Illusory Contour Detection Using MRF Models

Suthep Madarasmi[1], Ting-Chuen Pong[1] and Daniel Kersten[2]
[1]Department of Computer Science, [2]Department of Psychology
University of Minnesota, Minneapolis, MN 55455

Abstract— This paper presents a computational model for obtaining relative depth information from image contours. Local occlusion properties such as T-junctions and concavity are used to arrive at a global percept of distinct surfaces at various relative depths. A multi-layer representation is used to classify each image pixel into the appropriate depth plane based on the local information from the occluding contours. A Bayesian framework is used to incorporate the constraints defined by the contours and the prior constraints. A solution corresponding to the maximum posteriori probability is then determined, resulting in a depth assignment and surface assignment for each image site or pixel. The algorithm was tested on various contour images, including two classes of illusory surfaces: the Kanizsa and the the line termination illusory contours.

I. INTRODUCTION

Occluding contours refer to the image contours that correspond to object boundaries. Such contours are an integral part of the construction of the 2.5-D sketch [8] and, therefore, are important in computer vision research. We present an algorithm that uses contour information to segment an image into meaningful regions and to provide a relative depth ordering of the partitioned regions. A model for obtaining shape from occluding contours should be able to explain illusory (subjective) contour perception since these illusions arise from artificially arranged cues to occlusion. The primary objective of our model is to account for such illusions through a cooperative process of computing depth and its segmentation.

Illusory contours have been used as the intuitive proof of several models of perceptual organization. On the other hand, there are several theories which have been proposed to account for illusory contours.

An important question to address is whether illusory contours provide any insight into the computational mechanism of the brain. Equivalently, an interesting issue is if such mechanisms are needed for building a robust vision system. We take the position that illusory contours serve to strengthen edges (or even create edges) where other segmentation properties such as depth suggest their appearance.

Kanizsa [5] provides a thorough analysis of T-junctions and how surfaces are perceived for various arrangements of T-junctions. He demonstrated the perception of terminating lines as a weak form of T-junction and many terminators as a cue for an occluding surface such as in figure 1a. Similarly, corners with aligned edges are viewed as degenerate forms of T-junctions as in the Kanizsa Pac-man illusion in figure 1b. Such figures offer valuable insight into the computational processes involved in the human visual system to complete contours given partial evidence. Kanizsa suggests that a non-local process with knowledge of surfaces and occlusion is involved in the image completion process; thus, accounting for such illusions.

Grossberg and Mingola [4] (see also [3]) use a neural model to explain how segmentation may arise across image regions without any luminance differences. Network interactions between three components namely, the Boundary Contour (BC) system, the Feature Contour (FC) system, and the Object Recognition (OR) system are used to explain the illusory contours and related phenomena. The BC system acts as a hypothesis generator to generate perceptual boundaries for the FC system. They suggest that the visible percepts are the result of the hypothesis confirmation by the FC system using input from the BC system. The OR system is connected to the BC system and provides top-down learned template signals for the conversion of various BC features into FC features. Thus, accounting for the perception of illusory contours. Our algorithm can be related to that of Grossberg and Mingola at this abstraction level.
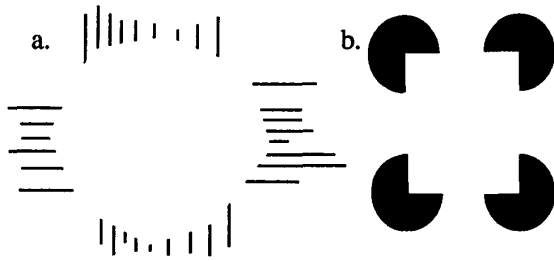
Figure 1: *Subjective contours formed by (a) terminating lines and (b) corners*

II. MULTI-LAYER REPRESENTATION

An important feature in our approach lies in the underlying representation for segmentation and surface ordering. Our model uses a different output representation from the commonly used edge or contour based methods. To illustrate our representation for image grouping, consider the input range image of two overlaid surfaces such as a textured, transparent cylinder shown in figure 2a using intensity to code the depth. Figure 2b shows the depth profile of such an image. If the segmentation relied on edges, the result would be multiple spurious edges for such an image due to the local discontinuities as shown in figure 2d. Clearly, this is not desirable since we clearly perceive two overlaid surfaces for such a display. Our multi-surface representation produces an explicit piece-wise smooth segmentation of the image into surfaces while also computing the relative depth of the surfaces. This representation is, in spirit, similar to the one proposed by Darrell and Pentland [1]. Our multi-layer representation for shape has been successfully used in various other low-level vision modules [6, 7].

III. ALGORITHM

In this section we discuss our segmentation algorithm using the depth information present in occluding contours. Since we are interested in applying this approach to obtain illusory contour perception, we make an assumption that all surfaces in the scene are piece-wise constant or fronto-parallel. To illustrate the modified multi-layer segmentation output representation used here, consider the input image shown in figure 3a. The output shown in part b of the figure consists of three intensity coded labels with layer 1 (background) shown in black, layer 2 (closer surface) shown in gray, and layer 3 (fore-
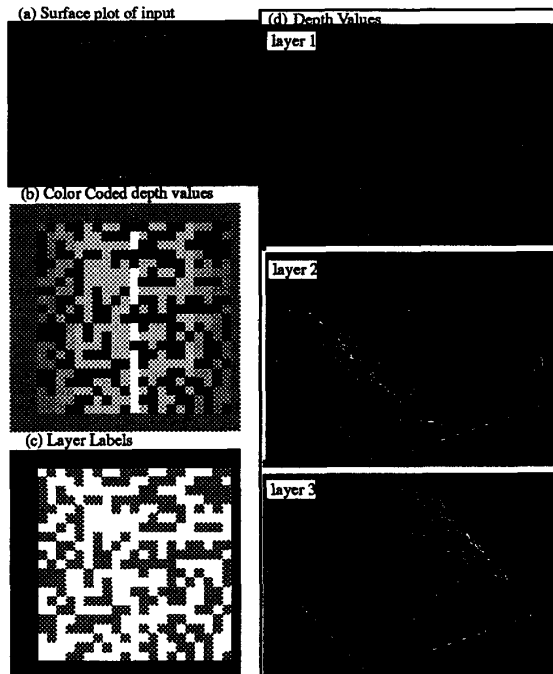

Figure 2: *The multi-layer representation: An example*

ground surface) shown in white. The layer label also represents the depth value so that layer N has a depth of N where N=1 represents the background. Since each depth layer contains a constant value due to our assumption of piece-wise constant surfaces, their values need not be shown. Stated more formally, given a set of depth values $D_{\{p\}}$, the image segmentation representation is modified so that the produced output is $C_{\{p\}}, U_{1\{p\}}, U_{2\{p\}}, ..., U_{L\{p\}}$ where C is a layer classification image, p is the image domain, and each U_l now has a constant depth value of l. Since each U_l has a constant value, we could drop its representation in the output and only look at the layer labels at each pixel given by $C_{\{p\}}$. Thus, the desired depth image and the labeling image are the same:

$$D_{\{p\}} = C_{\{p\}}$$

Our algorithm for segmenting contour images for illusory contour perception involves the confirmation of various local hypothesis. For example, any surface such as the Pac-man in figure 4a may be interpreted as either a circle partly occluded by the corner of a rectangle or as a simple Pac-man shaped object. Line terminations can either be due to an
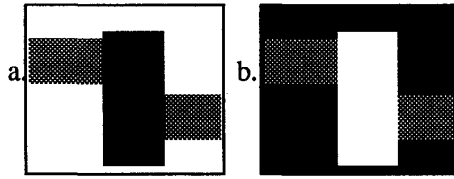
4344

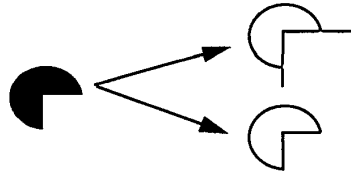Figure 3: *The multi-layer representation for depth from occluding contours*

Figure 4: *Examples of the occlusion hypothesis.*

occluding surface or due to a naturally terminating line as shown in figure 4b. In general, the shape of any visible surface may be partly due to an occluding surface or may be due to the surface having exactly that underlying shape to begin with. We define a local cost based on convexity and corner properties so that certain surfaces are preferred to others. For example, surfaces with convex boundaries have lower cost than those with concave boundaries. Surfaces with a corner boundary formed by the meeting of contours at less than 180 degrees angle are preferred. Thus, illusory surfaces are formed by the grouping of hypothesized local evidences of occlusion constraints to form a set of salient surfaces each with a cost.

For a single Pac-man the square region may be hypothesized as an occluding surface. This occluding surface would trigger a grouping in the frontal depth plane by extending the corner of the rectangle. Thus, the four corners may successfully be grouped into a rectangle. Note that the Pac-man image is interpreted as comprised of an illusory, salient rectangle occluding 4 surfaces (discs), instead of 4 high-cost, non-salient Pac-man shapes. Our treatment of the perception of illusory contours involves the minimization of a cost function that involves this type of saliency measure.

Our model for illusory contour perception can be outlined by the following steps:

1. *Image feature extraction.* In this step the edges are obtained and grouped to find features such as T-junctions, corners, convex contours, and line terminations.

2. *Layer and depth assignment to each pixel.* In this step each pixel is assigned an appropriate layer label and, thus, a depth value to minimize a cost defined by the local constraints.

3. *Layer and depth assignment to grouped regions.* Here, the pixels are assumed grouped by the previous step and are moved as regions to minimize the same cost function as in the previous step.

4. *Object completion in the visible depth plane.* Here, extrinsic edges in the visible depth plane are extended to group objects. This is primarily responsible for the grouping of illusory contours.

In our model steps 2,3, and 4 are performed simultaneously, with steps 3 and 4 occurring after longer time intervals than step 2.

A. Bayesian Approach

We use a Bayesian approach to compute the scene attribute, namely the layer assignment l which also has an associated relative depth of l:

$$p(l|I) = \frac{p(I|l)p(l)}{p(I)}$$

where I is the intensity input image data. Each constraint is expressed as a local cost function using the Markov Random Field assumption [2], that pixels values are conditional only on their nearest neighbors. Using the Gibbs-MRF equivalence, the energy function can be written as a probability function:

$$p(x) = \frac{1}{Z}e^{-\frac{E(x)}{T}}$$

where Z is the normalizing constant, T is the temperature, E is the energy cost function, and x is a random variable.

B. The Cost Function

We start with a L layer system each layer representing a different fronto-parallel depth plane. Initially, each pixel is assigned a random layer value and, therefore, a corresponding depth value. After the computations are completed, each pixel is assigned to the layer consistent with the 3-D occlusion cues in the input image.

Using a small $3x3$ window we obtain a feature measure. Using the window we obtain two constraints on the depth of nearest 8 neighboring pixels with respect to the center pixel: Above (A), Below (B) and Same (S). For the T-junction, if the center pixel is at the the occluding part of the junction, all the neighboring pixels with different intensity are constrained to be below. Thus, a cost is assigned to it if the constraint is not met. Neighboring pixels with different contrasts are constrained to be in
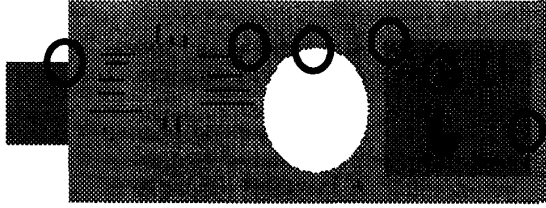
4345

Figure 5: *Examples of regions with occlusion constraints. The circled constraints from left to right include T-junction, line termination, convex contour, corner, illusory L junction and illusory T junction.*

different layers. In addition, we have a constant intensity constraint C that forces neighbors with the same intensity value to have the same depth. Note that this constraint discourages the formation of illusory contours. Another very important cost is what we call the illusory junction cost J. This cost is high if the T-junction or corner is partly (or completely) formed by illusory contours and does not a consistent depth assignment. For example, if a single Pac-man creates an illusory boundary that meets its borders, the T-junction formed would require that the border be above the illusory surface. Since the background is below the illusory surface in this case, the illusory junction is assigned a rather high cost. This cost ensures that a single Pac-man does not form illusory boundaries with the border of the image.

The local occlusion constraints used in our simulations are T-junctions, surround occlusion, and line terminations with examples shown in figure 5. The surround occlusion example includes convex contours and corners that are constrained to be above the background. This is because closed contours are frequently perceived as the inner region of a surface that, therefore, occludes the background. These constraints can be thought of as mere hypothesis which could be rejected to minimize the total system energy. For example, for the single Pac-man shown in figures 4, the hypothesis that an edge of a square occludes the disk both of which are low cost surfaces (i.e. two convex surfaces) may be rejected since it produces illusory junctions and the system may settle at perceiving a high cost surface (i.e. a non-convex surface shaped like a Pacman).

The data constraints A, B, S, C, J costs at each pixel can be converted into an energy function representation as:

$$V_D = \lambda_A V_A + \lambda_B V_B + \lambda_S V_S + \lambda_C V_C + \lambda_J V_J$$

where the λ's are the weighting factors, and V_A, V_B, V_S, V_C, and V_J are the energy associated with the above, below, same, no-edge, and illusory junction constraints, respectively. This energy function is defined locally for each pixel as in the example above.

The V_A constraint is given by:

$$\sum_i^M \sum_{j \in N_i} \delta(d_i, d_j) A_{ij}(\sigma(l_i, l_j) + \beta\sigma(l_j, l_i))$$

where i is the index for each image pixel, j is the nearest neighbor of i, d_i is the input luminance value at pixel i, A_{ij} is the local Above constraint A determined initially by the various $3x3$ window configurations. The various functions defined above are given by:

$$\delta(a, b) = \begin{cases} 1 & \text{if } a - b \neq 0 \\ 0 & \text{otherwise} \end{cases} \quad \sigma(a, b) = \begin{cases} 1 & \text{if } a - b > 0 \\ 0 & \text{otherwise} \end{cases}$$

In plain English, the above constraints ensures that if the local intensity defines an Above relationship, the layer assignment should enforce the same.

The V_B constraint is analogous to the V_A constraint with some of the signs reversed:

$$\sum_i^M \sum_{j \in N_i} \delta(d_i, d_j) B_{ij}(\sigma(l_j, l_i) + \beta\sigma(l_i, l_j))$$

where B_{ij} is the local *Below* constraint B.

The V_S constraint ensures that if there is an intensity edge, the layer and depth assignments must also be discontinuous,

$$\sum_i^M \sum_{j \in N_i} \delta(d_i, d_j) S_{ij}(1 - \delta(l_i, l_j))$$

where S_{ij} is the local *Same* constraint set to a value of 1 in our experiments.

The V_C constraint penalizes the formation of illusory contours by ensuring that if neighboring intensity is the same, then the layer and depth assignments should also be the same:

$$\sum_i^M \sum_{j \in N_i} (1 - \delta(d_i, d_j))\delta(l_i, l_j)\phi_i * \sigma(\phi_j, \phi_i)$$

where ϕ is a feature measure at each pixel. This constraint is active only when the neighboring feature measure ϕ_j is greater than the current ϕ_i. Since ϕ is proportional to the distance to an occluding feature, the regions with constant intensity rely more on their neighbor's values than on their own. The V_J constraint penalizes illusory junctions that have incorrect depth relations.
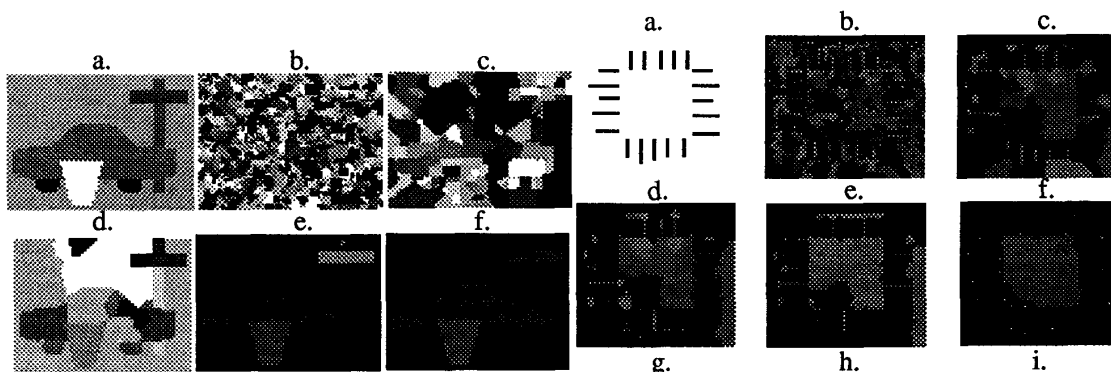
Figure 6: *(a) Input: Cartoon image with depth information. (b-f) Results from selected iterations.*
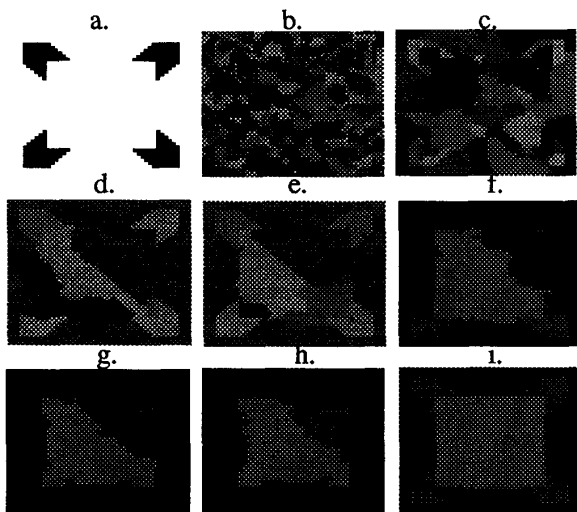

Figure 7: *(a) Input: Kanizsa type of Illusory contours. (b-i) Results from selected iterations.*

Figure 8: *(a) Input: Illusory contour from line terminations. (b-i) Results from selected iterations.*

IV. EXPERIMENTAL RESULTS

Various test images were used in our simulations including those with and without illusory contours. In figure 6 is the cartoon-like image of various objects with depth relations. Note that to use the minimum number of layers, the tires and the pole are placed in the same depth plane. Since other cues to depth such as familiarity are not available to the system, the objects are placed in the depth plane based purely on their information from occluding contours.

Images with illusory contours were also used in our simulations. Figure 7 shows the segmentation result for a Kanizsa type illusory contour. Note that
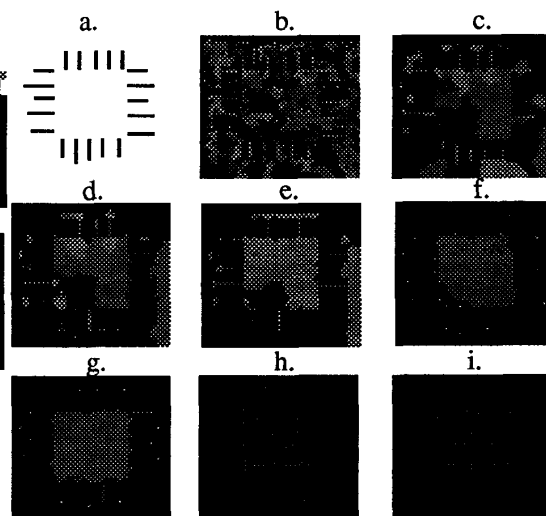
the 4 dark objects eventually get assigned to a single layer. The V_L cost is primarily responsible for bringing the regions down to the same depth plane. The pixel-based depth assignment allows for new surfaces to be generated while the region-based assignment is responsible for merging these regions or moving the regions to the appropriate depth plane to minimize the global cost. Thus, the corners of the occluding rectangle places the pixel above its background. This depth value is propagated to the various neighbors due to the confidence weighting ϕ and eventually meets in the middle region. The parts are then grouped by the region-based depth assignment process. Note that the boundary saliency term V_E eventually plays a role in assigning the labels to bias the formation of straight line boundaries.

A second class of illusory contours is formed by the termination of lines. Figure 8 shows the input to our segmentation algorithm. The result is the formation of a square illusory surface shown in the figure. The background eventually gets assigned to layer 0, the lines get assigned to layer 1, and the center illusory surface to layer 2.

V. CONCLUSION

We have shown that the depth planes, multi-layer representation which is based on surface completion instead of contour extension is appropriate for the segmentation of illusory contours. The study of illusory contours provides insight into the cooperative mechanisms involved in shape perception and object

4347

grouping. Our model can account for the various known classes of illusory contours and the representation is appropriate for the modeling of each.

REFERENCES

[1] T. Darrell and A. Pentland. Discontinuity models and multi-layer description networks. M.I.T. Media Lab Technical Report no. 162, 1991.

[2] S. Geman and D. Geman. Stochastic relaxation, gibbs distribution, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.

[3] S. Grossberg and E. Mingolla. Neural dynamics of form perceptions: boundary completion, illusory figures, and neon color spreading. *Psychology Review*, 92, 1985.

[4] S. Grossberg and E. Mingolla. Neural dynamics of perceptual grouping: Textures, boundaries, and emergent segmentations. *Perception and Psychophysics*, 38(2), 1985.

[5] G. Kanizsa. *Organization in vision*. Praegar, New York, 1979.

[6] S. Madarasmi, T. C. Pong, and D. Kersten. Stereopsis for opaque and for transparent surfaces. In C.L Giles, S.J. Hanson, and J.D. Cowan, editors, *Advances in Neural Information Processing Systems 5*. San Mateo, CA: Morgan Kaufmann Publishers, 1993.

[7] S. Madarasmi, T.C. Pong, and D. Kersten. Multi-layer surface segmentation through energy minimization. Transactions of the IEEE Conference on Computer Vision and Pattern Recognition 93, 1993.

[8] D. Marr and T. Poggio. Cooperative computation of stereo disparity. *Science*, 194:283–287, 1976.