Computing Layered Surface Representations:
An Algorithm for Detecting and Separating Transparent Overlays

Manish Singh
Psychology and Center for Cognitive Science
Rutgers University – New Brunswick
Piscataway, NJ 08854
manish@ruccs.rutgers.edu

Xiaolei Huang
Computer and Information Sciences
Rutgers University – New Brunswick
Piscataway, NJ 08854
xiaolei@cs.rutgers.edu

Abstract

The biological visual system possesses the ability to compute layered surface representations, in which one surface is represented as being viewed *through* another. This ability is remarkable because, in scenes involving transparency, the link between surface topology and image topology is greatly complicated by the collapse of the photometric contributions of two distinct surfaces onto image intensity. Previous analysis of transparency has focused largely on the role of different kinds of junctions. Although junctions are important, they are not sufficient to predict layered surface structure. We present an algorithm that propagates local junction information by searching for chains of polarity-preserving junctions with consistent 'sidedness,' and then propagates the transparency labeling into interior regions. The algorithm outputs a layered representation specifying (i) the distinct surfaces, (ii) their depth ordering, and (iii) their surface attributes. We demonstrate the results of the algorithm on a number of images—both synthetic and real. We end by considering implications for related domains, such as shading.

1. Introduction

Computational accounts of visual surface construction typically assume a one-to-one mapping from environmental surface patches to image patches: With the exception of occluding boundaries, each image point is assumed to have projected from a single surface point, and the goal is to compute various intrinsic attributes of the surface at that point. This assumption is violated in situations involving partially-transmissive surfaces: Here, the photometric contributions of two distinct surfaces—a farther opaque surface and an occluding partially-transmissive one—collapse onto a single intensity at each image pixel. As a result, a local image patch can correspond to the combined projection of two surface patches on distinct surfaces—thereby complicating the link between surface topology and image topology. Despite
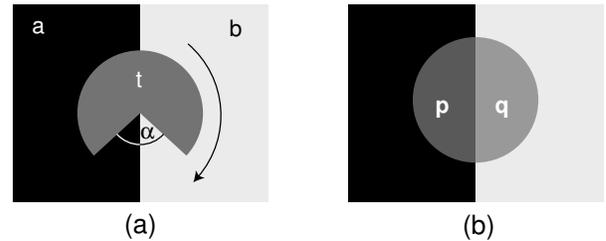

Figure 1: **Metelli's episcotister model of transparency.**

this, the biological visual system is able to decompose image intensities and compute representations of two distinct surfaces layered in depth. Indeed, many recent techniques in medical imaging and data visualization rely on this ability [1], [2]. In this paper, we present an algorithm that can detect and separate transparent surfaces in static, achromatic images.

1.1. Image Constraints in Transparency

In his influential theory, Metelli [3] modeled transparency using an 'episcotister': A disk with an open sector of relative area α, and reflectance t, is rotated in front of a bipartite surface, with reflectances a and b. With sufficiently fast rotation, the episcotister appears as a homogeneous overlying transparent layer (see Fig 1). The resulting 'color mixing' is thus described by:

$$p = \alpha a + (1 - \alpha)t, \qquad (1)$$

$$q = \alpha b + (1 - \alpha)t. \qquad (2)$$

These equations yield unique solutions for α and t:

$$\alpha = \frac{p - q}{a - b}, \qquad (3)$$

$$t = \frac{aq - bp}{a + q - b - p}. \qquad (4)$$

Gerbino et al. [4] showed that the same equations follow in the luminance domain, under the assumption of uniform
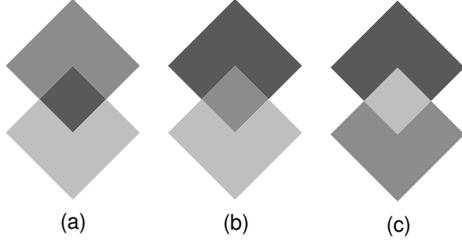
Figure 2: **X-junction classification: (a) non-reversing, (b) single-reversing, and (c) double-reversing.**
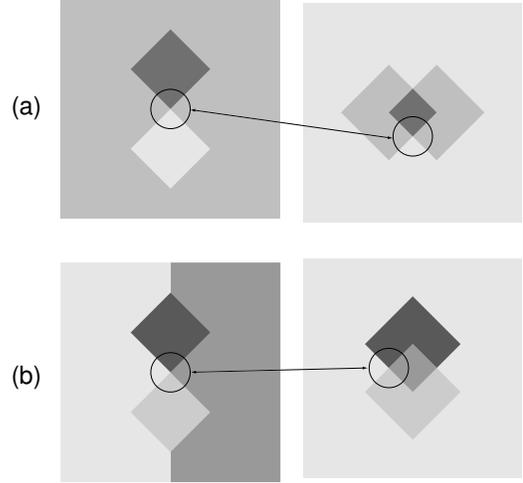

Figure 3: **Demonstrating the insufficiency of local junctions for transparency using (a) a non-reversing junction, and (b) a single-reversing junction.**

illumination. This luminance formulation is more natural from the point of view of visual computation, since a vision system is given luminances, not reflectances. Despite their simplicity, Metelli's equations also provide a reasonable approximation to more complex cases, including fog [5].

Based on restrictions on the solution (3) for α, Metelli derived two *qualitative constraints* for transparency. First, because α cannot be negative, $p - q$ must have the same sign (i.e., contrast polarity) as $a - b$:

Polarity constraint: $a \geq b \iff p \geq q$.

In other words, the presence of a putative transparent layer must preserve contrast polarity across an underlying contour. Second, because α cannot exceed 1 (being a proportion), the magnitude $|p - q|$ must not exceed the magnitude $|a - b|$:

Magnitude constraint: $|p - q| \leq |a - b|$.

Similarly, based on the restriction that the term t in solution (4) cannot be negative, Singh & Anderson [6] derived an additional constraint:

Michelson-contrast constraint: $\frac{|p-q|}{|p+q|} \leq \frac{|a-b|}{|a+b|}$.

Recent image analysis of transparency has adopted a qualitative approach, developing simple constraints that may be tested locally at junctions [7], [8]. A transparent overlap leads, generically, to X junctions in the projected image.[1] Adelson & Anandan [7] classified X junctions into non-reversing, single-reversing, and double-reversing, depending on whether both, one, or none of the two segments of the X junction preserves contrast polarity. A non-reversing junction is consistent with two different interpretations of depth layering (see Fig 2(a)): either of the two segments may be seen as bounding an overlying transparent layer. A single-reversing junction is consistent with a unique transparency interpretation (see Fig 2(b)); whereas a double-reversing junction is not consistent with transparency (Fig 2(c)). Anderson [8] has articulated a rule that combines the contributions of contrast polarity and magni-

tude, along with geometric continuity: *When two aligned contours undergo a discontinuous change in contrast magnitude, but preserve contrast polarity, the lower-contrast region decomposes perceptually into two causal layers.*

Although the Adelson & Anandan classification scheme and the Anderson rule articulate important constraints for the perception of transparency, these constraints are, by design, purely local. The computation of surface structure, on the other hand, requires that local constraints, applied to junctions, be integrated in a mutually consistent manner. Indeed, one can demonstrate that these local constraints are not by themselves sufficient to predict percepts of transparency. Fig 3(a), for example, shows the same non-reversing junction placed in two different image contexts. In the display on the right, this junction is perceived to result from a transparent overlap, whereas no such transparency interpretation is perceived at the identical junction in the left display. Fig 3(b) similarly demonstrates the insufficiency of a single-reversing junction in predicting a layered surface percept. What is required, therefore, is a scheme for propagating the influences of local junctions, and searching for circuits (and maximal chains) of mutually-consistent junctions. These circuits and chains then constitute boundaries of candidate transparent overlays, and the transparency interpretation can then be verified in interior regions.

2. Computing Transparent Overlays

Our algorithm consists of the following main steps:

1. Construct a scene graph data structure by detecting regions, junctions, and edges in an input image, and representing the relationships between them.

---

[1]T junctions can also be produced by transparent overlay [8], but only in special situations—e.g., if the color of the transparent layer happens to match the color of one of the underlying surface regions.
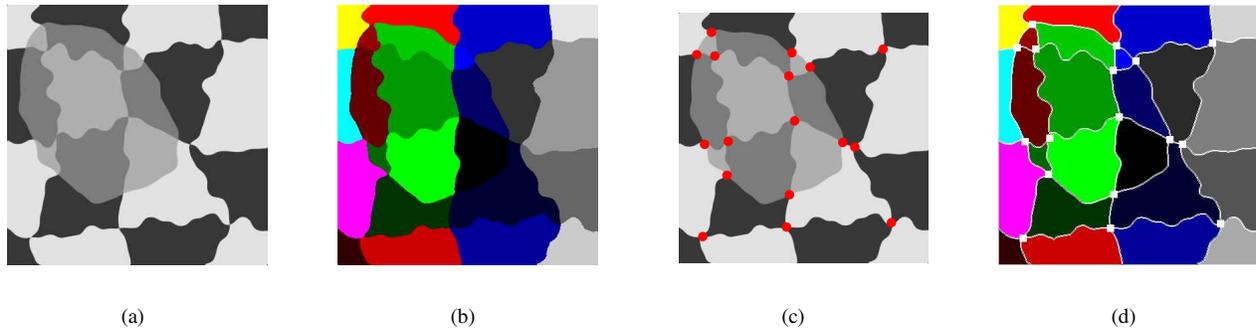
Figure 4: **Demonstrating the initial steps of the algorithm: (a) Original image. (b) Result of image segmentation. (c) Estimated X junction clouds. (d) Scene graph consisting of edges, junctions, and regions.**

2. Detect X junctions in the image, identify the polarity-preserving ones, and classify them into non-reversing and single-reversing.

3. Detect the boundaries of candidate transparent overlays by searching for all possible circuits and maximal chains of polarity-preserving junctions with consistent lower-contrast 'sidedness.' The regions adjacent to the circuit—on its lower-contrast side—receive a tentative transparency labeling.

4. Propagate the transparency labeling into interior regions, by verifying the transparency interpretation in regions on the lower-contrast side of the circuit, but not immediately adjacent to it.

5. Compute a layered surface representation by separating the transparent overlays from the background surface, and assigning surface attributes to both layers.

We present each step in turn, using the example display shown in Fig 4(a).

2.1. Constructing the scene graph

We use a scene graph data structure to represent the scene elements in an image, and the relationships between them. The scene elements include *edges*, *junctions*, and *regions*. Given an input image, the following algorithms are applied to detect the scene elements:

1. Image Segmentation. We partition an image into regions of similar color statistics. The image segmentation algorithm we use is based on active contour models ('snakes') [9] with balloon forces [10]. Around the boundary of a small patch of seed pixels, an active contour is initialized and it inflates to include neighboring pixels until it nears a region boundary, where the image gradient forces and balloon forces reach equilibrium. The process is repeated originating from different patches of seed pixels until all sufficiently large

regions are found. At the end of this operation, some pixels in the boundary areas may not be included in any region. We then apply Dilation morphological operation to all regions in parallel, in order to allow the regions to grow into those pixels. This results in a fully segmented image in which each pixel has a region label, as shown in Fig 4(b). (Each distinct region is rendered using a different color.)

2. Junction Detection. A pixel location is considered to be an X (or 4) junction candidate if pixels in its local neighborhood have exactly four distinct region labels. Applying this definition yields a cloud of candidate pixels around each "true" junction. We then use a local window with adaptive radius around each candidate cloud to localize the cloud centers, and take these centers to be a detected X junctions. The X junctions thus obtained in the example image are shown in Fig 4(c).

3. Edge Detection. The closed contour defining the perimeter of each region is disconnected at the junction locations, to yield one or more edges. (Non-closed edges are thus bounded by two junction terminators.) An edge thus identified is aware of which region it belongs to. For every boundary between two adjacent regions, there are thus two side-by-side edges—one from each region. Figure 4(d) shows all the edges and junctions for the example image.

After the scene elements are detected, we construct a scene graph data structure to represent their relationships (Fig 4(d)). Each of the element types, *Edge, Junction*, and *Region* consists of a set of attributes, specified as follows:

Edge: edge ID label, the chain of pixels on the edge, region label of the region it belongs to, junction labels of its two terminators (if it is a non-closed edge).

Junction: junction ID label, type (here we restrict the analysis to X junctions), center location, edge labels of its incident edges, region labels of its adjacent regions.
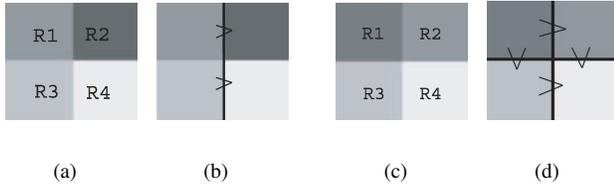
Figure 5: **(a) A single-reversing X junction. (b) Its polarity-preserving border, marked in black, and unique depth ordering, indicated by arrows (as in [11]). (c) A non-reversing X junction. (d) Its two polarity-preserving borders, and the associated depth orderings.**

Region: region ID label, mean intensity value of the region, its perimeter consisting of edges and junctions in counter-clock-wise order, e.g., "$edge1\_id \rightarrow junction1\_id \rightarrow edge2\_id \rightarrow junction2\_id \rightarrow ...$".

2.2. Finding polarity-preserving X junctions

With the scene graph data structure in place, we test for contrast polarity relationships at each X junction, and classify them accordingly. If both edge pairs preserve contrast polarity, the X junction is non-reversing. If only one preserves polarity, it is single-reversing. Otherwise it is double-reversing. Next, for each edge pair that preserves contrast polarity, we determine which of its two sides has lower contrast. Based on the *magnitude constraint* and the *Michelson-contrast constraint* (see Section 1.1), we require that *both* luminance differences and Michelson contrast be lowered[2] in order to make a local transparency interpretation.

Let the four adjacent regions at an X junction be R_1, R_2, R_3 and R_4, with mean intensity values L_1, L_2, L_3 and L_4, respectively (see Fig 5). We test for contrast polarity and magnitude relationships across the border B_{12-34} (which separates R_1, R_2 from R_3, R_4) and, similarly, across the border B_{13-24}:

Sign Tests:

    (I) Is $Sign(L_1 - L_2) = Sign(L_3 - L_4)$?

    (II) Is $Sign(L_1 - L_3) = Sign(L_2 - L_4)$?

Magnitude Tests:

    (IA) Is $|L_1 - L_2| \leq |L_3 - L_4|$ and $\frac{|L_1 - L_2|}{|L_1 + L_2|} \leq \frac{|L_3 - L_4|}{|L_3 + L_4|}$?

    (IB) Is $|L_1 - L_2| \geq |L_3 - L_4|$ and $\frac{|L_1 - L_2|}{|L_1 + L_2|} \geq \frac{|L_3 - L_4|}{|L_3 + L_4|}$?

    (IIA) Is $|L_1 - L_3| \leq |L_2 - L_4|$ and $\frac{|L_1 - L_3|}{|L_1 + L_3|} \leq \frac{|L_2 - L_4|}{|L_2 + L_4|}$?

    (IIB) Is $|L_1 - L_3| \geq |L_2 - L_4|$ and $\frac{|L_1 - L_3|}{|L_1 + L_3|} \geq \frac{|L_2 - L_4|}{|L_2 + L_4|}$?

---

[2]This holds automatically for single-reversing junctions (see Appendix C in [6]), but not necessarily for non-reversing junctions.

The sign tests determine which of the two edge pairs in an X junction could correspond to the boundary of a transparent layer, whereas the magnitude tests determine which *side* of such a boundary could contain the transparent layer. For example, the junction in Fig 5(a) satisfies Sign test (II), but not Sign test (I). Thus only the vertical border preserves contrast polarity. Then, applying the Magnitude tests (IIA) and (IIB), we find that the junction satisfies test (IIA), but not test (IIB). Thus, if this X junction maintains the status of a transparent overlap in the global interpretation, it would have a unique transparency interpretation (depicted in Fig 5(b))—with the vertical border bounding a transparent layer on the left side (i.e., corresponding to regions R_1 and R_3).

Having applied the Sign and Magnitude tests, we generate the object type *polarity-preserving X junction*, which is specified in the scene graph data structure as follows:

Preserving-Xjunc: junction ID label, subtype (single-reversing or non-reversing), for each polarity-preserving border: the labels of the two regions on its lower contrast side, and the labels of the two (topologically) aligned edges that belong to these two regions.

2.3. Finding transparent overlays

In order to find candidate boundaries of transparent overlays, the algorithm integrates local junction information by searching for all possible circuits, and maximal chains, of polarity-preserving X junctions with consistent polarity and consistent lower-contrast 'sidedness.' The transparency interpretation of these candidate boundaries is then verified by photometric checks on interior regions—i.e., regions that are not bounded by any of the edges in the circuit, but that lie on its lower-contrast side.

2.3.1 Initiating an X-junction chain

The junction integration process originates from polarity-preserving X junctions, with single-reversing junctions having higher priority than non-reversing ones. Starting from a single-reversing X junction, there is only one chain to be traced. The chain goes out along one edge—out of the pair of aligned edges, on the lower contrast side of the polarity-preserving border. Starting from a non-reversing X junction, there are two such chains to be traced, since there are two polarity-preserving borders.

2.3.2 Criteria for chain continuation and termination

Following an edge currently being traced, several possibilities can occur at the other end of the edge; see Figs 6(a)-6(d). In all displays, the edge currently being traced is E_1,
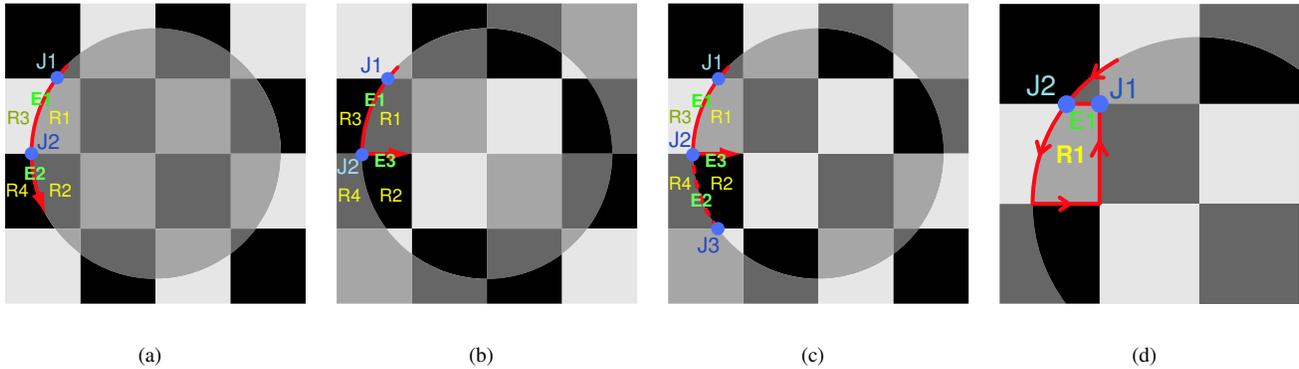
Figure 6: **Demonstrating criteria for chain continuation (a)-(c), and chain termination (d).**

and it belongs to region R_1. The edge originates at junction J_1, and ends at junction J_2.

The scenarios for *chain continuation* are illustrated in Figs 6(a)-6(c).

(a) In Fig 6(a), the following conditions hold:

   - R_1 is on the lower-contrast side of junction J_1.
   - J_2 is a polarity-preserving X junction (in this case, with a single polarity-preserving border B_{12-34}).
   - E_1 is (topologically) aligned with edge E_2, along J_2's polarity preserving border.
   - Region R_2—which contains edge E_2—is on the lower-contrast side of junction J_2.

   With these conditions holding, junction J_2 locally supports a transparency percept that is consistent with J_1. The chain thus continues across J_2 along the aligned edge E_2 (which belongs to region R_2).

(b) In Fig 6(b), the junction J_2 is not a polarity-preserving X junction. In this situation, the chain still continues, but follows the contour of the current lower-contrast-side region R_1 to E_3, rather than going across junction J_2, to the topologically aligned edge.

(c) The scenario shown in Fig 6(c) can occur only if J_2 is a non-reversing X junction. Here E_1 is aligned with one of J_2's polarity-preserving borders, i.e., B_{12-34}, but the 'lower contrast' side at this junction is ambiguous (i.e., luminance difference and Michelson contrast do not yield consistent results). In this case, the chain tentatively follows E_1's topologically aligned edge E_2 to the next single-reversing X junction J_3. At J_3, it becomes clear that the lower-contrast side has reversed from that at junction J_1. Thus the three adjacent junctions J_1, J_2, and J_3 cannot support a consistent transparency interpretation. Therefore, the chain continues as it did in situation (b), i.e., following along the perimeter of R_1, to edge E_3.

The scenarios for *chain termination* are as follows.

(d) Let the originating X junction of the chain be J_0 and the first edge traced on the chain be E_0. Suppose that:

   - the junction at the end of the edge currently being traced, is the same as the originating junction, J_0.
   - the edge currently being traced, is topologically aligned with the originating edge E_0.

   This will occur, for example, in Fig 6(a) once the chain has gone all the way around the disk; and in Fig 6(b) once the chain has circumscribed the top-left quarter disk. In this case, we have successfully found a closed global circuit, consisting of polarity-preserving X junctions with consistent 'sidedness.' We save the circuit's information using an Object *Circuit*:

   Circuit: circuit ID label, the closed chain of junctions and edges in the format "$junction0\_id \rightarrow edge0\_id \rightarrow junction1\_id \rightarrow ...,$" the regions adjacent to the chain, on its lower-contrast side, in the format of "$region0\_id \rightarrow region1\_id \rightarrow ....$"

(e) If, during the tracing, the chain comes to include all of the edges of region $R1$ (which contains the current edge $E1$), the chain is stopped (see Fig 6(d)). Although a loop is formed, it is not recognized as a successful global circuit. (This is why no successful circuit is obtained in the displays on the left of Fig 3.) If this chain contains the topologically aligned edges of at least two X junctions, with consistent sidedness, these edges and junctions are saved as an open maximal chain (along with the regions adjacent to it, on its lower-contrast side). Such a chain may *locally* generate a percept of transparency, but no globally-consistent transparent-surface interpretation is possible (e.g., the top-left section of the disk's perimeter in Fig 6(c) or 6(d)).
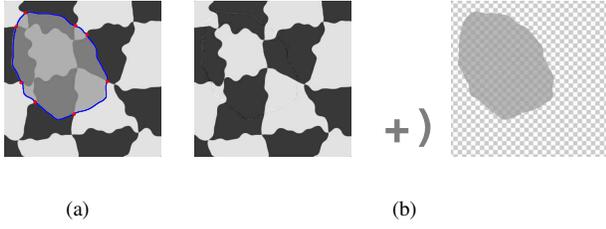
(a)   (b)

Figure 7: **(a) The computed circuit for the example image. (b) The layered decomposition: recovered background plus transparent overlay.**

(f) If a chain encounters an edge that belongs to a previously discovered circuit, the chain is terminated and discarded. This prevents the same chain from being traced repeatedly, starting from different X junctions.

2.3.3 Verifying the transparency interpretations in interior regions

A circuit is computed—by definition—based on image regions that are immediately adjacent to it (i.e., that own at least one of its edges). Before we assign a detected circuit the status of a boundary of a transparent overlay, we must verify that interior regions, that are not adjacent to the circuit (e.g., the four squares in the center of the display in Fig 6(a)), are also consistent with a transparency interpretation. In particular, the contrast generated by these interior regions must not greatly exceed the contrast values generated by the regions adjacent to the circuit. We currently use a conservative way to ensure this, by requiring that the mean intensity L of each interior region satisfies $l_{min} \leq L \leq l_{max}$, where l_{min} and l_{max} are the minimum and maximum of the mean intensity values, computed over the set of image regions that own the edges of the circuit (i.e., on its lower-contrast side). This criterion is clearly satisfied for the circuits computed in Figs 6(a) and 7(a)—thus yielding a global interpretation of a transparent overlay in these displays.

2.4. Computing layered surface representation

Once we have detected a transparent surface, we can separate it from the background surface, and quantitatively assign surface properties to both layers.

2.4.1 Transmittance and lightness of the overlay

Setting $f = (1 - \alpha)t$ in Metelli's equations makes it clear that the mapping from surface luminances in plain view (L) to luminances projected through a transparent layer (L_t) is a linear mapping (see, e.g., [14]):

$$L_t = \alpha L + f$$

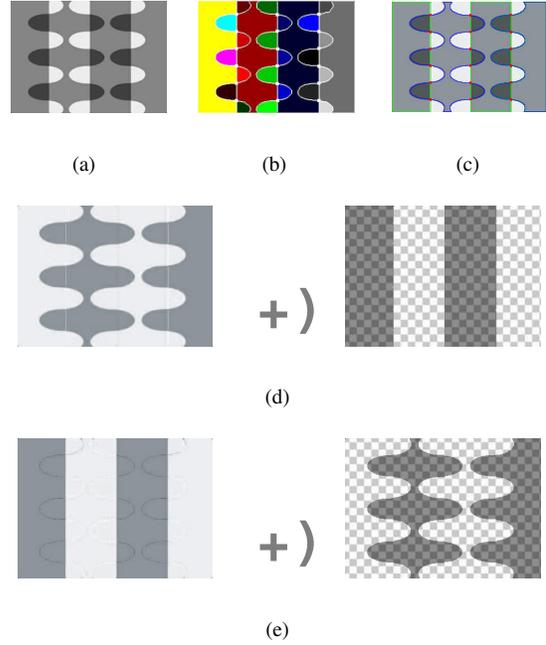(a)   (b)   (c)

(d)

(e)

Figure 8: **Result images for the "snake illusion" [14]. (a) The original display. (b) The scene graph. (c) The four detected circuits. (d-e) Two different interpretations of transparency and depth layering**

The parameters α and f can be computed at any X junction along the circuit. For example, for the single-reversing junction in Fig 5(a)[3]:

$$\alpha = \frac{L_1 - L_3}{L_2 - L_4} \qquad f = \frac{L_2 L_3 - L_1 L_4}{L_2 - L_4}$$

If a global circuit contains N polarity preserving X junctions, we obtain N sets of values: $\alpha_1, f_1, \alpha_2, f_2, ..., \alpha_N, f_N$. If all sets of values recovered at the local junctions are sufficiently close, we recognize a balanced (i.e., homogeneous) transparent overlay enclosed by the circuit, with estimated transmittance and luminance as:

$$\alpha = \frac{1}{N} \sum_{i=1}^{N} \alpha_i \qquad t = \frac{1}{N} \sum_{i=1}^{N} \frac{f_i}{1 - \alpha_i}$$

If, on the other hand, the sets of values differ from each other, the transparent overlay enclosed by the circuit is unbalanced, and we need to compute a dense surface property map for the overlay. (Humans can readily perceive such inhomogeneous transparency; see, e.g., [6].) Since we know

---

[3]It should be noted that the human perception of transmittance deviates systematically from the physical solution (3), and is determined instead by the ratio of contrasts; $\alpha = \frac{p-q}{p+q} / \frac{a-b}{a+b}$ (see [6]). Thus, either formula for transmittance may be used, depending on the application.
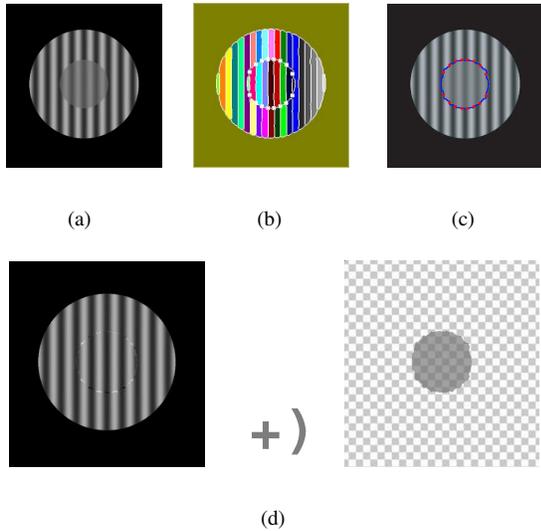
(a)　　　　　(b)　　　　　(c)

(d)

Figure 9: **Result images for the sine-wave display [6]. (a) The original display. (b) The scene graph. (c) The detected circuit. (d) The layered representation.**

the properties of the overlay over those regions that are immediately adjacent to the chain, we can compute the values in interior regions using simple interpolation methods.

2.4.2 Background surface properties

At any pixel location p that contains a transparent overlay, the background surface luminance (i.e., the luminance it would project when seen in plain view) is determined by $L_p = \frac{L_{tp} - f_p}{\alpha_p}$, where α_p and f_p are the computed surface properties of the transparent overlay at that pixel location. (Human perception of underlying surface lightness accords reasonably well with Metelli's prediction; see [13].)

Thus, we are now in a position to separate the background surface from the overlying transparent layer, and assign them separate attributes. Figure 7(b) shows the layered decomposition for the example input image in Fig 4(a).

3. Experimental Results

We have applied the algorithm, with successful results, to a number of images—including many of the commonly used transparency displays as well as some real images. Some typical results are shown in Figs 8 – 10.

1. In displays containing nonreversing junctions, such as Fig 2(a), the algorithm outputs both possible interpretations of transparency and depth layering. Fig 8 demonstrates this behavior on a more challenging display, Adelson's *snake illusion,* in which either the rect-
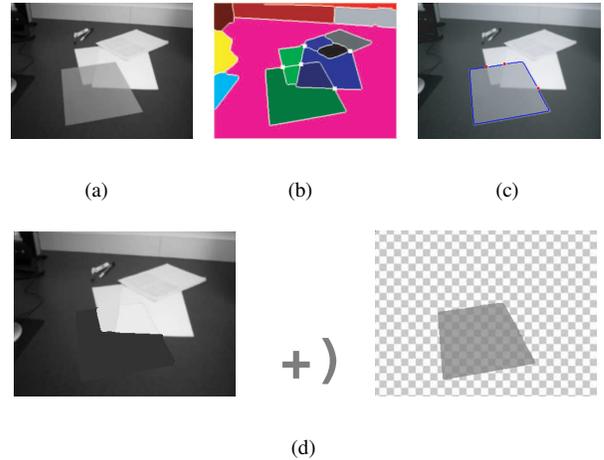

(a)　　　　　(b)　　　　　(c)

(d)

Figure 10: **Detecting transparent overlays in a real image. (a) The original image after Gaussian smoothing. (b) The computed regions, edges, and X junctions. (c) The detected circuit. (d) The layered representation.**

angular bars or the undulating "snakes" may be perceived to be the transparent overlays.

2. Fig 9 demonstrates the results on an image containing a sinusoidally-varying background, rather than sharply-defined edges, junctions, and regions.

3. Fig 10 demonstrates the results of the algorithm on a real image containing a transparent overlay—a stack of overhead transparencies covering sheets of paper on a desk. The algorithm finds and separates the transparent overlay properly. However, the recovery of the background luminance is not as good as in the synthetic images, because of the very gradual variations in intensity within each segmented region.

4. Conclusions

We have presented an algorithm that can detect transparent overlays in images, separate the overlays from background surfaces, and recover their surface properties. The algorithm finds boundaries of candidate transparent overlays by searching for circuits of polarity-preserving X junctions with consistent 'sidedness,' and then verifies their transparency interpretation by photometric tests on interior regions. Once transparent overlays have been detected, their surface properties, and those of the background surfaces, are assigned using known formulas.

The algorithm performs well on standard transparency displays—including those with multiple interpretations of transparency and depth layering (e.g., Figs 2(a) and 8(a)). It is also able to detect transparent overlays in real images. The image segmentation algorithm we use (active contour models with balloon forces; see [9], [10]) is robust

to noise in natural images: The inflating balloon forces enable the evolving contours to surpass small spurious structures inside a region, while the internal forces (instantiating smoothness constraints) prevent the contour from getting trapped within gaps on boundaries. This robust segmentation allows for successful detection of transparent overlays. However, the recovery of background luminance can suffer somewhat due to gradual intensity variations within segmented regions.

Given the simple, deterministic nature of the propagation scheme, these results are very encouraging. Because the algorithm relies solely on X junctions, however, it is not currently able to deal with cluttered scenes in which portions of the transparent-layer boundary (edges and/or junctions) are occluded behind other opaque surfaces, or where part of the boundary is camouflaged by a background surface region that happens to have the same color as the transparent layer (see, e.g., [8]). Both of these situations generate T junctions, and a more sophisticated, probabilistic, propagation scheme would be required to distinguish between cases in which a T junction arises from opaque occlusion, and cases in which it arises from camouflaged transparency. Efficient schemes for dealing with partly-occluded surfaces exist already (e.g., [11], [12]), and in future work these would need to be integrated into the algorithm.

Future directions also include looking at implications for shadows and shading. These domains are intimately related to transparency because they also generate polarity-preserving X junctions. (The main photometric difference being that Michelson contrast is preserved.) As a suggestive example, consider the results of the algorithm on a shading display by [15] (see Fig 11). The algorithm finds two circuits that correspond to reflectance change and shading, respectively. Note that, for Adelson & Pentland's class of displays, a simple additional rule can distinguish between these two cases: If the circuit meets the occluding contour of the surface at a Y junction, interpret a shading boundary; if it meets it at a T junction, interpret a reflectance change.

Acknowledgments

References

[1] R. Kasrai, F. A.A. Kingdom, and T. M. Peters, "The Perception of Transparency in Medical Images," *Medical Image Computing and Computer-Assisted Intervention (MICCAI'99)*, pp. 726-733, 1999.

[2] R. M. Taylor II, "Visualizing Multiple Scalar Fields on the Same Surface," *IEEE Computer Graphics and Applications*, pp. 6-10, Mar-Apr, 2002.

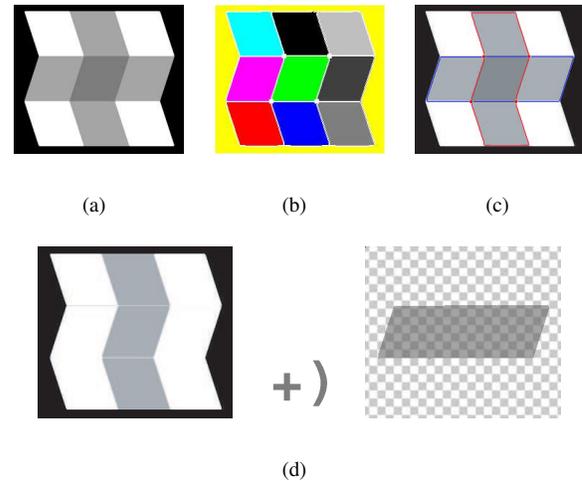[3] F. Metelli, "The Perception of Transparency," *Scientific American*, Vol. 230, pp. 90-98, 1974a.

(a)   (b)   (c)

(d)

Figure 11: **Results of the algorithm on Adelson & Pentland's display [15], suggesting implications for shading.**

[4] W. Gerbino, C. I. Stultiens, J. M. Troost, and C. M. de Weert, "Transparent layer constancy," *J. Exp. Psychology: Human Perception and Performance*, Vol. 16, pp. 3-20, 1990.

[5] J. Hagedorn, and M. D'Zmura, "Color appearance of surfaces through fog," *Perception*, Vol. 29, pp. 1169-1184, 2000.

[6] M. Singh, and B. L. Anderson, "Toward a Perceptual Theory of Transparency," *Psychological Review*, Vol. 109, pp. 492-519, 2002.

[7] E. H. Adelson, and P. Anandan, "Ordinal Characteristics of Transparency," *AAAI Workshop on Qualitative Vision*, pp. 77-81, 1990.

[8] B. L. Anderson, "A Theory of Illusory Lightness and Transparency in Monocular and Binocular Images: the Role of Contour Junctions, " *Perception*, Vol. 26, pp. 419-453, 1997.

[9] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active Contour Models," *International Journal of Computer Vision*, Vol. 1, pp. 321-331, 1988.

[10] L. D. Cohen, and I. Cohen, "Finite-Element Methods for Active Contour Models and Balloons for 2-D and 3-D Images," *IEEE PAMI*, Vol. 15, No. 11, pp. 1131-1147, 1993.

[11] E. Saund, "Perceptual Organization of Occluding Contours of Opaque Surfaces," *Computer Vison and Image Understanding*, Vol. 76, No. 1, pp. 70-82, 1999.

[12] L. R. Williams, and A. R. Hanson, "Perceptual Completion of Occluded Surfaces," *Computer Vision and Image Understanding*, Vol. 64, No. 1, pp. 1-20, 1996.

[13] M. Singh, "Lightness Constancy through Transparency," *Manuscript submitted for publication*, 2003.

[14] E. H. Adelson, "Lightness Perception and Lightness Illusions," *The New Cognitive Neurosciences*, 2nd ed., M. Gazzaniga, ed. Cambridge, MA:MIT Press, pp. 339-351, 2000.

[15] E. H. Adelson, and A. P. Pentland, "The perception of shading and reflectance," *Perception as Bayesian inference*, D. Knill, and W. Richards, eds. New York: Cambridge University Press, pp. 409-423, 1996.