

A Model of Contour Integration in Early Visual Cortex

T. Nathan Mundhenk, Laurent Itti

University of Southern California, Computer Science Department
Los Angeles, California, 90089-2520, USA – <http://iLab.usc.edu>

Abstract. We have created an algorithm to integrate contour elements and find the salience value of them. The algorithm consists of basic long-range orientation specific neural connections as well as a novel group suppression gain control and a fast plasticity term to explain interaction beyond a neurons normal size range. Integration is executed as a series of convolutions on 12 orientation filtered images augmented by the nonlinear fast plasticity and group suppression terms. Testing done on a large number of artificially generated Gabor element contour images shows that the algorithm is effective at finding contour elements within parameters similar to that of human subjects. Testing of real world images yields reasonable results and shows that the algorithm has strong potential for use as an addition to our already existent vision saliency algorithm.

Introduction

We are developing a fully integrated model of early visual saliency, which attempts to analyze scenes and discover which items in that scene are most salient. The current model includes many visual features that have been found to influence visual salience in the primate brain, including luminance center-surround, color opponencies and orientation contrast (Itti & Koch, 2000). However, many more factors need to be included; one such factor is the gestalt phenomenon of contour integration. This is where several approximately collinear items, through their alignment, enhance their detectability. Figure 1 shows two examples where a circle is formed by roughly collinear Gabor elements. The current paper outlines our progress in building a computational model of contour integration using both currently accepted as well as novel techniques.

Over several years the topic of contour integration has yielded several known factors that should be used in shaping a model. The first is that analysis of an image for contour integration is not global, but seems to act in a global manner. That is, the overlap of neural connections in primary visual cortex (V1) rarely exceeds 1.5mm (Hubel and Weisel, 1974), which severely limits the spatial extent of any direct interaction. However, several studies have shown that contour saliency is optimal for contours with 8-12 elements, with a saturation at 12, which is longer than the spatial range of direct interaction, typically corresponding in these displays to the inter-element distance (Braun, 1999). In addition, if the contours are arranged in such a way that they form a closed shape such as a circle, saliency is significantly enhanced (Braun, 1999; Kovacs and Julesz, 1993). This suggests not only a non-local

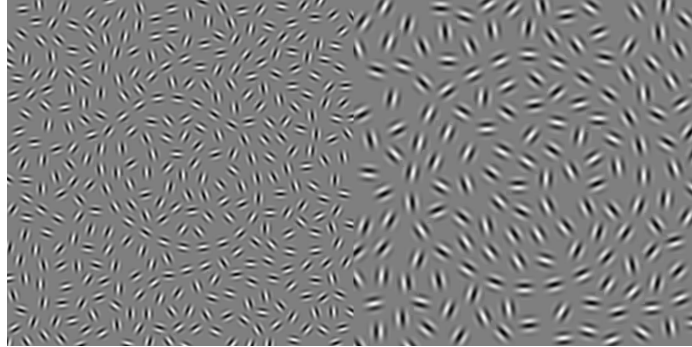
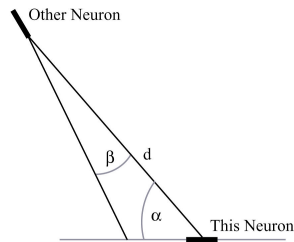


Fig. 1. Two examples of contours comprised of roughly collinear Gabor elements created by Make Snake(a Gabor element is the product of a 2D Gaussian and a sinusoidal grating).

interaction, but also a broader-range synergy between interacting neurons such that two neurons can affect each other without being directly connected.

Another noted factor playing a role in contour integration is the separation of elements usually measured in λ separation, which is the distance in units of the wavelength of the Gabor elements in the display. Studies by Polat and Sagi (1994) as well as Kapadia *et al.* (1995) indicate that an optimal separation exists for the enhancement of a central Gabor element by flanking elements. Polat and Sagi, using three Gabor elements (a test element and two flankers), found that a separation of approximately 2λ was optimal.

From these known factors several computational models have been proposed. Most start with a butterfly shape of neural connections. That is, elements are connected locally in such a way that the closer or more collinear another element is, the more the elements tend to stimulate each other (Braun, 1999). In addition many models add



Excitation is basically a function of a neurons relation to another neuron. α is the angle to the other neuron, β is the degree to which the other neurons preferred orientation points at this one. d is the distance to the other neuron. Simple excitation can be expressed as the product of the functions of α, β and d .

Fig. 2.A. The strength of interaction between two neurons is a product of α , β and d . The result is a set of 144 kernels (12 possible orientations, at each of two locations)

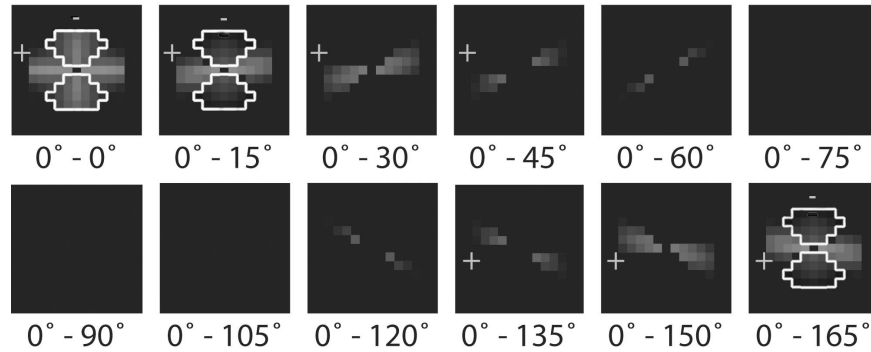


Fig. 2.B. 12 of the 144 kernels used by CINNIC are represented here. Each one in the figure show the weights of connections between a neuron with 0° preferred orientation and neurons with all other preferred orientations. The areas surrounded by a white boarder represent suppression while the other areas represent excitation. Lighter areas represent greater strength. neural suppression whereby parallel elements suppress each other. This has the effect of allowing smaller contours to be suppressed more than larger contours. Figure 2a shows how these factors combine and 2b shows what the butterfly pattern looks like in our model.

In addition to simple local connections, several other behaviors have been used in models in an attempt to explain observed long range interactions. Such methods include temporal synchronization (Yen and Finkel, 1998) and cumulative propagation (Li, 1998). It has also been suggested by Braun (1999) that a form of fast plasticity (<250 ms) may enhance synaptic transmission along contours.

The Model

The current model which we have named CINNIC (Carefully Implemented Neural Network for Integrating Contours) starts with the basic butterfly local connections, but in addition to this, we have added the use of multiscale analysis, a local group suppression gain control and fast plasticity for long range effects. Figure 3 shows a schematic description of our model, which is also described by equations (1-5). The first step is analyzing the input image using Gabor filters turned for 12 different angles. This produces 12 images that represent elements from the original image at increments of 15 degrees. A noise factor of approx. 2% is introduced at this stage. These 12 images are then reduced into three different scales 64x64, 32x32 and 16x16 pixels in resolution, which are run separately and do not interact. A 4D convolution is run to simulate the interaction between the different orientation images. The convolution is done using a set of 144 kernels that represent all possible interactions between pairs of the 12 orientation images. These kernels specify the excitation and suppression that should occur between two elements in the images. The kernels take into account the colinearity of two elements as well as their separating distance. For simplicity, interaction strength decreases as a ramp function as two elements are further separated. The kernels are statically specified at the beginning of a program run by input parameters and do not interact.

Each scale is run separately from the other. Each element is convolved against every other element within its range in such a way that collinear elements tend to excite, while parallel elements tend to suppress each other (eq. 1). This is expressed as $x_{ij\alpha}$ being the source image pixel at location (i, j) and orientation α , and $x_{kl\beta}$ being the other image pixel at location (k, l) and orientation β then taking the product of these two by the kernel $k_{\alpha\beta(k-l)(l-j)}$ (12 of which are pictured in fig. 2.B). It should be noted that m and n equal the image scale for instance 64,32 or 16. Further, $(S_{ij})^t$ is a group suppression term for the current group (detailed below) with t being the current iteration. $(P_{ij\alpha})^t$ is a plasticity term (also detailed below). The resulting potential from a single iteration is sent to a saliency map $(V_{ij})^{t+1}$. Each pixel in the saliency map represents a column of pixels from each of the twelve orientation images summed. The saliency map itself is made up of leaky integrator neurons, which lose

$$(v_{ij\alpha})^{t+1} = (S_{ij})^t (P_{ij\alpha})^t (x_{ij\alpha}) \sum_{\substack{k \in [[0, m]] \\ l \in [[0, n]] \\ \beta \in [[0, 11]]}} (x_{kl\beta}) (k_{\alpha\beta(k-l)(l-j)}) \quad (1)$$

$$(V_{ij})^{t+1} = \sum_{\substack{k \in [[0, m]] \\ l \in [[0, n]] \\ \alpha \in [[0, 11]]}} (v_{kl\alpha})^{t+1} - L \quad (2)$$

$$(S_{ij})^t = \upsilon \left[\sum_{(k,l) \in N_i \times N_j} ((V_{kl})^t - (V_{kl})^{t-1}) \right] - T \quad (3)$$

with

$$N_i = [[i-(m/8); i+(m/8)]]$$

$$N_j = [[j-(m/8); j+(m/8)]]$$

$$(P_{ij\alpha})^t = (v_{ij\alpha})^t (C) \quad (4)$$

$$I_{ij} = \text{sig}((V_{ij})^t) \quad (5)$$

some constant potential L from one iteration to the next (eq. 2). To form a final saliency map for one of the three image resolutions, the potential from the leaky integrator neurons are fed through a sigmoidal function that simulates neural firing patterns (eq. 5) with I_{ij} being the final saliency map pixel for this scale.

Non-linearities are introduced in the form of the group suppression gain control (eq. 3) where T is the threshold constant and $(V_{kl})^t$ is the potential for a neuron in this group which are all summed for that group with υ as a constant multiplied by that sum. m and n represent the image size at that scale. The suppression is based upon the rate of the change of excitation. Fast plasticity (eq. 4) is introduced as $v_{ij\alpha}$ being the potential this neuron had multiplied by a constant C . The fast plasticity works by

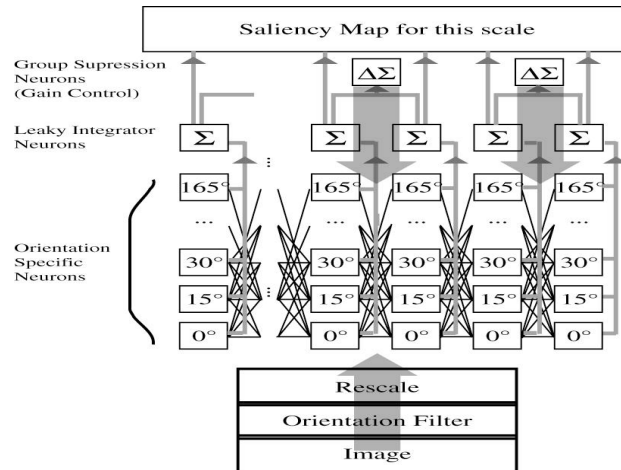


Fig. 3. This is a basic representation of the CINNIC algorithm. An input image is filtered, rescaled then interacted with images at other orientations including itself. The output goes to a saliency map of leaky integrator neurons. Group suppression is fed back from the change in group potential.

increasing all weights for a single simulated neuron, proportionally to the excitation it received in the previous iteration. That is, neurons that are stimulated more tend to stimulate collinear neighbors more as well as to suppress parallel neighbors more. This function is introduced to re-create non-local interactions that are observed in human subjects in an attempt to account for observed contour closure effect. The fast plasticity used here is bounded to 5 times the original connection strength for any given neuron.

IN our model the usage of fast plasticity was chosen for several reasons. The first was the suggestion by Braun (1999) that other methods that attempted to explain contour closure either occurred too fast such as cumulative propagation (Li, 1998), or were too slow such as temporal synchronization (Yen and Finkel, 1998) as to explain the time it takes for closure effect to happen which Braun measured at about 250 ms. Second we wished to test the idea of fast plasticity and find if it was a viable mechanism for explaining closure effect.

Another non-linearity introduced is a simple local gain control using group suppression. Neurons are grouped into local neighborhoods of size $1/8 \times 1/8$ pixels of the image size at the current scale (e.g. 8×8 pixels for a 64×64 pixel scale). If the total change in potential from a group surpasses a threshold then the neurons increase their suppression of parallel neighbors proportionally to the increase past threshold. The group includes all neurons in all orientation maps for a given visual location, which report to the same image location. There is no current cap on how much additional suppression can be added using this method.

The algorithm runs for eight iterations, which was a number chosen based upon its observed optimality. After the final iteration, the three scales are brought back together and combined using a fixed weighted average. This average is the total saliency map of contours for the input image. The entire process takes approximately

2 minutes using an Athlon 1400 MHz based PC running Linux. The time is mostly due to the enormous amount of computation needed to compute interactions between neurons from all possible pairs of the 12 images using 144 2D kernels.

Testing on Artificial Images

To tune our algorithm to human vision we are currently using a special program called Make Snake provided and created by J Braun (1999), to generate test images in which a salient contour is embedded among noise elements. Using these stimuli, we tested under which conditions our algorithm would detect the contour as being the most salient image element.

Make Snake creates images like the one presented in figure 1. The output is several Gabor patches aligned with randomized phase into a circular contour. The circle itself is carefully morphed by the program using energy to flex the joints of an “N-gon” to create a variety of circular contour shapes. The circles made up of foreground elements are controlled for the number of elements as well as the spacing in λ sinusoidal wavelengths. The elements can also be specified in terms of size and period. Background noise Gabors are added randomly. They are placed in such a way that they are moved like particles in liquid to a minimum spacing specified by the user. Gabors are added and floated until minimum spacing requirements are satisfied. The end result can also create accidental smaller contours among the noise background elements.

Test images were created using two different Gabor sizes, a small Gabor (70 pixels wide with a 20 pixel period) and a large Gabor (120 pixels wide with a 30 pixel period). The background elements were kept at a constant minimum spacing (48 for the smaller Gabors and 72 for the larger Gabors). Spacing for larger Gabors foreground elements was varied between 2 and 3.5λ in steps of 0.1666. This was constrained since values above 3.5 made the circle larger than the images frame itself. The smaller Gabors had more leeway and could be varied from 1.5 to 6λ in steps of 0.5. For both Gabor sizes, the minimum size is set the way it is because below this, the foreground elements begin to overlap. It should be noted that the ratio of foreground separation to the minimum background separation was the same for both large and small Gabor patch conditions given the same λ .

For each condition, Gabor size and foreground spacing, 20 images were created. An output mask was also created representing where foreground elements were positioned. This was used for later statistical analysis. In all, 400 images were run.

Statistical analysis was done by taking the output salience map from CINNIC, which always ran with identical model parameter settings for all images, and comparing it to the mask; this was done by looking for the top most salient points in the salience image. When a salient point was found, the local region was flooded to prevent the same element area from being counted twice. Salient points were marked as first, second, third and so on depending on its value in the salience map. Analysis was done by finding the most salient point in an image, which was also found within the foreground element mask. The rank of the most salient point also within the mask was the rank given to the image. The number of images of each rank was summed to

find out, for instance, how many images had their most salient point also lie within the mask (ranked as 1st).

As can be seen in figure 4, for the larger gabor images the most salient point falls on the foreground circle in 19 out of 20 images for separations from 2.33 to 2.833 λ , with the most salient point being found on all circles at a separation of 2.833 λ . For smaller elements, in 19 out of the 20 images the most salient point was found in the foreground at a separation of 2.5 λ . It should also be noted that the optimal results were obtained for the large Gabor size set with a ratio of 1.181 between the foreground element separation and background separation. The ratio for the smaller elements was optimal at 1.041. This means that optimal results were obtained with a slightly greater distance between foreground contour elements than background elements. Bumps in figure 4 can probably be accounted for as an artifact of the

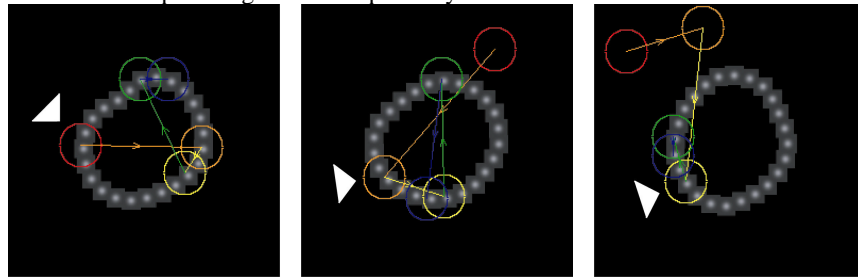


Fig. 4.A The top row of images shows contour image masks from make snake super imposed with what CINNIC found as the 5 most salient contour points. The arrow shows the most salient point CINNIC found that also lied on a contour circle. The first image is ranked as a 1st rank image since the most salient point in the image also lies on a contour. The second image is ranked as a 2nd rank image since the second most salient point is the first point to fall on a contour circle. Continuing this example the right most image is ranked as 3rd rank since the most salient point to be found on a contour circle is the third most salient point in the image

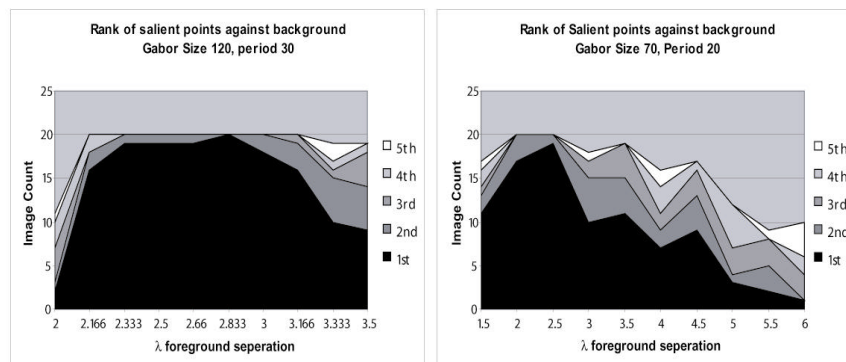


Fig. 4.B The bottom row image illustrates that as separation of foreground elements increases the likelihood off a contour element being found most salient also decreases for both Gabor sizes. Image count shows how many of the 20 images tested in each condition fall into one of the five different category ranks of saliency, or were not salient within the top five ranks.

discrete nature of the interactions between kernel and the image pixels.

A question raised by our results is that of why there seems to be an optimal separation distance in the data while an optimal distance is not explicitly defined in the neural connection weights (remember that weights decay linearly with distance). Further experimentation revealed that this was due to the group suppression gain control setting. We found that at a higher gain control threshold that saliency in the smaller Gabor size images was reduced dramatically due to an increase in noise between irrelevant Gabors. Going in the opposite direction shows that the optimal distance for the larger Gabor size increases with a lowered threshold for the gain control. This is due to the closer elements over exciting past the lower threshold. These results are interesting in that they not only explain why we obtain optimal distances, but it allows our algorithm to agree with research by Polat and Sagi (1994) who also found an optimal distance between Gabor elements.

Testing on real world images

Part of the goal of our project has been to be able to incorporate the CINNIC algorithm to our more general bottom-up visual saliency model. Thus, CINNIC must be able to analyze real world images much the same as the current saliency model does. At this point testing on real world images has been constrained to running the algorithm and inspecting the outputs to make sure that they seem reasonable. Although of a purely subjective and qualitative nature, such experimentations are particularly useful to estimate the applicability of our model to more general classes

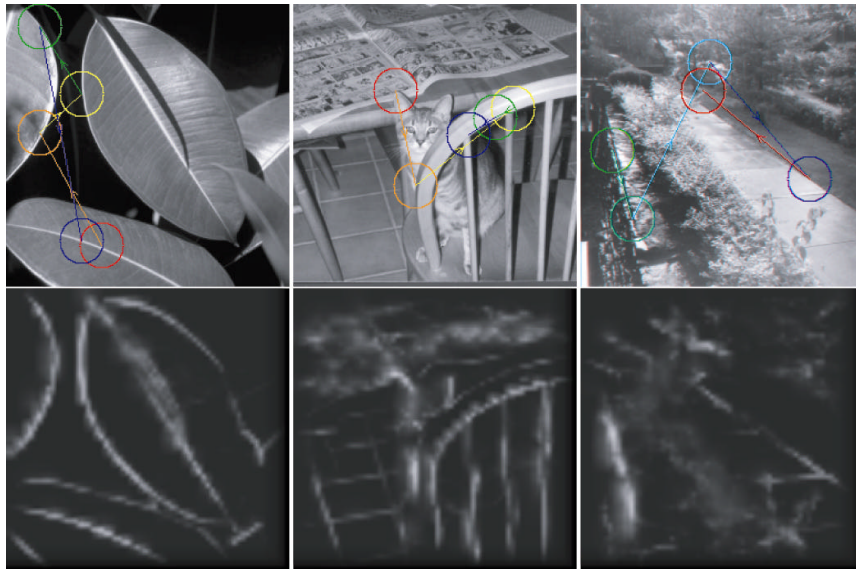


Fig. 5. The top row are real world images with the top 5 most salient points circled by the algorithm. The bottom row represents the raw saliency map for each image.

of stimuli. We found that in most cases, the algorithm behaves in a reasonable fashion when processing real world images. Noise is generally filtered out and the most salient points tend to lie on reasonable line elements and contours. Over 100 images were inspected in all; each image contained a different subject (e.g. plants, city pictures, wilderness images). This was done by selecting one image at random from each image subject category on two image library CD's. Figure 5 shows some typical outputs on real world images. The top five most salient points are circled in the images.

From the three images presented, the most salient contours can be found in 5(a) on the edges and stem of the leaves, on the cats ear and the chair rim in 5(b) and in 5(c) the stone wall, sidewalk and street possess the most salient contours. This agrees with a common sense idea that the most salient contours are found on objects with long collinear continuity and is supported by data from Braun (1999) and Hess and Field (1999) which shows that longer contours with smoother continuity and more elements tend to be more salient than shorter more jagged ones.

Discussion

On artificially-generated contours CINNIC performs very well. Performance for identifying generated contours drops as foreground elements are separated relative to background elements. This is to be expected as the same thing is observed in human subjects (Braun, 1999). Not only does it perform well but it also has an optimal distancing between foreground Gabor elements, which agrees with Polat and Sagi (1994) and the performance of detection begins to drop at a foreground-to-background ratio of about 1.25, which is what is observed by Braun (1999). The fall-off for the model is complete at a spacing of about 6λ which is consistent with the spacing range proposed by Hess and Field (1999), which they estimate to be about 4 to 6λ at maximum.

The model so far is also successful because all elements included in the model are biologically plausible. This is because the model is built upon basic long-range neural interactions (Gilbert et al., 2000). Our additions to this basic model include a previously untested neural plasticity and novel local gain control factors. It is our opinion that all of these factors are plausible and could help explain how contour integration occurs. It should also be noted that our use of fast plasticity may better explain long range interactions than previous models relying on synaptic transmission (Li, 1998; Pettet *et al.*, 1998), temporal synchronization (Yen and Finkel, 1998), or NMDA-mediated plasticity (Braun *et al.*, 1994) since, as Braun states, the time required for their mechanisms do not closely match observed times needed for salience in contour integration under contour closure. (Braun, 1999). That is, synaptic transmission would cause contour closure far sooner than the approximately 250 ms observed by Braun, while temporal synchronization and NMDA-mediated plasticity take slightly too long.

It should also be noted that our novel usage of a group suppression was successful. As we found, it seemed to have an optimal value where by removing it, turning it up too high or in general adjusting it too high or too low yielded sub-optimal results. This suggests to us that it is useful in our approach.

Acknowledgements

We would like to acknowledge Jochen Braun, Christof Koch and Mike Olson for their help and suggestions. This research is supported by the National Imagery and Mapping Agency, the National Science Foundation, the National Eye Institute, the Zumberge Faculty Innovation Research Fund and the Charles Lee Powell Foundation.

References

- Braun J, Niebur E, Schuster H G, Koch C, 1994, Perceptual contour completion: a model based on local anisotropic, fast-adapting interactions between oriented filters, *soc. Neurosci. Abstr.*, **20** 1665
- Braun J, 1999, On the detection of salient contours, *Spatial Vision*, **12**(2):211-225
- Gilbert C, Ito M, Kapadia M, Westheimer G, 2000, Interactions between attention, context and learning in primary visual cortex, *Vision Research*, **40**(10-12):1217-26
- Hess R, Field D, 1999, Integration of contours: new insights, *Trends in Cognitive Science*, **3**(12):480-486
- Hubel D H, Wiesel T N, 1974, Uniformity of monkey striate cortex: a parallel relationship between field size, scatter and magnification factor, *Journal of Comparative Neurology*, **158**:295-306
- Itti L, Koch C, 2000, A saliency-based search mechanism for overt and covert shifts of visual attention, *Vision Research*, **40**(10-12):1489-1506
- Kapadia M K, Ito M, Gilbert C D, Westheimer G, 1995, Improvement in visual sensitivity by changes in local context: Parallel studies in human observers and in V1 of alert monkeys, *Neuron*, **15**:843-856
- Kovacs I, Julesz B, 1993, A closed curve is much more than an incomplete one: Effect of closure in figure-ground segmentation, *Proceedings of the National Academy of Science USA*, **90**:7495-7497
- Li Z, 1998, A neural model of contour integration, *Neural Computation*, **10**:903-940
- Pettet M W, McKee S P, Grzywacz N M, 1998, Constraints on long-range interactions mediating contour integration, *Vision Research*, **38**:865-879
- Polat U, Sagi D, 1994, The architecture of perceptual spatial interactions, *Vision Research*, **34**(1):73-78
- Yen S, Finkel L H, 1998, Extraction of perceptually salient contours by striate cortical networks, *Vision Research*, **38**(5):719-741