Improved Techniques for Automatic Image Segmentation

Hai Gao, Wan-Chi Siu, and Chao-Huan Hou

***Abstract*—Mathematical morphology is very attractive for automatic image segmentation because it efficiently deals with geometrical descriptions such as size, area, shape, or connectivity that can be considered as segmentation-oriented features. This paper presents an image-segmentation system based on some well-known strategies. The segmentation process is divided into three basic steps, namely: simplification, marker extraction, and boundary decision. Simplification, which makes use of area morphology, removes unnecessary information from the image to make it easy to segment. Marker extraction identifies the presence of homogeneous regions. A new marker-extraction design is proposed in this paper. It is based on both luminance and color information. The goal of boundary decision is to precisely locate the boundary of regions detected by the marker extraction. This decision is based on a region-growing algorithm which is a modified watershed algorithm. A new color distance is also defined for this algorithm. In both marker extraction and boundary decision, color measurement is used to replace grayscale measurement and L*a*b* color space is used to replace the more straightforward spaces such as the RGB color space and YUV color space.**

***Index Terms*—Image segmentation, marker extraction, morphology.**

I. Introduction

OBJECT-ORIENTED processing has recently been introduced to image and video processing. MPEG-4 [1] provides a new video compression standard for multimedia applications using object based coding, i.e., each video object is coded in a separate data stream. Object-based coding provides more functionality to video, like object manipulation, hyperlinks, scene composition, and the combination of natural and synthetic video. MPEG-7 [2] is a set of standardized descriptors and description schemes, coding schemes for descriptions, and a description definition language for multimedia content. Examples of visual object-based descriptors are contour shape description or object motion trajectory. Possible applications of MPEG-7 are multimedia databases or authentication systems based on automatic (unsupervised) or semiautomatic (supervised) image segmentation. Both standards use image-segmentation information, but they do not describe how the segmentation masks are derived. Thus, image or video segmentation is an important field of research in video processing.

Manuscript received August 31, 2000; revised October 2, 2001. This paper was recommended by Associate Editor M. G. Strintzis.

H. Gao was with the Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering, Hong Kong Polytechnic University, Hong Kong. He is now with the Department of Digital System Integration, Institute of Acoustics, Chinese Academy of Sciences, Beijing, 100080, China (e-mail: haigao@hotmail.com).

W.-C. Siu is with the Department of Electronic and Information Engineering, Faculty of Engineering, Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong (e-mail: enwcsiu@polyu.edu.hk).

C.-H. Hou is with the Institute of Acoustics, Chinese Academy of Sciences, Beijing, 100080, China (e-mail: houch@public.bta.net.cn).

Publisher Item Identifier S 1051-8215(01)11034-7.

In this paper, we mainly deal with image segmentation, but the techniques are applicable to video signals.

II. Segmentation Strategies

A. Watershed Algorithm and Its Extension to Deal With Markers

Several techniques have been developed for image segmentation. The watershed algorithm, which is an important morphological tool for image segmentation, has been widely used in recent years [3]–[5]. The watershed technique [6] is a region-growing algorithm that analyzes an image as a topographic surface. It detects minima of the gradients of the gray-level image and grows these minima according to the gradient values. It can be viewed as a flooding process. Points of contact between the propagation originating from different minima are defined as the boundaries of the regions and are used to create the final partition. However, the watershed algorithm often leads to extreme over segmentation [6]. There are two approaches to reduce this problem. The first approach involves merging adjacent regions according to some criteria after the use of the watershed algorithm. Some recent articles such as [3]–[5] belong to this approach.

The second approach extends the watershed algorithm to deal with markers to reduce oversegmentation [7], [8]. Markers are a set of components marking flat regions of an image; i.e., each marker indicates the presence of an object. A marker represents the interior of an object. If the object interiors (markers) are set to 0, and the uncertainty areas are set to 1, we get a binary marker image. The binary marker image can be regarded as a first estimation of the partition. The first estimation is not a partition itself. It contains a set of components (markers) marking the core regions, and a large number of pixels may remain unassigned. The next step is then to optimize the initial partition to get the final partition. The watershed algorithm is used for partition optimization. We can impose the extracted markers as minima of the gradient function and suppress all other gradient minima [7]. Points of contact between the propagation originating from different minima are defined as the boundaries of the regions and are used to create the final partition. In this paper, we follow the principles of the second approach and also propose a new marker-extraction design based on both luminance and color information.

B. Boundary Decision With a Modified Watershed Algorithm

As discussed in Section II-A, once the markers have been defined, the decision can be taken by the watershed algorithm. The classical way of using the watershed algorithm to get object contours is to apply it to the morphological gradients of the signals. However, this method results in a great loss of information. So

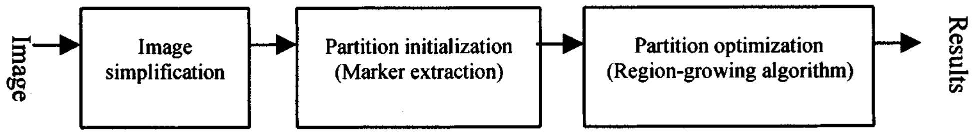


Fig. 1. Block diagram of image segmentation.

the use of gradients for segmentation is problematic. In particular, if the original signal involves transitions, its morphological gradient is either biased or too thick (more than two pixels).

The idea of using the watershed algorithm directly on the signal for segmentation was proposed in [9] to deal with color images. It is a region-growing algorithm. Our approach follows these principles with a modification on color distance measure.

C. Image Segmentation

A fairly general approach to morphological segmentation [8] involves three steps: image simplification, marker extraction, and contour definition. The scheme of this process is presented in Fig. 1.

In the first step, images are simplified for ease of segmentation; in the second step, markers are extracted; then in the last step, a region-growing algorithm is used to decide the boundaries. The second step, marker extraction, which will get the first estimation of the partition, is the most important step of the whole process, because the general view of the final partition is decided by this step on the whole. The marker-extraction step is also the most difficult step of the whole process, because it is not an easy task to extract markers exactly: too many markers will lead to extreme oversegmentation and too fewer markers will merge different objects.

In this paper, we simplify images by area morphological operators [10], [11]. This method is different from standard morphology which uses a structuring element, and avoids its disadvantages. We propose a new design for marker extraction, which makes use of both luminance and color information. In the third step, we use a region-growing algorithm to assign the remaining pixels, which follows the principles of [9] with a modification on color distance definition.

III. Image Simplification

In image segmentation, images are first simplified for ease of segmentation. Many standard morphological filters are often designed to smooth noisy gray-level images by a determined composition of opening and closing with a given structuring element. A standard morphological opening (res. closing) simplifies the original image by removing the *bright (res. dark)* components that do not fit within the structuring element. However, object boundaries may be distorted according to the shape of the structuring element used. Boundary distortion can lead to false segmentation results.

In our work, we use area morphological operators [10], [11]. An area open (res. close) operator on an image will remove all *bright (res. dark)* connected components that do not have a minimum area of s. Area openings and closings depend only on an area parameter and do not depend on structuring element shape. Area operators, therefore, avoid the associated problems of imposing the structuring element shape on a processed image.

Let us elaborate briefly the implementation of area openings. Note that a similar procedure is required for area closings. First, let us recall the definition of *regional maximum* [12]. A regional maximum m of a grayscale image I is a connected component of pixels with the same intensity value, such that every pixel in the neighborhood of m has a strictly lower value. The general principle of area openings [10], [11] is to decrease the intensity of all the "bright" connected components of the image. In order to specify all the "bright" connected components of the image, we begin with the extracted regional maxima. We consider *successively* all the extracted regional maxima of the original image. For each maximum, an iso-intensity line is drawn around it and the intensity of the iso-intensity line is progressively decreased until the area of the region which is enclosed by the iso-intensity line is larger than s, a given area parameter. Note that the enclosed region is a "bright" connected component, and the intensity of all the pixels of this component will be modified to the intensity of the iso-intensity line.

The data structure for area operators is the *heap* [10]. A pixel heap is basically a balanced tree of pixel pointers, which satisfies the heap condition: the grayscale value of any heap pixel is larger than the value of its children.

Let I be the original grayscale image. The successive steps of area openings are as follows.

1) Extract the regional maxima of I [12].
2) For each maximum m of I, do the following.
    a) Initialize the heap and add all neighbor pixels of m to the heap.
    b) Do the following until the number of the "marked" pixels (including the pixels belonging to maximum m) is larger than s, the given area parameter.
        i) Extract a pixel p from the heap.
        ii) If the intensity of p is not larger than the last extracted and "marked" pixel, pixel p will be "marked". Then scan all neighbor pixels of p, and the pixels that have not been added to the heap are put into the heap.
        iii) If the intensity of p is larger than the last "marked" pixel, it means that the scanning will lead to an unprocessed maximum. Pixel p will not be "marked", and the scanning for maximum m will be broken and no more pixels will be marked.
    c) Modify the original image: give all the "marked" pixels (including pixels belonging to maximum m) the intensity of the last "marked" pixel.

Note that the operations of adding a pixel to the heap and extracting a pixel from the heap include also rearranging the

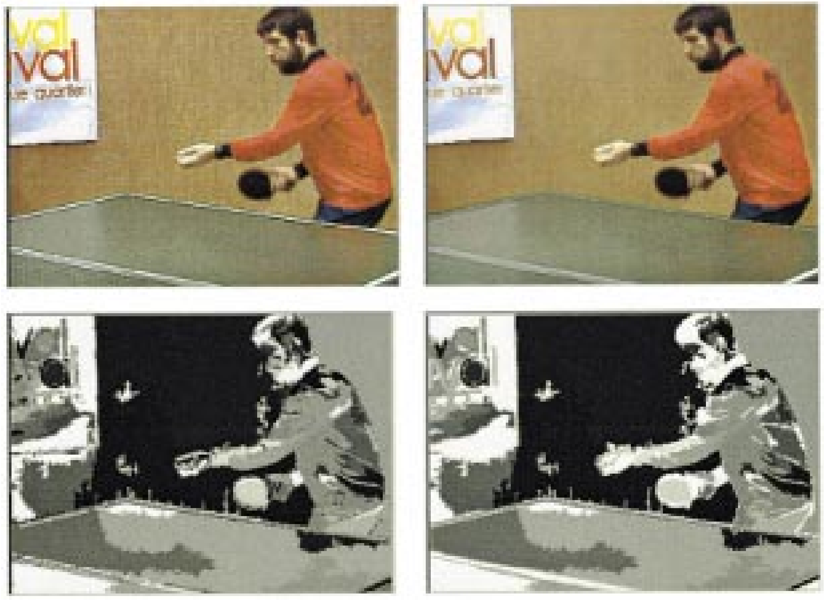


Fig. 2. Labeling of flat regions using the method given in [7]. First row: original and simplified image. Second row: labeling of all flat regions (2121 regions) and labeling of flat regions larger than 100 pixels (78 regions).

heap elements in order to preserve the pixel heap condition. For image simplification, area operations are taken on all the Y, Cr, and Cb components.

IV. Partition Initialization

The basic idea for this part is to extract flat regions of the simplified image. In [7], Salembier and Pardas introduced a method to extract flat regions of a simplified image based on "constrained region labeling" and "morphological contrast". This method is rather complicated. "Constrained region labeling" aims at finding the connected components of the space where the function is of constant gray-level value. Once a flat region has been labeled, if its size (number of pixels) is smaller than the given minimum size, its label is removed and the region is considered to be an uncertainty area. This labeling algorithm is viewed as the constrained labeling algorithm. However, the constrained region labeling itself is not good enough for the task of marker extraction. Marker extraction must be completed by another technique. In [7], "morphological contrast" is used to complete marker extraction. An example of labeling flat regions is given in Fig. 2.

In this paper, we propose a very simple but very efficient method to extract the flat regions of the simplified images based on both luminance and color information. If we set all the flat regions to 0 and set other regions to 1, we get the binary marker image. The marker image can be regarded as an initial partition, or a first estimation of the partition. As we have pointed out in Section II, the first estimation is not a partition itself. It contains a set of components marking the core regions (markers), and a large number of pixels may remain unassigned.

Flat regions can be simply identified by using the morphological gradient. Indeed, the areas where the morphological gradient has a low gray-level value correspond to flat regions. The morphological gradient by erosion and dilation is defined as follows.

Definition 1: If $f(x,y)$ denotes an input signal and M_n denotes a window or flat structuring element of size n, the erosion and dilation by the flat structuring element M_n are given by

$$\begin{aligned}\textbf{Erosion: } & \varepsilon_n(f)(x,y)\\ &= \mathrm{Min}\,\{f(x+x_0,y+y_0),(x_0,y_0)\in M_n\}\\ \textbf{Dilation: } & \delta_n(f)(x,y)\\ &= \mathrm{Max}\,\{f(x-x_0,y-y_0),(x_0,y_0)\in M_n\}. \end{aligned} \tag{1}$$

Definition 2: The morphological gradient is given by

$$\textbf{Morphological gradient: } \quad g(f)(x,y) = \delta_1(f)(x,y) - \varepsilon_1(f)(x,y). \tag{2}$$

The gradient is usually calculated on the luminance component. Sometimes luminance variations are very small along the borders of the adjacent objects. In such cases, we may get false markers due to the resulting small gradients. To overcome this problem, we incorporate color information into gradient computation.

Let $g_{\mathrm{Y}}(x,y)$ denote the gradient obtained from luminance information and $g_{\mathrm{C}}(x,y)$ denote the gradient obtained from color information, then the incorporated gradient $g(x,y)$ is given by

$$g(x,y) = \max\left(g_{\mathrm{Y}}(x,y), g_{\mathrm{C}}(x,y)\right). \tag{3}$$

$g_{\mathrm{Y}}(x,y)$ is calculated on the Y component in the YcbCr color space using (2), and g_{C} is calculated as follows:

$$g_{\mathrm{C}}(x,y) = \sqrt{\left(g_{\mathrm{L}^*}(x,y)\right)^2 + \left(g_{\mathrm{a}^*}(x,y)\right)^2 + \left(g_{\mathrm{b}^*}(x,y)\right)^2} \tag{4}$$

where $g_{\mathrm{L}^*}(x,y)$, $g_{\mathrm{a}^*}(x,y)$, and $g_{\mathrm{b}^*}(x,y)$ are calculated on the L*, a*, b* components in the L*a*b* color space using (2), respectively.

In order to get the gradient from color information, we use a nonlinear color space, the L*a*b* color space [13], which is

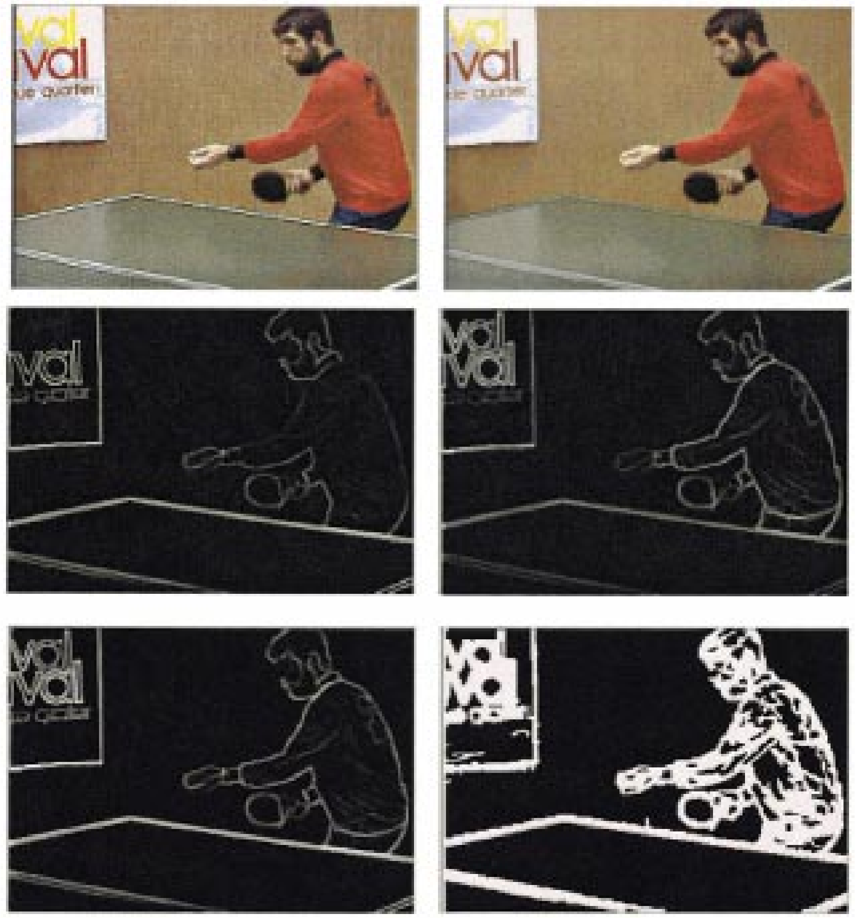


Fig. 3. Example of extraction of flat regions. First row: original and simplified image. Second row: gradient from luminance information and gradient from $\mathrm{L^*a^*b^*}$ color information. Third row: incorporated gradient image and marker image.

related to the CIE XYZ standard observer through a nonlinear transformation. The $\mathrm{L^*a^*b^*}$ color space is a suitable choice for this purpose because it is a perceptually equalized color space, i.e., the numerical distance in this space is proportional to perceived color difference.

The transformation of YCbCr (with ITU-R Rec. 624-4 specs) to RGB can be represented as follows [14]:

$$\begin{bmatrix} \mathrm{R} \\ \mathrm{G} \\ \mathrm{B} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1.4022 \\ 1 & -0.3456 & -0.7145 \\ 1 & 1.7710 & 0 \end{bmatrix} \begin{bmatrix} \mathrm{Y} \\ \mathrm{Cb} \\ \mathrm{Cr} \end{bmatrix}.$$

Note that in the above formula, we assume that Y is within the range $[0, 255]$, and Cb, Cr are within the range $[-128, 127]$. The transformation of RGB to $\mathrm{L^*a^*b^*}$ is represented as follows [14]:

1) R, G, B (with D65 white point) values are converted into X, Y, Z tristimulus values:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412\,453 & 0.357\,580 & 0.180\,423 \\ 0.212\,671 & 0.715\,160 & 0.072\,169 \\ 0.019\,334 & 0.119\,193 & 0.950\,227 \end{bmatrix} \begin{bmatrix} \mathrm{R} \\ \mathrm{G} \\ \mathrm{B} \end{bmatrix}.$$

2) $\mathrm{L^*}$, $\mathrm{a^*}$, $\mathrm{b^*}$ values are obtained from the following transformation:

$$L^* = \begin{cases} 116 * \left(\frac{Y}{Y_n}\right)^{1/3}, & \text{for } \frac{Y}{Y_n} > 0.008\,856 \\ 903.3 * \frac{Y}{Y_n}, & \text{otherwise} \end{cases}$$

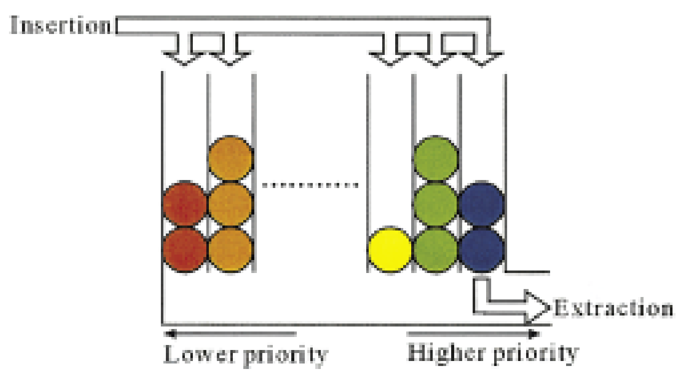


Fig. 4. Structure of the hierarchical queue [7].

$$\begin{aligned} a^* &= 500 * \left(f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right)\right) \\ b^* &= 200 * \left(f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right)\right) \end{aligned}$$

$$\text{where } f(t) = \begin{cases} t^{1/3}, & \text{for } t > 0.008\,856 \\ 7.787 * t + \frac{16}{116.0}, & \text{otherwise} \end{cases}.$$

Here X_n, Y_n, and Z_n are the tristimulus values of the reference white, which are selected as 255 for 8-bit image data representation. In many practical cases, the resulting gradient images exhibit many noisy gradients. In order to prevent this, gradient im-

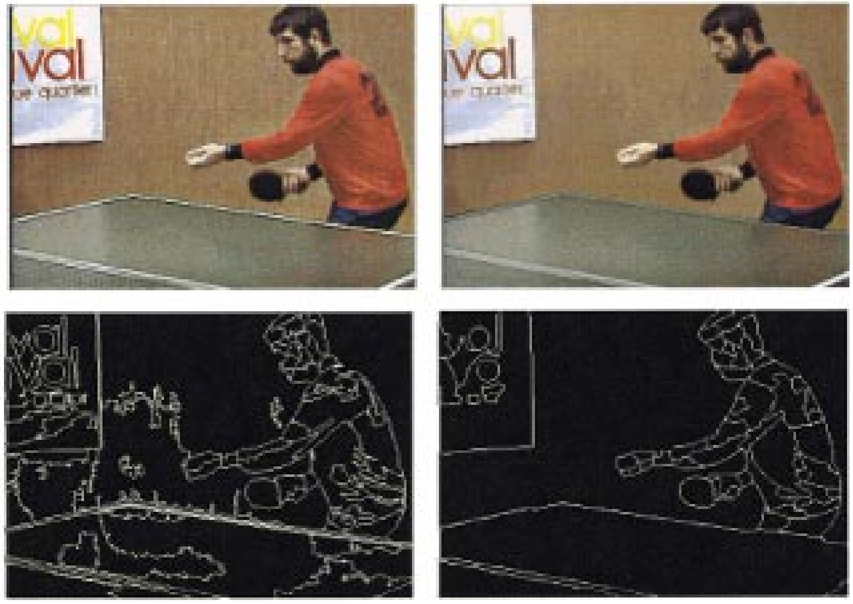


Fig. 5. First row: original image, and image simplified by area operators. Second row: segmentation result using the "constrained region labeling" marker extraction [7] and our proposed region growing, and segmentation result using our proposed marker extraction and region growing.

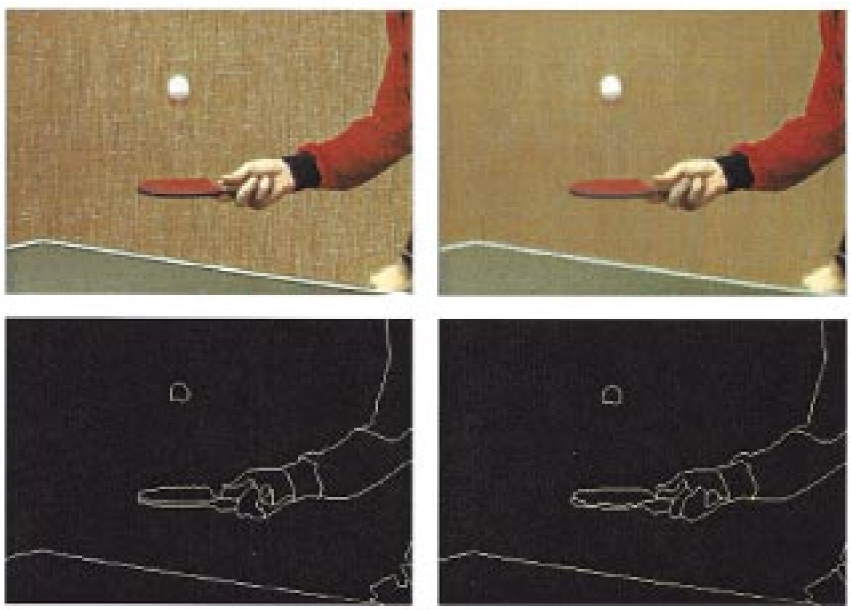


Fig. 6. First row: original image, and image simplified by area operators. Second row: segmentation result using our proposed marker extraction and Meyer's region growing [9], and segmentation result using our proposed marker extraction and region growing.

ages are open–closed by an area open–close operator to remove all bright and dark connected components that do not have a minimum area of s_0 (s_0 is much smaller than s, which is chosen for image simplification).

We then threshold the open–closed gradient image. All connected regions of the open-closed gradient image, which have a gray level lower than a given threshold are considered as markers (object interiors). The object interiors are set to 0, and the uncertainty areas are set to 1, giving us the binary marker image. The binary marker image can be regarded as the first estimation of the partition. Fig. 3 shows an example of marker extraction.

V. Partition Optimization

For partition optimization, as discussed in Section II-B, we follow the principles of [9] with a modification on color distance measure. To get the object contour, with respect to the classical way of using the watershed algorithm (which is to work on the morphological gradient of the signal to segment), the idea of using the watershed algorithm directly on the signal to segment has two distinctive characteristics. First, the pixels that are processed by the algorithm are not pixels of the gradient but pixels of the signal itself. Second, the priority of a pixel is not defined by its gray-level value, but defined as the opposite of its color

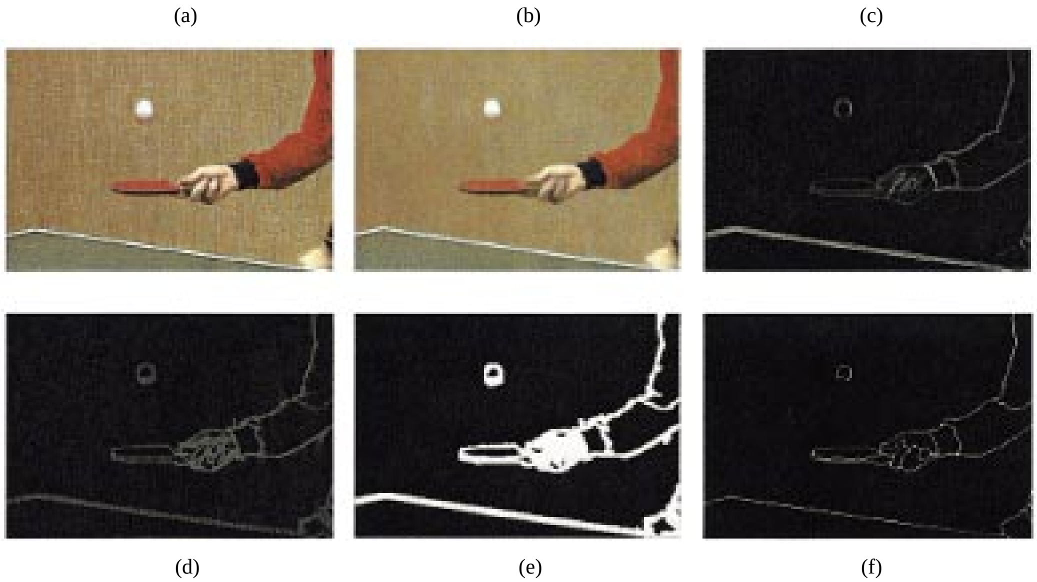


Fig. 7. Illustration of the whole process of the segmentation of the first frame of the table tennis sequence. (a) Original frame 000. (b) Simplified image 000. (c) Gradient image. (d) Open–closed gradient. (e) Marker image. (f) Segmentation result.

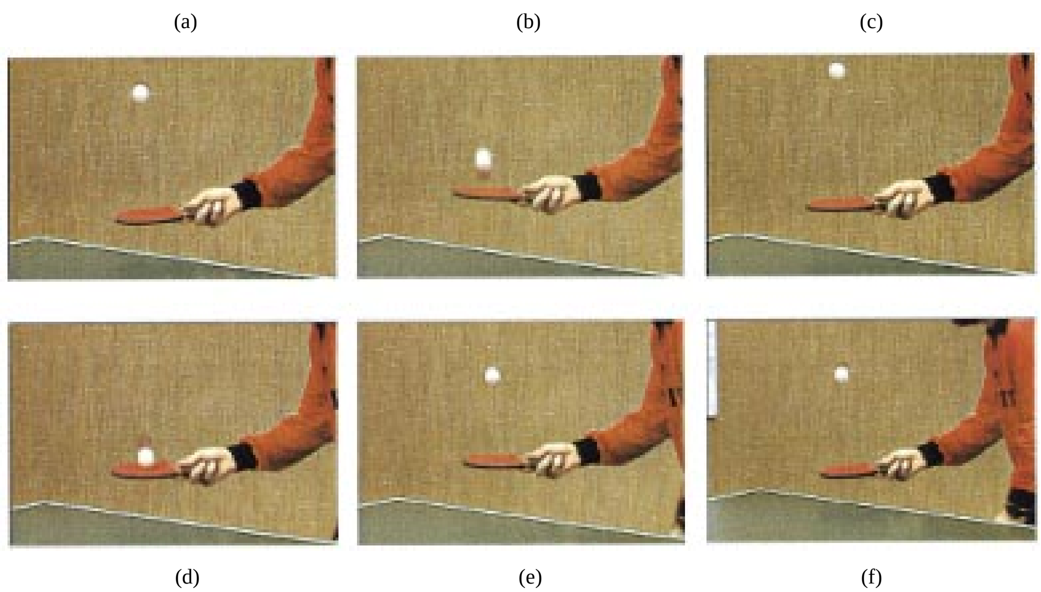


Fig. 8. Original frames of the table tennis sequence. (a) Original frame 006. (b) Original frame 012. (c) Original frame 018. (d) Original frame 024. (e) Original frame 030. (f) Original frame 036.

distance to a given region. This implies that a high (low) priority is assigned to a pixel with short (long) distance to a given region.

In [9], the color difference between two pixels $\mathbf{x}$ and $\mathbf{y}$ was defined in RGB color space as

$$\mathrm{Max}\left(|r_x - r_y|, |g_x - g_y|, |b_x - b_y|\right). \tag{5}$$

The author also used HLS color space for color distance definition and obtained better results as compared to the RGB color space.

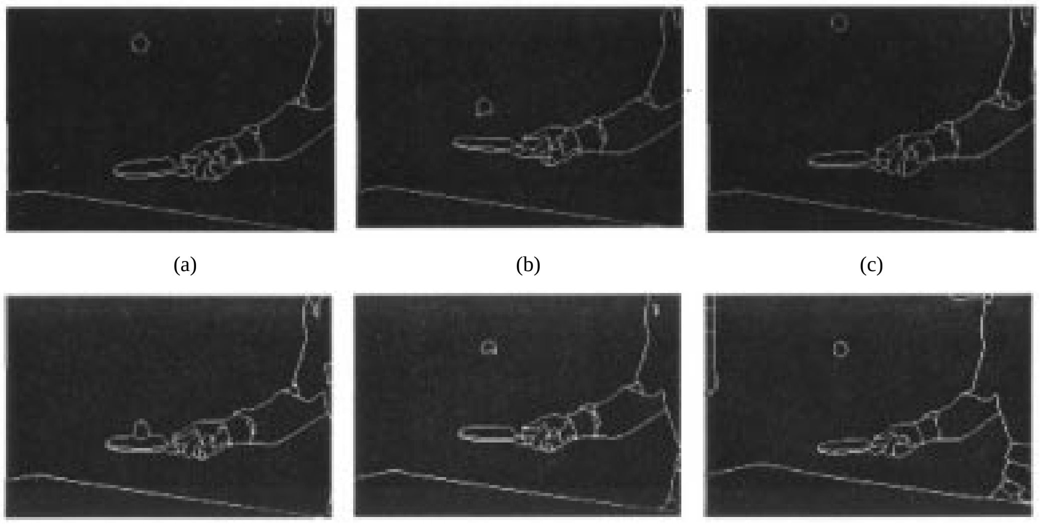
In this paper, we use $\mathrm{L^*a^*b^*}$ color space [13] for color distance definition, because $\mathrm{L^*a^*b^*}$ color space is a perceptually equalized color space, i.e., the numerical distance in this space is proportional to the perceived color difference. The color distance between two pixels $\mathbf{x}$ and $\mathbf{y}$ is defined as

$$\sqrt{\left(L_x^* - L_y^*\right)^2 + \left(a_x^* - a_y^*\right)^2 + \left(b_x^* - b_y^*\right)^2}. \tag{6}$$

The pixel priority of pixel p (to a region r) is defined as

$$\text{PRIORITY}\;(p) = \frac{1}{\sqrt{\left(L_p^* - L_r^*\right)^2 + \left(a_p^* - a_r^*\right)^2 + \left(b_p^* - b_r^*\right)^2}}. \tag{7}$$

The data structure used by the region-growing algorithm is the hierarchical queue (also called the ordered queue) [9]. It consists of a set of queues [7] with different priorities as shown



(a) (b) (c)

(d) (e) (f)

Fig. 9. Segmentation results of the frames of the table tennis sequence. (a) Segmented frame 006. (b) Segmented frame 012. (c) Segmented frame 018. (d) Segmented frame 024. (e) Segmented frame 030. (f) Segmented frame 036.

Fig. 10. Original frames of the Claire sequence.

in Fig. 4. The priority of each pixel is determined by its color distance to a given region.

Adopting the hierarchical queue and the pixel priority described above, the region-growing algorithm [9] involves two steps: *initialization of the hierarchical queue* and *growing of the markers*. The initialization step starts the process by giving the hierarchical queue the locations of all neighbor pixels of the markers with pixel priorities. The growing step repeats the following process until the queue is empty.

1) Extracts a pixel from the hierarchical queue.
2) This pixel has at least one labeled region in its neighborhood. If it has only one labeled region in its neighborhood, it is assigned to this region; if it has more than one labeled regions in its neighborhood, it is assigned to the labeled region with the shortest color distance to this pixel.
3) The neighbor pixels of this pixel that have not been labeled and still outside the hierarchical queue are put into the hierarchical queue with pixel priorities.

When the queue is empty, every pixel in the image is associated with an object. To show segmentation results more clearly, we represented them by boundaries.

VI. Experimental Results

We have performed a large number of experiments to test our new approach. In this section, we give some of these experimental results. In Section IV, we have presented our design for marker extraction, which has been implemented in our segmentation algorithm. Fig. 5 gives experimental results which compare our segmentation algorithm and the techniques described

Fig. 11. Segmentation results of the frames of the Claire sequence.

in [7]. Fig. 6 shows the results using different color distance definitions. We can see that our definition gives better results.

Figs. 7–11 are segmentation examples of the table tennis sequence and the Claire sequence by our techniques. Let us describe the whole process of the segmentation of the first frame of the table tennis sequence in some detail, and present the segmentation of other sample frames briefly. Fig. 7(a) is the original first frame of the table tennis sequence. In the first step, this image is simplified by the area open–close operator $(s = 80)$; Fig. 7(b) shows the simplified image. Fig. 7(c) is the gradient image obtained from (3). Fig. 7(d) is the open–closed gradient. Fig. 7(e) is the marker image and Fig. 7(f) is the final segmentation results. Figs. 8 and 9 show the segmentation of other frames of the table tennis sequences. Figs. 10 and 11 show the segmentation of the frames of the Claire sequences.

VII. Conclusion

In this paper, an unsupervised image-segmentation algorithm based on morphological tools has been presented. It involves three steps: simplification, marker extraction, and boundary decision.

Simplification removes disturbing components of the image while retaining the contours of the remaining elements. Marker extraction takes advantage of simplification, which identifies flat regions. Marker-extraction output is an image indicating the presence of homogeneous regions whose contours are not precisely defined. The last step decides accurate object boundaries by the watershed algorithm. As demonstrated, this unsupervised segmentation technique is robust. Our experimental work also shows that we can obtain very good spatial segmentation results from this technique.

References

[1] *MPEG-4 Visual Fixed Draft International Standard*, ISO/IEC 14 496–2, Oct. 1998.

[2] “MPEG-7 Requirements Document,” MPEG Requirements Group, 1998.

[3] J. M. Gauch, “Image segmentation and analysis via multiscale gradient watershed hierarchies,” *IEEE Trans. Image Processing*, vol. 8, pp. 69–79, Jan. 1999.

[4] K. Haris, S. N. Efstratiadis, N. Maglaveras, and A. K. Katsaggelos, “Hybrid image segmentation using watersheds and fast region merging,” *IEEE Trans. Image Processing*, vol. 2, pp. 1684–1699, Dec. 1998.

[5] A. Tremeau and P. Colantoni, “Regions adjacency graph applied to color image segmentation,” *IEEE Trans. Image Processing*, vol. 9, pp. 735–744, Apr. 2000.

[6] F. Meyer and S. Beucher, “Morphological segmentation,” *J. Vis. Commun. Image Rep.*, vol. 1, no. 1, pp. 21–46, Sept. 1990.

[7] P. Salembier and M. Pardas, “Hierarchical morphological segmentation for image sequence coding,” *IEEE Trans. Image Processing*, vol. 3, pp. 639–651, Sept. 1994.

[8] P. Salembier and F. Marques, “Region-Based representations of image and video: Segmentation tools for multimedia services,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 1147–1169, Dec. 1999.

[9] F. Meyer, “Color image segmentation,” in *Proc. IEE Int. Conf. Image Processing and its Applications*, The Netherlands, 1992, pp. 303–306.

[10] L. Vincent, “Grayscale area openings and closings, their efficient implementation and applications,” in *Proc. 1st Workshop on Mathematical Morphology and Its Applications to Signal Processing*, Barcelona, Spain, May 1993, pp. 22–27.

[11] ——, “Morphological area openings and closings for gray-scale images,” in *NATO Advanced Research Workshop on “Shape in Picture: Mathematical Description of Shape in Grey-Level Images*, Driebergen, The Netherlands, 1992, pp. 197–208.

[12] ——, “Morphological grayscale reconstruction in image analysis: applications and efficient algorithms,” *IEEE Trans. Image Processing*, vol. 2, pp. 176–201, Apr. 1993.

[13] P. Colantoni and B. Laget, “Color image segmentation using region adjacency graphs,” in *Proc. (IEE) 6th Int. Conf. Image Processing and Its Applications*, vol. 2, 1997, pp. 698–702.

[14] R. S. Berns, *Principles of Color Technology*, 3rd ed. New York: Wiley, 2000.