MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

A.I. Memo No. 1550 August, 1995

Extracting Salient Curves from Images: An Analysis of the Saliency Network

T. D. Alter **Ronen Basri**

This publication can be retrieved by anonymous ftp to publications.ai.mit.edu.

Abstract

The Saliency Network proposed by Shashua and Ullman [20] is a well-known approach to the problem of extracting salient curves from images while performing gap completion. This paper analyzes the Saliency Network. The Saliency Network is attractive for several reasons. First, the network generally prefers long and smooth curves over short or wiggly ones. While computing saliencies, the network also fills in gaps with smooth completions and tolerates noise. Finally, the network is locally connected, and its size is proportional to the size of the image.

Nevertheless, our analysis reveals certain weaknesses with the method. In particular, we show cases in which the most salient element does not lie on the perceptually most salient curve. Furthermore, in some cases the saliency measure changes its preferences when curves are scaled uniformly. Also, we show that for certain fragmented curves the measure prefers large gaps over a few small gaps of the same total size. In addition, we analyze the time complexity required by the method. We show that the number of steps required for convergence in serial implementations is quadratic in the size of the network, and in parallel implementations is linear in the size of the network. We discuss problems due to coarse sampling of the range of possible orientations. We show that with proper sampling the complexity of the network becomes cubic in the size of the network. Finally, we consider the possibility of using the Saliency Network for grouping. We show that the Saliency Network recovers the most salient curve efficiently, but it has problems with identifying any salient curve other than the most salient one.

This report describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology and at the Weizmann Institute of Science. Support for the Artificial Intelligence Laboratory's research is provided in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract N00014-91-J-4038. TDA was supported by Office of Naval Research AASERT Reference No. N00014-94-1-0128.

1 Introduction

In line drawings, certain shapes attract our attention more than others. For example, these shapes may be the ones that are smooth, long, and closed (see for example Fig. 1). Shashua and Ullman [20] proposed a method, which attracted considerable attention, to extract such shapes from a line drawing. They defined a function that evaluates the "saliency" of a curve. Their function has the following properties. First, when all other parameters are held constant, it monotonically increases with the length of the evaluated curve. In addition, it decreases monotonically with the *energy* (the total squared curvature) of the curve. Thirdly, the function evaluates fragmented curves, in which case it penalizes according to the amount of fragmentation. Finally, the saliency measure is the sum of contributions from different sections of the curve, where these contributions decay with the sections' accumulated energy and gap length from the beginning of the curve. Using this saliency function, Shashua and Ullman defined the "saliency map" of an image to be an image in which the intensity value of each pixel is proportional to the score of the most salient curve emanating from that pixel.

A network of locally connected elements (the Saliency Network) was proposed for computing the saliency map. The Saliency Network's computation involves local interactions between image locations, and its size is proportional to the size of the image. The network implements a relaxation process that optimizes the saliency measure. As a consequence of the optimization, the network can identify the most salient location in the image, which could lie on either an open or closed curve. Additionally, the method attempts to fill in gaps smoothly while simultaneously overcoming noise.

The problem of marking salient locations in images ("attention") is also addressed in the work of Guy and Medioni [7] and Subirana and Sung [21, 22]. Subirana and Sung extend Shashua and Ullman's method to find skeletons of regions. Using a different method from Shashua and Ullman's, Guy and Medioni also produce a saliency map from an edge image. In Guy and Medioni's scheme, each point in the image receives a saliency value equal to a weighted sum of contributions from the individual edge elements. The weight assigned to an element is based on a circular-arc completion between it and the image point; the weight decreases with the total curvature of the arc, preferring straighter and shorter completions. Unlike Shashua and Ullman, however, there is no attempt to optimize a measure of saliency over the set of image curves.

Identifying salient structures in images is one of the objectives of *perceptual grouping*. By perceptual grouping, we refer to the (bottom-up) process of grouping together structures in the image that are likely to belong to a single object. Other tasks in perceptual grouping are image segmentation and gap completion. For instance, [9, 11, 12, 13, 14, 15, 17, 18, 25, 26] extract contours from the image according to certain optimization criteria, [23, 19, 1, 10, 3] compute optimal curves for filling in gaps, and [2, 24, 5, 6, 8, 16, 27] identify occluded and subjective contours.

In this paper we provide an analysis of Shashua and Ullman's method. We examine both the measure of saliency and the computational performance of the Saliency Network. Motivated by both perceptual and computational reasons we identify below three properties which we believe a measure of saliency should satisfy. We then analyze Shashua and Ullman's measure with respect to these properties. The properties are:

Fidelity. To be consistent with examples such as Figure 1, the saliency map should highlight the locations in the image that lie on perceptually salient curves. In particular, the most salient location in the saliency map should lie on the most perceptually salient curve. Thus, for example, in Figure 1 the most salient location in the saliency map should be on the circle rather than on any of the surrounding line segments.

Invariance. In different images objects often appear in different positions and orientations or in different sizes. A saliency measure for curves should be insensitive to such variations. In particular, the measure should be invariant to 2D rigid transformations of the curve. In addition, the measure should be consistent over different scales. That is, given two curves Γ_1 and Γ_2, if Γ_1 is considered more salient than Γ_2, then Γ_1 should remain more salient when the curves are scaled uniformly.

Performance on gaps. In Figure 1 as the size of gaps between edge elements is increased, our perception of the circle fades. We therefore expect the measure of saliency to degrade with gaps. Furthermore, we require a saliency measure to penalize large gaps more than few small gaps of the same total size. This requirement is motivated by psychophysical studies performed by Elder and Zucker [4] which demonstrate that, when a fraction of the boundary of an object is missing, humans' recognition ability is hindered more when the missing fraction is contained all in one gap than when spread over several gaps.

The Saliency Network is an efficient and elegant method, well suited to locating salient structures in images. However, we found cases in which the network violates each of the above three properties. On the issue of fidelity, the network indeed locates the perceptually salient curves, so that long, smooth, closed curves are preferred over short, wiggly, open ones. Nonetheless, our analysis reveals cases in which the most salient location in the saliency map is not on the perceptually most salient curve. For example, if there are short line segments touching a salient curve, then often the short segments shall be judged more salient than the closed curve. In this situation, the most salient location in the network will not lie on the closed curve, but it will draw its saliency from the closed curve.

Since the saliency measure depends only on length and curvature, it is invariant to rigid transformations. We show, however, that at times the measure changes its preferences when the curves are scaled uniformly. For instance, consider a straight line and a circle of the same

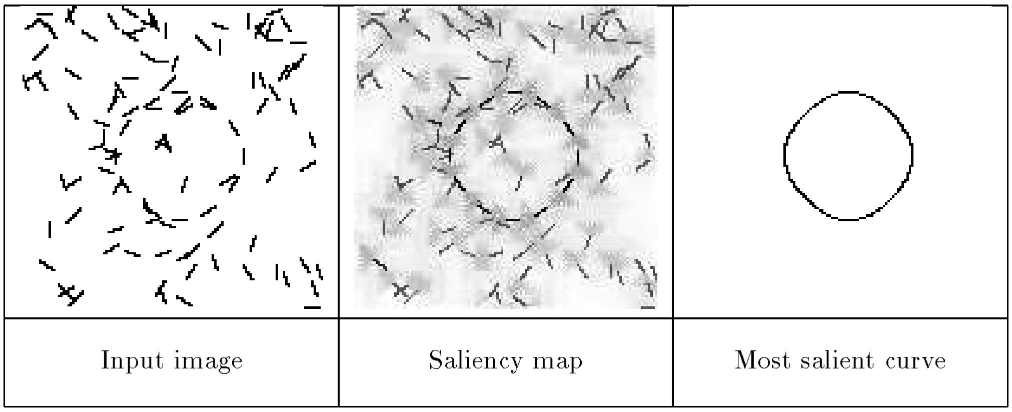

Figure 1: A fragmented circle in the middle of noise. The global shape of the circle is apparent.

length. For lengths less than a certain value, the line is preferred over the circle, whereas for larger lengths this preference reverses. Shashua and Ullman's rankings of curves, therefore, are not invariant to uniform scaling of the image.

Finally, the saliency measure can be applied to fragmented curves, in which case it will attenuate with gap length. However, our analysis indicates that, when circles of both the same size and gap length are compared, the measure prefers a circle with one long gap over a circle with few small gaps of the same total size.

In addition to studying properties of the saliency measure, we also examine the computational properties of the Saliency Network. In particular, we analyze the convergence rate of the network and show that the run-time complexity of the network in serial implementations is quadratic in the number of elements. We then discuss problems due to coarse sampling of the range of possible orientations. We show that, when the range of possible orientations is sampled too coarsely, undesirable effects may occur in which corners are preferred over straight lines. With proper sampling the complexity of the network becomes cubic in the size of the image.

Finally, we consider the possibility of using the Saliency Network for grouping. We note that, in contrast to other existing methods for grouping that search over the exponentially large space of all possible image curves (e.g., [9, 11, 17, 26]), the Saliency Network recovers the most salient curve in time complexity that is polynomial in the size of the image. However, the network must take a single choice at every junction, and as a consequence has problems with identifying salient curves other than the most salient one.

The paper proceeds as follows. Section 2 contains definitions. Section 3 includes an analysis of the different properties of the saliency measure. Section 4 analyzes the time complexity of the network computation. Section 5 analyzes the effects of sampling on the computation. Finally, Section 6 discusses the issue of using the output of the network for grouping.

2 Definitions

Shashua and Ullman defined their saliency measure as follows. For every pixel in the image, there is a fixed set of "orientation elements" connecting the pixel to neighboring pixels (Fig. 2-left). Each orientation element is called "actual" or "real" if it lies on an edge in the underlying image, and otherwise it is called "virtual" or "gap" (see Fig. 3). Given a curve Γ composed of the $N+1$ orientation elements $p_i, p_{i+1}, ..., p_{i+N}$ (Fig. 2-right), the saliency of Γ is defined by

$$\Phi(\Gamma) = \sum_{j=i}^{i+N} \sigma_j \rho_{ij} C_{ij}, \tag{1}$$

with

$$\sigma_j = \begin{cases} 1, & \text{if } p_j \text{ is actual} \\ 0, & \text{if } p_j \text{ is virtual} \end{cases}$$

$$\rho_{ij} = \prod_{k=i}^{j} \rho_k = \rho^{g_{ij}},$$

where $\rho_{ii} = 1$ and where ρ is some constant in the range $[0, 1)$.[1] (Shashua and Ullman set ρ to 0.7.) σ_j ensures that only actual elements will contribute to the saliency measure. g_{ij} is the number of gap elements between p_i and p_j, and ρ_{ij} reduces the contribution of an element according to the total length of the gaps up to that element. Further,

$$C_{ij} = e^{-K_{ij}},$$

with

$$K_{ij} = \int_{p_i}^{p_j} \kappa^2(s) ds,$$

where $\kappa(s)$ is the curvature at position s. K_{ij} reduces the contribution of elements according to the accumulated squared curvature from the beginning of the curve.

[1] The formula for ρ_{ij} appeared in [20] as $\rho_{ij} = \prod_{k=i+1}^{j} \rho_k$, but the computation actually performed by the network (which is given by Eq. 5) implements the modified formula given here.

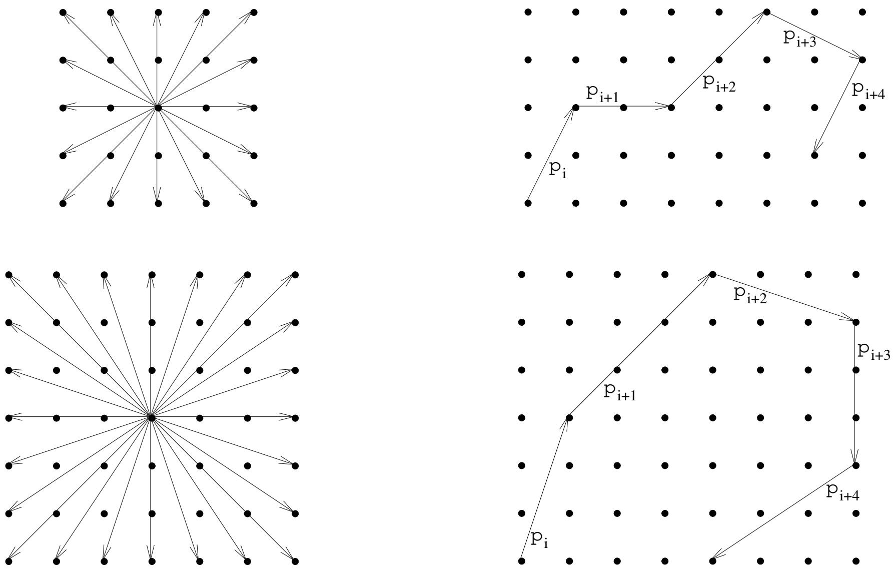

Figure 2: Example of the connectivity of Shashua and Ullman's Saliency Network, for the cases of sixteen and twenty-four orientation elements per pixel. In the left pictures, the neighbors of a pixel (x, y) are $\{(x+\Delta x, y+\Delta y) | \max(|\Delta x|, |\Delta y|) = \Delta e\}$, where $\Delta e = 2$ for 16 elements per pixel and $\Delta e = 3$ for 24 elements per pixel. Given the pixel neighborhoods in the left pictures, the right pictures show examples of five-element curves.

The saliency of an element p_i is defined to be the maximum saliency over all curves emanating from p_i:

$$\Phi(i) = \max_{\Gamma \in \mathcal{C}(i)} \Phi(\Gamma), \tag{2}$$

where $\mathcal{C}(i)$ denotes the set of curves emanating from p_i. Shashua and Ullman showed how to compute $\Phi(i)$ on a network of locally connected elements. Denote by $\Phi_N(i)$ the saliency of the most salient curve of length $N+1$ or less emanating from p_i. The measure $\Phi_N(i)$ satisfies

$$\Phi_N(i) = \max_{p_j \in \mathcal{N}(i)} F(i, j, \Phi_{N-1}(j)), \tag{3}$$

where $\mathcal{N}(i)$ is the set of all neighboring elements of p_i, and where $F()$ is a function of $\Phi_{N-1}()$ and constants stored at elements p_i and p_j. Shashua and Ullman referred to this type of measure as "extensible."[2] In the Saliency Network,

$$F(i, j, \Phi_{N-1}(j)) = \sigma_i + \rho_i C_{ij} \Phi_{N-1}(j), \tag{4}$$

which gives

$$\Phi_N(i) = \sigma_i + \rho_i \max_{p_j \in \mathcal{N}(i)} C_{ij} \Phi_{N-1}(j). \tag{5}$$

Note that this recurrence relation updates each element's saliency by taking a maximum over its neighbor's, but does not allow an element to retain its current saliency. This observation raises the question of whether the saliencies are optimal over all curves that are less than or equal to N elements long or only over curves that are exactly N elements long. In fact the former is true, which we now show. First, note that the saliency measure in Eq. 1 is monotonically non-decreasing with the number of elements N on a curve. Consequently, at iteration $N+1$ every element has the option of choosing the same neighbor as it chose at iteration N, and thus obtain a new saliency that is no less than its current saliency. Therefore, it is sufficient to not include an element's current saliency when taking the maximum, because there will be at least one neighbor through which the element can obtain a new saliency that is as great as its own.

To make the saliency measure Φ independent of the particular implementation, we introduce a continuous version of Φ. Given a curve $\Gamma(s)$ of length l ($0 \le s \le l$, s denotes arc length), we define Φ by

$$\Phi(\Gamma) = \int_0^l \sigma(s)\rho(0, s)C(0, s)ds, \tag{6}$$

where

$$\begin{aligned} \sigma(s) &= \begin{cases} 1, & \text{if } \Gamma(s) \text{ is actual} \\ 0, & \text{if } \Gamma(s) \text{ is virtual} \end{cases} \\ \rho(s_1, s_2) &= \rho^{g(s_1, s_2)} \\ C(s_1, s_2) &= e^{-K(s_1, s_2)}, \end{aligned}$$

where $g(s_1, s_2)$ is the total gap length of Γ between s_1 and s_2 and $K(s_1, s_2)$ is the energy of the curve between s_1 and s_2, which are defined by

$$g(s_1, s_2) = \int_{s_1}^{s_2} (1 - \sigma(t))dt, \tag{7}$$

$$K(s_1, s_2) = \int_{s_1}^{s_2} \kappa^2(t)dt. \tag{8}$$

A useful tool in computing saliencies is the following rule. Given a curve Γ which is composed of two smoothly

[2] Note that this definition of extensibility is different from that used by Brady et al. [1].

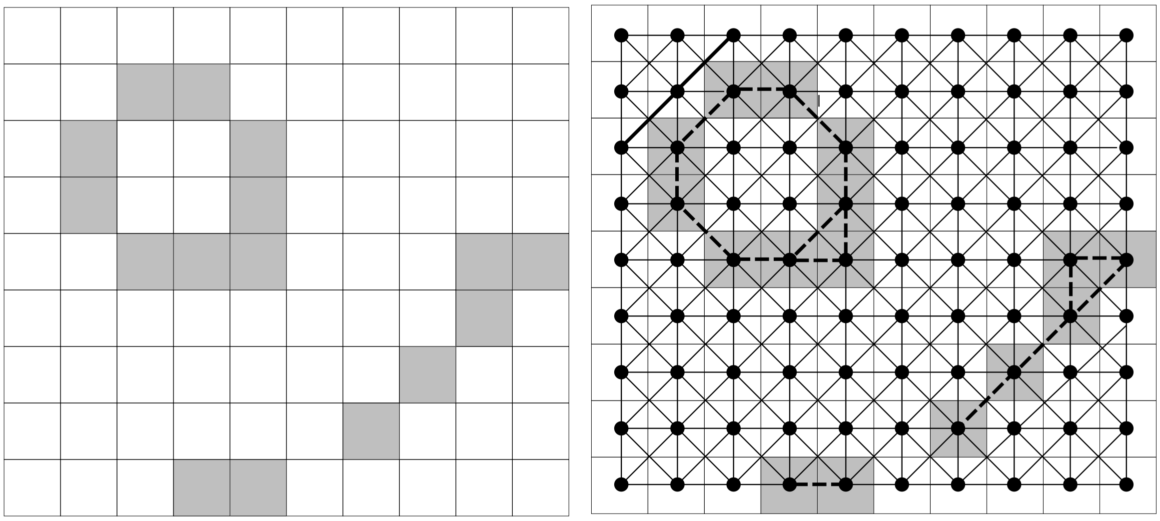

Figure 3: Left: Input image is a binary edge map. In the picture the black squares represent edge pixels. Right: The Saliency Network is defined on top of the edge map. The network is composed of locally connected elements which are called "active" if they lie on edges and "gaps" if they do not. In the right picture, the dashed line segments between eight-connected pixels represent active elements, and the remaining line segments represent gaps. For viewing purposes, every element was set to have eight neighbors, although in Shashua and Ullman's implementation every element had sixteen neighbors, and in our implementation every element had twenty-four neighbors.

concatenated sections, Γ_1 and Γ_2, the saliency of Γ is given by

$$\Phi(\Gamma) = \Phi(\Gamma_1) + \rho^{g(\Gamma_1)} e^{-K(\Gamma_1)} \Phi(\Gamma_2), \qquad (9)$$

where $g(\Gamma_1)$ is the total gap length and $K(\Gamma_1)$ is the energy of Γ_1.

Over the remainder of the paper, we will present examples of the Saliency Network on simulated and real images. Our implementation replicates Shashua and Ullman's original implementation, except that we increased the number of orientation elements per pixel to obtain greater accuracy. We used twenty-four orientation elements per pixel, whereas Shashua and Ullman used sixteen elements per pixel. Also we set $\rho = .7$ as in the original implementation.

3 Properties of the Saliency Measure

We begin our analysis by examining the saliency measure proposed by Shashua and Ullman. Section 3.1 below discusses the treatment of cycles. Section 3.2 analyzes the behavior of the measure when applied to simple curves. Lastly, Section 3.3 analyzes the behavior of the measure when applied to curves that include gaps.

3.1 Cycles

The measure of saliency proposed by Shashua and Ullman is a positive function that increases monotonically with the lengths of the curves in the image. Closed curves (cycles) are considered to have infinite length, even though they form finite structures in the image. Shashua and Ullman showed that their network is guaranteed to converge when applied to closed curves. The reason it converges is that the contribution to the saliency from remote elements attenuates geometrically with the curvature accumulated from the beginning of the curve. In cycles this generates a geometric series that converges to a finite value.

Formally, given a closed curve Γ, denote by Φ the saliency of an element of Γ that is obtained by starting from that element and then proceeding once around the curve. Denote by K the total squared curvature of the cycle and by g the cycle's total gap length. Then by repeatedly applying Eq. 9 we obtain

$$\begin{aligned} \Phi(\Gamma) &= \Phi + \rho^g e^{-K} \Phi + \rho^{2g} e^{-2K} \Phi + \dots \\ &= \frac{\Phi}{1 - \rho^g e^{-K}} . \end{aligned} \qquad (10)$$

When the network is applied to an open curve, after going once along the curve it is possible for the network to then take a 180° turn and walk back along the curve. The saliency of the resulting closed curve could be considered to be the saliency of the open curve at convergence. As we shall see next, the attenuation due to the 180° is so high that the additional score is negligible. Let Γ be an open curve. Let Φ_f and Φ_b be the saliencies of Γ measured by going once along the curve in the forward and backward directions, respectively. Denote by K the total squared curvature of Γ. Then the saliency of Γ is given by

$$\begin{aligned} \Phi(\Gamma) &= \Phi_f + e^{-K-\pi^2} \Phi_b + e^{-2K-2\pi^2} \Phi_f \\ &\quad + e^{-3K-3\pi^2} \Phi_b + \dots \\ &= \frac{\Phi_f + e^{-K-\pi^2} \Phi_b}{1 - e^{-2K-2\pi^2}} . \end{aligned} \qquad (11)$$

If Γ is symmetric then $\Phi_f = \Phi_b$ and we obtain

$$\Phi(\Gamma) = \frac{\Phi_f}{1 - e^{-K-\pi^2}} \approx \frac{\Phi_f}{1 - .000051723\, e^{-K}} . \qquad (12)$$

The largest increase in saliency is obtained for a straight line ($K = 0$), where the saliency becomes

$$\Phi(\Gamma) = \frac{\Phi_f}{1 - e^{-\pi^2}} \approx 1.000051725\,\Phi_f. \tag{13}$$

One can see that the additional saliency obtained by wrapping around an open curve is very small and practically can be ignored. As a consequence, the network is likely to prefer connecting the curve through gaps to other curves, or even around to itself, since such connections often will result in higher saliencies.

3.2 Straight lines and circles

In this section we compute the saliencies of a few simple curves. We then use these simple curves to examine the issues of fidelity and invariance. In general, we will only be interested in the measure of saliency obtained for the most salient element of the curve. Throughout this section we shall use the continuous definition of the saliency measure (Eq. 6). We consider only curves with no gaps (we will analyze curves with gaps in Section 3.3); hence $\sigma(s) = 1$ and $\rho(0, s) = 1$ for all s. Eq. 6 therefore becomes

$$\Phi(\Gamma) = \int_0^l C(0, s)ds, \tag{14}$$

where

$$C(0, s) = e^{-\int_0^s \kappa^2(t)dt}.$$

The examples below demonstrate some of the problems with Shashua and Ullman's saliency measure. In particular, we compare the saliency of a line segment of length l to that of a circle of perimeter l. We show that for small values of l, the straight line is preferred over the circle, and that this preference reverses for large values of l. The saliency function, therefore, ranks curves differently when these curves are scaled uniformly. In another example, we analyze the results of applying the saliency measure to a picture containing a circle and short line segments connected to it. We see that a short line segment increases its saliency value by connecting to the circle. As a result of this increase, it is not unusual for a short segment to become more salient than a circle. The saliency of the short line segment in this case represents the saliency of the circle, but the most salient element is in fact not part of the circle.

We begin by deriving explicit formulas for the saliency of straight lines and curves. For straight lines $C(0, s) = 1$ for all s. Therefore, ignoring the possibility that a line can wrap around itself (see Section 3.1), a straight line of length l will obtain the score $\Phi(\Gamma) = l$. The saliency of a straight line, therefore, grows linearly with the length of the line.

For a circle of radius r, the curvature is constant, $\kappa = 1/r$, and so for a circular arc of length s,

$$C(0, s) = e^{-\int_0^s \frac{1}{r^2}dt} = e^{-\frac{s}{r^2}}. \tag{15}$$

The saliency attributed for the circular arc is

$$\begin{aligned} \Phi(0, s) &= \int_0^s C(0, t)dt \\ &= \int_0^s e^{-\frac{t}{r^2}}dt \\ &= r^2\left(1 - e^{-\frac{s}{r^2}}\right). \end{aligned} \tag{16}$$

At convergence ($s = \infty$), the saliency of the circle is given by

$$\Phi(\Gamma) = \lim_{s\to\infty} r^2\left(1 - e^{-\frac{s}{r^2}}\right) = r^2. \tag{17}$$

The score of a circle, therefore, grows quadratically with the radius (and thus also with the perimeter) of the circle.

The fact that the saliency of a straight line grows linearly with its length, whereas the saliency of a circle grows quadratically with its perimeter, suggests that the network may treat the two differently when they are scaled. Consider a straight line of length l and a circle of perimeter $l = 2\pi r$. These two entities will have exactly the same saliency when $l = 0$ and when $l = 4\pi^2 \approx 39.48$. (The saliencies in the two cases are 0 and $4\pi^2$, respectively.) When $0 \le l \le 4\pi^2$ the line will be more salient than the circle, whereas when $l > 4\pi^2$ the circle will be more salient. Fig. 4 shows an example of three images, each of which contains a straight line and a circle of the same length. Consistent with our analysis, the Saliency Network found the straight line to be more salient than the circle at shorter lengths, and found the circle to be more salient at longer lengths.

A different problem is encountered in the case of a circle connected to short line segments. Consider the picture in Fig. 5-left. The circle seems to be the most perceptually salient curve in this image. Counterintuitively, the most salient element computed by the network is on one of the line segments connected to the circle, thus violating the fidelity requirement. The reason is that a neighboring line segment may increase its saliency by connecting to the circle, without affecting the saliency of the circle. Consider, for example, a circular arc of length 1 and curvature κ connected smoothly to a circle of radius r (which corresponds to a single element connected smoothly to the circle via curvature κ). Using Eq. 9 we obtain that the saliency of the first element on the arc is

$$\Phi_e = \Phi(\Gamma) + e^{-\kappa^2}\Phi_c, \tag{18}$$

where Γ represents the circular arc and Φ_c is the saliency of the circle. Now, using Eq. 15,

$$\Phi(\Gamma) = \int_0^1 C(0, s)ds = \frac{1 - e^{-\kappa^2}}{\kappa^2}. \tag{19}$$

Combining Eqs. 17-19, we obtain that

$$\Phi_e = \frac{1 - e^{-\kappa^2}}{\kappa^2} + e^{-\kappa^2}r^2. \tag{20}$$

If we now compare the saliency of the element, Φ_e, to that of the circle $\Phi_c = r^2$ (Eq. 17), we obtain that $\Phi_e > \Phi_c$ when

$$\frac{1 - e^{-\kappa^2}}{\kappa^2} + e^{-\kappa^2}r^2 > r^2, \tag{21}$$

so that

$$|\kappa| < \kappa_c, \tag{22}$$

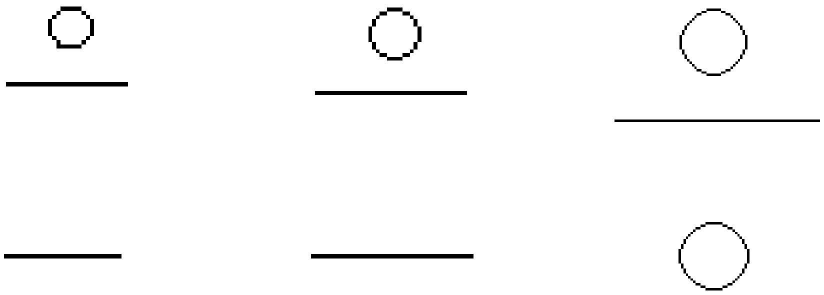

Figure 4: Lack of scale invariance in the Saliency Network. Top figures: three images that contain a straight line and a circle of roughly the same length. Bottom figures: the most salient curves that were found in these images. Lengths are 27 (left), 39 (middle), and 84 (right). The saliency values obtained for the circles are 15.39 (left), 33.60 (middle), and 132.05 (right), and for the lines are 27.06 (left), 39.00 (middle), and 84.00 (right).

where $\kappa_c = 1/r$. That is, the element will be more salient than the circle if and only if it connects to the circle at a curvature that is less than the curvature of the circle. This is consistent with the network's preference for straight curves. Notice that if the element is a line tangential to the curve ($\kappa = 0$) the element will be more salient than the circle regardless of the circle's radius.

This phenomenon, that curves connecting to a circle may increase their saliencies due to these connections and actually beat the circle, is more likely to occur for longer curves. Suppose a curve Γ connects to a circle C such that the total squared curvature of Γ, including the connection point, is K. Then the saliency of the element on Γ that is most distant from the circle is given by Eq. 18, where κ^2 is replaced by K, namely,

$$\Phi_e = \Phi(\Gamma) + e^{-K}\Phi_c. \tag{23}$$

The longer Γ is, the more likely it is to become more salient than the circle. Suppose for example that Γ is a straight line of length l that connects to the circle via curvature κ. We have that $\Phi(\Gamma) = l$ and $K = \kappa^2$, which implies

$$\Phi_e = l + e^{-\kappa^2}\Phi_c. \tag{24}$$

Now $\Phi_e > \Phi_c$ when

$$l + e^{-\kappa^2} r^2 > r^2. \tag{25}$$

Substituting $r = 1/\kappa_c$, this implies that

$$\kappa_c^2 > \frac{1 - e^{-\kappa^2}}{l} \approx \frac{\kappa^2}{l}, \tag{26}$$

when κ^2 is small, or

$$\kappa_c > \frac{|\kappa|}{\sqrt{l}}. \tag{27}$$

Clearly, the longer the line is, the more likely it is to become more salient than the circle.

Fig. 5-left shows a picture of a circle with a few short curves connected to it. When the Saliency Network is applied to this picture, the most salient element does not lie on the circle, although most of its saliency is due to the circle. Indeed, if we disconnect these short curves from the circle, then the circle becomes the most salient structure in the image.

3.3 Curves with gaps

One of the most important properties of Shashua and Ullman's saliency network is its ability to fill in gaps while computing the saliencies. The network handles gaps by using virtual elements, which compute the saliencies of curves emanating from their locations and transfer these saliencies to their neighboring elements. Via these transfers, actual elements evaluate the saliencies of curves that emanate from their locations and contain any number of gaps. The network avoids curves with large gaps by attenuating the scores of curves exponentially with gap size.

In this section we analyze the performance of the saliency network in the presence of gaps. Due to the saliency measure attenuating exponentially with gap size, the network is capable of overcoming small gaps, but is unlikely to overcome large ones. As an example, consider the problem mentioned in Section 3.2, that a short line segment in the neighborhood of a circle may increase its saliency by connecting to the circle. One consequence of the fast attenuation is that this problem almost disappears when the segment is not physically connected to the circle. On the other hand, we show below that, due to the exponential decay, very long structures (straight lines and circles) obtain very low scores even when only a small fraction of the curves are gaps.

Finally, we explore the question of whether the network prefers fragmented curves (dashed lines) over curves with single gaps of the same total size. At first glance Shashua and Ullman's saliency measure appears indifferent to this property, because the total size of gaps is taken into account, irrespective of the fragmentation. In fact, for open curves there is no clear preference between a curve having many small gaps or a few long gaps. For closed curves, however, we show that a curve with

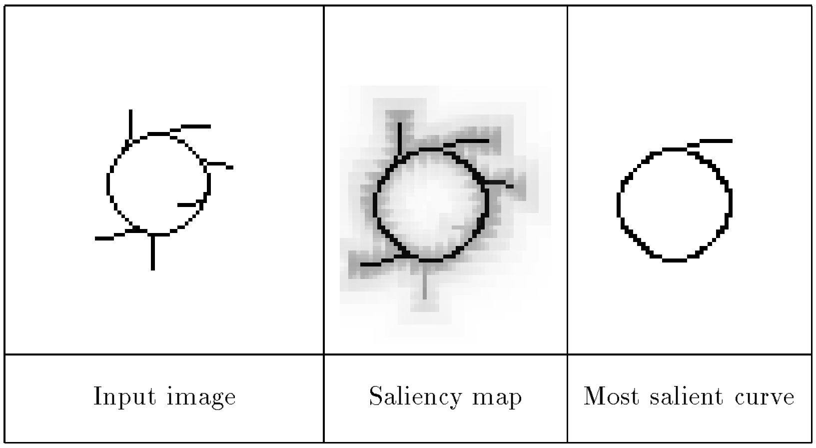

Figure 5: An example of a circle with a few short curves connecting to it. The most salient element (for which $\Phi = 136.63$) was not on the circle, although its saliency came mostly from the circle (the saliency of the circle is 130.74). If short gaps were added between the curves and the circle, the circle would become the most salient curve in the image.

a single large gap is preferred over the same curve with several small gaps of the same total size; this preference is inconsistent with the psychophysical experiments of Elder and Zucker [4].

In computing the saliency of a fragmented curve, gaps affect the total score in two ways (see Eq. 1). First, gap elements themselves do not contribute at all to the total score (since $\sigma_j = 0$ for virtual elements). Secondly, the actual elements of the curve that lie on the other side of a gap are attenuated by a factor ρ^g, where g is the total gap length accumulated from the beginning of the curve. Consider for example a curve Γ with one gap of length g. Denote the first part of the curve (before the gap) by Γ_1 and the second part of the curve (after the gap) by Γ_2, and denote by $K(0, m)$ the total squared curvature of Γ_1 plus the gap. The saliency of Γ is given by (Eq. 9)

$$\Phi(\Gamma) = \Phi(\Gamma_1) + \rho^g e^{-K(0,m)} \Phi(\Gamma_2) \qquad (28)$$

From this formula, Γ_1 contributes to the saliency of Γ as if there were no gap, the gap elements contribute nothing, and the contribution of Γ_2 is attenuated by ρ^g. Clearly, the longer the portion of Γ before the gap (Γ_1), the less the saliency of Γ will be attenuated. If the gap appears near the end of the curve the saliency of Γ is hardly attenuated. If the gap appears at the beginning, the entire saliency of Γ is attenuated by the factor ρ^g. Notice that since the network evaluates open curves starting from both endpoints, if a curve contains a relatively smooth section on one of its sides and a relatively wiggly section on its other side, then the highest score will be obtained when the gaps are distributed along the wiggly side.

Consider now a straight line Γ with gaps distributed uniformly along the line. Let p ($0 \le p \le 1$) be the fraction of the line containing the actual elements, and let $q = 1 - p$ be the fraction of the line which is virtual. We can thus set $\sigma(s) = p$. The gap length g of a line segment of length l is given by ql. Since we are dealing with a straight line, $C(0, s) = 1$ for all s. Consequently, the expected saliency of a straight line of length l with fraction q in uniform gaps is given by (Eq. 6)

$$\Phi(\Gamma) = p \int_0^l \rho^{qs}\, ds = \frac{p}{q \ln \rho}(\rho^{ql} - 1)\,. \qquad (29)$$

This score converges as l approaches infinity to

$$\Phi_\infty = -\frac{p}{q \ln \rho}\,. \qquad (30)$$

Thus, the saliency of an infinitely long straight line with uniformly distributed gaps is always finite and, in fact, proportional to p/q. Note that, since the saliency measure monotonically increases with the length of a curve, the score of an infinitely long straight line with uniform gaps provides an upper bound on the score of any finitely long line segment with the same distribution of gaps.

Examples for the values assumed by Φ_∞ as a function of p and ρ are given in Table 1. Consider $\rho = 0.7$: When 95% of the line includes actual elements (5% gaps), the score is only 53.27, and when 90% of the line includes actual elements (10% gaps), the score drops to 25.23. This means that a straight line of length 54 will be better than any line that contains 5% gaps. Similarly, a straight line of length 26 will always be better than an infinite line with 10% gaps.

A similar analysis can be performed for a circle with uniformly distributed gaps. Unlike the infinite straight line, here the circle has finite size. Given a circle with radius r and fraction p actual elements and $q = 1 - p$ virtual elements, we set $\sigma(s) = p$ for all s, $g(0, s) = qs$ and, using Eq. 15, $C(0, s) = e^{-s/r^2}$. Thus, the saliency of Γ is given by

$$\Phi(\Gamma) = p \int_0^\infty \rho^{qs} e^{-\frac{s}{r^2}}\, ds = p \int_0^\infty e^{\left(q \ln \rho - \frac{1}{r^2}\right)s}\, ds, \quad (31)$$

which, since $q \ln p < 0$, simplifies to

$$\Phi(\Gamma) = \frac{p}{\frac{1}{r^2} - q \ln \rho}\,. \qquad (32)$$

Examples for the values assumed by $\Phi(\Gamma)$, for $\rho = 0.7$, are given in Table 2. Similar to the case of straight

$p \setminus \rho$	0.1	0.3	0.5	0.7	0.9
0.5	0.43	0.83	1.44	2.80	9.49
0.7	1.01	1.94	3.37	6.54	22.15
0.9	3.91	7.48	12.98	25.23	85.42
0.93	5.77	11.03	19.17	37.25	126.10
0.95	8.25	15.78	27.41	53.27	180.33
0.97	14.04	26.86	46.65	90.65	306.88
0.99	43.00	82.23	142.83	277.56	939.63
1	∞	∞	∞	∞	∞

Table 1: Φ_∞ for a straight infinite line with uniform gaps as a function of p and ρ. Note that the score for infinite lines gives an upper bound on the score of finite ones.

$p \setminus r$	1	2	4	8	16
0.5	0.42	1.17	2.07	2.58	2.74
0.7	0.63	1.96	4.13	5.71	6.31
0.9	0.87	3.15	9.17	17.55	22.74
0.93	0.91	3.38	10.63	22.91	32.21
0.95	0.93	3.55	11.82	28.39	43.70
0.97	0.96	3.72	13.25	36.85	66.41
0.99	0.99	3.90	14.98	51.58	132.48
1	1	4	16	64	256

Table 2: The saliency values of circles with uniform gaps as a function of p and r (for $\rho = 0.7$).

lines, the saliency of circles attenuates very fast with gap size. For example, the saliency of a circle of radius 16 that contains no gaps is 256. With 5% gaps its saliency reduces to 43.70. This saliency (43.70) is identical to the saliency of a gap-free circle of radius 6.61. Similarly, with 10% gaps the saliency of the same circle reduces to 22.74, which corresponds to the saliency of a gap-free circle of radius 4.77.

Next, we analyze the case of a short curve, Γ, that lies near a circle such that the two are not touching. Again, we shall ask whether such a curve may become more salient than the circle by using the saliency of the circle. Let $\Phi(\Gamma)$ denote the saliency of Γ, let g be the gap length between Γ and the circle, and let K be the total squared curvature of Γ plus the gap to the circle. The saliency Φ_e of the first element on Γ is given by

$$\Phi_e = \Phi(\Gamma) + \rho^g e^{-K} \Phi_c. \tag{33}$$

We obtain that $\Phi_e > \Phi_c$ (recall that $\Phi_c = r^2$) when

$$\Phi(\Gamma) + \rho^g e^{-K} r^2 > r^2, \tag{34}$$

which implies that

$$\frac{1}{r^2}\Phi(\Gamma) > 1 - \rho^g e^{-K}. \tag{35}$$

Note that since $\rho < 1$ the right-hand side grows larger as the gap size increases. Consequently, the chance of an element becoming more salient than a circle by taking its saliency from the circle decreases with the gap size. Suppose finally that Γ is a straight line of length l such that its continuation is tangential to the circle, in which case $\Phi(\Gamma) = l$, $K = 0$. The condition (Eq. 35) becomes

$$\frac{l}{r^2} > 1 - \rho^g. \tag{36}$$

For $l = 1$ and $\rho = .7$ we obtain that Γ is almost never more salient than the circle:

$$\frac{1}{r^2} > 1 - .7^g \tag{37}$$

or

$$r < \frac{1}{\sqrt{1 - .7^g}}. \tag{38}$$

From this equation, r must be extremely small to allow an element to win with gaps: For $g = 1$, we have $r < 1.826$, and for $g = 2$, we have $r < 1.400$. As l increases the likelihood of Γ becoming more salient increases.

The final issue we discuss is the saliency measure's preference for how gaps are distributed along a curve. Elder and Zucker [4] conducted experiments which demonstrate that, when a fraction of the boundary of an object is missing, humans' recognition ability is hindered more when the missing fraction is contained all in one gap than when spread over several gaps. For any curve, the saliency measure encourages gaps to be as far as possible from the starting point. For an open curve with a fixed total gap length, the best and worst cases are when the curve has one large gap at the start (worst) or end (best). Since the network evaluates the saliency of curves from all possible starting points it prefers that gaps are pushed as far as possible from the smooth sections of the curve.

While for open curves there is no clear preference for a single long gap versus a few short gaps, for closed curves such a preference does exist. Consider a circle Γ with one large gap. Let Γ_1 be the open curve corresponding to the part of the circle that is actual, and let Γ_2 be the gap. The most salient element on the circle will be the first element of Γ_1, since the saliency measure prefers gaps to be as far as possible from the start of the curve. So the most salient curve will go first through Γ_1, then through Γ_2, and then loop back to Γ_1. Let αr denote the length of gap Γ_2. Since only the actual elements contribute to the saliency of a curve, the saliency obtained by going once around the circle is simply $\Phi(\Gamma_1)$. Using Eq. 10 the saliency of the circle becomes

$$\Phi(\Gamma) = \frac{\Phi(\Gamma_1)}{1 - \rho^{\alpha r} e^{\frac{2\pi}{r}}}. \tag{39}$$

If the circle now contains, say, two gap sections of the same total length αr, then the saliency obtained by going once around the circle will be reduced. This is because a gap will be closer to the start of the curve. As a consequence, the numerator in Eq. 39 will become smaller. The denominator, however, will remain unchanged since the total gap length and curvature do not change. This analysis clearly applies when the circle is fragmented by more than two gaps. Consequently, the saliency of the circle will become smaller as a result of fragmentation. An example is given in Fig. 6. The figures shows three circles of the same radius and with the same total gap size. The network prefers the one that contains one long gap over the ones in which the gaps are fragmented. This behavior disagrees with Elder and Zucker's results.

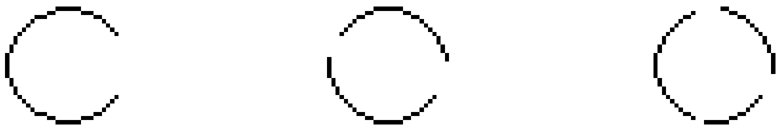

Figure 6: Three circles of the same radius with the same total gap size. Using Shashua and Ullman's network the saliency values are 46.85 (left), 27.93 (middle), and 23.27 (right).

4 Complexity and Convergence Analysis

In this section we analyze the complexity of Shashua and Ullman's saliency network. Denote the total number of pixels in the image by p and the number of discrete orientation elements at every pixel by b. The network has pb elements. At each iteration every element has to evaluate all the saliencies obtained from elements connected to it. The complexity of each iteration therefore is pb^2. The question then is how many iterations are required before the network converges. Clearly, if we did not allow cycles the longest curve may be of length p, and so the total complexity of the computation would be at most p^2b^2. But when cycles are considered, we show below that the network converges in a linear number of iterations, and so the total complexity is indeed $O(p^2b^2)$.

Given a cycle Γ, denote by Φ_n the score obtained after going n times around the cycle, by K the energy of Γ, and by g the total gap size. Then from Eq. 10 the saliency of Γ is $\Phi = \Phi_1/(1-\rho^g e^{-K})$. After going n times around the cycle, the accumulated score becomes (this is the finite sum of the geometric series in Eq. 10)

$$\Phi_n = \frac{1-\rho^{ng} e^{-nK}}{1-\rho^g e^{-K}} \Phi_1. \tag{40}$$

Define the relative error by

$$E = \frac{\Phi - \Phi_n}{\Phi} = \rho^{ng} e^{-nK}. \tag{41}$$

We can now compute the number of cycles, n, needed to achieve an $E = \epsilon$ error:

$$\ln \epsilon = n(g \ln \rho - K) \quad \Longrightarrow \quad n = \frac{\ln \epsilon}{g \ln \rho - K}. \tag{42}$$

Assume Γ is a circle of radius r with no gaps. Then $K = \frac{2\pi}{r}$ and

$$n = -\frac{r \ln \epsilon}{2\pi}. \tag{43}$$

The number of cycles around the circle is $O(r)$. Assuming one iteration covers one unit of arc length, the number of iterations for each cycle is $2\pi r$. Thus, from Eq. 43 the total number of iterations needed to achieve an ϵ error is

$$N = 2\pi r n = -r^2 \ln \epsilon. \tag{44}$$

Consequently, the total number of iterations required is $O(r^2) = O(p)$, where p is the size of the image. As an example, the number of cycles required to achieve 1% error ($\epsilon = 0.01$, $\ln \epsilon \approx -4.605$) is $n \approx 2.303\, r/\pi \approx 0.733r$, and therefore the number of iterations is $N \approx 4.605r^2$.

Fig. 7 shows an image of a gap-free and a fragmented circle on a noisy background. As expected, the Saliency Network chooses the gap-free circle as the most salient curve. Using Eq. 44, we could predict the number of iterations for the network to converge on the gap-free circle: The radius of the circle is $r \approx 11.39$, and one iteration covers an arc length $\Delta s \approx 2.983$ (r and Δs are discussed in the next section). So to obtain 1% error, Eq. 44 gives

$$N = \left(\frac{2\pi r}{\Delta s}\right) n \approx \frac{4.605 r^2}{\Delta s} \approx 200.4. \tag{45}$$

We ran the Network for 200 iterations on the left image in Fig. 7, and the maximum saliency converged to 130.8. This generally agrees (for a 1% relative error) with $r^2 = 129.8$, as predicted by Eq. 17, and with the exact saliency of the circle under discretization, 132.1 (see next section).

In Fig. 7, the input image has dimensions 128×128, and the example was run on a network with 24 orientation elements per pixel. The 200 iterations took 54 minutes using C code on a Sun SPARCstation 5 with 32M of memory. Note that this time for convergence is independent of the number of background elements. So the execution time would be the same if the gap-free circle were alone in the image. To illustrate this point, Fig. 14 shows an example of two circles, the larger of which is the gap-free circle from Fig. 7. The input image contains no clutter, but, nevertheless, as before the Saliency Network took 55 minutes to converge.

By taking the maximal possible circle in the image, we account for the worst case complexity of the network. This is because any larger closed curve must accumulate comparable energy in order not to exceed the boundaries of the image. We can therefore conclude that the worst case complexity of the network is $O(p^2b^2)$, which is the squared number of elements in the network.

5 Discrete Implementations

Our analysis of Shashua and Ullman's method has concentrated on the theoretical, continuous version of their

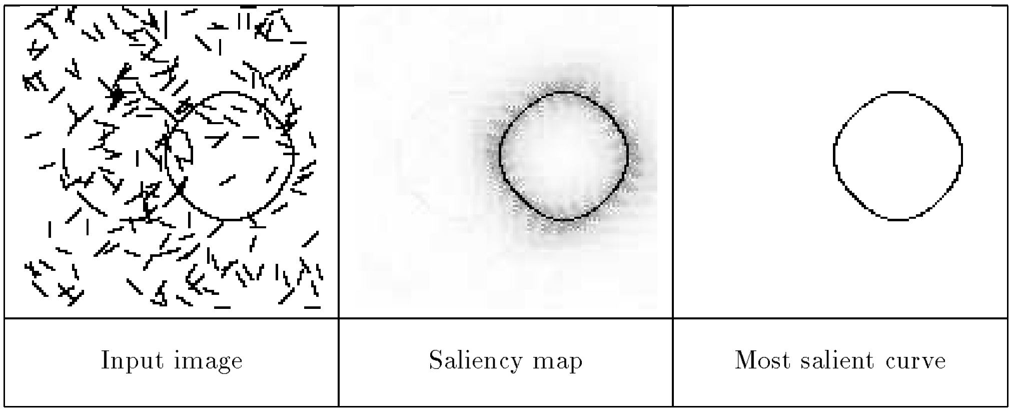

Figure 7: Running the Saliency Network on an image with gap-free and fragmented circles and a background of 200 random line segments (at the left). The saliency map and most salient curve image are shown in the center and right pictures, respectively. After 200 iterations, the maximum saliency was 130.8. The time to convergence and the maximum saliency are independent of the number of background elements.

saliency measure. Shashua and Ullman proposed to compute this measure using a network of finitely many, locally connected elements. In this section we analyze the effect of computing the saliency measure on discrete networks. We show in particular that the network is extremely sensitive to the number of discrete orientation elements allocated per pixel.

Shashua and Ullman's network has the following structure. Let p be the number of pixels in the image, and let b be the number of orientation elements at each pixel. (Shashua and Ullman set $b = 16$.) The network contains $p \times b$ processors, a processor for every orientation element at every pixel in the image. A continuous arc is assigned between every two elements that meet at the same pixel in the underlying image. The local curvature κ corresponding to such an arc is approximated using the formula

$$\kappa = \frac{2 \tan \frac{\alpha}{2}}{\Delta e}, \tag{46}$$

where α denotes the angle between the neighboring elements and Δe denotes the length of an orientation element. This formula represents the curvature of a circular arc that joins the midpoints of two elements of the same length. As an example, the gap-free circle of Fig. 7 was generated using a 24-sided regular polygon with one element per side and with $\Delta e = 3$. Then $\alpha = \pi/12$, and Eq. 46 gives $\kappa \approx .08777$. Therefore, the radius of the circular-arc approximation is $r = 1/\kappa \approx 11.39$, which gives the arc length and total squared curvature covered by one iteration to be $\Delta s = \alpha r \approx 2.983$ and $K = \alpha / r \approx .02298$, respectively.

Shashua and Ullman set Δe to be constant, and hence ignored the different sizes of elements of different orientations. As a result, a horizontal or vertical line of length l obtains the same saliency as a diagonal line of length $l\sqrt{2}$. Shashua and Ullman's implementation therefore encourages curves that are aligned with the main axes of the image.

A more critical issue is the number of orientation elements in the network. Consider for example a nearly horizontal straight line segment. Due to aliasing, the line may be cut in the middle so that one part of the line is raised up by one pixel (see Fig. 8). Let $2l$ be the length of the line. The saliency of the first element along the line is given by

$$\Phi_e = l + e^{-K} + (l-1)e^{-2K}, \tag{47}$$

where K is the total squared curvature over the change in orientation α corresponding to raising the line up by one pixel (which is also the total squared curvature for when the line returns to horizontal).

Consider now a pair of lines of length l meeting at a corner such that they form the same orientation change α. Since a corner forms only one turn the obtained saliency will be

$$\Phi_c = l + le^{-K}. \tag{48}$$

Consequently, we obtain the paradoxical result that the corner is more salient than the nearly straight continuation. Hence straight lines oriented such that they deviate slightly from horizontal will often be less salient than corners.

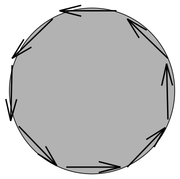

Figure 9: Discretizing a circle with a regular polygon.

The discretization problem is carried over to other, more complicated examples. Consider a circle of radius r. When r is sufficiently small, the circle can be approximated by a regular polygon where each side includes a single orientation element (Fig. 9). Let K be the total squared curvature corresponding to a turn $\alpha = 2\pi/n$, where n is the number of sides of the polygon. The discrete saliency of such a regular polygon is given by

$$\Phi \;=\; 1 + e^{-K} + e^{-2K} + \ldots$$

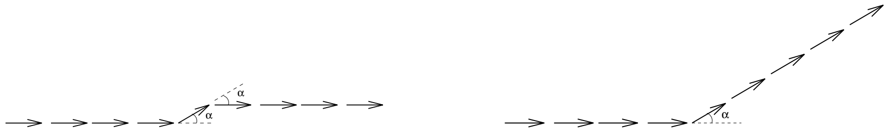

Figure 8: Discretization effect on a straight line. Left figure: the discretization of a straight line. Right figure: a corner. The saliency value obtained for a perfectly horizontal line of length 20 is 20.00, the saliency value for a straight line of the same length is 18.41, and the saliency value of a corner is 19.10.

$$
\begin{aligned}
&= \frac{1}{1-e^{-K}} \\
&= \frac{1}{1-e^{-\frac{\Delta s}{r^2}}}, \qquad (49)
\end{aligned}
$$

where Δs is the arc length of the circle that is covered in one iteration. Returning again to the gap-free circle in Fig. 7, for this circle $r = 11.39$ and $\Delta s = 2.983$ (see above), and so under discretization its saliency is 132.1. When $\Delta s / r^2$ is small,

$$
\frac{1}{1-e^{-\frac{\Delta s}{r^2}}} \approx r^2, \qquad (50)
$$

The approximation in this equation improves as r increases; this happens when the number of sides in the polygonal approximation increases and as a result fits a circle more closely. When r is big so that a good approximation by a regular polygon would require finer orientation changes (less than $2\pi/b$), a faithful discretization of the circle would involve many inflections (that is, clockwise turns balanced by counter-clockwise turns). These inflections would be penalized unduly by the network.

We could improve the saliency of the discretization if we instead represent the circle by a regular polygon with b sides; each side now contains more than one element. This, however, will still not result in a reasonable approximation to the continuous saliency of the circle. This can be seen by the following observation. Eq. 49 gives the saliency of a regular polygon with b sides, each of unit length, in terms of K, the total squared curvature assigned for a turn of $\frac{2\pi}{b}$. The saliency of a similar regular polygon in which every side is of length l is given by

$$
\Phi_l = \frac{l}{1-e^{-K}} = l\Phi_1. \qquad (51)
$$

The saliency of a regular b-sided polygon, therefore, increases linearly with the length of each side, l. Since l is directly related to the radius of the circumscribed circle, the saliency of the polygon also increases linearly with the radius of that circle. Since the continuous saliency of a circle grows quadratically with the radius of the circle (Eq. 17), we obtain that, as r grows, the saliency of the polygon will considerably underestimate the saliency of the circle.

The results shown in this section establish that the Saliency Network faces serious difficulties due to discretization of the range of orientations. A faithful implementation of the continuous saliency measure would require a very fine discretization. The number of orientation elements needed to completely avoid the problems mentioned in this section is of the order of $\sqrt{p}$, where p is the total number of pixels in the image. With this number of orientation elements the overall time complexity of the network (see Section 4) becomes $O(p^2b^2) = O(p^3)$.

6 Applications to Grouping

The Saliency Network is viewed by many people not only as a mechanism for shifting attention to salient structures, but also as a method for the initial grouping of curves. The problems of identifying salient structures and the grouping of curves are not identical. Saliency can be viewed as the problem of identifying the "odd man out," whereas grouping is the problem of identifying image structures that are likely to belong to a single object. The criteria of length and straightness can separate a smooth object from a background of short, broken curves (e.g., a disc on a background of grass), but they may be inappropriate for segmenting equally-smooth objects in cluttered scenes, since long smooth curves often will traverse a few objects. For example, Fig. 10 shows a case where the Saliency Network successively finds a curve that belongs to an object of interest, but Fig. 11 shows another case where the most salient curve traverses more than one object. Nevertheless, in many cases the salient curves may lie on objects of interest, and so may be useful for grouping.

The Saliency Network computes, for every element, the saliency of the most salient curve emerging from that element. For grouping, we would like to recover the curves that made those locations salient. In fact, we show below that, after the network converges, the most salient curves can be extracted in the following simple way, which was proposed by Shashua and Ullman. To extract the optimal (most salient) curve emerging from an element, during the computation one has to store for every element p a single pointer $\pi(p)$ which points to the second element on the optimal curve emerging from p. At the end of the computation, the best curve from p can be retrieved by tracing these pointers starting from p. To obtain the most salient curve in the image, we would trace from the most salient element.

This tracing procedure follows from the property of extensibility. The basic idea of extensibility, which is illustrated in Fig. 12, is that at convergence any suffix of an optimal curve is optimal as well. The following argument shows that at convergence the tracing procedure produces the optimal curves. At any iteration N, we know from the definition of extensibility (Eq. 3) that $\Phi(p)$ is the saliency of the most salient curve emerging from p among all curves leaving p of length less than or equal to N, and we know that $\pi(p)$ points to the next element on that most salient curve. Therefore, at

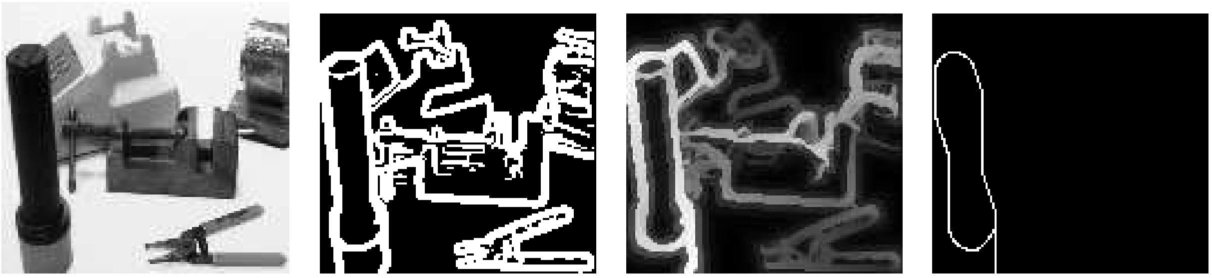

Figure 10: Shashua and Ullman's saliency map for a cluttered scene. The scene image (on the left) was smoothed with a Gaussian of standard deviation 1 and then the gradient magnitude was thresholded to get a binary image (second picture from the left). This edge image was the input to the network. The third picture displays Shashua and Ullman's saliency map, and the fourth shows the curve (71 elements) emanating from the most salient element, for which $\Phi = 263.5$.

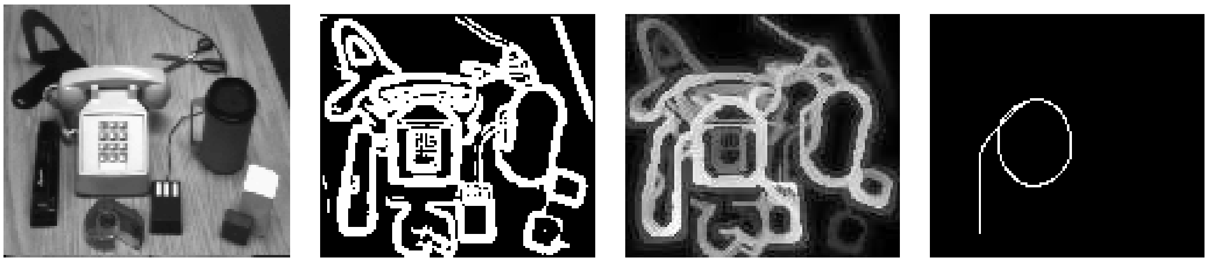

Figure 11: Shashua and Ullman's saliency map for a cluttered scene. From left to right, the first picture is the scene image and the second is an edge image obtained from the scene image by thresholding the gradient magnitude. The edge image was the input to the network. The third picture displays Shashua and Ullman's saliency map, and the fourth shows the curve (51 elements) emanating from the most salient element, for which $\Phi = 210.1$.

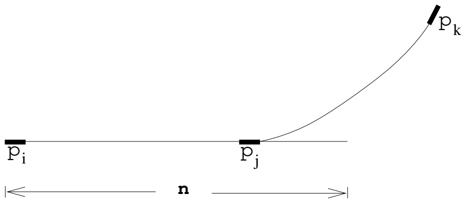

Figure 12: Characterization of extensible functions. If the most salient curve from p_i goes through p_j then, at convergence, the most salient curve from p_i must coincide with the most salient curve from p_j. At any finite time, however, the most salient curves from p_i and p_j may not overlap anywhere except at p_j. In particular, after n iterations the most salient curve from p_i will be the straight line of length n, but the most salient curve from p_j could be along the curved segment from p_j to p_k.

$N = \infty$ (i.e., at convergence), $\Phi(p)$ is the saliency of the most salient curve emerging from p among all possible curves, and $\pi(p)$ points to the next element on that curve. We will assume for simplicity that the optimal curve from p is unique. Let $\Gamma = \langle p_0, p_1, p_2, \ldots \rangle$ be the optimal curve from some element p_0. Then for any suffix Γ_i of Γ ($\Gamma_i = \langle p_i, p_{i+1}, p_{i+2}, \ldots \rangle$, $i \geq 0$), Γ_i must be the optimal curve from p_i — otherwise, if a different curve Γ_i^* were more salient than Γ_i, then from Eq. 5 we could substitute Γ_i^* for Γ_i and obtain a new curve from p_0 that is more salient than Γ. But if Γ_i is optimal, then $\pi(p_i)$ must equal p_{i+1}, since $\pi(p_i)$ points to the next element on the optimal curve from p_i. Thus following the pointers traces out the optimal curve.

The fact that the tracing procedure discussed above supplies the optimal curves has serious implications for grouping. When two curves share a common section (as in Fig. 13), the elements on the common section must decide between the two curves. So if two different objects are touching, then always the best curve through one of the objects will merge into the other; this situation is illustrated by the real image example in Fig. 11, where the boundary curves of two objects (a flashlight and a telephone) merge together.

The two-circle example can also be problematic for grouping due to the problem of leeching. Leeching can cause non-salient curves next to a salient one to include the salient curve as part of them. We have already seen an example in which, due to this property, a non-salient

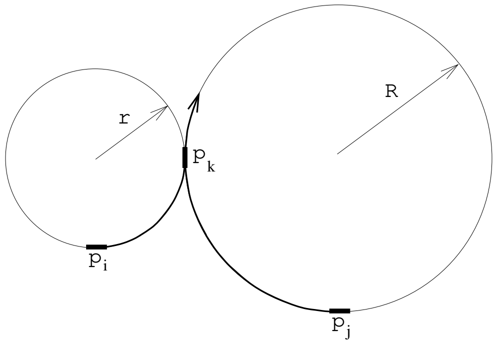

Figure 13: The problem of leeching. Each element of a curve chooses one neighboring element with which to combine. Consequently the shared element must choose between the two shapes, and so the best curves emerging from p_i and p_j will merge together. The larger circle is the most salient curve, and, for all elements p_j on the larger circle, $\Phi(p_j) = R^2$. The elements on the smaller circle draw their saliencies from the larger circle, and the saliencies decrease as the elements get further from the junction element. For every element p_i on the smaller circle, $r^2 \leq \Phi(p_i) \leq R^2$.

curve becomes salient unduly (Section 3.2). Another example is shown in Fig. 13, in which the elements on the smaller circle draw their saliencies from the larger circle, and as a result the most salient curves emanating from these elements combine with the larger circle. In addition, we show next that the smaller circle can only be traced from the least salient element over both curves; this could be problematic if a grouping system wishes to recover both circles. Consider an element p_i on the smaller circle, and let s be the arc length from p_i to the connecting element between the two circles (denoted by p_k). The saliency of the larger circle at convergence is R^2, according to Eq. 17. From Eq. 16, the saliency of a circular arc of extent s on the smaller circle is $r^2(1 - e^{-\frac{s}{r^2}})$. Finally, using Eq. 9 we can derive the saliency of p_i:

$$\begin{aligned} \Phi(p_i) &= r^2(1 - e^{-\frac{s}{r^2}}) + e^{-\frac{s}{r^2}} R^2 \\ &= r^2 + (R^2 - r^2)e^{-\frac{s}{r^2}} \end{aligned} \tag{52}$$

It can be readily seen that $\Phi(p_i)$ decreases as s, the arc length from p_i to p_k, increases. Therefore, the saliencies of the elements on the smaller circle decrease as the elements get further away from the junction element, with the constraint that $r^2 \leq \Phi(p_i) \leq R^2$. As a consequence, if a grouping system were to try and recover the smaller circle, it would have to trace the curve from the least salient element on both curves.

Fig. 14 shows the results of the Saliency Network on an analogous two-circle example. To get the optimal curves, we first traced the curve from the most salient element (for which $\Phi = 130.8$), which gave the larger circle. To compute the second most salient curve, we ignored the elements on the most salient curve and selected among the remaining elements the next most salient element. We then traced the curve from this element. The traced curve emerged from the selected element and the went around the larger circle. We repeated this process to obtain the third most salient curve. The new curve resembled the second most salient curve again, except that it was one element longer. As discussed above, elements near the most salient cycle tend to merge with the cycle and then draw their saliencies from it. The saliencies of these elements attenuate as they become further away. To extract the smaller circle, we would have to trace the curve from the least salient active element. Fig. 15 shows another example in which the same phenomenon occurs, except this time the larger circle is less salient because of gaps. Similar to the previous case, tracing from the least salient element in the image is needed to recover the larger circle.

Thus far our analysis has concentrated on the asymptotic behavior of the Saliency Network. In their experiments, Shashua and Ullman demonstrated that good results could be obtained already after a few dozen iterations. In this they relied on the property that after the n'th iteration the score attributed by the network to every element represents the saliency of the best curve of length $n+1$ emanating from the element. There is a drawback to this approach, however. Whereas after running the network for a small number of iterations the saliency values obtained for short curves already approximate their asymptotic saliencies, long curves still are undervalued significantly. This is particularly problematic when closed curves are considered, because their asymptotic scores benefit from being considered infinitely long. Thus, when the network is run for a relatively small number of iterations, closed curves are evaluated as if they were short, open curves, and as a result closure would not be encouraged by the network.

Furthermore, when the network is not run to convergence the tracing procedure is not guaranteed to extract the best curve. Consider for instance the picture in Figure 12. The picture contains a straight line of length n emerging from an element p_i, and it contains a curved segment between elements p_j and p_k, which merges into the straight line. We choose the curved segment so that after n iterations it is more salient (due to having greater length) than the portion of the straight line to the right of p_j. Consequently after n iterations $\pi(p_j)$ will point to the curved segment. As well, after the n iterations the best curve emerging from p_i will be the straight line of length n (and its current saliency will be n). But if we now trace the pointers starting from p_i, we will mistakenly think that the best curve of length n contains a portion of the curved segment between p_j and p_k. This problem could be avoided if the entire history of the computation were stored, but that of course would increase the storage space required by the method considerably.

To conclude, Shashua and Ullman's Saliency Network may be used for grouping, because it both is efficient and is guaranteed to find the optimal curves in an image, according to a measure-of-fit that prefers length, straightness, and few gaps. When the network reaches convergence, the optimal curves in the image can be extracted through a straightforward tracing procedure. The algorithm is efficient because, as we have shown in Section 4, it searches the exponential space of possible image curves

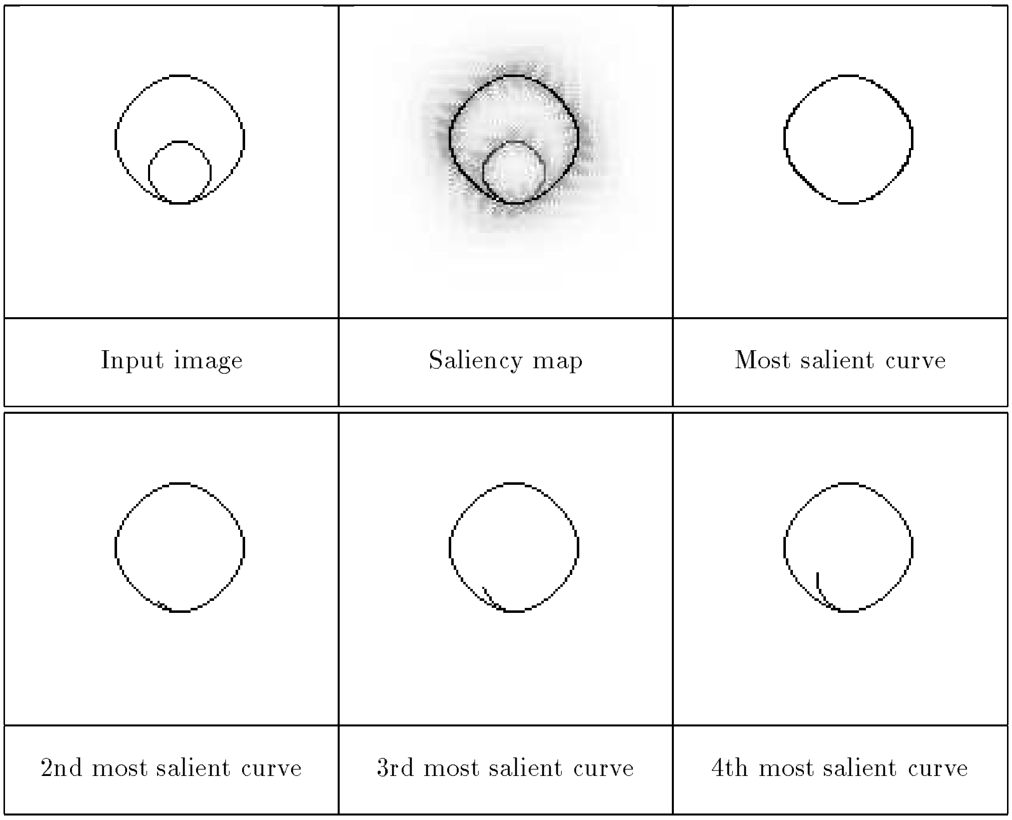

Figure 14: Shashua and Ullman's method at image junctions. The second, third, and fourth most salient curves start from the open curve attached to the circle and then proceed around the circle. The saliencies of the top four curves are 130.8, 122.1, 114.1, and 106.9. The saliency of the least salient active element is 68.2.

in time that is polynomial (either quadratic or cubic) in the size of the image. To speed-up the computation even further, Shashua and Ullman recommended running the network for a small number of iterations. If not run to convergence, however, the network is no longer guaranteed to provide the optimal curves, and for longer curves the computed saliencies can be significantly undervalued. Even if run to convergence, due to curve junctions the method still has serious problems in extracting any salient curve other than the best. The Saliency Network, therefore, may be useful for directing attention to a single object, but will be unsuitable in cluttered images for extracting a number of different objects.

7 Conclusion

The Saliency Network is a mechanism for identifying salient curves in images based on length and straightness. The method is attractive for several reasons. First, the measure of saliency generally prefers long and smooth curves over short or wiggly ones. In addition, the network is guaranteed to find the most salient structure according to the measure. While so doing, the network fills in gaps with smooth completions and tolerates noise. The network itself is locally connected and its size is proportional to the size of the image. The locality is further emphasized since the contribution of remote elements to the score of a given element attenuates with the curvature and gap length separating the remote elements from the given element.

Our analysis revealed, however, certain weaknesses with the method. We found cases in which the most salient element does not lie on the perceptually most salient curve. Furthermore, we showed cases in which the saliency measure changes its preferences when curves are scaled uniformly. Finally, we found that for certain fragmented curves the measure prefers large gaps over a few small gaps of the same total size.

We believe that the weaknesses of the Saliency Network are due largely to two important properties of the saliency measure which are imposed by the Network's computation. The two properties are (1) extensibility and (2) geometric convergence for cycles. Extensibility implies that an optimal curve must be composed of subcurves that are themselves optimal. Due to extensibility, saliencies can be computed efficiently using a procedure of recursive optimization (dynamic programming). One of the benefits of extensibility is that, although the Saliency Network finds the element from which the best curve emanates rather than extracting the best curve itself, the best curve can be extracted through a simple tracing procedure. Also due to extensibility, however, the method has difficulties at junctions; this leaves unclear how one could use the method for grouping in cluttered images.

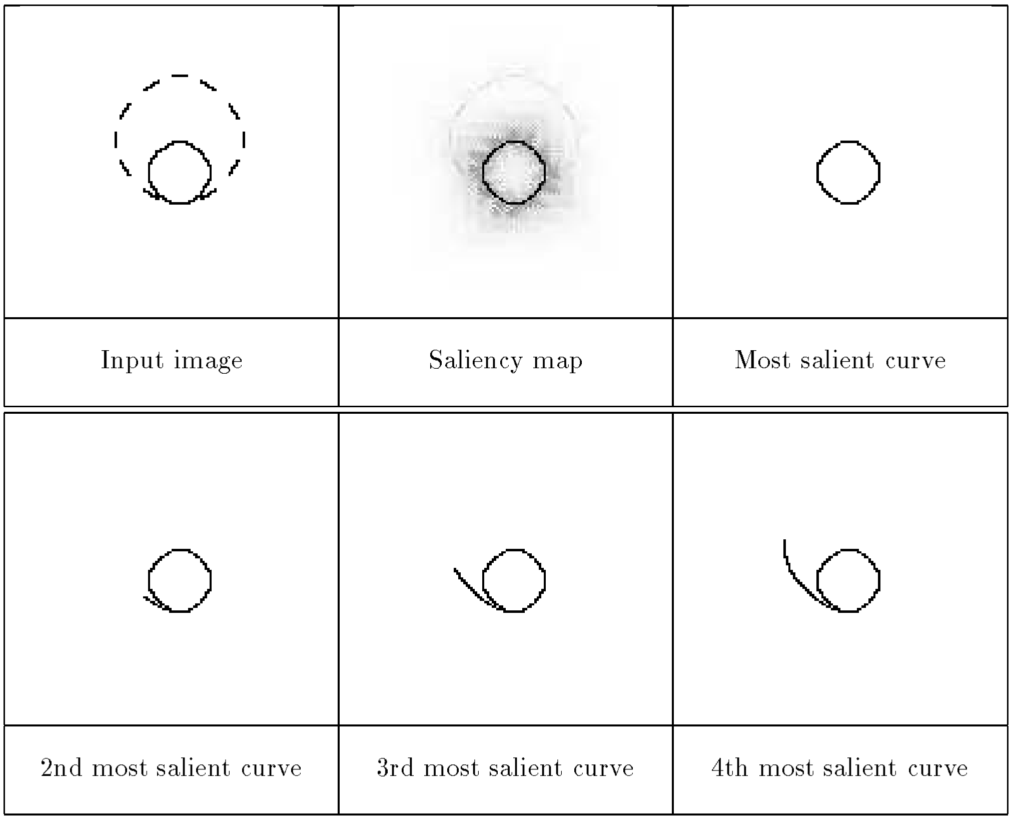

Figure 15: Shashua and Ullman's method at image junctions. The gaps in the larger circle cause it to be less salient than the smaller circle. The saliencies of the top four curves are 33.3, 13.9, 7.54, and 5.47. The saliency of the least salient active element is 4.46.

The second property exhibited by the saliency measure is that the measure decreases in a geometric series when evaluated along a cycle. This property, which is essential for convergence, was used in this paper to compute the network's time complexity. In particular, we showed that the number of iterations is linear in the size of the image, and as a consequence the overall complexity in serial implementations is quadratic in the size of the network. This complexity result is based on the assumption that the number of discrete orientations per pixel is independent of the size of the image. On the other hand, we also showed the network's rankings of curves can be significantly altered when the range of possible orientations is coarsely sampled. With proper sampling, however, the complexity of the network becomes cubic in the size of the image.

In sum, extensibility and geometric convergence enable the saliency measure to be optimized and the optimal curves to be recovered efficiently (in polynomial time), but at the same time they restrict the set of possible functions that can be used as measures of saliency. Partly due to this restriction, the chosen measure has certain properties which counter that which we believe is expected in a measure of saliency. It remains to be seen whether variations of the current measure can be defined that remedy some of its weaknesses while still allowing the saliency map and most salient curves to be computed efficiently.

Code availability

We have made available our C-code implementation of the Saliency Network. To retrieve the code, ftp to "ftp.ai.mit.edu," then log in as "anonymous," then cd to "pub/users/tda/," and then get and uncompress "susal-code.tar.Z."

Acknowledgment

The authors thank David W. Jacobs for many fruitful discussions.

References

[1] Brady, M., W. E. L. Grimson, and D. J. Langridge, "Shape Encoding and Subjective Contours," *Proc. First Annual Conf. Artif. Int.*, 1980.

[2] Brady, M., and W. E. L. Grimson, "The Perception of Subjective Surfaces," *MIT A. I. Memo No. 666*, November 1981.

[3] Bruckstein, A. M., and A. N. Netravali, "On Minimal Energy Trajectories," *Computer Vision, Graphics, and Image Processing*, 49, pp. 283–296, 1990.

[4] Elder, J., and S. Zucker, "The Integration of Figure Fragments into Representations of Planar Shape,"

McGill Research Centre for Intelligent Machines TR-93-2, February 1993.

[5] Finkel, L. H., and P. Sajda, "Object Discrimination Based on Depth-from-Occlusion," *Neural Computation*, 4, pp. 901–921, 1992.

[6] Grossberg, S., and M. Mingolla, "Neural Dynamics of Surface Perception: Boundary Webs, Illuminants, and Shape-from-Shading," *Computer Vision, Graphics, and Image Processing*, 37, pp. 116-165, 1987.

[7] Guy, G., and G. Medioni, "Inferring Global Perceptual Contours from Local Features," *Proc. DARPA Image Understanding Workshop*, Washington, D. C., pp. 881-892, 1993.

[8] Heitger, F., and Rüdiger von der Heydt, "A Computational Model of Neural Contour Processing: Figure-Ground Segregation and Illusory Contours," *Proc. of ICCV*, pp. 32–40, 1993.

[9] Herault, L., and R. Horaud, "Figure-Ground Discrimination: A Combinatorial Optimization Approach," *IEEE Trans. on Pattern Analysis and Artificial Intelligence*, 15(9), 1993.

[10] Horn, B. K. P., "The Curve of Least Energy," *ACM Trans. on Math. Soft.*, vol. 9, no. 4, pp. 441-460, December 1983.

[11] Jacobs, D. W., "Recognizing 3-D Objects Using 2-D Images," MIT Artif. Intell. Lab. Tech. Report 1416, 1993.

[12] Martelli, A., "An Application of Heuristic Search Methods to Edge and Contour Detection," *Comm. ACM*, 19(2), February 1976.

[13] Mohan, R., and R. Nevatia, "Perceptual Grouping for the Detection and Description of Structures in Aerial Images," *Proc. DARPA Image Understanding Workshop*, pp. 512–526, 1988.

[14] Mohan, R., and R. Nevatia, "Perceptual Grouping for Scene Description and Segmentation," *Trans. on Pattern Analysis and Machine Intelligence*, 14(6), 1992.

[15] Montanari, U., "On the Optimal Detection of Curves in Noisy Pictures," *Comm. ACM*, 14(5), May 1971.

[16] Mumford, D., "Elastica and Computer Vision," *Algebraic Geometry and Its Applications*, Chandrajit Bajaj (ed.), Springer Verlag, pp. 491–506, 1994.

[17] Parent, P., and S. W. Zucker. "Trace Inference, Curvature Consistency, and Curve Detection," *Trans. on Pattern Analysis and Machine Intelligence*, 11(8), 1989.

[18] Pavlidis, T., and Y.-T. Liow, "Integrating Region Growing and Edge Detection," *Trans. on Pattern Analysis and Machine Intelligence*, 12(3), pp. 225–233, March 1990.

[19] Rutkowski, W. S., "Shape Completion," *Computer Vision, Graphics, and Image Proc.*, vol. 9, pp. 89–101, 1979.

[20] Sha'ashua, A., and S. Ullman, "Structural Saliency: The Detection of Globally Salient Structures Using a Locally Connected Network," in *2nd ICCV*, pp. 321–327, December 1988, also Weizmann Institute of Science Report CS88-18, October 1988.

[21] Subirana-Vilanova, J., "Curved Inertia Frames: Visual Attention and Perceptual Organization Using Convexity and Symmetry," *MIT A. I. Memo No. 1137*, Dec. 1991.

[22] Subirana-Vilanova, J., and K. K. Sung, "Multi-Scale Vector-Ridge-Detection for Perceptual Organization Without Edges," *MIT A. I. Memo No. 1318*, Dec. 1992.

[23] Ullman, S., "Filling-In the Gaps: The Shape of Subjective Contours and a Model for Their Generation," *Biological Cybernetics*, 25, pp. 1–6.

[24] Webb, J. A., and E. Pervin, "The Shape of Subjective Contours," *Proc. National Conf. on Artificial Intelligence*, pp. 340–343, Aug. 1984.

[25] Weiss, I., "3D Shape Representation by Contours," *CGVIP*, 41, pp. 80–100, 1988.

[26] Williams, L. R., and A. R. Hanson, "Perceptual Completion of Occluded Surfaces," *Proc. IEEE Conf. Computer Vision Pat. Rec.*, Seattle, WA, June 1994. Also to appear in *Computer Vision and Image Understanding*.

[27] Williams, L. R., and D. W. Jacobs, "Stochastic Completion Fields: A Neural Model of Illusory Contour Shape and Salience," *Proc. Fifth Inter. Conf. on Computer Vision*, pp. 408–415, June 1995.