From Fragments to Salient Closed Boundaries: An In-Depth Study

Song Wang, Jun Wang, and Toshiro Kubota
Department of Computer Science and Engineering
University of South Carolina, Columbia, SC 29208
songwang@cse.sc.edu, wang286@cse.sc.edu, kubota@cse.sc.edu

Abstract

This paper conducts an in-depth study on a classical perceptual-organization problem: finding salient closed boundaries from a set of boundary fragments detected in a noisy image. In this problem, a saliency boundary is formed by identifying and connecting a subset of fragments according to the simple Gestalt laws of closure, continuity, and proximity. Our specific interest is focused on the methods that aim to achieve boundary closure, an important global property of perceptual salient boundaries. In this paper, we analyze and compare three such methods that are developed in recent years: (a) Elder and Zucker's method based on the shortest-path algorithm, (b) Williams and Thornber's method combining the spectral-analysis and the strongly-connected-component algorithms, and (c) Wang, Kubota, and Siskind's method based on ratio-contour algorithm. Both theoretic analysis and experimental study show that, with a unified setting of fragment saliency, Wang, Kubota, and Siskind's method more appropriately constrains the search space for the closed boundaries, and usually produces better performance than or at least comparable performance as the other two methods. Particularly, Wang, Kubota, and Siksind's method can always guarantee the boundary closure and simplicity, which may not be always hold in the other two methods. We construct and collect a variety of synthesized and real images for this comparison.

1. Introduction

Extracting perceptually salient boundaries from noisy images is a fundamental problem in computer vision. According to Gestalt laws, *closure*, *proximity*, and *continuity* are three critical properties in measuring the boundary saliency. These properties can be modelled more conveniently with a set of pre-extracted boundary fragments than with image pixels. The boundary fragments, or simply *fragments*[1], can be obtained by applying some edge-detection

---

[1]Most literatures use the terminology *edge* instead of *fragment*. However, in this paper *edge* has other specified meaning in a graph model.

algorithms on the input image. From these fragments, the boundary-extraction problem is to identify and connect a subset of fragments for a salient boundary. In this formulation, *Closure* means that the resulting boundary should be closed and fully partition an object out of the background. *Proximity* means that the gap between two neighboring fragments should be small. *Continuity* means that the resulting boundary should be smooth.

Most methods follow three steps in designing their boundary-extraction algorithms. (a) Define a local saliency for each fragment and/or gap between two neighboring fragments, usually based on the smoothness and length of the involved fragments and/or gaps, to encode the local continuity and proximity properties. (b) Define a saliency measure for each valid boundary in terms of the local saliency defined in step (a). For example, we can measure the saliency of a boundary by summing up the local saliency of all the fragments and gaps along this boundary. (c) Develop an optimization algorithm to find the boundary that maximizes the selected boundary-saliency measure. Graph models are useful in this problem formulation and algorithm design [5, 12, 9].

As a global property, boundary closure is usually more difficult to be encoded into the saliency measure than the local properties of proximity and continuity. An intuitive way to achieve boundary closure is to impose it as a hard constraint: i.e., the search space only consists of closed boundaries. With this constraint, the boundary-extraction problem is converted into an optimal cycle-finding problem in a graph. Elder and Zucker [5] define the boundary saliency as the product of local saliency along the boundary and then use the shortest-path algorithm to find the optimal one. Wang, Kubota and Siskind [12] define a new boundary-saliency measure with a normalization over the boundary length, and present a ratio-contour algorithm to find the optimal cycle in polynomial time. Williams and Thornber [14, 9] develop a spectral-analysis algorithm to encode the global boundary-closure property into the local saliency of the fragments. This technique was compared with five other relevant methods [11, 7, 10, 6, 13] with improved performance in enhancing the local saliency of many

fragments along the desired closed boundaries. With this saliency enhancement, a strongly-connected-component algorithm [9] is developed to extract the final closed boundaries. Other related work includes Alter and Basri [1], Jacobs [8], and Amir and Lindenbaum [2].

This paper conducts an in-depth study on the methods of computing boundary closure by analyzing and comparing the three typical methods mentioned above: Elder and Zucker's method (EZ) [5], Williams and Thornber's method (WT) [14, 9], and Wang, Kubota, and Siskind's method (WKS) [12]. Our interesting findings include: (a) EZ has a bias to produce boundaries with fewer fragments while WT and WKS do not have such a bias, (b) the boundary-saliency measures used in WT and WKS are very similar to each other, while WKS further prunes the search space to guarantee the boundary closure, (c) EZ and WKS find the real global optima in terms of the selected saliency measure, while the algorithms in WT are only related to its boundary-saliency measure in some special cases [13], (d) WT may produce boundaries that are not closed or simple, while WKS does not, (e) the spectral-analysis algorithm used in WT can enhance the local saliency of many fragments along the most salient boundary, but this enhancement does not improve the performance of WKS in a noticeable way. In general, with a unified setting of local saliency, WKS usually performs better than or at least comparably to WT and EZ.

This paper is organized as follows. In Section 2, we give a brief introduction to WKS, WT and EZ. In Section 3, we analyze the intrinsic similarities and differences of them. Section 4 provides an experimental comparison for evaluation, followed by a brief conclusion in Section 5.

2. Background

2.1. Problem Formulation

As shown in Fig. 1(a), we start with a set of non-crossing fragments, each of which is a continuous open curve segment with two endpoints. Our goal is to identify and connect a subset of fragments into a closed boundary with a maximum perceptual saliency as shown in Fig. 1(b). To achieve a closed boundary, another set of fragments (dashed curves in Fig. 1(b)) are constructed to fill the gap between initial fragments. To distinguish these two kinds of fragments, we name the initial fragments (shown in Fig. 1(a)) *real* fragments and those gap-filling ones *virtual* fragments. A virtual fragment is usually derived from its two neighboring real fragments using some curve-completion techniques. Associated with each fragment, we define a *fragment saliency* (or *fragment cost*) that describes its likeliness (or unlikeness) to be along a salient closed boundary according to some local information. There are various ways to define the fragment saliency (or fragment cost)

[5, 13, 12]. Most of them are subject to the Gestalt laws of proximity and continuity: (a) real fragments are more salient than virtual fragments, (b) the longer a virtual fragment, the less its local saliency, and (c) the smoother a fragment, the larger its local saliency.
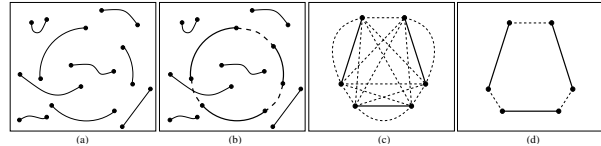

Figure 1. An illustration of extracting a salient boundary from some fragments: (a) real fragments, (b) salient closed boundary by connecting some real/virtual fragments, (c) a solid-dashed graph G, and (d) an alternate cycle in G.

A closed boundary can then be constructed by connecting a subset of real fragments and virtual fragments, sequentially and alternately, as shown in Fig. 1(b). We are only interested in *simple* boundary, where no fragments is traversed more than once. In the remainder of the paper, all boundaries without a qualifier 'nonsimple' always mean simple boundaries. Associated with each closed boundary, we can define a *boundary saliency $S(\cdot)$* (or its negation, a *boundary cost $L(\cdot)$*) as a function of the fragment saliency (or fragment cost) of all the involved fragments. Finally, we need to develop an optimization algorithm to search for a closed boundary that maximizes the boundary saliency or minimize the boundary cost. In essence, various boundary-extraction methods differ mainly in their boundary-saliency/cost definitions and/or the optimization algorithms.

2.2. Wang, Kubota & Siskind's Method (WKS) [12]

In WKS, the problem is formulated in an undirected graph $G = (V, E)$. A unique vertex is constructed for each fragment endpoint. Two different kinds of edges, *solid* edges and *dashed* edges, are then constructed between vertices for representing real fragments and virtual fragments, respectively. An example is shown in Fig. 1(c), which consists of 3 solid edges and all 15 possible dashed edges. Graph G always has even number of vertices, as each real fragment has two endpoints. Furthermore, no two solid edges are incident from the same vertex. In [12], such a graph is called an (undirected) *solid-dashed (SD)* graph. In this graph, an *alternate cycle* is defined as a cycle that traverses solid edges and dashed edges alternately and a *simple* cycle as a cycle that does not traverse a vertex more than once. In the remainder of this paper, all cycles without a qualifier 'nonsimple' always mean simple cycles. Examples of a SD graph and an alternate cycle are illustrated in

Fig. 1(c) and (d). It is easy to see that simple closed boundaries are exactly modelled by alternate cycles in G.

In WKS, a weight function $w(e)$ is defined for each edge e to describe the fragment cost of $\Gamma(e)$, the fragment modelled by e. In [12], the edge weight is defined as $w(e) \triangleq \int_{\Gamma(e)} [\sigma(e) + \kappa^2(t)] dt$, where $\sigma(e) = 0$ for solid edges and $\sigma(e) = 1$ for dashed edges. $\kappa(t)$ is the curvature along the fragment. Furthermore, let $l(e)$ be the length of the fragment $\Gamma(e)$. For the closed boundary modelled by the alternate cycle $C \subset G$, the boundary cost is defined as

$$L_{wks}(C) = \frac{\sum_{e \in C} w(e)}{\sum_{e \in C} l(e)}. \tag{1}$$

In [12], a polynomial-time ratio-contour algorithm, which consists of three main reductions, is developed to search for an optimal alternate cycle C^* with minimum cost $L_{wks}(C)$, and it provides the most salient closed boundary.

2.3. Williams & Thornber's Method(WT) [14, 9]

In WT, the problem is formulated in a *directed* graph $\hat{G} = (\hat{V}, \hat{E})$ and the spectral graph theory [3] is applied to encode the global boundary-closure property into the local fragment saliency. All real fragments are simplified to be straight line segments with infinitesimal length. Therefore, each real fragment is determined by its location and its slope angle. In the graph construction, each real fragment is represented by two separate vertices i and $\bar{i}$ to cover the two possible directions of this real fragment in the resulting closed boundary. Each virtual fragment is then represented by two directed edges as shown in Fig. 2(b). Associated with each (directed) edge (i, j) is a weight $p_{ij} \in (0, 1)$ that is the fragment saliency of the virtual fragment connecting i to j sequentially[2]. The affinity matrix $\mathbf{P} = [p_{ij}]_{2n \times 2n}$ is not symmetric, but time-reversal symmetric, i.e., $p_{ij} \neq p_{ji}$, but $p_{ij} = p_{\bar{j}\bar{i}}$, where n is the number of real fragments.

The fragment saliency p_{ij} in WT is estimated using the stochastic-completion fields [13], which also models the proximity and continuity between two real fragments. Fragment saliency associated with real fragments is ignored initially. To encode the boundary-closure property, the affinity matrix $\mathbf{P} = [p_{ij}]_{2n \times 2n}$ is updated to $\mathbf{C} = [c_{ij}]_{2n \times 2n}$ using the eigenvalue decomposition of $\mathbf{P}$. In [14], it is shown that many fragments along the salient boundaries are greatly enhanced in the new affinity matrix $\mathbf{C}$. To finally achieve a closed boundary, a strongly-connected-component algorithm is used to find a directed cycle $\hat{C} \subset \hat{G}$ that traverses the edges with large fragment saliency c_{ij} [14]. The update from $\mathbf{P}$ to $\mathbf{C}$ is also related to maximizing the boundary

Figure 2. An illustration of the directed graph $\hat{G}$ used in WT and EZ: (a) initial real/virtual fragments and SD graph G, (b) directed real/virtual fragments, and (c) the resulting directed graph $\hat{G}$ equivalent to (a). This figure was adapted from Fig. 3 in [14].

saliency

$$S_{wt}(\hat{C}) = \left(\prod_{(i,j) \in \hat{C}} p_{ij} \right)^{\frac{1}{|\hat{C}|}}, \tag{2}$$

but only in some special cases [13]. Here $|\hat{C}|$ is the number of edges in cycle $\hat{C}$.

2.4. Elder & Zucker's Method (EZ) [5]

In EZ, the graph structure is the same as the one used in WT, i.e., $\hat{G} = (\hat{V}, \hat{E})$ where vertices represent real fragments and edges represent virtual fragments. Also similar to WT, a fragment saliency $p_{ij} \in (0, 1)$ is defined as weight for each edge, by combining the properties of proximity, continuity, and the neighboring-area's intensity distribution. A closed boundary is also modelled by a directed cycle $\hat{C}$, but the boundary saliency is defined as

$$S_{ez}(\hat{C}) = \prod_{(i,j) \in \hat{C}} p_{ij}, \tag{3}$$

and the final goal is to find an optimal cycle $\hat{C}^*$ that maximizes the boundary saliency (3). Taking a logarithm on (3), we can see that $\hat{C}^*$ should minimize the boundary cost

$$L_{ez}(\hat{C}) = \sum_{(i,j) \in \hat{C}} |\log p_{ij}|, \tag{4}$$

as $p_{ij} \in (0, 1)$. If we redefine the edge weight as $\hat{w}_{ij} = |\log p_{ij}|$, $\hat{C}^*$ is then a (directed) cycle with the minimum total weights. This optimization problem can be solved in polynomial time using the classic shortest-path algorithms like Dijkstra's algorithm [4].

3. Analysis

In this section, we analyze some relations among these three methods in terms of their choices of local saliency,

---

[2]In both WT and EZ, p_{ij} is named as transition probability to reflect that edges are directed. However, it does not satisfy the strict definition of transition probability in Markov networks, because $\sum_j p_{ij}$ may not be 1.
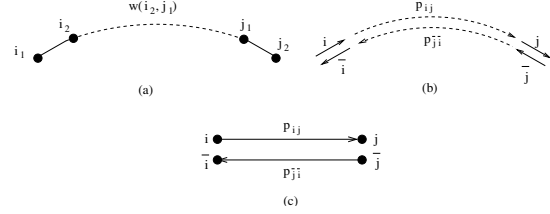
graph models, search spaces, boundary saliency, and optimization algorithms. Based on this analysis, we choose a unified setting for the experimental comparison, which will be reported in Section 4.

First, the structures of the undirected SD graph G used in WKS and the directed graph $\hat{G}$ used in WT and EZ are actually equivalent to each other as described in Section 2.3. Real fragments are represented by solid edges in G and vertices in $\hat{G}$, while virtual fragments are represented by dashed edges in G and directed edges in $\hat{G}$. Both graphs fully describe the real and virtual fragments and their connection relations.

Second, there are some intrinsic similarities among the edge-weight functions used in these three methods. To represent fragment saliency or cost, all three methods define the edge-weight functions based on the Gestalt laws of proximity and continuity, although different formulae are used to measure them. In WKS, edge weight $w(\cdot)$ represents fragment cost that is negatively related to the fragment saliency p_{ij} used in WT and EZ. These two can be unified by setting $w(i_2, j_1) = |\log p_{ij}|$, because undirected dashed edge (i_2, j_1) in G and the directed edge (i, j) in $\hat{G}$ model the same virtual fragment, as shown in Fig. 2. Another difference is that the saliency of the real fragments is explicitly considered in WKS, but not in EZ and WT. Finally, in WKS, the length of the fragments, or $l(e)$, is explicitly used to measure the boundary length, while the number of edges, or $|\hat{C}|$, is used to measure the boundary length in WT and EZ. To achieve a fair comparison only in terms of their boundary saliency and optimization algorithms, we use the following unified setting in constructing G for WKS and $\hat{G}$ for WT and EZ.

Unified Setting:

1. The initial real fragment are preprocessed so that all of them are of infinitesimal length and represented by its location and its slope angle.

2. Calculate the edge weights p_{ij} in $\hat{G}$ using the stochastic-completion-field method with parameters $\gamma = 0.15$, $T = 0.004$ and $\tau = 5.0$ [13, 9].

3. Calculate the dashed-edge weights in G using $w(i_2, j_1) = |\log p_{ij}|$.

4. Set $l(e) = 1$ for all dashed edges in G and $w(e) = 0, l(e) = 0$ for all solid edges in G.

Third, the search space employed in WKS is different from the one employed in WT and EZ. In WKS, the search space $\mathcal{B}$ is made up of the boundaries corresponding to all the alternate (simple) cycles $C \subset G$, which exactly cover all the simple closed boundaries we are interested in. In WT and EZ, the search space $\hat{\mathcal{B}}$ consists of boundaries corresponding to all directed cycles $\hat{C} \subset \hat{G}$. Because $\hat{C}$ traverses edges and vertices alternately in $\hat{G}$, the resulting boundary also traverses real and virtual fragments alternately, as in WKS method. However, a directed simple cycle $\hat{C} \subset \hat{G}$ may traverse both edges (i, j) and $(\bar{j}, \bar{i})$, which in fact represent the same virtual fragment. Therefore, $\hat{\mathcal{B}}$ contains strongly-connected components, as shown in Fig. 3(a), which correspond to nonsimple boundaries, as shown in Fig. 3(c). In the undirected SD graph G, such a nonsimple boundary is represented by a nonsimple alternate cycle that may traverse a vertex or an edge more than once. From this analysis, we see that $\mathcal{B} \subset \hat{\mathcal{B}}$ is a tighter search space covering exactly all the simple closed boundaries.

To see whether WT can guarantee to select a cycle corresponding to a simple boundary, we further study closely the strongly-connected-component algorithm of WT [9]. In essence, this algorithm starts from a vertex $k \in \hat{G}$ and locally searches for a cycle $\hat{C}_k$. Then it starts from $\bar{k}$ and searches for another optimal cycle $\hat{C}_{\bar{k}}$, from which a new cycle $\hat{C}'_{\bar{k}}$ is constructed as its time-reversal version, i.e., $\hat{C}'_{\bar{k}} = \{\bar{i} | i \in \hat{C}_{\bar{k}}\}$. Finally, the subgraph $\hat{C}_k \cap \hat{C}'_{\bar{k}}$ is returned for deriving boundaries. While both $\hat{C}_k$ and $\hat{C}'_{\bar{k}}$ may correspond to nonsimple boundaries in $\hat{\mathcal{B}}$, it is expected that $\hat{C}_k \cap \hat{C}'_{\bar{k}}$ always corresponds to a simple closed boundary. However, this algorithm may result in a boundary that is not closed. For example, for vertices k and $\bar{k}$, and cycle $\hat{C}_k$ shown in Fig. 3(a), $\hat{C}_k$ contains both k and $\bar{k}$ and $\hat{C}_k$ and $\hat{C}_{\bar{k}}$ are the same cycle. In this case, the subgraph $\hat{C}_k \cap \hat{C}'_{\bar{k}}$ corresponds to an open curve segment as shown in Fig. 3(c). In the next section, we will see such results in the experiments using WT.
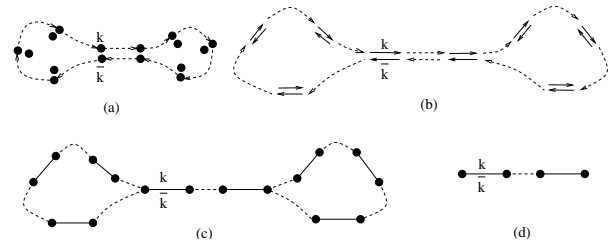

Figure 3. An illustration of invalid boundaries produced by WT: (a) a directed cycle in $\hat{G}$ that traverses both k and $\bar{k}$, (b) the same cycle represented by directed fragments, (c) the nonsimple boundary corresponding to the cycle in (a) and (b), and (d) an open curve segment produced by the strongly-connected-component algorithm of WT.

Fourth, the boundary saliency measures used in WT and WKS are similar while the measure used in EZ is different from the other two. The difference of the boundary saliency used in EZ and WT is obvious from (3) and (2):

EZ uses the product of the fragment saliency while WT takes the geometric mean of this product. As mentioned in [14], EZ's boundary saliency decreases quickly with the increasing of the number of fragments along the boundary. Therefore, it can be expected that EZ favors a shorter boundary, or a boundary with less number of fragments. For WT and WKS, their boundary saliency are very similar. As mentioned in Section 2.4, we can redefine the edge weight $\hat{w}_{ij} = |\log p_{ij}|$ in directed graph $\hat{G}$ to represent fragment cost. Then the cycle $\hat{C}^*$ maximizing the boundary saliency (2) is exactly the one minimizing the boundary cost

$$L_{wt}(\hat{C}) = \frac{\sum_{(i,j)\in\hat{C}} \hat{w}_{ij}}{|\hat{C}|}, \qquad (5)$$

which is of the same form as the WKS boundary cost (1), under the unified setting mentioned above. We can see that $L_{wt}(\hat{C}) = L_{wks}(C)$ if $C \subset G$ and $\hat{C} \subset \hat{G}$ model the same simple closed boundary. From this, we expect that the performance of WT and WKS to be similar to each other. However, we have shown that the search space $\mathcal{B}$ in WKS is a subset of the search space $\hat{\mathcal{B}}$ in WT. Therefore, even with the unified setting, the results from WT and WKS may still be different.

Finally, EZ and WKS are guaranteed to find the optimal solutions in their respective search spaces while WT is not. In WKS, a ratio-contour algorithm is developed to find a cycle C^* minimizing L_{wks} in polynomial time, with a provable global optimality. Similarly in EZ, a shortest-path algorithm is used to find a globally optimal cycle minimizing L_{ez} in polynomial time. In WT, however, a local-search-based algorithm is developed to find a strongly-connected component in $\hat{G}$. This algorithm is not shown to minimize the boundary cost L_{wt} globally. Only the fragment-saliency update from **P** to **C** is related to the boundary cost L_{wt} in some special cases [14]. In the next section, we conduct some experiments to further compare these three methods.

4. Experiments

Without specific declarations, all the experiments reported in this section are conducted under the unified setting mentioned in Section 3. Virtual fragments are produced using the same stochastic-completion field [13] as one used for estimating fragment saliency p_{ij} in the unified setting.

4.1. Toy Examples

We first test the methods using two toy examples, where the desired boundaries are circles. In the first example, a circle of radius 10 is located in the center of the image. The circle is represented by a set of 16 real fragments placed uniformly along the circumference. While all the real fragments are of infinitesimal length in the unified setting, each
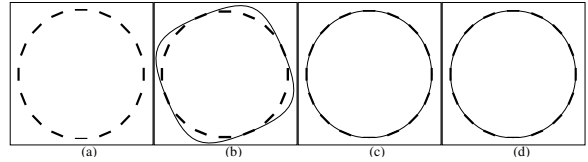

Figure 4. A toy example of boundary extraction: (a) real fragments, (b-d) boundary-extraction results from EZ, WT, and WKS, respectively.

of them is displayed as a straight line segment with short length in the figures so that we can identify both locations and slope angles. The results are shown in Fig.4, from which we see that both WT and WKS extract all 16 real fragments along the circle while EZ only extracts 4 of them. This toy example clearly shows that EZ favors boundaries with less number of fragments.
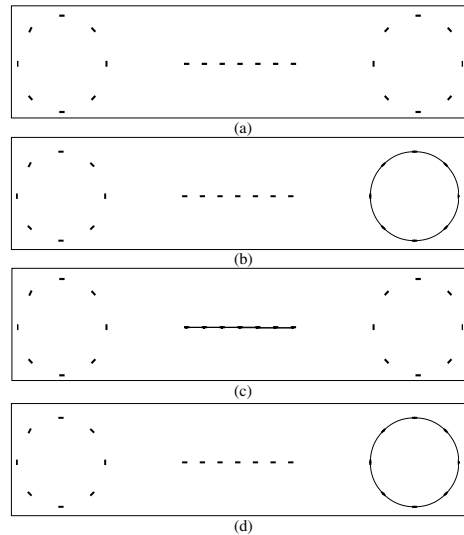

Figure 5. Another toy example of boundary extraction: (a) real fragments, (b-d) boundaries extracted using EZ, WT, and WKS, respectively.

The second toy example demonstrates that WT does not always guarantee boundary closure and simplicity. As shown in Fig. 5(a), two circles are located in the image and each of them is sampled by 8 real fragments. Another set of 7 real fragments is placed along a straight line between these two circles. Fig.(b), (c), and (d) show the most salient boundaries extracted using EZ, WT, and WKS, respectively. We see that, while EZ and WKS extract one of the circles as the most salient boundary, WT extracts an open line segment connecting the 7 real fragments lying between the two circles. This experiment verifies the analysis on search space in Section 3.

4.2. Synthesized Data

We also use the some synthesized data in [14] to test these three methods. In those data, the input real fragments contain two patterns. One is the salient pattern extracted from one of the five fruits: apple, banana, lemon, pear, and tamarillo, and the other is a noisy pattern extracted from one of the five textures: water, wood, sand, stone, and fabric. All 25 combinations of a salient pattern and a noisy pattern are tested. We found that WT and WKS successfully extracted the salient patterns in all 25 images, while EZ failed in all of them. Four examples are shown in Fig. 6, where each row illustrates an example. It is clear that the main problem of EZ comes from its bias toward boundaries with fewer fragments or, in many cases, shorter boundaries.
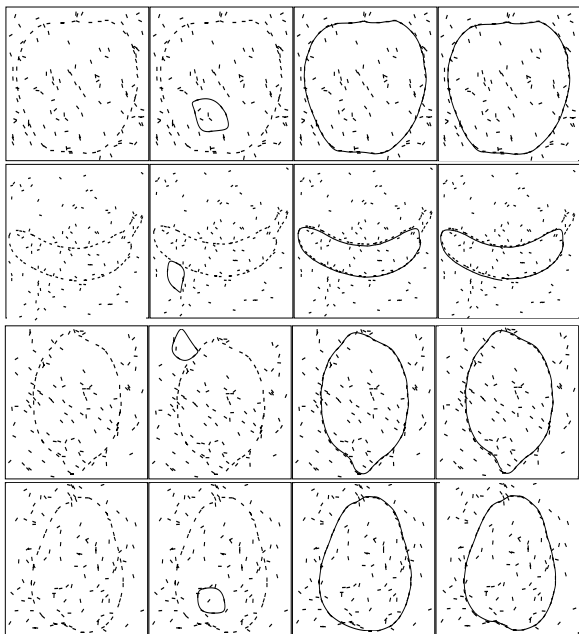

Figure 6. Extracting salient boundaries from synthesized data. From leftmost column to the rightmost column are the real fragments, and the boundaries extracted using EZ, WT, and WKS, respectively. From the top row to the bottom row are the mixed patterns of apple & sand, banana & wood, lemon & fabric, and pear & stone, respectively.

4.3. Real Images

We collect some real images from the Corel database and some medical images for testing these three methods. Real fragments are extracted from those images by Canny edge detectors using the default setting in Matlab software.
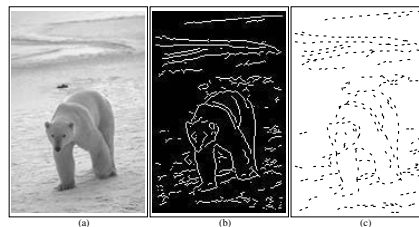

Figure 7. An illustration of image processing for real fragments: (a) original image, (b) Canny edge detection, and (c) resulted real fragments.

Then we perform a preprocessing of real-fragment sampling. Each detected fragment from Canny edge detectors is sampled into real fragments with an interval of 8 pixels. An example of this processing is shown in Fig.7.

We then apply the three methods on those real fragments and results are shown in Fig. 8. We find that EZ fails in many examples by extracting shorter and smoother boundaries, like in the second and fifth images, and the two medical images shown in the figure. We also see some examples where WT extracts open curve segments instead of closed boundaries, like in the polar-bear image shown in the 10th row of the figure. In general, WKS performs better than EZ and WT. In some cases, neither of these three methods can extract the salient boundaries successfully, such as the cow image shown in the fourth row. In most failure cases, there usually exist multiple perceptually salient patterns mixed and entwined in one image and therefore, the resulted closed boundary may mistakenly traverse fragments from different patterns.

4.4. Update P to C for WKS?

An intuitive way to achieve an improvement is to combine the merits of WT and WKS by using the updated fragment saliency in C for WKS, i.e., setting the edge weight $w(i_2, j_1) = |\log c_{ij}|$ in the SD graph G and then applying WKS to acquire final boundary closure. For simplicity, we denote this combined method by C-WKS. In this new method, on one hand, global boundary closure is encoded into fragment saliency (edge weight $w(\cdot)$ in G). On the other hand, the search space is limited to $\mathcal{B}$ that contains only simple closed boundaries. We test C-WKS and compare its performance to the standard WKS with $w(i_2, j_1) = |\log p_{ij}|$. We find that there is no noticeable improvement by using C-WKS instead of WKS. Therefore, the global boundary-closure property encoded in WT has been effectively incorporated in the standard WKS by limiting the search space to simple closed boundaries only.

In Fig. 9, we try a toy example, which contains a circle and 100 noisy fragments. We decrease the number of sampled real fragments along the circle and check whether C-
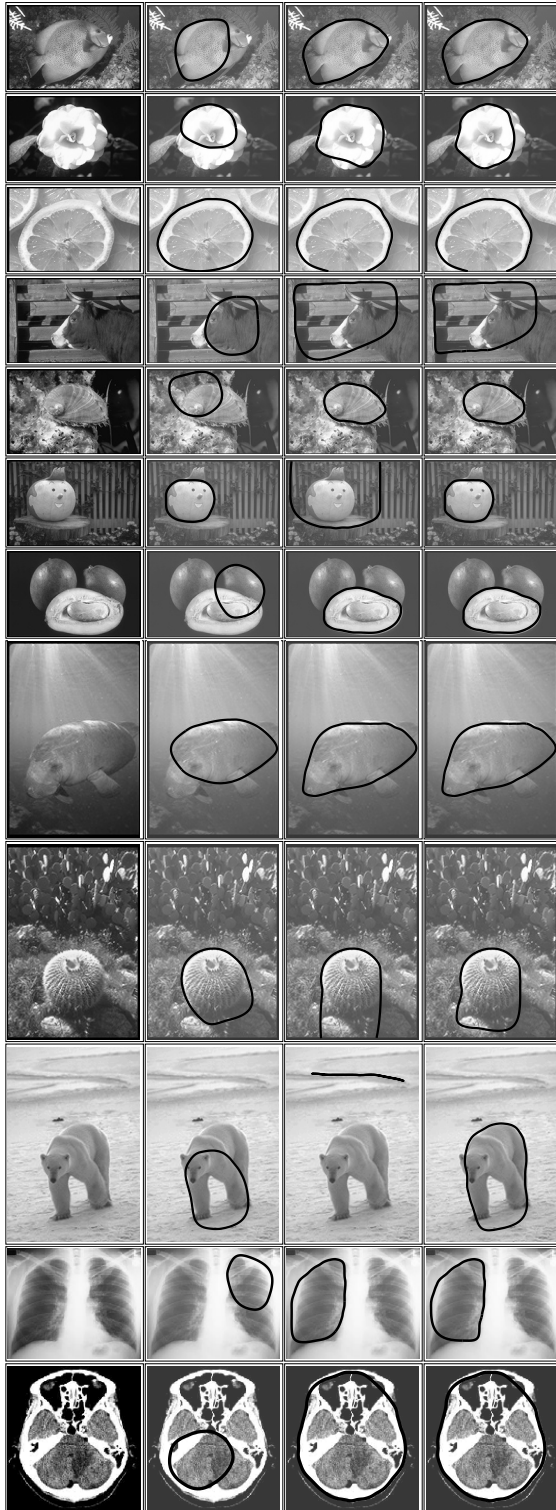
Figure 8. An illustration of salient-boundary extraction on real images. From the leftmost column to the rightmost column are: the original images, and the boundaries extracted using EZ, WT, and WKS, respectively. Some closed boundaries are cropped by the image perimeter when producing this figure.
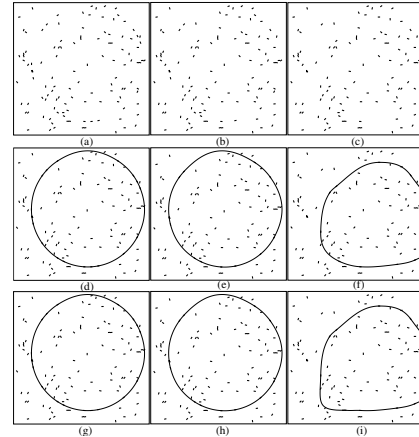

Figure 9. A toy example used for testing C-WKS. (a-c) real fragments, out of which 18, 12, and 6 are on the salient circle, respectively. (d-f) boundaries extracted using the standard WKS on (a-c), respectively. (g-i) boundaries extracted using C-WKS on (a-c), respectively.

WKS is able to extract the circle represented by less number of real fragments. Unfortunately, we find that C-WKS has similar performance with the standard WKS. In the cases where the standard WKS succeeds in extracting the salient circle, so does C-WKS. When the standard WKS fails, C-WKS also fails. We try all the 25 synthesized data collected in Section 4.2 and a set of real images from the Corel database, without seeing any statistically meaningful improvement. Part of these experiment results are shown in Fig. 10 and Fig. 11.

5. Conclusion

This paper conducted an in-depth study on the problem of finding salient closed boundaries from a set of boundary fragments. In this paper, we analyzed and compared three typical methods that had been developed for closed-boundary extraction: (a) Elder and Zucker's method (EZ), (b) William and Thornber's method (WT), and (c) Wang, Kubota, and Siskind's method (WKS). Our interesting findings from this comparison include: (a) EZ has a bias to boundaries with fewer fragments while WT and WKS do not have such a bias, (b) the boundary-saliency measures used in WT and WKS are very similar to each other, while WKS further prunes the search space to guarantee the boundary closure, (c) EZ and WKS find the real global optima in terms of their selected saliency measures, while the algorithms in WT are only related to its boundary-saliency measure in some special cases, (d) WT may produce boundaries that are not closed or simple, while WKS does not, (e) the spectral-analysis algorithm used in WT can enhance
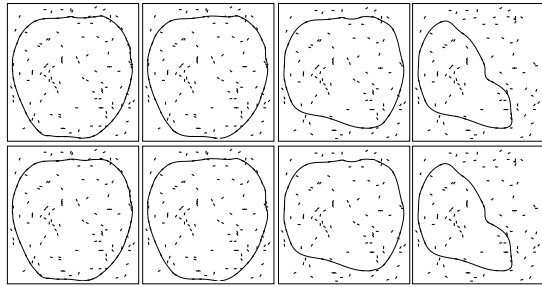
Figure 10. Using synthesized data, sampled from an apple pattern and a wood texture, for testing C-WKS. The wood texture contributes the same 100 real fragments in all of them while the apple pattern was sampled into 47, 35, 20 and 15 fragments from leftmost column to rightmost column. The first row shows the most salient boundaries extracted from the fragments shown in the first rows, using the standard WKS on the P matrix. The second row shows the most salient boundaries extracted using C-WKS.
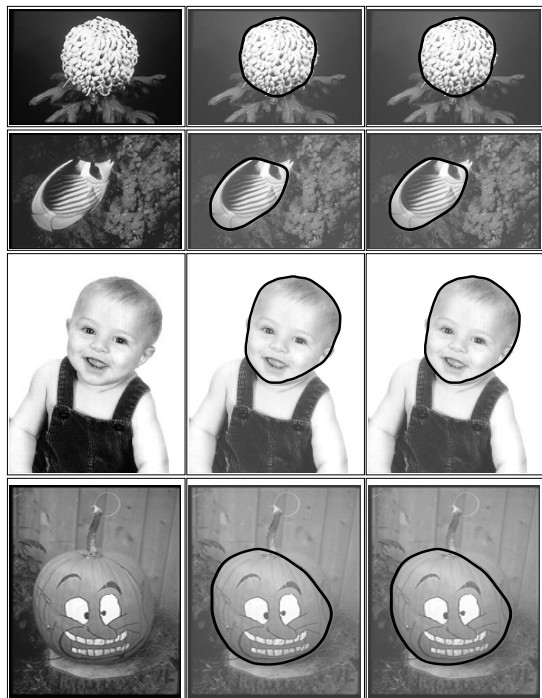

Figure 11. Using real images for testing C-WKS. From leftmost column to the rightmost column are the original images, the boundaries extracted using the standard WKS, and the boundaries extracted using C-WKS, respectively.

the local saliency of many fragments along the most salient boundary, but this enhancement does not improve the performance of WKS in a noticeable way. In general, with a unified setting of fragment saliency, WKS performs more reliably than WT and EZ.

References

[1] T. Alter and R. Basri. Extracting salient contours from images: An analysis of the saliency network. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 13–20, 1996.

[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] F. R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, Providence, 1997.

[4] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. Cambridge: MIT Press/New York: McGraw Hill, 1990.

[5] J. Elder and S. Zucker. Computing contour closure. In *European Conference on Computer Vision*, pages 399–412, 1996.

[6] G. Guy and G. Medioni. Inferring global perceptual contours from local features. *International Journal of Computer Vision*, 20(1):113–133, 1996.

[7] L. Herault and R. Horaud. Figure-ground discrimination: A combinatorial optimization approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15:899–914, 1993.

[8] D. Jacobs. Robust and efficient detection of convex groups. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(1):23–27, 1996.

[9] 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.

[10] S. Sarkar and K. Boyer. Quantitative measures of change bvased on feature organization: Eigenvalues and eigenvectors. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 478–483, 1996.

[11] A. Shashua and S. Ullman. Structural saliency: The detection of globallly salient structures using a locally connected network. In *International Conference on Computer Vision*, pages 321–327, 1988.

[12] S. Wang, T. Kubota, and J. M. Siskind. Salient boundary detection using ratio contour. In *Neural Information Processing Systems Conference*, 2003.

[13] L. Williams and D. Jacobs. Stochastic completion fields: A neural model of illusory contour shape and salience. *Neural Computation*, 9:849–870, 1997.

[14] 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.