Ben-Shahar, O., Huggins, P.S., and Zucker, S.W., "On computing visual flows with boundaries: The case of shading and edges," Biologically Motivated Computer Vision, Tubingen, November 2002; LNCS 2525, Springer-Verlag, 189 - 198.

On Computing Visual Flows with Boundaries: The Case of Shading and Edges

Ohad Ben-Shahar, Patrick S. Huggins, and Steven W. Zucker

Department of Computer Science and Interdisciplinary Neuroscience Program,
Yale University,
New Haven, Connecticut 06520, USA
{ben-shahar,huggins,zucker}@cs.yale.edu

Abstract. Many visual tasks depend upon the interpretation of visual structures that are flow fields, such as optical flow, oriented texture, and shading. The computation of these visual flows involves a delicate tradeoff: imaging imperfections lead to noisy and sparse initial flow measurements, necessitating further processing to infer dense coherent flows; this processing typically entails interpolation and smoothing, both of which are prone to destroy visual flow discontinuities. However, discontinuities in visual flows signal corresponding discontinuities in the physical world, thus it is critical to preserve them while processing the flow. In this paper we present a computational approach motivated by the architecture of primary visual cortex that directly incorporates boundary information into a flow relaxation network. The result is a robust computation of visual flows with the capacity to handle noisy or sparse data sets while providing stability along flow boundaries. We demonstrate the effectiveness of our approach by computing shading flows in images with intensity edges.

1 Introduction

Many visual structures, including optical flow [7], oriented texture [14], and shading [4], appear as flow fields in the image plane. Accurate knowledge of the geometry of these flows is a key step toward the interpretation of visual scenes, both two dimensional [13, 14] and three dimensional [19, 21].

Perceptually, visual flow fields are characterized by their dense, smoothly varying (almost everywhere) oriented structure. Formally a flow is an orientation function $\theta(x, y)$ over the image plane. Initial measurements of a flow field, made locally with imperfect sensors (like V1 receptive fields), are likely to suffer from noise and perhaps even fail altogether in some regions of the image. Hence, an effective computational process for the inference of coherent visual flow must be able to do so from a noisy, incomplete set of measurements. Furthermore, as we discuss below, it should localize singularities, reject non-flow structures, and behave appropriately along line discontinuities and boundaries. We have developed a computational model that addresses all these issues within a framework that is inspired by the columnar architecture of the primary visual cortex [2, 1].

The requirement that flow discontinuities be preserved is particularly important since their presence typically indicates physical discontinuities in the world. However, there is a delicate trade-off between handling sparse and noisy data sets and achieving stability along boundaries. To solve this some nonlinear interaction between the flow and its boundaries needs to be made explicit. In this paper we describe a biologically motivated way of achieving that goal.

Of the different types of visual flows, one – the *shading flow field* [4] – has a special relationship to its corresponding visual boundaries. The relationship between the geometry of the shading and that of the bounding edge provides a basis for classifying edges [4], and can be used to resolve occlusion relationships [10]. Since shading flow boundaries (i.e., the curves along which the flow should be segmented into coherent parts) are defined as intensity edges, shading is a clear example of a visual flow for which both the flow field and its boundaries can be directly measured from the image. Thus in this paper we focus on the case of shading and edges.

2 The Differential Geometry of Coherent Visual Flows

Given that the initial measurements of a visual flow field may contain spurious or missing values, we would like to refine the flow field so as to counteract these effects. Interpolating and fitting [18], smoothing [17], and diffusing [20] the orientation function $\theta(x,y)$ corresponding to the flow are commonly used approaches to achieving this goal, but they also prone to affect the underlying geometry of the flow in undesirable ways. In particular, they can distort flow singularities that must be preserved to correctly interpret visual scenes [1,2]. To overcome this problem we process the visual flow by enforcing local coherence, that is, by ensuring that each local measurement of the flow field is consistent with its neighboring measurements. Toward this we first examine the differential geometry of visual flows.

A natural representation of a visual flow which highlights its intrinsic geometry is its *frame field* [15]. Here a local frame $\{E_T, E_N\}$ is attached to each point q of the flow, with E_T tangent and E_N normal to the flow. Small translations in direction V from the point q rotate the frame, a change which is characterized through the covariant derivatives $\nabla_V E_T$ and $\nabla_V E_N$ of the underlying pattern. Cartan's connection equation [15] expresses these covariant derivatives in terms of the frame field itself:

$$\begin{pmatrix} \nabla_V E_T \\ \nabla_V E_N \end{pmatrix} = \begin{bmatrix} 0 & w_{12}(V) \\ -w_{12}(V) & 0 \end{bmatrix} \begin{pmatrix} E_T \\ E_N \end{pmatrix} . \tag{1}$$

The connection form $w_{12}(V)$ is a linear function of the tangent vector V and can thus be represented by two scalars at each point. In the basis $\{E_T, E_N\}$ these scalars are defined as $\kappa_T \triangleq w_{12}(E_T)$ and $\kappa_N \triangleq w_{12}(E_N)$, which we call the *tangential curvature* and the *normal curvature* - they represent the rate of change of the flow's dominant orientation in the tangential and normal directions,

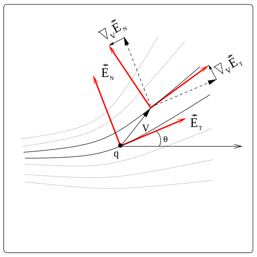


Fig. 1. The frame field representation of visual flows. The local behavior of the frame is described by its covariant derivatives $\nabla_V E_T$ and $\nabla_V E_N$ which are always normal to E_T and E_N, respectively. Since the connection form – the operator that describes the frame's rotation for any direction V – is linear, it is fully characterized by two numbers computed as projections on two independent directions. In the basis of the frame this yields the curvatures κ_T and κ_N.

respectively. In terms of $\theta(x, y)$ and its differential, these curvatures become:

$$\begin{aligned} \kappa_T &= d\theta(E_T) = \nabla\theta \cdot E_T = \nabla\theta \cdot (\cos\theta, \sin\theta) \\ \kappa_N &= d\theta(E_N) = \nabla\theta \cdot E_N = \nabla\theta \cdot (-\sin\theta, \cos\theta). \end{aligned} \tag{2}$$

Knowledge of E_T, E_N, κ_T, and κ_N at a point q enables us to construct a local approximation to the flow which has the same orientation and curvatures at q; we call such an approximation an *osculating flow field.* The osculating flow field is important in that it predicts flow values in the neighborhood of q. Comparing these predictions to the measured flow values indicates how consistent the measured values of the flow at q are with those at its neighbors and suggests how to update them to be consistent. An analogous technique to refine curve measurements using cocircularity was developed in [16].

There are an infinite number of possible osculating flow fields to choose from. However, there exist criteria for "good" osculating flow fields. One such criterion is the minimization of the harmonic energy $E[\theta] = \int\int \|\nabla\theta\|^2 dxdy$ associated with the orientation function of the flow, as is used in orientation diffusion [20]. Viewing the orientation function as a surface in $\mathbb{R}^2 \times \mathcal{S}^1$, however, suggests that the osculating flow field should minimize the surface area $A[\theta] = \int\int \sqrt{1 + \theta_x^2 + \theta_y^2} dxdy$. Finally, the duality of the curvatures κ_T and κ_N suggests that the osculating flow field should exhibit unbiased curvature covariation. Surprisingly, there is a unique osculating flow field which satisfies all of these criteria simultaneously [1, 2]. In the space $\mathbb{R}^2 \times \mathcal{S}^1$ of orientations over the image plane it takes the form of a *right helicoid* (Fig. 2):

Proposition 1. *Assume (w.l.o.g) that a visual flow $\theta(x, y)$ satisfies $q = (0, 0)$ and $\theta(q) = 0$, $\kappa_T(q) = K_T$, and $\kappa_N(q) = K_N$. Of all possible orientation functions $\theta(x, y)$ around q that satisfy these constraints, the only one that extremizes both $E[\theta]$ and $A[\theta]$, and has curvature functions that covary* identically *(i.e., $\frac{\kappa_T(x,y)}{\kappa_N(x,y)}$ =const= $\frac{K_T}{K_N}$) is the right helicoid $\theta(x, y) = \tan^{-1}(\frac{K_T x + K_N y}{1 + K_N x - K_T y})$.*

Armed with a model of the local structure of visual flow we are in a position to compute a globally coherent flow, the procedure for which is described in the next section.

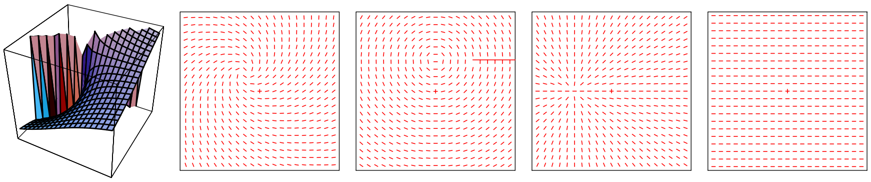


Fig. 2. Examples of right helicoidal visual flows, both in (x, y, θ) space (left) and the image plane. Note that tuning to different curvatures at the origin (point marked with a cross) produces qualitatively different coherent behaviors in its neighborhood.

3 Computing Coherent Visual Flows

The advantage of having a model for the local behavior of visual flow lies in the ability to assess the degree to which a particular measurement is consistent with the context in which it is embedded. This, in turn, can be used to refine noisy measurements, remove spurious ones, and fill in "holes" so that global structures become coherent. A framework in which one can pursue this task by iteratively maximizing the average local consistency over the domain of interest is *relaxation labeling* [11]. We have developed such a network for the organization of coherent visual flows [2]. The following is a short overview of that system.

A direct abstraction of the relaxation process for visual flow should involve an image-based 2D network of nodes $i = (x, y)$ (i.e., pixels) whose labels are drawn from the set $\Lambda = \{\text{no-flow}\} \cup \{(\theta, \kappa_T, \kappa_N) \mid \theta \in (-\frac{\pi}{2}, \frac{\pi}{2}] \ , \ \kappa_T, \kappa_N \in [-K, K]\}$ after the appropriate quantization. To allow for the representation of either "no-flow" or multiple flows at a pixel, we replace this abstraction with a 5D network of nodes $i = (x, y, \theta, \kappa_T, \kappa_N)$ whose labels are either $TRUE$ (T) or $FALSE$ (F). For each node i, $p_i(T)$ denotes the confidence that a visual flow of orientation θ and curvatures κ_T, κ_N passes through pixel (x, y). Since $p_i(F) = 1 - p_i(T)$ we need to maintain and update the confidence of only one label at each node.

The geometrical compatibilities $r_{ij}(\lambda, \lambda')$ that drive our relaxation process are computed from the osculating flow field as defined by the right helicoid. Measurement quantization implies that every possible node i represents an equivalence class of measurements, each of which induces a field of compatible labels in the neighborhood of i. In the continuum, the union of all these fields forms a consistent 5D "volume" that after quantization results in a set of excitatory labels. See Fig. 3

With the network structure, labels, and compatibilities all designed, one can compute the support $s_i(\lambda)$ that label λ at node i gathers from its neighborhood. $s_i(\lambda)$ is typically the sum of the individual support of all labels j in the neighborhood of i

$$s_i(\lambda) = \sum_j \sum_{\lambda'} r_{ij}(\lambda, \lambda') p_j(\lambda') \ . \tag{3}$$

Having computed the support for a label, $s_i(\lambda)$ is then used to update the confidence $p_i(\lambda)$ by gradient ascent, followed by non-linear projection. Under the

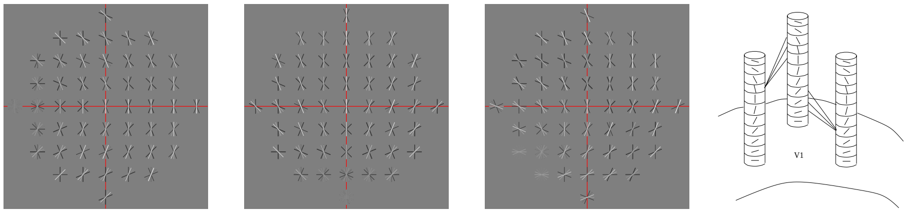


Fig. 3. Examples of compatibility structure (for different values of θ, κ_T and κ_N) projected onto the image plane (brightness represents degree of compatibility, black segments represent an inhibitory surround). As is illustrated on the right, these structures are closely related to long range horizontal connections between orientation columns in V1.

2-label paradigm and the appropriate weighing of negative (F) versus positive (T) evidence [2], the projection operator takes a particularly convenient form and the update rule reduces to

$$p_i(\lambda) \leftarrow \Pi_0^1 \left(p_i(\lambda) + \delta s_i(\lambda) \right) \tag{4}$$

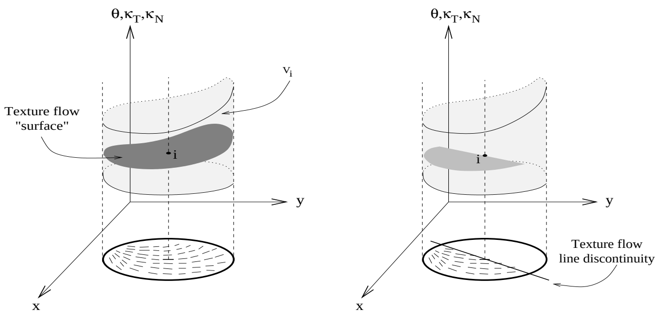
where $\Pi_0^1(x)$ projects its operand to the nearest point on the interval $[0,1]$ and δ is the step size of the gradient descent.

While the relaxation labeling network described is an abstraction based on the differential geometry of flow fields, it is motivated by the architecture of the primary visual cortex. The columnar structure of V1 clearly lends itself to the representation of orientation fields [9], and is capable of the necessary curvature computations [6]. Considerable speculation surrounds the functional significance of long-range horizontal connections [8] between orientation columns; we posit that they may play a role not unlike the compatibility structures of our network (Fig. 3, right panel).

3.1 Stability at Discontinuities

In computing coherent visual flows it is important to respect its discontinuities, as these often correspond to significant physical phenomena. The relaxation process described above does not destroy these structures because in the high dimensional space in which it operates the flow structures that meet along a line discontinuity, either in orientation or curvature, are separated and thus do not interact. However, without proper tuning, the relaxation process will quickly shrink or expand the flow in the neighborhood of boundaries. It is this behavior we seek to suppress.

To achieve stability we normalize the compatibility function, and thus the support function $s_i(\lambda)$, to account for reduced support in the neighborhood of a discontinuity. Given the compatibility volume V_i which corresponds to a particular node i, we compute the maximal support a node can receive, s_{max}, as the integral of the compatibility coefficients assuming a consistent flow traverses



(a) Maximal expected support (b) Minimal acceptable support

Fig. 4. Practical stability of the relaxation labeling process at line discontinuities in the flow can be achieved through the normalization of the support function. (a) At each node i, s_{max} is determined by integrating the support gathered from a full confidence, compatible flow that traverses the entire compatibility volume V_i. (b) The minimal accepted support s_{min} of a flow of some minimally accepted confidence $\rho_{min} < 1$ (depicted here by the brighter surface intensity) that terminates along a line that intersects i.

V_i with all supporting nodes at full confidence (Fig. 4). It is clear that the closer i is to a flow discontinuity, the less context supports it. At the discontinuity, the flow should neither grow nor shrink, leading us to define the minimal level of support for which no change in confidence occurs, s_{min}. Observe that s_{min} depends on both the geometry of the discontinuity and the minimally accepted confidence of the supporting nodes. For simplicity we assume the discontinuity (locally) occurs along a straight line. The support from neighboring nodes of minimally accepted average confidence ρ_{min} (Fig. 4) can be approximated as $s_{min} = \frac{\rho_{min} s_{max}}{2}$. Normally ρ_{min} would be set to 0.5, which is the minimal confidence that cannot be disambiguated as the $TRUE$ label. In the context of the two-label relaxation labeling paradigm and the gradient ascent update rule (Eq. 4), a decrease in the confidence of a label occurs only if $s_i < 0$. Thus, it remains to normalize the support values by mapping the interval $[s_{min}, s_{max}]$ to the unit interval $[0, 1]$ via the transformation $s_i \leftarrow \frac{s_i - s_{min}}{s_{max} - s_{min}}$ before applying the update rule.

The result of the normalized relaxation process is usually very good (Fig. 5). Nevertheless, the fact that both the support function (Eq. 3) and the normalization are linear creates a delicate balance: while better noise resistance suggests smaller s_{min}, it also implies that at discontinuities the flow will eventually grow uncontrollably. Some solutions to this problem are discussed in [2]. However, in the case of shading flow fields, discontinuities are intensity edges and thus can be explicitly identified by edge detection. As we discuss below, this information can be directly embedded into the network to decouple the handling of discontinuities from the support normalization.

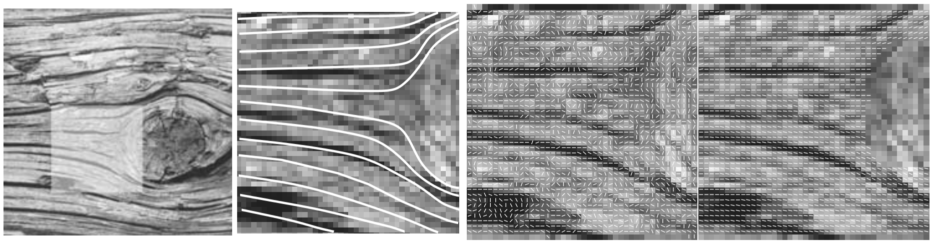


Fig. 5. Visual flow organization based on right helicoidal compatibilities. Shown (left to right) are: Tree bark image and a region of interest (ROI), perceptual structure (drawn manually), initial flow measurements (gradient based filter), and the relaxed visual flow after few iterations of relaxation labeling with the right helicoidal compatibilities. Compare the latter to the perceptual structure and note how the non-flow region was rejected altogether.

4 Edges as Shading Flow Boundaries

Edges in images are important because they signify physical changes in a scene; hence the numerous efforts to detect them. The physical nature of an edge is often discernible from the appearance of the edge in the image. In particular, the relationship between the edge and the shading flow field in the neighborhood of the edge can be used to identify the physical cause of the edge. The shading flow field is defined as the unit vector field aligned with the iso-brightness contours of the image [4]. For example, the shading flow field is continuous across an edge caused by an abrupt albedo change but discontinuous across an edge caused by a cast shadow [4].

Significantly, occlusion edges can be distinguished on the basis of the shading flow field as well. At an occlusion edge of a smooth object, the edge results from the object's surface curving away from the viewer; we call this type of edge a *fold*. At a fold, the shading flow field is generically tangent to the edge due to the projective geometry of the situation (Fig. 6). On the occluded side of the edge the shading flow has an arbitrary relationship to the edge and is generically non-tangent; we call this side of the edge a *cut* [10].

The ability to compute the flow field structure in the neighborhood of the edge is exactly what we are looking for to classify the edge. However, techniques that compute flow field structure without explicitly accounting for edges can destroy the relationship between the flow field and the edge and thus prevent the correct interpretation and classification of the edge. What we describe next is how we endow the connectivity structure of our relaxation labeling network with the ability to explicitly consider edge information and thus prevent the problem just mentioned. Naturally, this places some dependence on the edge detector used; however this is clearly preferable to completely ignoring the edge.

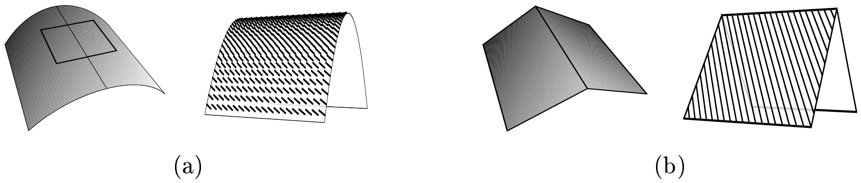


Fig. 6. Illustration of shading flow in the neighborhood of an edge. When a shaded surface is viewed such that an edge appears, the shading flow field takes on different appearances depending on the nature of the edge. A fold occurs (a) when the surface bends smoothly away from the viewer (the typical occlusion case), and the shading flow field appears tangent to the edge. At a cut (b), the surface is discontinuous (or occluded), and shading flow is generally non-tangent to the edge.

5 Edges as Nonlinear Inhibition

Due to its physical nature, an edge can be thought of as dividing the shading flow field domain into distinct regions, implying that the computation of the shading flow on either side of the edge can and should be done separately. This is an intuitive but powerful argument: incorporating edges into the relaxation labeling network to regulate the growth of flow structure obviates the trade-off between high resistance to noise and strict stability along discontinuities we mentioned in Section 3.

To implement this idea in the framework of relaxation labeling, what is needed is a specialized set of interactions between edge nodes and nearby shading flow nodes. These interactions would block the flow input if it comes from across the edge. With this input blocked, and so long as s_{min} is positive, the flow on one side of the edge will not extend across the edge, because the total support contributed to the other side will never exceed zero. This frees the selection of s_{min} from stability considerations and allows us to determine it solely on the basis of noise resistance and structural criteria. A cartoon illustrating these interactions appears in Fig. 7. Interestingly, a nonlinear veto mechanism that is reminiscent of the one proposed here also exists in biological systems in the form of shunting inhibition [3].

We have tested this adaptive relaxation labeling network on a variety of synthetic and natural images, two of which are shown in Fig. 8. We used the Logical/Linear [12] and the Canny [5] edge detectors and the shading flow fields were measured using standard differential operators.

6 Conclusions

In this paper we have described a computational approach that integrates boundary and visual flow cues for the computation of coherent shading flow fields in images. It is important to capture this interaction between flows and boundaries accurately as it indicates the geometry of the scene underlying the image. Based

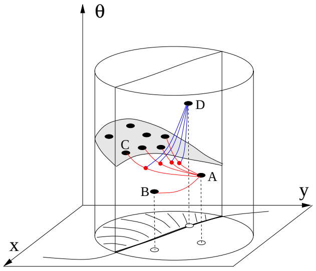


Fig. 7. Edge-flow interactions for boundary stability. Assume the flow structure in the image plane is bounded by the indicated edge. *Flow* cell A is connected to a set of other cells (B and C) which are a part of the same coherent flow. Although A is not active (there is no flow in its corresponding retinotopic position), its facilitory interaction with the cells on the other side of the edge may eventually raise its activity level. To prevent cell C from affecting A, an active *edge* cell D blocks the facilitory inputs from C, thus effectively limiting A's context to cell B only. Unless enough of these cells are also active, A will not reach its activation potential, and thus will not signal any flow.

on a geometrical analysis, our computation is carried out in a relaxation labeling network whose nodes are tuned to position, orientation, and two flow curvatures. Boundary information is used to adaptively alter the context which influences a given node, a mechanism which enables the network to handle noisy and sparse data sets without affecting the flow's discontinuities. Both the flow computation and the incorporation of edges as boundary conditions are motivated by the columnar architecture of the primary visual cortex and neurophysiological shunting inhibition. While here we applied our system to shading flow fields and edges, the same ideas can be used for other flow-like visual cues like motion, texture, and color.

References

1. O. Ben-Shahar and S. Zucker. On the perceptual organization of texture and shading flows: From a geometrical model to coherence computation. In *CVPR*, pages 1048–1055, 2001.
2. O. Ben-Shahar and S. Zucker. The perceptual organization of texture flow: A contextual inference approach. *IEEE PAMI.*, 2002. In press.
3. L. Borg-Graham, C. Monier, and Y. Fregnac. Visual input evokes transient and strong shunting inhibition in visual cortical neurons. *Nature*, 292:369–373, 1998.
4. P. Breton and S. Zucker. Shadows and shading flow fields. In *CVPR*, 1996.
5. J. Canny. A computational approach to edge detection. *IEEE PAMI*, 8(6):679–698, 1986.
6. A. Dobbins, S. Zucker, and M. Cynader. Endstopped neurons in the visual cortex as a substrate for calculating curvature. *Nature*, 329(6138):438–441, 1987.
7. J. Gibson. *The Perception of the Visual World.* The Riverside Press, 1950.
8. C. Gilbert. Horizontal integration and cortical dynamics. *Neuron*, 9:1–13, 1992.

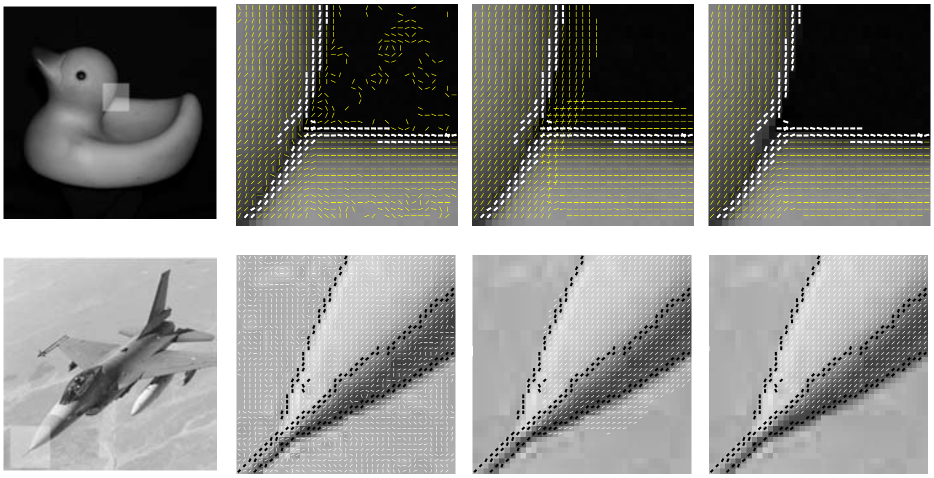


Fig. 8. Examples of shading flow field relaxation with edges as boundary conditions. Shown are (left to right) image and ROI, initial shading flow (thin segments) and edges (thick segments), relaxation without boundaries, and relaxation with boundaries. Note that while both relaxations compute correct flow structure, the one without boundaries extends the flow beyond the edge, making classification more difficult. On the other hand, edge classification and occlusion relationship is trivial based on the result using edges as boundary conditions.

9. D. Hubel and T. Wiesel. Functional architecture of macaque monkey visual cortex. In *Proc. R. Soc. London Ser. B*, volume 198, pages 1–59, 1977.
10. P. Huggins and S. Zucker. Folds and cuts: how shading flows into edges. In *ICCV*, 2001.
11. R. Hummel and S. Zucker. On the foundations of the relaxation labeling proceeses. *IEEE PAMI*, 5:267–287, 1983.
12. L. Iverson and S. Zucker. Logical/linear operators for image curves. *IEEE PAMI*, 17(10):982–996, 1995.
13. G. Kanizsa. *Organization in Vision: Essays on Gestalt Perception*. Praeger Publishers, 1979.
14. M. Kass and A. Witkin. Analyzing oriented patterns. *CVGIP*, 37:362–385, 1987.
15. B. O'Neill. *Elementary Differential Geometry*. Academic Press, 1966.
16. P. Parent and S. Zucker. Trace inference, curvature consistency, and curve detection. *IEEE PAMI*, 11(8):823–839, 1989.
17. P. Perona. Orientation diffusion. *IEEE Trans. Image Processing*, 7(3), 1998.
18. A. Rao and R. Jain. Computerized flow field analysis: Oriented texture fields. *IEEE PAMI*, 17(7):693–709, 1992.
19. K. Stevens. The line of curvature constraint and the interpretation of 3d shape from parallel surface contours. In *Proc. IJCAI*, pages 1057–1061, 1983.
20. B. Tang, G. Sapiro, and V. Caselles. Diffusion of general data on non-flat manifolds via harmonic maps theory: The direction diffusion case. *IJCV*, 36(2):149–161, 2000.
21. J. Todd and F. Reichel. Visual perception of smoothly curved surfaces from double-projected contour patterns. *J. Exp. Psych.: Human Perception and Performance*, 16(3):665–674, 1990.