Part of IS&T/SPIE Conf. on VCIP'99, SPIE Vol.3653, Part Two, pp.937–945 (1999 1)

Image segmentation and object extraction based on geometric features of regions

Toru Tamaki[a], Tsuyoshi Yamamura[b] and Noboru Ohnishi[ac]

[a] Dept. of Information Eng., Graduate School of Eng., Nagoya Univ., Japan.

[b] Faculty of Information Science and Technology, Aichi Prefectural Univ., Japan.

[c] Bio-Mimetic Control Research Center (BMC),
The Institute of Physical and Chemical Research (RIKEN), Nagoya, Japan

ABSTRACT

We propose a method for segmenting a color image into object-regions each of which corresponds to the projected region of each object in the scene onto an image plane. In conventional segmentation methods, it is not easy to extract an object-region as one region. Our proposed method uses geometric features of regions. At first, the image is segmented into small regions. Next, the geometric features such as inclusion, area ratio, smoothness, and continuity, are calculated for each region. Then the regions are merged together based on the geometric features. This merging enables us to obtain an object-region even if the surface of the object is textured with a variety of reflectances; this isn't taken into account in conventional segmentation methods. We show experimental results demonstrating the effectiveness of the proposed method.

Keywords: image segmentation, object extraction, object-region, geometric feature, split and merge

1. INTRODUCTION

Image segmentation is a fundamental process in computer vision, and various methods have been studied. The usual image segmentation method divides an image into homogeneous regions each of which is a set of pixels with similar features — intensity, color, variance, texture, etc. Thus, a textured object in an image is segmented into several regions by conventional segmentation methods.

Ideally, an image should be segmented so that one object in the scene holds one region in the image. We call this an object-region. If such ideal segmentation was possible, then other subsequent processes could easily use the result of segmentation for scene description and recognition. However, in dividing an image into homogeneous regions, conventional segmentation methods tend to divide one object region into several regions. We call this phenomenon over-segmentation. The reasons why this over-segmentation occurs are described below.

One is optic factor such as shading, shadow, highlights, interreflection, and gradation of the intensity in the image formation. These cause multiple homogeneous regions on the object surface with uniform reflectance. To overcome this defect, a conventional segmentation method "split-and-merge"[1,2] is often used. In the beginning of the method, it splits an image into many regions and then merges them with various criteria, such as color similarity of two regions, boundary intensity between regions, and Minimum Description Length (MDL).[3,4]

The other is physical factor; the structure of a textured object whose surface consists of regions with different spectral reflectances. Generally, very few objects have only one feature. Nevertheless, since conventional segmentation methods divide an image based on the similarity of features, one object is usually divided into several regions and fails to be extracted as the corresponding object-region in an image.

Thus, over-segmentation is inevitable when a region in segmentation is defined as a set of pixels which have a similar feature. This is one of the reasons why image processings, such as object tracking and recognition in an image, are not performed so well.

Correspondence to: T. Tamaki (address : Furo-cho, Chikusa-ku, Nagoya 464-8603 JAPAN, phone : +81 52 789-3811, fax : +81 52 789-3814, e-mail : tamaki@ohnishi.nuie.nagoya-u.ac.jp, WWW : http://www.ohnishi.nuie.nagoya-u.ac.jp/~tamaki/)

In this paper, we propose a new method for image segmentation in order to obtain an object-region composed of several regions each of which has a similar feature but different one from the others. This segmentation method uses splitting method to eliminate the optic factor of the over-segmentation and merging method to remove the physical factor based on geometrical relationships between regions, which is not a pictorial information such as color or intensity. This merging enables us to obtain an object-region even if the surface of the object is textured with a variety of reflectances; this isn't taken into account in conventional segmentation methods. We define the object-region and describe how to obtain the object-region in section 2, give the algorithm of the proposed image segmentation in section 3, and provide the results of a real image in section 4. We conclude this paper in section 5.

2. REGION WHICH SHOULD BE EXTRACTED

2.1. Definition of object-region

We define an object-region as a projection of a three-dimensional object onto an image plane. For example, in a picture containing human figure there are multiple regions with uniform texture, such as the face, hair, and clothes. These regions are parts of the human body, and compose the image projection of the whole human body, i.e., the object-region.

Those partial regions have the following properties;

- locate inside the outer contour of the corresponding object.
- relatively small in comparison with a projection of the object.
- some have rugged or bended boundary corresponding not to the real shape of the object but to a change of reflectances.

Partial regions with these properties must be correctly merged into the corresponding object-region. Then in this study we assume that an object region has a closed and smooth boundary on an image plane. And we employ a split-and-merge strategy that divides an image into many small partial regions and merges* neighboring two regions to generate larger one until obtaining the object-regions.

2.2. Geometric features between partial regions

Based on the assumption described above, we define the following four features of boundaries; continuity, smoothness, inclusion, and area ratio. We call these features geometric features, which haven't been considered in the conventional segmentation method. When two adjacent regions satisfy any one of these features, they are merged together.

continuity Both have smoothly connected boundaries.

If an object is occluded by another object, its contour is hidden by the occluding one, and the orientation of the boundary of the region changes abruptly. Otherwise the boundary continuously changes its direction[7] (**Figure 1**). In cases where regions comprise the same surface of an object, even if their color differs greatly, the boundaries may be smoothly connected. These situation is simplified and illustrated in Fig.1.

Of course there are regions which have bending boundary, e.g. a square that is a projection of a cube. However it is very rare that a boundary of an object placed behind the cube lies exactly upon a side of the cube, they seldom have their common tangent at the vertex (**Figure 2**). Therefore there is little possibility that the square region corresponding to the cube is merged with the region of the object (sphere) behind the cube.

*It is possible that two separate regions could belong to the same object-region. One case is that an object is occluded by another, for example, the object has a hole, or is placed in front of another. Another case is that a region is perceived but doesn't have not its real contour. Such phenomenon is called perceptual grouping or subjective contour, and they are due to the configuration of regions. This case has been examined on a line drawing[5] or edge.[6] However the separated case is not considered in this paper because it isn't easy to merge separate regions.

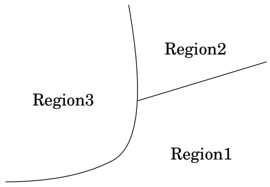

Figure 1. Continuity. Region1 and Region2 are smoothly connected, but Region3 is not connected smoothly to neither Region1 nor Region2.

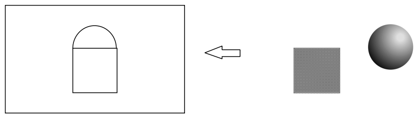

Figure 2. Special case that merging by continuity yields a wrong result. The square is a projection of the cube, and the circle is that of the sphere.

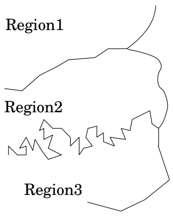

Figure 3. Smoothness. The boundary between Region2 and Region3 is jagged, and the boundary between Region1 and Region3 is smooth.

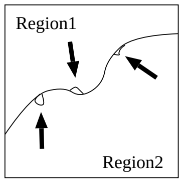

Figure 4. The regions indicated by the arrows are small as compared with Region1 and Region2.

smoothness Both don't have a smooth common boundary.

A gradual change of intensity yields a false boundary, and it is not smooth but jagged. Examples of these boundaries are shown in **Figure 3**. There are objects having jagged contour, but the contour appearance depends on viewing distance. For example, when we look at the whole of a tree from a distance, the contour looks smooth due to being out of focus. On the contrary, when we observe the tree from a nearer point, we can't see the contour of the tree but that of the leaves. So a jagged boundary is regarded as the result of the optic factor.

inclusion One is included by the other.

A region included by the other region corresponds to a part of a three-dimensional object; for example, eyes and mouth regions in a face.

area ratio One is much larger than the other.

Figure 4 shows that small regions are adjacent to larger regions. It is unrealistic that an image has many object-regions whose size is one pixel. A very small region compared with its adjacent ones may be considered as noise or a part of a larger object-region.

2.3. Formulation

Here we formulate the four geometric features described above to decide which two neighboring regions are merged prior to others. Each feature is a maximum or minimum value calculated for each region R_i against its adjacent region R_j ($j \in Q(i)$). Here $Q(i)$ is a set of region numbers adjacent to the region R_i. Feature values of the region R_i are denoted by $V_{feature}(i)$.

- inclusion

$$V_{inclusion}(i) = N(Q(i)), \tag{1}$$

where $N(\cdot)$ denotes the number of elements of the set $Q(i)$. $V_{inclusion}(i)$ is the number of adjacent regions and larger than or equal to 1. $V_{inclusion}(i) = 1$ when the adjacent region is only one, i.e., the region is included by another.

- area ratio

$$V_{area}(i) = \max_{j \in Q(i)} \frac{S(R_j)}{S(R_i)}, \tag{2}$$

where $S(R)$ denotes the number of pixels of region R. $V_{area}(i)$ is the maximum of area ratios of all adjacent regions R_j ($j \in Q(i)$) to the region R_i. It is the value against the largest adjacent region, and it becomes larger than 1 when there is a bigger adjacent region than R_i.

- smoothness

$$V_{smoothness}(i) = \min_{j \in Q(i)} \frac{1}{L_{ij}} \int\limits_{\partial R_i \cap \partial R_j} \left\{ \frac{\|\boldsymbol{v}(s) - \boldsymbol{v}(s+\ell)\|}{\ell} \right\} ds, \tag{3}$$

where ∂R is the boundary of region R. As shown in **Figure 5**, $L_{ij} = \int\limits_{\partial R_i \cap \partial R_j} ds$ is the length of a part of the common boundary $\partial R_i \cap \partial R_j$ of the regions R_i and R_j. $\boldsymbol{v}(s)$ denotes the position of a point on the common boundary, and ℓ is a unit length along the boundary to examine the smoothness. $V_{smoothness}(i)$ ranges between 0 and 1, and $V_{smoothness}$ increases as the boundary becomes smooth because the straight distance between two points on the common boundary is divided by the constant length ℓ along the boundary[†]. $V_{smoothness}$ is 1 when the boundary is a straight line.

- continuity

$$V_{continuity}(i) = \min_{j \in Q(i)} \cos^{-1}\left(\frac{(\boldsymbol{t}_i(j), \boldsymbol{t}_j(i))}{\|\boldsymbol{t}_i(j)\| \, \|\boldsymbol{t}_j(i)\|} \right), \tag{4}$$

where $\boldsymbol{t}_i(j)$ is a vector tangent to the boundary ∂R_i at the T-junction due to two boundaries of the regions R_i and R_j (**Figure 6**), and $(\cdot, \cdot)$ represents the inner product. $V_{continuity}(i)$ ranges between 0° and 180° because it is the angle between two tangent vectors, and $V_{continuity}(i)$ decreases as the two boundaries' connection becomes smooth. Note that the direction of the tangent vector is determined so that its left side should be the interior of the region.

3. THE ALGORITHM OF SEGMENTATION

According to the previous discussion in subsection 2.1, the proposed method first splits a whole image into many small partial regions and then repeat merging two of them using the four geometric features mentioned in the previous section. Here we describe the method of initial splitting, the algorithm for merging, and how to extract object-regions from the merged regions.

3.1. Initial Splitting

Initial region splitting uses color-space clustering which is a conventional image segmentation method. At first an image is transferred from the RGB color space to the CIE $L^*a^*b^*$ color space[8] which is perceptually more linear presentation than the RGB space.

Next, k-means clustering is applied to the image represented with a five-dimensional vector (L^*, a^*, b^*, x, y) ; (x, y) is the position of each pixel in the image, and (L^*, a^*, b^*) is its color coordinate in the CIE $L^*a^*b^*$ color space. Since the clustering result depends on k, the number of clusters, preparatory clustering is performed to

[†]Although a sort of smoothness must be defined originally as a curvature of the boundary, we use this expression to calculate curvature of curve because it is difficult to accurately get curvature on a digitized image.

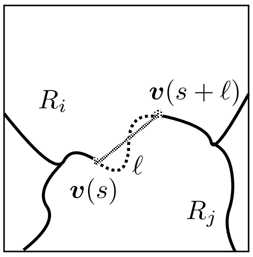

Figure 5. Smoothness of boundary. The dashed line between two points $\boldsymbol{v}(s)$ and $\boldsymbol{v}(s+\ell)$ represents the length along the common boundary, and the gray solid line shows the straight distance between them.

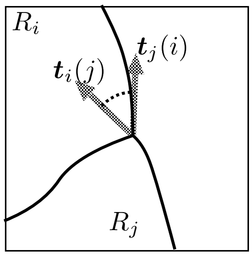

Figure 6. Continuity of boundaries. $\boldsymbol{t}_i(j)$ and $\boldsymbol{t}_j(i)$ represent the tangent vectors of the boundary of R_i and R_j, respectively, at the T-junction,

the reduced scale image in advance by changing k. Here the centers of the initial clusters are set at random. And k which minimizes the sum of variance of each cluster is selected, and used in the real clustering which is applied to the original scale image. As a result, the image is segmented into N regions R_i ($i = 1, \ldots, N$).

3.2. Feature-based merging algorithm

After the segmentation, the image is divided into many partial regions. They are merged if they have the geometric features described in subsection 2.3. We show a merging algorithm using the geometric features below.

Step 0. Set initial values of three thresholds Th_{area}, Th_{smooth}, $Th_{continuity}$.

Step 1. If the number of regions are two, stop the process. Otherwise, if $V_{inclusion}(i) = 1$, i.e., the region R_i is included by another region, merge it with the bigger region. Repeat this operation over i until no inclusion relation exists.

Step 2. Calculate the area ratio $V_{area}(i)$ for each region R_i. If $\max_i V_{area}(i) \leq Th_{area}$, i.e., there is no region all of whose adjacent regions are relatively large, then go to Step 3. Otherwise, merge the region R_i with the largest value of $V_{area}(i)$ to the adjacent region R_j with the most similar to R_i in mean color. Here, j is determined by $j = \operatorname*{argmin}_{j \in Q(i)} D(R_i, R_j)$, where $D(R_i, R_j)$ is CIE color difference of the mean color between regions R_i and R_j. The reason of employing color information is that merging only with size information causes unsmooth boundaries. Go to Step 1.

Step 3. Calculate the smoothness of boundary $V_{smooth}(i)$ for each region R_i. If $\min_i V_{smooth}(i) \geq Th_{smooth}$, i.e., boundaries of all regions are smooth enough, then go to Step 4. Otherwise, merge the region R_i with the smallest value of $V_{smooth}(i)$ to the adjacent region R_j which minimizes $V_{smooth}(i)$; here, $j = \operatorname*{argmin}_{j \in Q(i)} V_{smooth}(i)$. Go to Step 1.

Step 4. Calculate the continuity of boundary $V_{continuity}(i)$ for each region R_i. If $\min_i V_{continuity}(i) \geq Th_{continuity}$, i.e., any two adjacent regions are smoothly connected to each other, then go to Step 5. Otherwise merge the region R_i with the smallest value of $V_{continuity}(i)$ to the adjacent region R_j which minimizes the feature $V_{continuity}(i)$; here, $j = \operatorname*{argmin}_{j \in Q(i)} V_{continuity}(i)$. Go to Step 1.

Step 5. Update the thresholds; decrease Th_{area}, and increase Th_{smooth} and $Th_{continuity}$. However if each updated threshold reaches to the limitation, the updating won't be performed any more. If all of three thresholds are not updated, stop the merging process. Otherwise, go to Step 1.

3.3. Object-region extraction

The merging divides the image into two regions which are the object-region and the background, or more than two regions when it stops because of no threshold updating. Even if plural regions remained in the image after merging, we must select only one region because we assume only one object exists in the image of a scene. However if it is known that there are more than one objects, a modification is needed to properly select two or more regions.

In order to extract an object-region we define a weight function $W(x, y)$ represented by two-dimensional Gaussian $G(x, y)$, whose center is located at the center of the image with the width of w and height of h;

$$W(x, y) = G\left((x, y); (\tfrac{w}{2}, \tfrac{h}{2}), \tfrac{\min(w,h)}{4}\right) \tag{5}$$

where $G((x, y); (m_x, m_y), \sigma)$ is a isotropic 2-D Gaussian function, (m_x, m_y) its means and σ standard deviation: i.e., the covariance matrix is $\left[\begin{smallmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{smallmatrix}\right]$.

For all region R_i, the sums of the weights in each region are calculated, and finally we extract a region R_{i^*} which has the largest amount of weight.

$$i^* = \operatorname*{argmax}_i \sum_{(x,y)\in R_i} W(x, y) \tag{6}$$

4. EXPERIMENTAL RESULTS

We implemented the proposed method on a workstation and conducted experiments using several real images. In the following experiments, the thresholds Th_{area}, Th_{smooth} and $Th_{continuity}$ are 1200, 0.5 and 5, respectively, and by updating, Th_{area} is reduced to 0.9 times, and Th_{smooth} and $Th_{continuity}$ are increased to 1.01 times. The limit of the thresholds Th_{area}, Th_{smooth} and $Th_{continuity}$ are 30, 0.65 and 20, respectively. We select 20 as ℓ a unit length to examine the smoothness.

In **Figure 7**, (a) shows an original color image (432×301 size) of a portrait of a man in front of slightly blurred background, and an initial segmentation result producing 59 regions is illustrated in (b) by drawing the region boundaries with black line. Images (c)∼(k) show the snapshots of the region merging process. The number under each image denotes the number of remaining regions at the stage of merging, in other words, the number of iteration of the merging process (e.g., (c) is the result after $13 (= 59 - 46)$ times merging.)

In this figure, false boundaries in the background were removed by the stage (f), and small regions which arose between the object and the background were eliminated as the merging process proceeded (see (c) and (d)). Finally the image was segmented into two object-regions corresponding to the man and the background as shown in (k)[‡], and the extracted object-region is painted in black in (l). Note that in the final result, the parts of face, such as eyes, mouth and nose belong to the object-region of the man, and the highlighted and shadowed regions on the face and clothes are not segmented to different region.

Another experimental result is shown in **Figure 8**. (a) is a color input image (116×261 size) of a woman putting on blue and black clothes against a pink background, and this is the same image used by Zhu *et al.*[4] In the final result of merging shown in (j), the image is divided into four regions, the upper and lower halves of the body and two backgrounds, although the upper and lower body halves should belong to the same object-region corresponding to the body. The upper and lower halves are not merged because the arms are placed perpendicular to the body. So the relation of continuity cannot work well and updating thresholds was stopped due to the limitation. The result of ours, however, is considered better than that of Zhu *et al.* shown in (l) from the viewpoint of an object-region which we intend to extract.

‡Although these two regions may be merged next time with the inclusion relation, this result should occur because the merging process is terminated when the number of remaining regions is two.

Other experimental results are shown in **Figure 9**. In each image, the region expected to be segmented as an object-region is roughly extracted (for example, the human figures are partially extracted in (a), (b) and (c) even with various complex backgrounds). (d) is the simple case. The black eyes of a turtle are merged into its green body. This is what we intended the proposed method to do. The cases in (e) and (f), holed regions are extracted (strictly speaking the former is not holed but opening). Both results are due to the defectiveness of formulating inclusion. (g) shows a result against the image in which there are two books. The results shows only one object is extracted. We have to modify the assumption for extracting both objects. The difference between (j) and (k) is only whether a stapler exists or not. But the results are considerable different from each other, because the calculation of continuity and smoothness cannot be done effectively at quantized digital image.

We describe computation time needed to process an image. We implemented the proposed method by GNU C++ on AT compatible personal computer with 200MHz MPU. The time for initial segmentation is from about twenty seconds to at most one minute, and merging time is from three to five minutes. But the time depends on an image, especially complexity of texture.

5. CONCLUSION

We developed a geometric feature-based image segmentation and object extraction method that can obtain an object-region composed of partial regions with homogeneous features. We showed the experimental results for several real images and compare them with the other method. We will analyze the proposed method in more detail to improve its performance (for example, the initial and limit value of the thresholds were determined heuristically). In the merging process we get relationship among merged regions and generated one. Representing such relationship as a tree structure in each step of the process, we can develop methods for understanding a hierarchical relation among the three-dimensional objects.[9–11]

REFERENCES

1. R. Jain, R. Kasturi, and B. G. Schunck, *MACHINE VISION*, McGraw-Hill, New York, 1995.
2. Y. Shirai, *Three-Dimensional Computer Vision*, Springer-Verlag, Berlin, 1987.
3. T. Kanungo, B. Dom, W. Niblack, and D. Steele, "A Fast Algorithm for MDL-Based Multi-band Image Segmentation," in *Proc. of IEEE CVPR'94*, pp. 609–616, 1994.
4. S. C. Zhu and A. Yuille, "Region competition: Unifying Snakes, Region Growing, and Bayes/MDL for Multiband Image Segmentation," *IEEE Trans. on PAMI.* **18**(9), pp. 884–900, 1996.
5. A. Sha'ashua and S. Ulman, "Structural saliency: The detection of globally salient structures using a locally connected network," in *Proc. of IEEE ICCV'88*, pp. 321–327, 1988.
6. T. D. Alter and R. Basri, "Extracting Salient Curves from Images: An Analysis of Salient Network," in *Proc. of IEEE CVPR'96*, pp. 13–20, 1996.
7. P. J. Kellman and T. F. Shipley, "A Theory of Visual Interpolation in Object Perception," *Cognitive Psychology* **23**, pp. 141–221, 1991.
8. M. Groß, *Visual Computing*, Springer-Verlag, Berlin, 1994.
9. D. Arita, N. Tsuruta, R. Taniguchi, and M. Amamiya, "An Object Recognition System Based on a Model Generated from Image Examples," in *Proc. of ACCV'95*, vol. 3, pp. 161–165, School of EEE, IEEE Singapore Section, (Singapore), 1995.
10. K. L. Vincken, A. S. E. Koster, and M. A. Viergever, "Probabilistic Multiscale Image Segmentation," *IEEE Trans. on PAMI.* **19**(2), pp. 109–120, 1997.
11. A. S. E. Koster, K. L. Vincken, and M. A. Viergever, "Heuristic Linking Models in Multi-Scale Image Segmentation," *Computer Vision and Image Understanding* **65**(3), pp. 382–402, 1997.

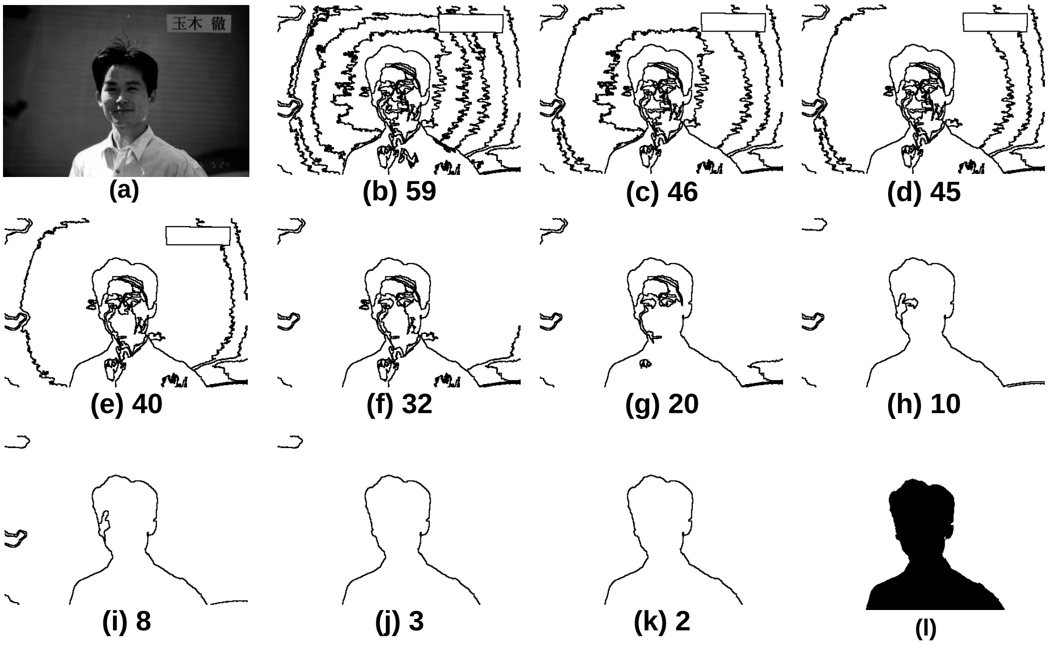

Figure 7. Experimental result. (a) Original image whose size is 432×301. (b) Initial segmentation result with 59 regions. (c)∼(k) Processes of merging. Number under each image is the number of regions at the stage of merge. (l) Extracted object-region shown as a black area.

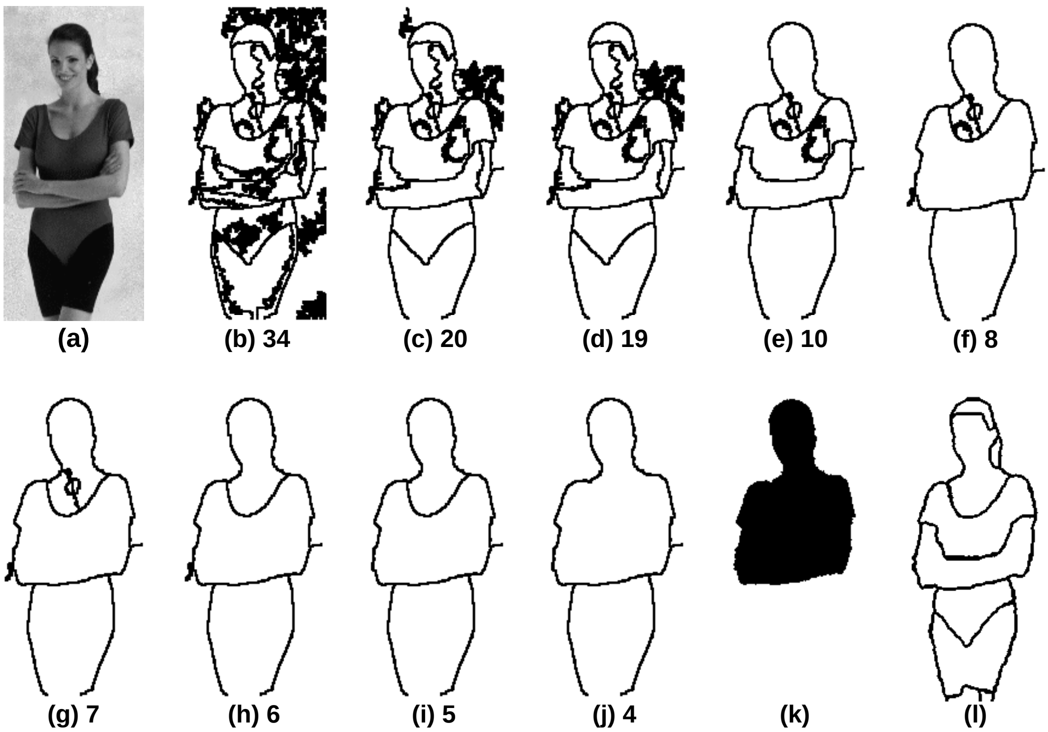

Figure 8. Experimental result. (a) Original image whose size is 116×261. (b) Initial segmentation result with 34 regions. (c)∼(k) Processes of merging. Number under each image is the number of regions at the stage of merge. Four regions remains after the merging. (l) Extracted object-region shown as a black area.

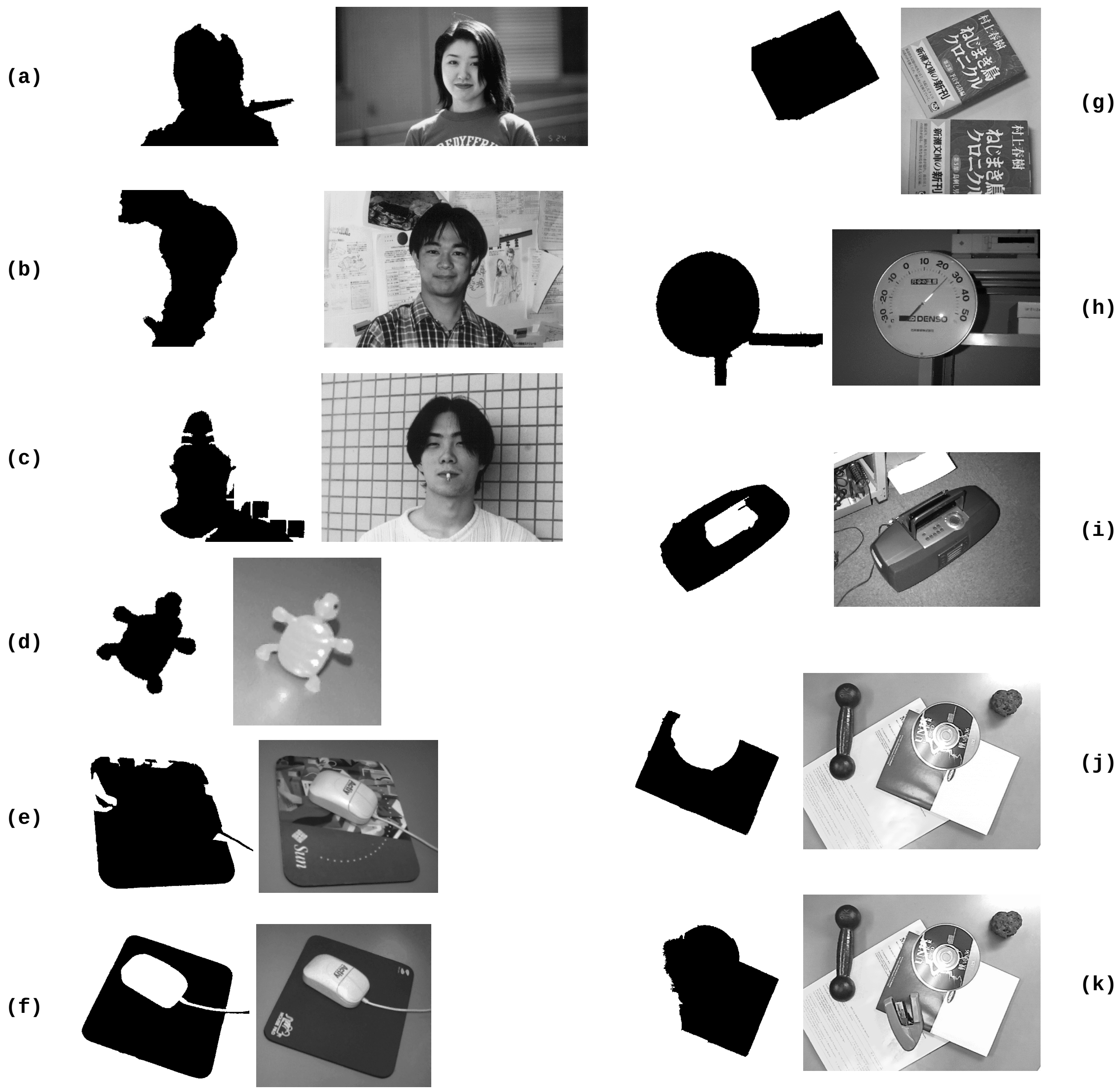

Figure 9. Other experimental results. Left image is the result extracted from each original image.