A Computational Approach to Illusory Contour Perception Based on the Tensor Voting Technique

Marcus Hund and Bärbel Mertsching [*]

University of Paderborn, Dept. of Electrical Engineering, GET-Lab,
Pohlweg 47-49, D-33098 Paderborn, Germany
{hund, mertsching}@get.upb.de

Abstract. A computational approach to the perception of illusory contours is introduced. The approach is based on the *tensor voting* technique and applied to several real and synthetic images. Special interest is given to the design of the communication pattern for spatial contour integration, called voting field.

1 Introduction

Illusory contours, also called virtual contours, are perceived contours that have no counterpart in the retinal image of the human vision system. Neurophysical studies have shown that the perception of illusory contours can be found in mammals, birds and insects [20]. The importance of illusory contours becomes obvious regarding the fact that the brains of these animals have developed independently throughout evolution. We can therefore assume that illusory contour perception is not just a malfunction of these visual processing systems, but instead is necessary for object border completion. Also for technical vision systems, the completion of object boundaries that are interrupted due to occlusions or low luminance contrast is an important issue.

For the human vision system, illusory contours have been studied by Gestalt psychologists from the early 20th century on [10, 2, 25]. Schuhmann was one of the first to mention this phenomenon in 1900 [24]. He described illusory contours as contours that are not "objectively present". In the following years the contributions to the field of illusory contour perception comprised the description of optical illusions based on contour perception rather than explaining these illusions. The most famous examples for optical illusions caused by illusory contour perception are the Kanizsa figures shown in Fig. 1 (see also [8]).

In the following we present a computational approach to illusory contour perception in natural scenes. The model uses the position and orientation of detected corners. We therefore developed an algorithm for threshold-free edge detection and a subsequent corner detection, which leads to the question of the

(a) Kanizsa Triangle   (b) Kanizsa Rectangle
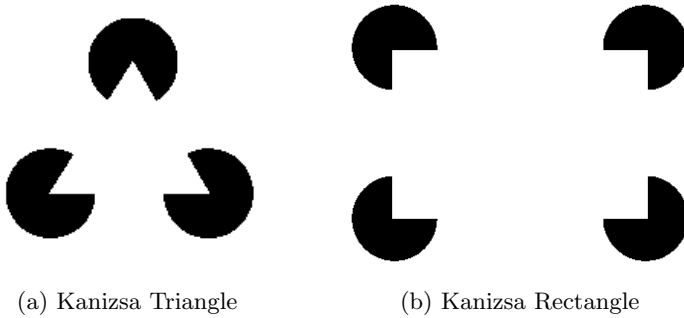
Fig. 1. Kanizsa figures: The black "pacmen" induce the perception of a triangle in (a) and a rectangle in (b)

dependency of the presented results on the preceding low level image processes. On the one hand it can be argued that results on real images suffer from the systematic errors in preceding steps, on the other hand a real image offers a much more complex and realistic test bed. Furthermore few attempts have been made so far to apply illusory contour perception to real images.

2   Related Work

The majority of approaches or models dealing with the problem of spatial contour integration use some kind of bipole connection scheme [6, 21], as introduced by Grossberg and Mingolla [23]. This perceptual grouping kernel usually consists of two symmetric lobes encoding the connection strength and orientation. In [26], Williams and Thornber address the comparison of different methods of aggregating the contributions of neighboring sites of the grouping kernel. For a detailed overview of contour integration approaches, see [5] or [19]. In [19], emphasis is placed on models including illusory contour perception, namely the model of Heitger et al. [7, 22] as a neurophysical inspired computational model and the approach of Zweck and Williams [28] which models the Brownian motion of a particle from source to sink.

The method proposed in this paper uses the tensor voting technique introduced by Medioni et al. [4]. Tensor voting was applied successfully to contour inference problems on synthetic and binary input images in [17]. In [13] and [12], this approach was extended to greyscale images as input, using gabor filtering as a preceding image processing step. In the tensor voting framework the grouping kernel, called stick voting field, is orientational, i.e. with angles from $0°$ to $180°$, and designed for contour inference. Considering illusory contour perception, the use of this stick voting field could make sense in the context of a unified treatment of all contour elements, but would lead to interference of contour elements, especially in the case of amodal completions behind a textured foreground. What is needed for illusory contours including amodal completions is a directional communication pattern (with angles from $0°$ to $360°$), e.g. one lobe,

which was already used in [15] and [16], but addressed to spontaneously splitting figures and binary input images.

3   Edge and Corner Detection

As a preprocessing step we have developed a method for threshold-free edge detection and a subsequent corner detection. This is achieved by applying a recursive search to edge candidates. The edge candidates are local extrema and zero-crossings of the responses of several Gaussian-based filter banks. For an overview of edge detection algorithms, see [3] and [27].

The kernels of our filter functions are shown in Fig. 2. In the case of the edge-filter, Fig. 2(a), the behaviour of the filter is similar to that of the first derivative of a Gaussian. For example, edges produce local extrema in the filter responses with corresponding orientation. Like the edge filters, the corner filters in Fig. 2(b) are defined for different orientation angles. The center-surround filter shown in Fig. 2(c) behaves like the Mexican Hat Operator [11]. Edges produce zero crossings, while lines result in local extrema of the filter responses.

(a)   (b)   (c)
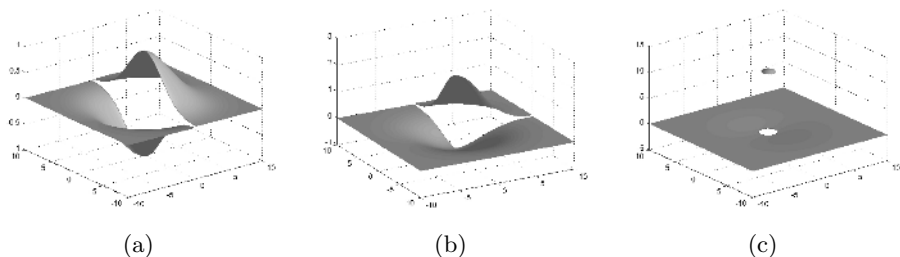
Fig. 2. Filter Masks: (a) Edge filter (b) Corner filter (c) Center-surround filter

In general, edge detection is performed by convolving one filter mask with the image data and in many cases, the filter mask is rotated to gain orientation-sensitive filter responses. Like all differential edge detection schemes these methods suffer from the necessity of defining a threshold. This makes the results of an edge detector highly dependent on the image and on its brightness. Furthermore, to avoid the influence of noise, the size of the convolution mask has to be sufficiently large. This often leads to rounded corners and poor localization. To avoid these disadvantages we use several convolution masks with different shapes and compare their filter responses with each other to decide whether a given point belongs to an edge.

Taken on their own, these convolution masks have several problems in detecting edges or corners, but in spite of these problems, some image positions can be labled as edges with a high probability. If an image position belongs to a zero crossing of the center-surround filter and to a local maximum of the edge filter and furthermore, if this maximum is higher than the corresponding corner

(a)   (b)

(c)   (d)
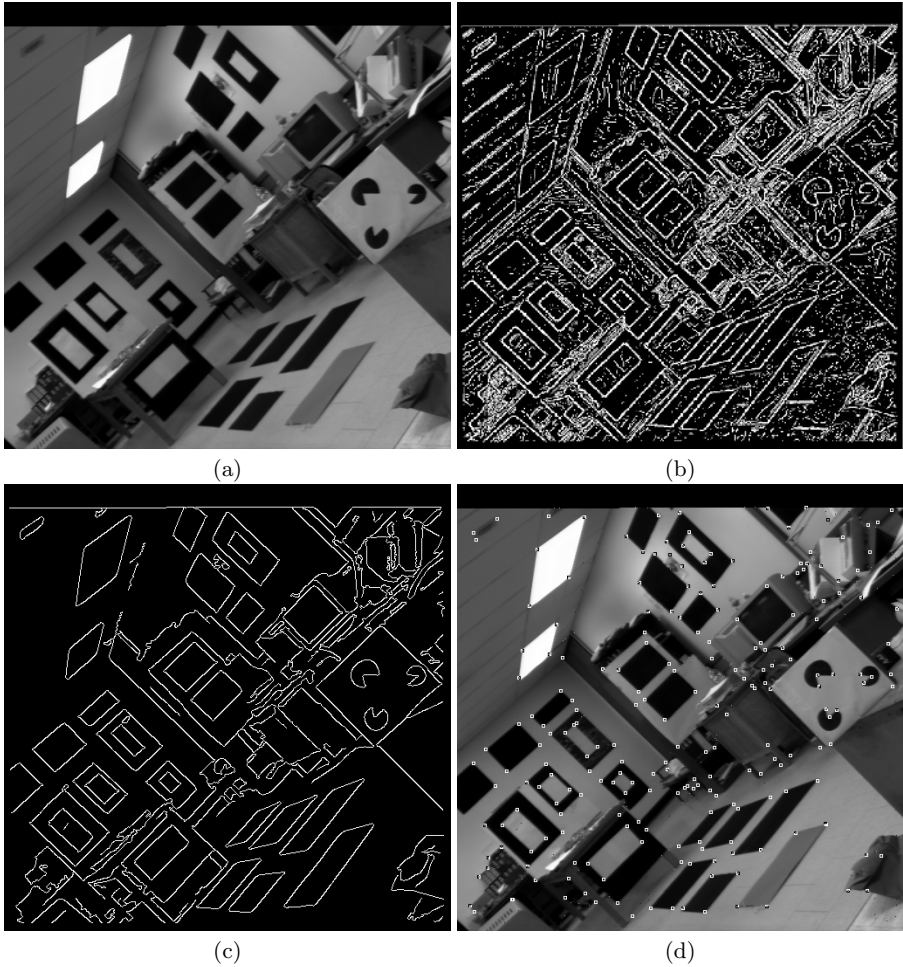
Fig. 3. Lab scene: (a) Input image (b) Edge Candidates (c) Detected edges (d) Detected corners

filter response, the position is very likely to belong to an edge. Starting from these "ideal" edges, we now recursively try to find a good continuation of the edge, using a ranking list that defines e.g. that a zero crossing combined with an edge filter maximum is preferred to a simple zero crossing.

With a search for local minima on the absolute center-surround filter responses, most of the corners are found correctly, except for those that lie on "ideal" edges. Here we have to use the classic approach and compute the orientation differences between neighboring edgels.

The result of the recursive search is shown in Fig. 3(c), detected corners are shown superposed to the input image in Fig. 3(d). A comparison to other corner

detectors is given in [18]. Note that the used method for corner and edge detection is not an integral part of the proposed computational approach to illusory contour perception and can therefore be replaced by any other corner detector providing not only the corner positions but also the associated orientation angles.

4   Tensor Voting

In [17], Medioni, Lee and Tang describe a framework for feature inference from sparse and noisy data called tensor voting. The most important issue is the representation of edge elements as tensors. In the 2D-case, a tensor over $\mathbb{R}^2$ can be denoted by a symmetric 2×2 matrix T with two perpendicular eigenvectors $\boldsymbol{e}_1, \boldsymbol{e}_2$ and two corresponding real eigenvalues $\lambda_1 > \lambda_2$. A tensor can be visualized as an ellipse in 2-D with the major axis representing the estimated tangent direction $\boldsymbol{e}_1$ and its length λ_1 reflecting the saliency of this estimation. The length λ_2 assigned to the perpendicular eigenvector $\boldsymbol{e}_2$ encodes the orientation uncertainty. The definition of saliency measures is deducted from the following decomposition of a tensor into $T = \lambda_1 \boldsymbol{e}_1 \boldsymbol{e}_1^\top + \lambda_2 \boldsymbol{e}_2 \boldsymbol{e}_2^\top$ or equivalently $T = (\lambda_1 - \lambda_2)\boldsymbol{e}_1 \boldsymbol{e}_1^\top + \lambda_2(\boldsymbol{e}_1 \boldsymbol{e}_1^\top + \boldsymbol{e}_2 \boldsymbol{e}_2^\top)$. Then, the weighting factor $(\lambda_1 - \lambda_2)$ represents an orientation in the direction of the eigenvector $\boldsymbol{e}_1$ and thus will be called *curve-* or *stick-saliency*. The second weight λ_2 is applied to a circle, hence it is called *junction-* or *ball-saliency* as its information about multiple orientations measures the confidence in the presence of a junction.
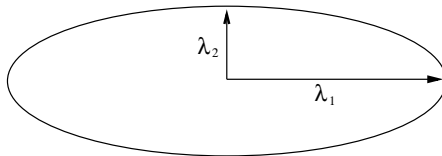
Fig. 4. Visualization of a tensor as an ellipse

Grouping can now be formulated as the combination of elements according to their stick-saliency or ball-saliency. In stick-voting, for each oriented input token the grouping kernel called stick-voting-field (see next section) is aligned to the eigenvector $\boldsymbol{e}_1$. In the following the input tokens consist of detected corners and their associated directions. All fields are combined by tensor addition, i.e. addition of the matrices and spectral decomposition of the sum into eigenvectors and -values. The field is designed to create groupings with neighboring tokens which fulfill the minimal curvature constraint. Hence the orientation of each token of the voting field is defined to lie on a cocircular path.

Note that for junctions or corners neither the tensor representation suffices to encode the at least two different orientations nor is the ball saliency a trustable measure for junctions, since it is highly dependent on the orientations of incoming edges [14].

5   Voting Fields

Given a point P with an associated tangent direction and a point Q with the orientation difference θ between the tangent direction and the direct connection of P and Q. Let ℓ be the distance between P and Q. Then, with

$$r = \frac{\ell}{2 sin\theta} \qquad \text{and} \qquad s = \frac{\ell \cdot \theta}{sin\theta} \quad ,$$

r is the radius of the tangent circle to P going through Q and s is the arc length distance along the circular path (radian).

   Most approaches to spatial contour integration define the connection strength V for P and Q and therefore the shape of the bipole connection scheme via $V = V_d \cdot V_c$ with a distance term V_d and a curvature term V_c. In [7], Heitger et al. use

$$V_{d1} = e^{-\frac{\ell^2}{2\sigma^2}} \qquad \text{and} \qquad V_{c1} = \begin{cases} \cos^k(\frac{\pi/2}{\alpha} \cdot \theta) & \text{if } |\theta| < \alpha \\ 0 & \text{otherwise} \end{cases}$$

with $k = 2n$, $n \in \mathbb{N}$ and an opening angle $2\alpha = \pi$. Hansen and Neumann also use V_{d1} and V_{c1}, but with $k = 1$ and $\alpha = 10°$ [6]. In [17], Medioni et al. define the proximity term V_{d2} and the curvature term V_{c2} as follows:

$$V_{d2} = e^{-\frac{s^2}{2\sigma^2}} \qquad \text{and} \qquad V_{c2} = e^{-\frac{c \cdot \rho^2}{\sigma^2}} \qquad \text{with} \qquad \rho = \frac{2 sin\theta}{\ell}$$

c is a positive constant and ρ is nothing else than the inverse radius of the osculating circle. This results in a curvature measure that is highly dependent on scale.
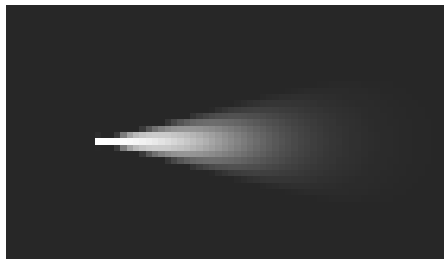

Fig. 5. Stick saliency for the half lobe stick voting field with $V = V_{d2} \cdot V_{c1}$, $k = 1$ and $\alpha = 15°$

   To achieve a clear separation of distance along the circular path and its curvature, we choose V_{d2} and V_{c1}. The results presented in the next chapter are computed with a directional one-lobe voting field and $V = V_{d2} \cdot V_{c1}$, $k = 1$ and $\alpha = 15°$, i.e. an opening angle of $30°$.

6   Results

Fig. 6(a), (d) and (g) show detected edges and corners and their associated orientations, (b), (e) and (f) show stick saliencies after tensor voting and (c), (f) and (i) show extracted illusory contours superposed to the previously detected contours. It is remarkable, that in Fig. 6(f) the amodal completions of the circles are found while this is not the case in Fig. 6(c) and (i). This is due to the acute connection angles in the latter two cases as can be seen in the images showing the stick saliencies.

(a)   (b)   (c)

(d)   (e)   (f)

(g)   (h)   (i)
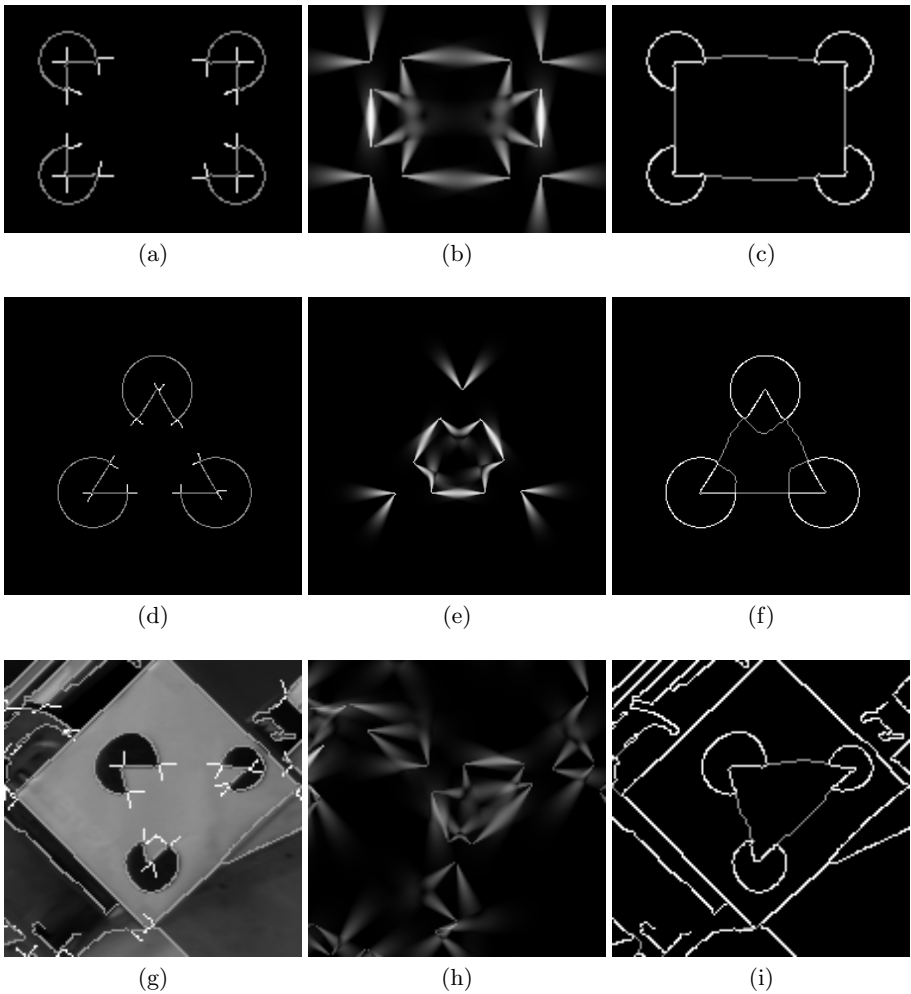
Fig. 6. Top row: Results for Fig. 1(b), middle row: results for Fig. 1(a), bottom row: results for the Kanizsa triangle in Fig. 3. For further description see text.
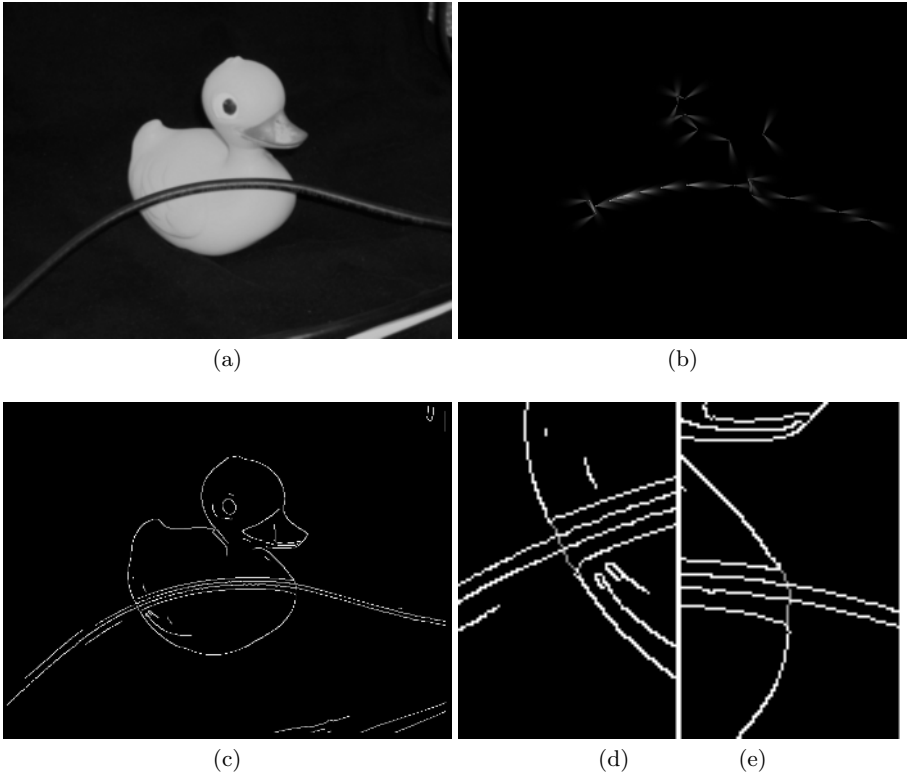
Fig. 7. Rubber duck image: (a) input image, (b) stick saliencies induced by corners and associated orientations, (c) illusory contours superposed to detected edges, (d) and (e) magnification of regions containing illusory contours

In Fig. 7, the rubber duck is partially occluded by a black cable. Note that there are some false assigned corners due to light reflections on the cable. The voting fields cast by these corners interfere with the fields generated at the duck's object boundary and hence compromise the correct detection of amodal completions (Fig. 7(b)). This illustrates that a unified treatment of edge segments and corners would disturb the perception of amodal completions, at least for this low level image processing step. Anyhow just the two desired amodal completions of the duck's object boundary are marked as illusory contours, see Fig. 7(d) and (e), so the correct virtual contours are found.

7   Conclusion

An approach to illusory contour perception has been introduced and successfully applied to synthetic and real images. There are some natural limitations to a low level image processing model for illusory contour perception. For a certain

stage of grouping a knowledge base is required which leaves the field of low level image processing. Furthermore, the human vision system derives its enormous capabilities not only from the "hardware implementation" as a parallel network but also from the fact that several cues like depth and motion are considered when detecting object boundaries.

With our approach we have shown that good results for illusory contour perception can be achieved even in a low level image processing step.

8   Future Work

Currently, our model does not distinguish between modal and amodal completions and the contours are not assigned to certain object boundaries. Consequently, unit formation will be substantial for future research. For further discussion about unit formation, see [9] and [1].

References

1. B. L. Anderson, M Singh, and R. W. Fleming. The interpolation of object and surface structure. *Cognitive Psychology*, 44:148–190, 2002.
2. Walter H. Ehrenstein, Lothar Spillmann, and Viktor Sarris. Gestalt issues in modern neuroscience. *Axiomathes*, 13(3):433–458, 2003.
3. Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Prentice Hall, 2002.
4. Gideon Guy and Gerard Medioni. Inferring global perceptual contours from local features. *International Journal of Computer Vision*, 20(1-2):113–133, 1996.
5. Thorsten Hansen. *A neural model of early vision: Contrast, contours, corners and surfaces*. PhD thesis, Universität Ulm, 2003.
6. Thorsten Hansen and Heiko Neumann. Neural mechanisms for representing surface and contour features. In *Emergent Neural Computational Architectures Based on Neuroscience - Towards Neuroscience-Inspired Computing*, pages 139–153. Springer-Verlag, 2001.
7. F. Heitger and R. von der Heydt. A computational model of neural contour processing: Figure-ground segregation and illusory contours. In *International Conference on Computer Vision*, pages 32–40, 1993.
8. G. Kanizsa, editor. *Organization in Vision*. Praeger, 1979.
9. P.J. Kellman, S. E. Guttman, and T. D. Wickens. Geometric and neural models of object perception. In *From fragments to objects: Segmentation and grouping in vision*. T. F. Shipley, Ed, and P.J. Kellman, Ed. . Oxford, UK: Elsevier Science, 2001.
10. K. Koffka. *Principles of Gestalt psychology*. Harcourt Brace, New York, 1935.
11. D. Marr. *Vision: a computational investigation into the human representation and processing of visual information*. W. H. Freeman, San Francisco, 1982.
12. A. Massad, M. Babos, and B. Mertsching. Application of the tensor voting technique for perceptual grouping to grey-level images. In L. van Gool, editor, *Pattern Recognition, 24th DAGM Symposium (DAGM2002)*, pages 306–313, 2002.
13. A. Massad, M. Babos, and B. Mertsching. Perceptual grouping in grey level images by combination of gabor filtering and tensor voting. In R. Kasturi, D. Laurendeau, and C. Suen, editors, *ICPR*, volume 2, pages 677–680, 2002.

14. A. Massad, M.Babos, and B. Mertsching. Application of the tensor voting technique for perceptual grouping to grey-level images: Quantitative evaluation. 2003. Intl. Symposium on Image and Signal Processing and Analysis.

15. A. Massad and G. Medioni. 2-D Shape Decomposition into Overlapping Parts. In C. Arcelli, L. Cordella, and G. Sanniti di Baja, editors, *Visual Form 2001, 4th International Workshop on Visual Form (IWVF 4)*, pages 398 – 409, Capri, Italy, January 2001.

16. A. Massad and B. Mertsching. Segmentation of Spontaneously Splitting Figures into Overlapping Parts. In B. Radig and S. Florcyk, editors, *Pattern Recognition, 23rd DAGM Symposium*, pages 25 – 31, January 2001.

17. G. Medioni, M.-S. Lee, and C.-K. Tang. *A Computational Framework for Segmentation and Grouping*. Elsevier Science, 2000.

18. Farzin Mokhtarian and Riku Suomela. Robust image corner detection through curvature scale space. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(12):1376–1381, 1998.

19. H. Neumann and E. Mingolla. Computational neural models of spatial integration in perceptual grouping. In T. Shipley and P. Kellman, editors, *From fragments to units: Segmentation and grouping in vision*, pages 353–400. Elsevier Science, Oxford, UK, 2001.

20. Andreas Nieder. Seeing more than meets the eye: processing of illusory contours in animals. *Journal of Comparative Physiology A: Sensory, Neural, and Behavioral Physiology*, 188(4):249–260, 2002.

21. Pierre Parent and Steven Zucker. Trace inference, curvature consistency, and curve detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(8):823–839, 1989.

22. Esther Peterhans and Friedrich Heitger. Simulation of neuronal responses defining depth order and contrast polarity at illusory contours in monkey area v2. *Journal of Computational Neuroscience*, 10(2):195–211, 2001.

23. W. D. Ross, S. Grossberg, and E. Mingolla. Visual cortical mechanisms of perceptual grouping: interacting layers, networks, columns, and maps. *Neural Netw.*, 13(6):571–588, 2000.

24. F. Schumann. Beiträge zur Analyse der Gesichtswahrnehmungen. Erste Abhandlung. Einige Beobachtungen über die Zusammenfassung von Gesichtseindrücken zu Einheiten. *Zeitschrift für Psychologie und Physiologie der Sinnesorgane*, 23:1–32, 1900. English translation by A. Hogg (1987) in The perception of Illusory Contours Eds S Petry, G.E. Meyer (New Yourk: Springer) pp 40-49.

25. Max Wertheimer. Untersuchungen zur Lehre von der Gestalt II. *Psychologische Forschung*, 4:301–350, 1923.

26. Lance R. Williams and Karvel K. Thornber. A comparison of measures for detecting natural shapes in cluttered backgrounds. *International Journal of Computer Vision*, 34(2/3):81–96, 2000.

27. D. Ziou and S. Tabbone. Edge detection techniques: an overview. *International Journal on Pattern Recognition and Image Analysis*, 8(4):537–559, 1998.

28. John W. Zweck and Lance R. Williams. Euclidean group invariant computation of stochastic completion fields using shiftable-twistable functions. In *ECCV (2)*, pages 100–116, 2000.