”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.18, no.9, pp.884-900, 1996.

[51] S.W. Zucker, ”Region growing : childhood and adolescence”, Computer Graphics and Image Processing, vol.5, no.3, pp.382-399, 1976.