

Edge Grouping for Detecting Salient Boundaries with Sharp Corners

Joachim S. Stahl

Department of Computer Science
Clarkson University, Potsdam, NY 13676
jstahl@clarkson.edu

Abstract. The Gestalt law of continuity, or smoothness, is an important factor in guiding the edge grouping. In practice, however, the boundaries of many salient structures of interest may not be smooth everywhere and may contain some sharp corners. This paper presents a globally optimal edge-grouping method that can detect perceptually salient boundaries with sharp corners. In this method, a set of straight-line fragments are first detected from an image by classical edge detection and line approximation. Then a modified ratio-contour method is applied to detect the most salient closed boundary by identifying and connecting a subset of these line fragments into a closed curve. The major contribution comes from the introduction of corner information into the boundary-saliency definition. The corner information is derived from the image by the widely used Harris corner detector. Experiments on some real images show that the proposed method performs better than the original ratio-contour method, when the boundaries of the salient structures of interest in the image contain some sharp corners.

1 Introduction

Grouping, or *perceptual organization*, is a fundamental problem in computer vision. In this problem, a set of tokens are first detected from an image and then clustered into a small number of desirable groups [11, 14, 5]. This paper is concerned with a special category of grouping methods called *edge grouping*, where the grouping tokens are a set of line fragments and the goal is to identify and connect a subset of these fragments into a closed curve¹, which is expected to align well with the boundary of some perceptually salient structures. Edge grouping has attracted high interests in the past decades [1, 2, 4, 6, 9, 12, 13, 15–18, 20] for its capability to incorporate some important Gestalt laws, which have been shown by many psychophysical studies to be very important in the human visual system.

In edge grouping, the three most widely used Gestalt laws are closure, proximity, and continuity. *Closure* requires that the boundary be a cycle. *Proximity*

¹ In some prior works [13], edge grouping is a more general concept, which includes the cases of clustering fragments into groups other than a boundary. An edge grouping for closed boundaries is sometimes called a *contour grouping* [8].

requires the gap between two neighboring fragments to be small. *Continuity* requires the resulting boundary to be smooth. Boundary continuity is particularly important in edge grouping, because only considering closure and proximity may lead to very small or thin boundaries that are usually not perceptually salient [9]. However, while continuity is assumed in almost all prior work on edge grouping, the boundary of the structure of interest in various practical applications may not be smooth everywhere. On the contrary, the boundaries of many salient structures may contain some sharp corners with very high curvatures. Examples include both man-made structures, such as furniture, planes and weapons, and natural structures, such as leaves, fishes, and birds. With the assumption of the full boundary continuity, most prior edge-grouping methods have difficulty in extracting these structures accurately. However, these sharp corners can usually be detected from the original image using some corner detectors, although the detection results usually contain some errors, such as missing some real corners and detecting some false corners. The major goal of this paper is to incorporate the information of the detected corners into edge grouping so that salient structures can be detected even their boundaries contain some sharp corners.

The proposed method is based on the ratio-contour method [18], which is an edge-grouping method that well integrates all three important Gestalt laws of closure, proximity, and continuity. This method has several desirable properties. (a) It is guaranteed to produce closed boundaries. (b) It is guaranteed to find the globally optimal boundary in terms of the selected boundary-saliency measure. (c) It has no bias to produce shorter or longer boundaries, as in some prior works based on the shortest-path algorithm [4]. A comparison study on both synthetic and real images shows that the ratio-contour method performs more favorably than two other typical edge-grouping methods [18, 20, 12, 4]. In this paper, a way to incorporate the corner information into the ratio-contour-based edge grouping is developed, to detect salient boundaries with sharp corners.

Specifically, the corners are detected from the original image using the Harris corner detector [7], which has been widely used for its simplicity and reliability. Then the boundary-saliency measure used in the ratio-contour method is adapted in a way that the boundary fragments around an identified sharp corner are not counted into measuring the continuity of this boundary. This adaptation of the boundary-saliency measure does not vary its underlying form and its optimization (i.e., finding the optimal boundary with the maximum saliency) can still be achieved by the ratio-contour algorithm, a globally optimal graph algorithm developed in [18].

The remainder of this paper is organized as follows. Section 2 describes the problem of the edge grouping with the corner information. Section 3 briefly introduces the ratio-contour method. In section 4, the method to incorporate the corner information into edge grouping is presented. Section 5 reports the experimental results on a set of real images. These results are compared to the ones from the original ratio-contour method. Section 6 briefly concludes the paper.

2 Problem Description

As in all prior edge-grouping methods, the first step is to construct a set of line fragments from the image. This step is usually achieved by edge detection and line approximation. Figure 1 shows an example of the fragment construction in which the Canny edge detector [3] is used for edge detection and Kovesi's package [10] is used for line approximation. Note that, line approximation is performed on the edge detection output.

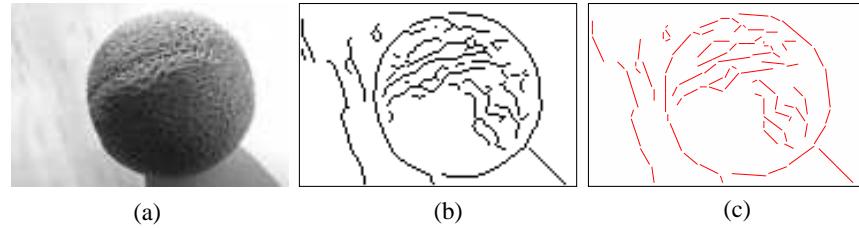


Fig. 1. An illustration of the fragment construction. (a) The input image, (b) Canny edge detection output, (c) line-approximation output.

Figure 2 illustrates the typical process of edge grouping on a set of line fragments. To avoid confusion, we refer to the line fragments constructed above as *detected fragments*. Based on them, a valid boundary is defined as a closed polygon that sequentially connects a subset of these fragments, as shown in Fig. 2(c). We can see that, in forming a boundary, a new set of line fragments (represented as dashed lines in Fig. 2) needs to be constructed to fill the gap between the neighboring detected fragments. We refer to these newly constructed fragments as *gap-filling fragments*. Without knowing the optimal boundary, we need to construct gap-filling fragments between each possible pair of the detected fragments, as shown in Fig. 2(b). In this setting, edge grouping can be defined as finding a valid closed boundary that has the maximum saliency (or the minimum cost), as shown in Fig. 2(c).

The most important step in edge grouping is the definition of the boundary saliency (or its negation, the boundary cost). Different edge-grouping methods usually define the boundary saliency in different ways and therefore, result in different optimization algorithms to find the most salient boundary. As mentioned above, prior edge-grouping methods usually consider all or some of the three important Gestalt laws: proximity, continuity, and closure in defining the boundary-saliency measure. In the above formulation, proximity means that the resulting boundary should contain fewer and shorter gap-filling fragments; closure means the boundary should be closed, which is always true in this formulation; continuity means that the resulting boundary should be as smooth as possible. We can see that the definition of continuity is nontrivial in this setting since the boundary is constructed as a polygon in this formulation. The quanti-

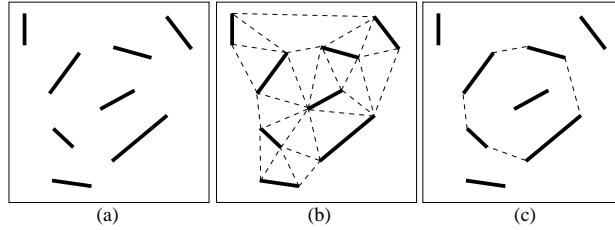


Fig. 2. An illustration of the edge-grouping process. (a) Detected fragments constructed from an image. (b) Constructing gap-filling fragments (dashed lines) to link detected fragments. (c) The salient boundary resulting from the grouping.

tative model of the boundary continuity will be discussed in detail in Section 3. However, intuitively, we know that a polygon is smooth if it does not contain sharp corners.

Figure 3 illustrates the problem addressed in this paper. By assuming the full continuity of resulting boundary, prior edge-grouping methods may produce a salient boundary as shown in Fig. 3(f), which misses the portions of the boundary with sharp corners. Some prior methods, such as [9], do not consider the boundary continuity at all. The problem of such methods is that they may produce many small or thin boundaries that are not perceptually salient [9]. To address this problem, we get back to the original image: a corner detector can be used to detect the sharp corners based on the intensity information in the original image, as illustrated in Fig. 3(g). Therefore, a very natural idea is to combine the corner information and continuity measure to achieve a more desirable edge grouping, as shown in Fig. 3(h). The basic idea is to remove the effect of a sharp corner in measuring the continuity of a boundary if this corner is detected from the original image. In this paper, this idea is implemented based on the ratio-contour method, which is briefly discussed in the next section.

3 Ratio-Contour Method

The ratio-contour method [18] is developed based on a setting that is slightly different from the one shown in Section 2. In the original ratio contour, both detected and gap-filling fragments are smooth curves and a gap-filling fragment always links the two neighboring detected fragments in G^1 -continuity, i.e., interpolating both locations and tangent directions at the fragment endpoints. This way, the resulting boundary is smooth everywhere and the boundary continuity, or smoothness, can be measured by its total curvature. In the ratio-contour method, a boundary B is parameterized according its arc length as $\mathbf{v}(t)$, $0 \leq t \leq |B|$, where $|B|$ is the length of the boundary B . Then the cost (negatively related to the saliency) of this boundary is defined as

$$\phi(B) = \frac{\int_B \sigma(t) dt + \lambda \cdot \int_B \kappa^2(t) dt}{|B|} \quad (1)$$

where $\sigma(t) = 1$ if $\mathbf{v}(t)$ is on a gap-filling fragment and $\sigma(t) = 0$, otherwise. $\kappa(t)$ is the curvature of the boundary at $\mathbf{v}(t)$. Clearly, the first term $\int_0^{|B|} \sigma(t)dt$ makes it biased to a boundary with longer detected fragments and shorter gap-filling fragments. This reflects the preference of better proximity. The second term $\int_0^{|B|} \kappa^2(t)dt$ reflects the favor of smoother boundaries, or better continuity. The denominator normalizes the cost by the boundary length $|B|$ to avoid a bias to shorter boundaries. $\lambda > 0$ is a regularization factor that balances the proximity and continuity in this cost. As in [18], A uniform $\lambda = 40$ is chosen for all the experiments.

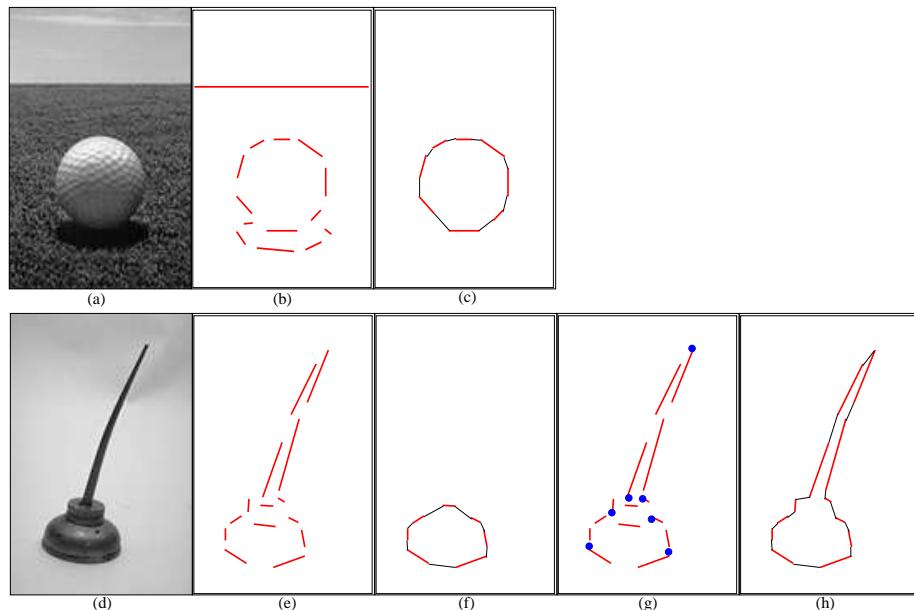


Fig. 3. An illustration of edge grouping with and without the corner information. (a) An image in which the structure of interest does not contain a sharp corner on its boundary. (b) Fragments constructed from (a). (c) The salient boundary detected from (b). (d) An image in which the structure of interest contains sharp corners on its boundary. (e) Fragments constructed from (d). (f) The salient boundary detected from (e) without considering the corner information. (g) Corners detected from the image shown in (d). (h) The salient boundary detected from (e) by incorporating the corner information.

In the ratio-contour method, a graph model is developed for finding the optimal closed boundary that globally minimizes the cost (1). Specifically, an *undirected* graph $G = (V, E)$ is constructed to model the fragments, where V is the vertex set and E is the edge set. In this graph, each vertex represents a fragment endpoint. For each detected fragment, a *solid* edge connecting the

corresponding two vertices is constructed. For each gap-filling fragment, a *dashed* edge connecting the corresponding two vertices is constructed. An example is shown in Fig. 2(b), where the constructed solid and dashed edges are represented by solid and dashed lines, respectively. In this graph, an *alternate* cycle is defined as a cycle that alternately traverses solid and dashed edges, as shown in Fig. 2(c). It is easy to see that a valid closed boundary corresponds exactly to an alternate cycle in G . Thus the problem of finding an optimal closed boundary from a set of fragments can be converted into the problem of finding an optimal alternate cycle in graph G .

To formulate the boundary cost (1) in the graph G , a weight $w(e)$ and a length $l(e)$ are defined for each edge e in G . For convenience, $B(e)$ is defined as the original fragment corresponding to the edge e . For each edge e ,

$$w(e) = \int_{B(e)} \sigma(t)dt + \lambda \cdot \int_{B(e)} \kappa^2(t)dt, \quad (2)$$

which is the un-normalized cost of $B(e)$, and

$$l(e) = |B(e)|,$$

which is the curve length of $B(e)$. The closed boundary with the minimum cost (1) then corresponds to an alternate cycle C that minimizes the *cycle ratio*

$$CR(C) = \frac{\sum_{e \in C} w(e)}{\sum_{e \in C} l(e)}.$$

In [19, 18], it was shown that such an optimal alternate cycle can be found from G in polynomial time in the worst case and developed such an algorithm.

However, in Section 2, fragments are constructed as line segments and boundaries as polygons, which are not smooth everywhere. Some minor adaptations are needed to apply the ratio-contour method to the problem formulated in Section 2. As shown in Fig. 2(c), the resulting boundary is a polygon that traverses detected and gap-filling fragments alternately. A polygon can easily be divided into a set of triple-segment polylines at the midpoints of the detected fragments. As shown in Fig. 4, the gap-filling fragment P_2P_3 links the detected fragments P_1P_2 and P_3P_4 on a boundary. Given the midpoints M_1 and M_2 on the two detected fragments, we can construct a triple-segment polyline $M_1P_2P_3M_2$, from which a smooth *arc* can be constructed using Bezier curves:

$$\mathbf{p}(u) = (1-u)^3 M_1 + 3u(1-u)^2 P_2 + 3u^2(1-u) P_3 + u^3 M_2,$$

with $0 \leq u \leq 1$. This way, a polygon boundary can be approximated by a closed curve that is smooth everywhere and we can apply the ratio-contour method mentioned above. To achieve this, only the following changes are needed:

1. The cost (1) needs to changed to

$$\phi(B) = \frac{\int_B \sigma(t)dt + \lambda \cdot \int_{B^s} \kappa^2(t)dt}{|B|}, \quad (3)$$

where B^s is the approximated smooth curve constructed from the closed boundary B .

2. The weight function defined in Eq. (2) needs to be changed to

$$w(e) = \begin{cases} 0 & \text{If } B(e) \text{ is a detected fragment} \\ \int_{B(e)} dt + \lambda \cdot \int_{B^s(e)} \kappa^2(t) dt & \text{If } B(e) \text{ is a gap-filling fragment.} \end{cases} \quad (4)$$

where $B^s(e)$ is the smooth arc constructed from the triple-segment polyline that contains the gap-filling fragment $B(e)$.

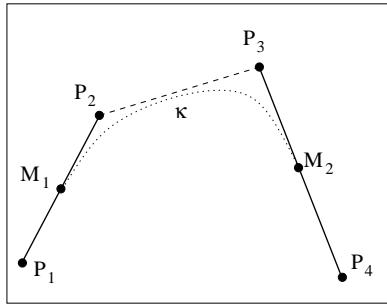


Fig. 4. An illustration of approximating a polygon boundary by a smooth curve.

4 Detecting Salient Boundaries with Sharp Corners

The problem of detecting salient boundaries with sharp corners is addressed by further adapting the boundary cost (3) to incorporate the corner information. Besides the fragments detected by edge detection and line approximation, a corner detector is applied to the original image to identify a set of corners, say Q_1, Q_2, \dots, Q_m ². Generally, a corner detector works by searching for pixels with abrupt intensity and direction changes on an image. It is expected that the detected corner points can align well with the sharp corners along structural boundaries. In practice, since real images usually contain various kinds of noise, the detected corners usually contain some errors: some real corners may be missed and some false ones may be detected. the edge-detection results and the corner-detection results are expected to complement each other to improve the edge grouping.

To consider the corner information, the boundary cost (3) is adapted to

$$\phi(B) = \frac{\int_B \sigma(t) dt + \lambda \cdot \int_{B^s} \kappa^2(t) \cdot \psi(t) dt}{|B|}, \quad (5)$$

² It is well known that most edge detectors perform very poorly at sharp corner. Therefore, it is usually unfeasible to figure out the corners from an edge-detection output.

where $\psi(t) = 1$ if there is no detected corner ($Q_i, i = 1, 2, \dots, m$) located within a neighboring area around the arc that contains $\mathbf{v}(t)$, and $\psi(t) = \alpha \in [0, 1]$, otherwise. The parameter α reflects the impact of the sharp corner on the boundary cost. A smaller α indicates higher impact, i.e., the boundary curvature around the detected corners is counted less towards the boundary cost. In this paper, $\alpha = 0$ for all the experiments.

Accordingly, the edge weight needs to be updated in Eq. (4) to

$$w(e) = \begin{cases} 0 & \text{If } B(e) \text{ is a detected fragment} \\ \int_{B(e)} dt + \lambda \cdot \int_{B^*(e)} \kappa^2(t) \cdot \psi(t) dt & \text{If } B(e) \text{ is a gap-filling fragment.} \end{cases}$$

Obviously, with this updated weight function, the boundary cost (5) can still be optimized by the ratio-contour algorithm based on the same graph model.

The only remaining problem is to define the search area for locating the nearby detected corners. As illustrated in Fig. 5(a), for a gap-filling fragment P_2P_3 , a search for corner points is done inside the circular region with the fragment P_2P_3 as the diameter. If a corner Q is located in this circular region, then $\psi(t) = \alpha = 0$ for this fragment, as mentioned above. In addition, the gap-filling fragment P_2P_3 is updated to a two-segment polyline P_2QP_3 , as shown in Fig. 5(b). This makes the boundary that traverses the gap between P_2 and P_3 to pass through the corner point Q . If there are multiple corners located in the search area, simply pick the “strongest” one that is usually determined in the corner detection.

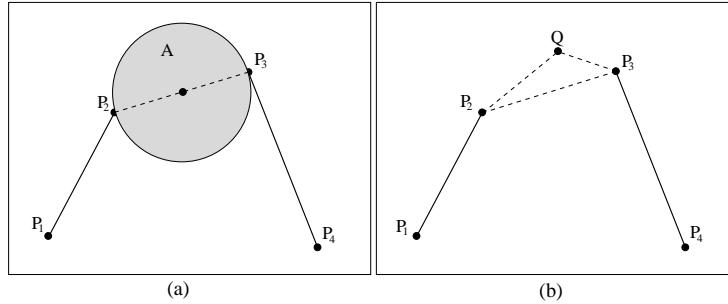


Fig. 5. An illustration of the search for corner points around a gap-filling fragment. (a) Circular search area. (b) Updating the gap-filling fragment to make it pass through the nearby corner.

5 Experiments

In the implementation, the ratio-contour package developed in [18] is used. For edge detection, the Canny edge detector (with default parameters) is used, provided in the Image Processing Toolbox (Version 5.0.2) in MATLAB 7.0.4.352

(R14 with SP2). Short edges are filtered, only keeping the 20 longest ones. For line approximation, the line-approximation package provided by Peter Kovesi [10] is used. In using this package, the allowed maximum deviation from an edge pixel to its approximated line segment is set to 2 pixels. For corner detection, the Harris corner detector [7] is used, in which the Gaussian-smoothing factor is set to 2, the thresholding value to 1000, and the radius for non-maximal suppression to 2. Figure 6 illustrates the step-by-step operations of the Harris corner detector. It first determines for each pixel in the image the likelihood of it being a corner, and then assigns to each pixel a *corner strength* according to this likelihood (See Fig. 6(b)). Finally a combination of thresholding and non-maxima suppression is used to determine the corner points. In the corner detection, only the 30 strongest corners are kept for further edge grouping in all the experiments, if the corner detector produces more than 30 corners. As mentioned above, the regularization parameter λ in the cost (5) is set to 40 in all the experiments. Also all images larger than 250×190 are resized to be as close as possible to this size, while maintaining the aspect ratio.

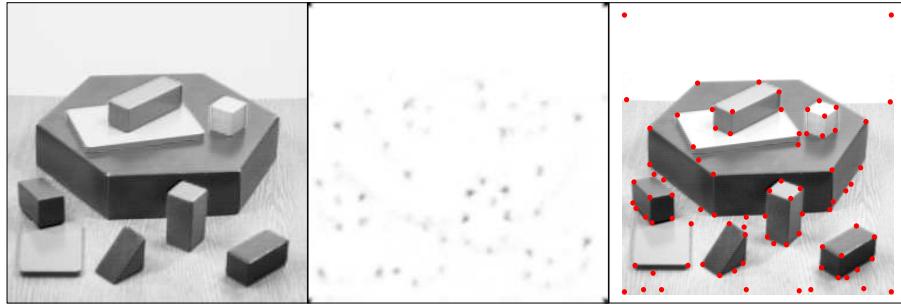


Fig. 6. The step-by-step operations of the Harris corner detector. From left to right are the original image, the corner-strength map (darker means a stronger likelihood of being a corner), and the detected corners, respectively.

Figure 7 shows the most salient boundaries detected from 10 real images using the proposed edge-grouping method. For comparison, also the results from the original ratio contour method are shown, without considering the corner information. We can see that the proposed method can produce boundaries that are better aligned with the salient structures, if the boundary of the salient structures contain some sharp corners. For example, in image (i), the quill contains a very sharp corner close to the left side of the image. This corner is successfully detected by the Harris corner detector (see column 3), facilitating the accurate detection of the whole boundary of the quill. On the other hand, the original ratio-contour method, with no corner information, fails to detect the boundary correctly.

As mentioned above, corner detection is not perfect and may produce errors. If an important corner on the desired boundary is missed in corner detection,

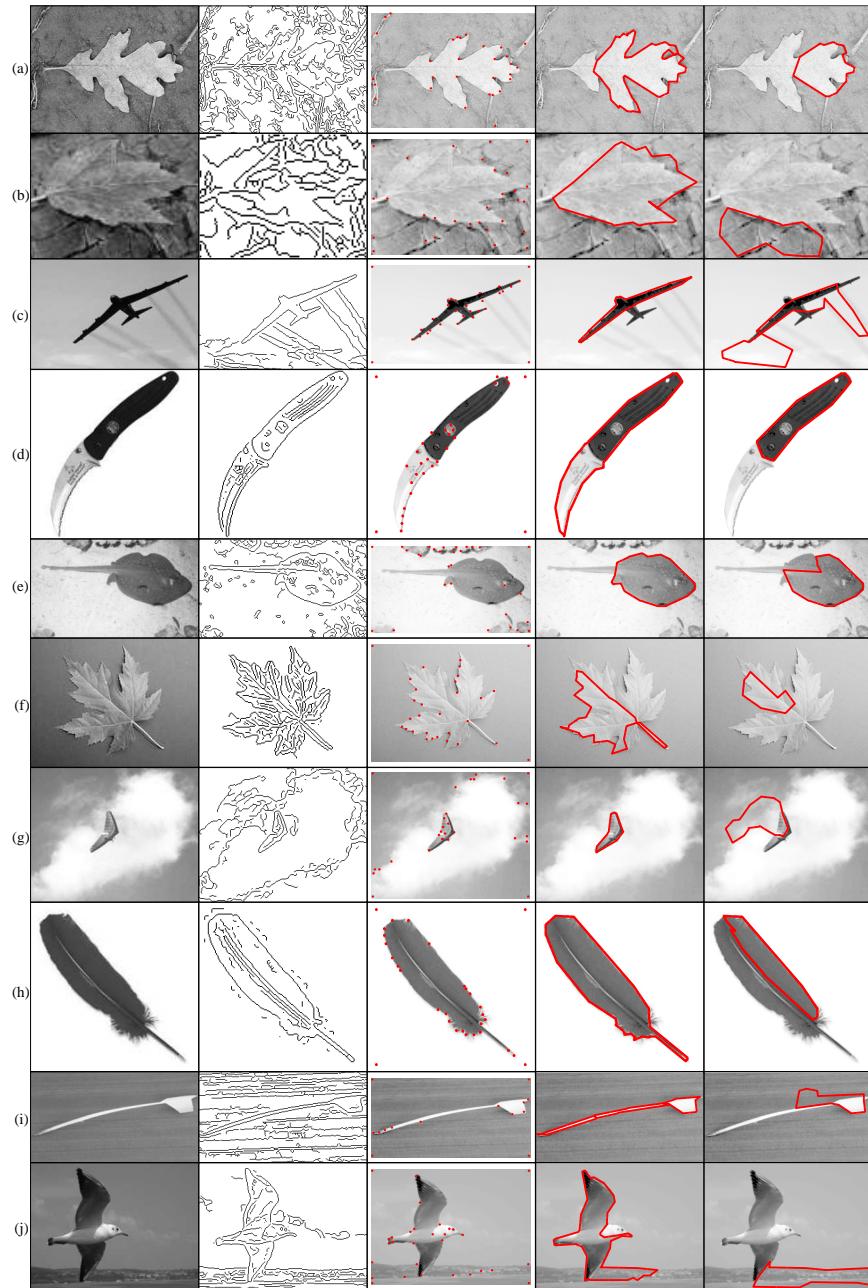


Fig. 7. Edge-grouping results on 10 real images. First column, original images; second column, Canny edge detection outputs; third column, Harris corner detector outputs; fourth column, boundaries detected by the proposed method; fifth column, boundary detection by the original ratio-contour method without considering the corner information.

the proposed method may not detect the portion of the boundary around this corner. For example, the tip of the tail of the fish shown in the image (e) and the apex of the leaf shown in image (f) are all important corner points. However, they are not detected by the corner detector and therefore, the proposed method fails to detect the whole desired boundary. On the other hand, some detected corners are not real corners, which may also affect the edge-grouping results. For example, some undesired corners are detected in the background in image (j), which leads to a boundary that contains fragments from both the foreground bird and the background. It is expected that a more reliable corner detector will improve the performance of the proposed edge-grouping method.

6 Conclusions

Boundary continuity, or smoothness, is a very important factor in edge grouping. The incorporation of this factor can avoid the detection of overly small and thin structures that may come from the noise or only cover part of a salient structure. However, assuming the full boundary smoothness also prevents many edge-grouping methods from correctly detecting salient structures with sharp corners. In this paper, an edge-grouping method was developed that combines the continuity factor and the corner information. This combination is achieved by adapting the boundary saliency defined in the ratio-contour method. The basic idea is to not count the boundary curvature around the detected sharp corners when measuring the boundary continuity. Experiments were conducted on some real images and the obtained results show that the proposed method can detect salient boundaries with sharp corners, while the original ratio contour has difficulty to do so. Experiments also show that the accuracy and reliability of the corner detection is important to the proposed method. Application and testing of more advanced corner detectors to the proposed method will be done in future work

References

1. T. Alter and R. Basri. Extracting salient contours from images: An analysis of the saliency network. In *International Journal of Computer Vision*, pages 51–69, 1998.
2. A. Amir and M. Lindenbaum. A generic grouping algorithm and its quantitative analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(2):168–185, 1998.
3. J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.
4. J. Elder and S. Zucker. Computing contour closure. In *European Conference on Computer Vision*, pages 399–412, 1996.
5. D. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Upper Saddle River, NJ: Prentice Hall, 2003.
6. G. Guy and G. Medioni. Inferring global perceptual contours from local features. *International Journal of Computer Vision*, 20(1):113–133, 1996.

7. C. J. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, pages 147–151, 1988.
8. A. K. J. H. Elder and L. A. Johnston. Contour grouping with prior models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(25):661–674, 2003.
9. D. Jacobs. Robust and efficient detection of convex groups. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(1):23–27, 1996.
10. P. D. Kovesi. MATLAB functions for computer vision and image analysis. School of Computer Science & Software Engineering, The University of Western Australia. Available from: <<http://www.csse.uwa.edu.au/~pk/research/matlabfns/>>.
11. D. G. Lowe. *Perceptual Organization and Visual Recognition*. Boston: Kluwer Academic Publishers, 1985.
12. S. Mahamud, L. R. Williams, K. K. Thornber, and K. Xu. Segmentation of multiple salient closed contours from real images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(4):433–444, 2003.
13. S. Sarkar and K. Boyer. Quantitative measures of change based on feature organization: Eigenvalues and eigenvectors. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 478–483, 1996.
14. S. Sarkar and K. L. Boyer. *Computing Perceptual Organization in Computer Vision*. Singapor: World Scientific, 1994.
15. A. Shashua and S. Ullman. Structural saliency: The detection of globally salient structures using a locally connected network. In *International Conference on Computer Vision*, pages 321–327, 1988.
16. J. S. Stahl and S. Wang. Edge Grouping Combining Boundary and Region Information. *IEEE Transaction on Image Processing*, pages 2590–2606, 2007.
17. J. S. Stahl and S. Wang. Globally Optimal Grouping for Symmetric Closed Boundaries by Combining Boundary and Region Information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 395–411, 2008.
18. S. Wang, T. Kubota, J. Siskind, and J. Wang. Salient closed boundary extraction with ratio contour. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005.
19. S. Wang, T. Kubota, and J. M. Siskind. Salient boundary detection using ratio contour. In *Neural Information Processing Systems Conference*, pages 1571–1578, 2003.
20. L. Williams and K. K. Thornber. A comparison measures for detecting natural shapes in cluttered background. *International Journal of Computer Vision*, 34(2/3):81–96, 2000.