MULTI-BAND SEGMENTATION USING MORPHOLOGICAL CLUSTERING AND FUSION — APPLICATION TO COLOR IMAGE SEGMENTATION

H. Xue[*]

Inner Mongolia Agricultural University
College of Computer and Information Engineering
306 zhao wu da road, Hohhot 010018
China

T. Géraud[†], *A. Duret-Lutz*

EPITA Research and Development Laboratory
14-16 Rue Voltaire, F-94276 Le Kremlin-Bicêtre
France

ABSTRACT

In this paper we propose a novel approach for color image segmentation. Our approach is based on segmentation of subsets of bands using mathematical morphology followed by the fusion of the resulting segmentation "channels". For color images the band subsets are chosen as RG, RB and GB pairs, whose 2D histograms are processed as projections of a 3D histogram. The segmentations in 2D color spaces are obtained using the watershed algorithm. These 2D segmentations are then combined to obtain a final result using a region split-and-merge process. The CIE $L^*a^*b^*$ color space is used to measure the color distance. Our approach results in improved performance and can be generalized for multi-band segmentation of images such as multi-spectral satellite images.

1. INTRODUCTION

Color image segmentation is basically a 3D image histogram clustering, since picture regions of similar colors tend to form clusters of points within the 3D space. Many methods for color image segmentation on 3D histograms have been developed so far. The method we present is partly based on the automatic morphological clustering method in the 3D color space [1] whose algorithm is briefly described as follows:

1. Compute a 3D histogram $H^{(0)}(c$ from an input RGB image I.

2. Magnify the smallest clusters w.r.t. bigger ones and invert the result. $H^{(1)}(c) = M - log(1 + H^{(0)}(c))$, where $M = \max_c log(1 + H^{(0)}(c))$ and c denotes a color.

3. Smooth/regularize $H^{(1)}$ using an isotropic Gaussian filter whose variance is σ.

   $H^{(2)}(c) = Gauss(H^{(1)}(c), \sigma)$.

4. Remove small local minima of $H^{(2)}$ with a morphological closing.

   $H^{(3)}(c) = Closing(H^{(2)}(c), r)$, where r is the radius of a spherical structuring element.

   At that point, a color cluster is a region of $H^{(2)}$ surrounding a minimum.

5. Label regions (get a classification) of the 3D space with the connected watershed transform.

   $H^{(4)}(c) = Watershed(H^{(3)}(c))$.

6. Get a segmentation S of I using color space labeling information.

Clustering a 3D histogram can be expensive because of the huge amount of data involved. Every step processes 3D images ($H^{(*)}$s) whose size is $(2^8)^3$ voxels, if color quantization is 8 bit per component. This method was thus memory consuming and computationally intensive. One remedy is to project the 3D color space into a lower dimensional space such as 2D [2] or even 1D space [3]. In this paper, we propose a novel clustering approach, yet based on mathematical morphology [4, 5, 6, 7, 1] (a discussed state of the art of morphological classification can be found in this last reference), but only relying on 2D spaces.

This approach consists in two main processes: first, several morphological classifications are performed on 2D histograms, which lead to several segmentation results (section 2). Then, a fusion of these segmentation results is performed (section 3) to get a final segmentation. Results are presented and discussed in section 4, and we conclude in section 5.

2. SEGMENTATION IN 2D COLOR SPACES

The first step of the method we propose here consists in applying the morphological segmentation process recalled

*This work was supported by a grant of the Chinese Government.

†Corresponding author. E-mail: thierry.geraud@lrde.epita.fr

above but in a slightly different manner. Instead of computing a 3D histogram in the 3D RGB space, we compute three 2D histograms, respectively in the 2D color spaces RG, RB, and GB. As a result we now have to handle $3 \times (2^8)^2$ pixels (if quantization is 8 bit per component) instead of the $(2^8)^3$ voxels of the original 3D process.

For instance, $H_{RG}^{(0)}$ is a 2D histogram of input image I which is only based on the red and green components. Magnification of smallest clusters, inversion, smoothing, closing and applying the watershed transform (steps 1 to 5 described above) are then processed as 2D images. These steps do not take into account information from the blue component. Conversely, step 6 reintroduces this part of information. This segmentation step is performed as follows. First, for each class of $H_{RG}^{(4)}$, we compute its center in 3D space. With p being a point of I, let us denote $I(p) = (r(p), g(p), b(p))$ and $l_{RG}(p) = H_{RG}^{(4)}(r(p), g(p))$, which is the label corresponding to p in the RG plane classification. The color center of the class l of $H_{RG}^{(4)}$ is given by:

$$c_{RG}^{l} = \frac{\displaystyle\sum_{p,\ l_{RG}(p)=l} I(p)}{\displaystyle\sum_{p,\ l_{RG}(p)=l} 1}.$$

It is thus a "true" 3D color. The resulting segmentation S_{RG} is then obtained as follows:

$$\forall p, \quad S_{RG}(p) = c_{RG}^{l_{RG}(p)},$$

that is, $S_{RG}(p)$ is set to the center color of the RG class that corresponds to p.

From the three 2D histograms, $H_{RG}^{(0)}$, $H_{RB}^{(0)}$, and $H_{GB}^{(0)}$, we output three segmented images, S_{RG}, S_{RB}, and S_{GB}. Figure 1 shows the segmentation results obtained for an image of Kandinsky's painting "*Composition X*" (only a small detail of the painting is depicted). We have chosen this painting because getting a good color classification is far from being easy: there are a lot of classes and most of them are close to each other.

It is worth (although obvious) to note that segmenting (clustering) a 2D projection of a 3D image will yield less classes than the clustering in 3D space: classes that are observable in 3D may overlap once projected in 2D. None of these three 2D segmentations is good enough to be the final resulting segmentation of the original color image. So a fusion process of segmentation results is required.

(a) Input image I. (b) S_{RG}; 30 classes.

(c) S_{RB}; 35 classes. (d) S_{GB}; 27 classes.

Fig. 1. Segmentation results based on 2D color spaces.

3. FUSION OF SEGMENTATION RESULTS

3.1. Fuzzy Matching Degree Between Two Classes

Let us symbolize by T a 2D projection (for instance, T can be RG, RB, or GB). l being a class label, we denote by $S_T^{(l)}$ the set of points assigned to class l in the segmentation S_T; in other words:

$$S_T^{(l)} = \left\{ p \mid S_T(p) = c_T^{l_T(p)} \right\}.$$

Now consider a couple of segmentation results obtained from two different projections T_1 and T_2, and a couple of resulting classes labeled l_1 and l_2 respectively from the process using T_1 and T_2. A partial similarity degree between both corresponding point sets is defined as:

$$\mu_{T_1 \to T_2}^{(l_1,l_2)} = \frac{card(\, S_{T_1}^{(l_1)} \cap S_{T_2}^{(l_2)} \,)}{card(\, S_{T_1}^{(l_1)} \,)}.$$

Obviously, the partial similarity degree has the following properties:

$$\forall l_1\ \forall l_2 \quad 0 \le \mu_{T_1 \to T_2}^{(l_1,l_2)} \le 1$$

$$\forall l_1 \quad \sum_{l_2} \mu_{T_1 \to T_2}^{(l_1,l_2)} = 1.$$

Here, two families of fuzzy sets, $\mu_{T_1 \to T_2}$ and $\mu_{T_2 \to T_1}$, are defined in order to estimate the degree of similarity between regions of S_{T_1} and S_{T_2}. Finally, to get a symmetrical matching degree, we set:

$$m_{T_1,T_2}^{(l_1,l_2)} = \mu_{T_1 \to T_2}^{(l_1,l_2)} \oplus \mu_{T_2 \to T_1}^{(l_2,l_1)},$$

where $\oplus$ can be any fuzzy T-norm operator. For our experiments, we have chosen the very simple operator $\max$.

Given a criterion m_{match}, with $0 < m_{match} < 1$, if $m_{T_1,T_2}^{(l_1,l_2)} > m_{match}$ we state that regions $S_{T_1}^{(l_1)}$ and $S_{T_2}^{(l_2)}$ match. Practically, we have set m_{match} to $0,8$.

3.2. Region Splitting

Projections can be ordered with respect to correlation between components. In the case of RGB images, it is well know that RG is the less correlated couple of components, and then come the couples RB and GB. The splitting process is iterative and follows such ordering:

for each T_i (taken in the given order)
    for each region l of $S_{T_1+..+T_{i-1}}$
        for each region l' of S_{T_i}
            if $m_{T_1+..+T_{i-1},T_i}^{(l,l')} > m_{match}$
                then region l is kept *as is* in $S_{T_1+..+T_i}$
                otherwise split region l in $S_{T_1+..+T_i}$.

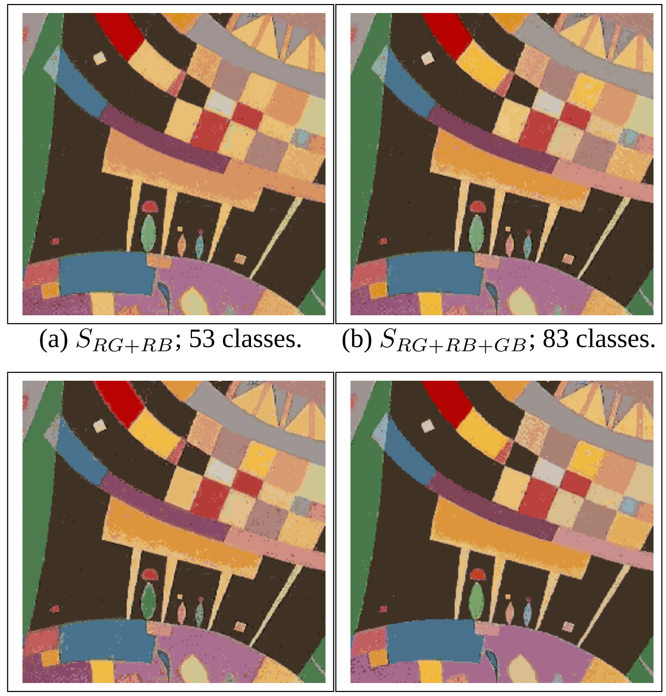
At every iteration of the main loop, a new segmentation image $S_{T_1+..+T_i}$ is created from the previous one ($S_{T_1+..+T_{i-1}}$) where some regions are split according to S_{T_i}. Therefore, this splitting process only takes into account spatial information about the initial segmentations given by morphological classifications. The splitting process, while combining information from these classifications, separates nearly all classes that overlap in 2D color spaces.

Figure 1 (b) shows the starting segmentation S_{RG} and 2 (a) and (b) illustrates the resulting segmentations, S_{RG+RB} and $S_{RG+RB+GB}$, after splitting respectively S_{RG} w.r.t. S_{RB} and S_{RG+RB} w.r.t. S_{GB}.

3.3. Region Merging

At the end of the splitting process, the result $S_{T_1+..+T_i}$ is an over-segmentation of the initial image I. Therefore, a region merging process is necessary. Since, we have only taken into account spatial information during the splitting process, we now introduce the notion of color distance within the merging process.

The CIE $L^*a^*b^*$ color system has more uniform perceptual properties than other spaces [3]; in particular, this space has better metric sensitivity for color differences and is very convenient to measure small color differences, whereas it is not the case of the RGB space. In CIE $L^*a^*b^*$ representation, given two colors $c_1 = (L_1, a_1, b_1)$ and $c_2 = (L_2, a_2, b_2)$, the color difference between them is:



(a) S_{RG+RB}; 53 classes. (b) $S_{RG+RB+GB}$; 83 classes.

(c) 2D (fusion) result; 23 classes. (d) 3D Result; 23 classes.

Fig. 2. Results of splitting and fusion processes.

$$d(c_1, c_2) = \sqrt{(L_1 - L_2)^2 + (a_1 - a_2)^2 + (b_1 - b_2)^2}.$$

In the merging algorithm, the regions with the smallest color difference are fist merged. The mean color values for new regions are updated. This process is repeatedly performed until $d(c_1, c_2)$ exceeds a given threshold. The final result corresponding to our textbook image is depicted by figure 2 (c).

4. EXPERIMENTATION AND RESULTS

We have employed a large variety of color images in our experiments. Some results of the proposed approach are shown in figures 1, 2, and 3. Image files can be fetched from `http://www.lrde.epita.fr/dload/papers/icip03` to better appreciate the differences between them. The corresponding results using automatic morphological approach in 3D space [1] are also listed in figure 2 (d). The latter is based on 3D histogram and no fusion is needed. It is obvious that some objects are not correctly segmented in 2D space before fusion as depicted by figure 1 (b) (c) (d). After the process of fusion (splitting and merging) all the main objects are correctly separated. The results are similar to the ones of the 3D method; see figure 2 (c) and (d). Furthermore, the execution time for the algorithm based on 2D space is less than that in 3D space.

Images	σ, R		Classes		Running time (s)	
	3D	2D	3D	2D	3D	2D
Comp10	3, 3	3, 3	23	23	63.16	6.53
Peppers	5, 4	4, 4	9	9	85.16	6.19
Woman	7, 6	7, 6	7	5	162.78	5.55
Mandrill	4, 3	4, 3	6	6	88.40	6.04
Lenna	4, 3	5, 3	9	8	73.65	14.70
House	5, 3	11, 4	7	6	63.83	5.38

Table 1. Comparison between 2D and 3D methods.

Some experimental results for the comparative performance between the 2D method proposed in this paper and the 3D method proposed in [1], are listed in Table 1.

The experimental results indicate that both results of the 3D method and the ones obtained from the 2D method are correct. However the latter is more efficient, particularly regarding running time.

Fig. 3. Results on a classical image.

5. CONCLUSION

A novel segmentation method for color image segmentation has been presented. The method is based on an efficient automatic morphological classification method in 2D color space and a fusion technique. It has been found that, for images with several classes or overlapping feature clusters, the 2D approach is even better than the 3D case. Even when the 3D performance is competitive with the 2D case, the latter is definitely more attractive because it requires less memory and running time (see table 1).

Our method can directly be extended to multi-band images. For n-bands images, C_n^2 segmentation maps in 2D space will be fused to obtain the final segmentation result.

Last, we do not yet have performed a rigorous study of our approach in terms of stability, robustness and accuracy of final segmentation results w.r.t. the different parameters; in addition, we do not have studied the impact of a more or less important correlation factor between projections.

Software

Source code of our method is available on the Internet from the location `http://www.lrde.epita.fr`. It has been developed using OLENA, our generic image processing library [8].

6. REFERENCES

[1] T. Géraud, P.Y. Strub, and J. Darbon, "Color image segmentation based on automatic morphological clustering," in *Proc. IEEE International Conference on Image Processing*, 2001, vol. 3, pp. 70–73.

[2] F. Kurugollu, B. Sankur, and A.E. Harmanci, "Color image segmentation using histogram multithresholding and fusion," *Journal of Image and Vision Computing*, vol. 19, no. 13, pp. 915–928, 2001.

[3] H.D. Cheng and Y. Sun, "A hierarchical approach to color image segmentation using homogeneity," *IEEE Trans. on Image Processing*, vol. 9, no. 12, pp. 2071–2082, 2000.

[4] R.D. Zhang and J.-G. Postaire, "Convexity dependent morphological transformations for mode detection in cluster analysis," *Pattern Recognition*, vol. 27, no. 1, pp. 135–148, 1994.

[5] P. Soille, "Morphological partitioning of multi-spectral images," *Journal of Electronic Imaging*, vol. 5, no. 3, pp. 252–265, 1996.

[6] S.H. Park, I.D. Yun, and S.U. Lee, "Color image segmentation based on 3D clustering: Morphological approach," *Pattern Recognition*, vol. 31, pp. 1061–1076, 1998.

[7] A. Bieniek and A. Moga, "An efficient watershed algorithm based on connected components," *Pattern Recognition*, vol. 33, pp. 907–916, 2000.

[8] J. Darbon, T. Géraud, and A. Duret-Lutz, "Generic implementation of morphological image operators," in *Proceedings of the International Symposium on Mathematical Morphology VI (ISMM)*, Sydney, Australia, April 2002, pp. 175–184.