IMAGE EDGES FROM AREA MORPHOLOGY

Scott T. Acton[†] *and Dipti Prasad Mukherjee*

[†]School of Electrical and Computer Engineering
Oklahoma State University
Stillwater, Oklahoma 74078 USA
sacton@okstate.edu

Electronics and Communications Sciences Unit
Indian Statistical Institute
India 700035
dipti@isical.ac.in

ABSTRACT

This paper introduces an edge detection process based on area morphology. Area open-close and area close-open operators are used to generate scaled image representations for feature extraction. The edges are defined by the boundaries of the scaled objects in the area-filtered images. From the area open-close and close-open operators, thin, closed contours suitable for image segmentation are produced. The edge maps allow exact specification of the minimum area for the extracted regions and are Euclidean invariant and causal through scale space. Results are given that demonstrate the effectiveness of the area operator-based edge detection. In contrast to traditional edge detectors, edge detection via area morphology provides well-localized boundaries and does not require thresholding.

1. INTRODUCTION

We propose an approach to edge detection based on *area operators*. Area operators, such as *area open* and *area close*, modify an image by removing connected components within the image level sets that do not meet a prescribed minimum area. Although area open and area close are morphological filters in the sense that they are idempotent and increasing operators, area open and area close depend only on an area (scale) parameter and do not depend on structuring element shape. In this manner, the area operators avoid the associated problems of imposing the structuring element shape on a processed image. With standard morphology, the structuring element shape can produce artifacts and leads to edge localization errors.

Area operators have been utilized in general filtering for image enhancement and image reconstruction [6], [7]. The filters have the desirable property of connected invariance – connected components in the image level sets (that may correspond to image objects) are never partially removed in filtering. As a result, the filters may be used for noise and detail removal without boundary movement, and a straightforward method to scale an image is provided. Although filtering and reconstruction applications have been successful, a suitable edge detection method that exploits the area operators has not been explored.

In this paper, we define edge detectors that utilize area operators to provide a scaled image representation. Edges are defined as boundaries between the objects that result from the area-based nonlinear filters. The edge detectors provide closed, thin contours that correspond to objects of a specified scale. The edge detection process is invariant to translation and rotation. Also, the edge maps are causal – new (false) edges are not produced as scale is increased. At the conclusion of the paper, comparative examples are given that demonstrate the salient properties of the area operator-based edge detectors.

2. THEORY AND ANALYSIS

The set $\mathbf{B}$ is defined on domain Ω. We may consider $\mathbf{B}$ as a binary image with values of $B(\mathbf{x}) \in \{0, 1\}$ at locations $\mathbf{x} \in \Omega$. With discrete-domain images, $\Omega \subset \mathbf{Z}^2$. We say that $\mathbf{x} \in \mathbf{B}$ if $B(\mathbf{x}) = 1$. The *connected component* at $\mathbf{x}$, $\mathbf{C_B}(\mathbf{x})$, is the set of locations $\mathbf{y}$ where there exists an unbroken path between $\mathbf{x}$ and $\mathbf{y}$: $\mathbf{C_B}(\mathbf{x}) = \{\mathbf{y}: \exists\, \mathbf{P_B}(\mathbf{x}, \mathbf{y})\}$ where the path $\mathbf{P_B}(\mathbf{x}, \mathbf{y})$ is a finite sequence of neighboring pixels $\{\mathbf{x}, \mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N, \mathbf{y}\}$ such that $N < \infty$ and $\mathbf{z}_1 \in \mathbf{N}(\mathbf{x})$, $\mathbf{z}_2 \in \mathbf{N}(\mathbf{z}_1)$, $\mathbf{z}_3 \in \mathbf{N}(\mathbf{z}_2)$, …., $\mathbf{y} \in \mathbf{N}(\mathbf{z}_N)$. Here, $\mathbf{N}(\mathbf{x})$ is the set of neighboring pixels for location $\mathbf{x}$. If $B(\mathbf{x}) = 0$, then the connected component $\mathbf{C_B}(\mathbf{x}) = \varnothing$.

For a set $\mathbf{B}$, we can define the area open operation by

$$\mathbf{x} \in \overset{a}{\circ}(\mathbf{B}) \text{ if } |\mathbf{C_B}(\mathbf{x})| \geq a, \tag{1}$$

0-7803-6293-4/00/$10.00 ©2000 IEEE.

where $|\mathbf{C_B(x)}|$ is the cardinality (area in the discrete sense) of the connected component, and a is the minimum area. This implies that

$$\mathbf{x} \notin \overset{a}{\circ}(\mathbf{B}) \text{ if } |\mathbf{C_B(x)}| < a. \quad (2)$$

Similarly, if $\mathbf{A} = \overset{a}{\circ}(\mathbf{B})$, we can say that the value of the area open is defined by

$$A(\mathbf{x}) = 1_{[\mathbf{x} \in \overset{a}{\circ}(\mathbf{B})]} \quad (3)$$

where $1_{[e]}$ is the indicator function; $1_{[e]} = 1$ when the expression e is true, and $1_{[e]} = 0$ otherwise.

Let the complement of the set $\mathbf{B}$ be denoted by $\mathbf{B}^c$. Where $B(\mathbf{x}) = 0$, $B^c(\mathbf{x}) = 1$, and where $B(\mathbf{x}) = 1$, $B^c(\mathbf{x}) = 0$. We refer to $\mathbf{B}$ as the *on-set* and to $\mathbf{B}^c$ as the *off-set*. To form the area close operation, we have

$$\mathbf{x} \in \overset{a}{\bullet}(\mathbf{B}) \text{ if } |\mathbf{C_{B^c}(x)}| \geq a, \quad (4)$$

where a denotes the minimum area of connected components in the off-set. This infers that

$$\mathbf{x} \notin \overset{a}{\bullet}(\mathbf{B}) \text{ if } |\mathbf{C_{B^c}(x)}| < a. \quad (5)$$

Concatenation of the area open and area close operators yields important scale-generating operators. Area open-close is denoted by $\overset{a}{\bullet}(\overset{a}{\circ}(\mathbf{B}))$, the area close-open operation is $\overset{a}{\circ}(\overset{a}{\bullet}(\mathbf{B}))$. Both operations remove connected components of area less than a in both the on-set and the off-set. Thus, area open-close (AOC) and area close-open (ACO) can be employed to control the minimum scale of objects in $\mathbf{B}$. However, $\overset{a}{\bullet}(\overset{a}{\circ}(\mathbf{B})) \neq \overset{a}{\circ}(\overset{a}{\bullet}(\mathbf{B}))$ in general.

To apply area open, area close, AOC and ACO to grayscale imagery, a stacking or threshold decomposition process is utilized. An image $\mathbf{I}$ can be decomposed into *level sets* $\mathbf{I}_t$, where $\mathbf{I}_t = \{\mathbf{x}: I(\mathbf{x}) \geq t\}$. The image intensity can be obtained from the level sets using

$$I(\mathbf{x}) = \max\,\{t: \mathbf{x} \in \mathbf{I}_t\}. \quad (6)$$

This level set definition allows the area operators to be applied to each level set (independently) – the image is then reconstructed using (6). For example, if

$$\mathbf{J} = \overset{a}{\bullet}(\overset{a}{\circ}(\mathbf{I})),$$

then

$$J(\mathbf{x}) = \max\,\{t: \mathbf{x} \in \overset{a}{\bullet}(\overset{a}{\circ}(\mathbf{I}_t))\}.$$

With multi-valued (grayscale) imagery, we can think of the area open operation as removing small (area $< a$) bright objects, while the area close removes small dark objects.

The goal of this paper is definition of edge detection using area operators. Here, we define potential edges as *level lines* – boundaries of connected components within the level sets. In a level set $\mathbf{I}_t$, a level line exists at locations within a connected component in $\mathbf{I}_t$ that have neighbors that are not members of the connected component. We define the set of level line locations for $\mathbf{I}_t$ as $\mathbf{L}(\mathbf{I}_t)$ where $\mathbf{x} \in \mathbf{L}(\mathbf{I}_t)$ if $\mathbf{x} \in \mathbf{I}_t$ and there exists a $\mathbf{y}$ such that $\mathbf{y} \in \mathbf{N}(\mathbf{x})$, but $\mathbf{y} \notin \mathbf{I}_t$. A level line exists at $\mathbf{x}$ in the image $\mathbf{I}$ if a level line exists in any level set $\mathbf{I}_t$. So, the set of all level lines for $\mathbf{I}$ is given by $\mathbf{L}(\mathbf{I})$ where $\mathbf{x} \in \mathbf{L}(\mathbf{I})$ if $\mathbf{x} \in \mathbf{L}(\mathbf{I}_t)$ for any t.

If we simply selected all level lines as edges, we would have an over-segmented edge map. Furthermore, we would introduce false edges. Consider the case of a ramp (slowly varying) edge. Each incremental change in intensity would produce a level line, and each ramp edge would produce a multitude of edges if the level lines were used directly as edges. Therefore, we must define boundaries between *objects*.

First, we define an object in the area operator sense. In this case, objects are classified as *ascending* or *descending* (brighter or darker than the surrounding pixels, respectively). Consider two sets:

$$\mathbf{A}(\mathbf{x}) = \{\mathbf{y}: J(\mathbf{y}) \geq J(\mathbf{x})\}$$

and

$$\mathbf{D}(\mathbf{x}) = \{\mathbf{y}: J(\mathbf{x}) \geq J(\mathbf{y})\},$$

where $\mathbf{x}$ is a member of both sets, $\mathbf{J} = \overset{a}{\bullet}(\overset{a}{\circ}(\mathbf{I}))$ in the case of AOC, and $\mathbf{J} = \overset{a}{\circ}(\overset{a}{\bullet}(\mathbf{I}))$ for ACO. The corresponding connected components containing $\mathbf{x}$ for $\mathbf{A}(\mathbf{x})$ and $\mathbf{D}(\mathbf{x})$ are $\mathbf{C_A}(\mathbf{x})$ and $\mathbf{C_D}(\mathbf{x})$, respectively. $\mathbf{C_A}(\mathbf{x})$ defines a connected group of pixels that have equal or greater intensity than $J(\mathbf{x})$. Likewise, $\mathbf{C_D}(\mathbf{x})$ defines a connected component of pixels that have equal or less intensity than $J(\mathbf{x})$. If $\mathbf{x}$ is a part of an ascending object, then $|\mathbf{C_A}(\mathbf{x})| < |\mathbf{C_D}(\mathbf{x})|$. The object at $\mathbf{x}$ is defined to be descending if $|\mathbf{C_A}(\mathbf{x})| > |\mathbf{C_D}(\mathbf{x})|$. When the object is ascending, $|\mathbf{C_A}(\mathbf{x})|$ represents the area (scale) of the ascending object. The scale of a descending object is given by $|\mathbf{C_D}(\mathbf{x})|$.

Hence, the boundaries between descending and ascending objects are found where $|\mathbf{C}_A(\mathbf{x})| = |\mathbf{C}_D(\mathbf{x})|$. Edges in an image that have been processed by AOC or ACO can be located by computing zero crossings in $\mathbf{Z}$ where $Z(\mathbf{x}) = |\mathbf{C}_A(\mathbf{x})| - |\mathbf{C}_D(\mathbf{x})|$. These zero crossing will represent significant level lines – the object boundaries. The scale of the edge map is determined by the area parameter, a, used in the AOC or ACO operation. Since AOC and ACO remove all connected components within the image level sets that do not have a minimum area of a, the minimum area for any object is a under $\overset{a}{\bullet}(\overset{a}{\circ}(\mathbf{I}))$ and $\overset{a}{\circ}(\overset{a}{\bullet}(\mathbf{I}))$. We call the set of edge locations, $\mathbf{E}(\mathbf{I}, a)$, the edge map for $\mathbf{I}$ at scale a.

In addition to the control of scale, the benefits of AOC and ACO-based edge detection are production of closed, thin contours, Euclidean invariance, and edge causality. Because each connected component within a level set has a closed boundary, the AOC and ACO-based edge detectors produce closed contours. Thus, the closed contours define regions and can be utilized in image segmentation, the subdivision of the image into constituent regions. The boundaries between ascending and descending objects are always single-pixel in thickness, since an edge in $\mathbf{E}(\mathbf{I}, a)$ represents a change in signal concavity (ascending to descending or *vice versa*). In the discrete-domain case, double edges may occur where an object is of one or two pixels in width. However, a single intensity transition (monotonic change) will not produce multiple edges in this framework.

The edge detector is invariant to translation and rotation. Because rotation and translation do not alter the area of the level set connect components (ignoring discretization error), the AOC/ACO edge operators are Euclidean invariant. Furthermore, new features are not introduced at coarser scales. Since the AOC and ACO operators only preserve and remove connected components in their entirety, the operators do not produce new regions at coarser scale. Therefore, we say that the edge detection operation is causal.

3. RESULTS AND CONCLUSIONS

Sample results from this edge detection method are shown in Figs. 1-3. In Fig. 1, notice the improvement in edge localization compared to the edge detection results given by isotropic diffusion (linear Gaussian filtering) and anisotropic diffusion [1]. Using Fig. 2(a) as input, the area morphology approach gives a more semantically meaningful edge detection (shown in Fig 2(b)) as compared to the result of the Laplacian-of-a-Gaussian (LoG) [4] in Fig. 2(c). The area morphology does not require a threshold on intensity, while the LoG technique must use a threshold to avoid over-segmentation.

Fig. 3 reveals the robustness of the area morphology approach in the presence of noise. Given the noisy image shown in Fig. 3(b), only the area morphology edge detector gives an edge map that corresponds to the object boundaries (see Fig. 3(d)). The LoG approach (Fig. 3(e)) and the Canny approach [3] (Fig. 3(g)) are sensitive to the noise, distort the boundaries, and require thresholding.

The benefits of edge detection using area operators are edge localization, causality, and Euclidean invariance. The approach produces thin, contiguous edges and avoids *ad hoc* thresholds. To extend this technique, we are examining methods of combining edge intensity information with the scale information obtained from area morphology. An example result of such an approach is shown in Fig. 4.

Currently, we are utilizing AOC and ACO for the generation of image scale space – a family of images that vary from fine to coarse. The scale space representations are used in image classification, hierarchical search processes, and feature extraction for content-based retrieval [2], [5].

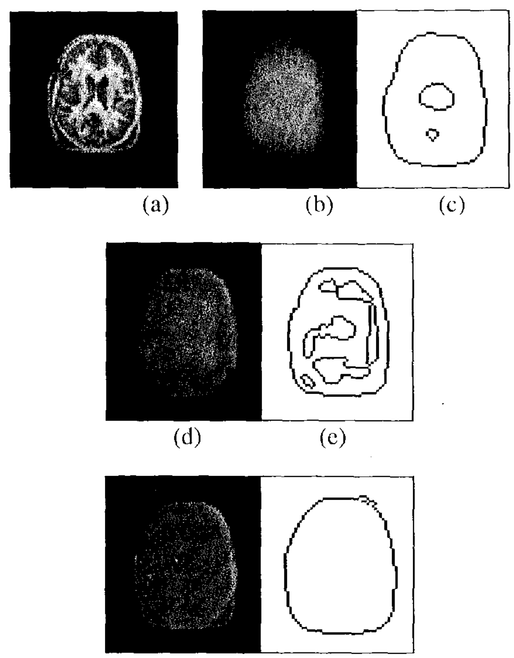
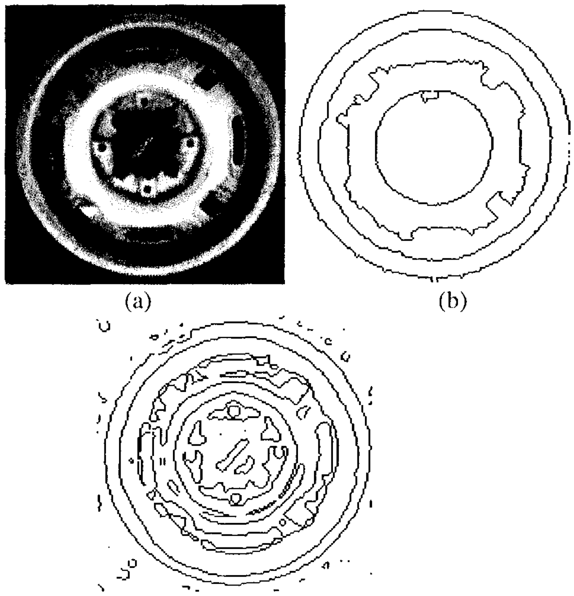
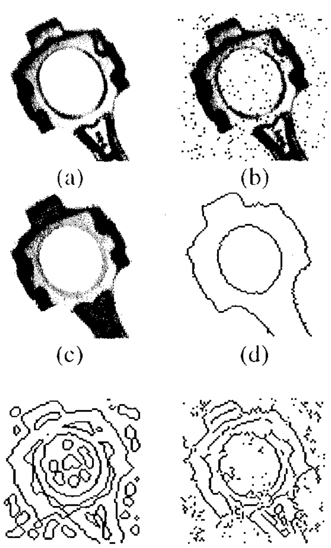


Fig 1: Extracting the coarse boundary contour from the brain image shown in (a); (b) using isotropic, linear diffusion (via Gaussian convolution); (c) edges found in (b); (d) using anisotropic, nonlinear diffusion [1]; (e) edges found in (d); (f) using AOC with $a = 300$; (g) edges found in (f).



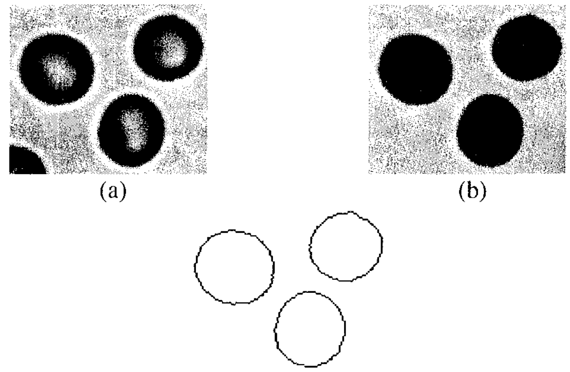
(a) (b) (c)

Fig. 2: (a) Rim image; (b) edge map of Fig. 2 (a) using AOC with a = 800; (c) LoG edge map at $\sigma = 2$.



(a) (b) (c) (d) (e) (f)

Fig. 3: (a) Original image; (b) corrupted image (5% salt and pepper noise); (c) AOC-scaled image at a=200; (c) LoG edge map at $\sigma = 2$; (f) edge map from Canny algorithm [3] at $\sigma = 1$.



(a) (b) (c)

Fig. 4: (a) Cells image; (b) after AOC operation with a = 500; (c) edge map combining area and intensity based edge maps.

4. REFERENCES

[1] S.T. Acton, “Multigrid anisotropic diffusion,” *IEEE Trans. Image Processing*, vol. 7, pp. 280-291, 1998.

[2] S.T. Acton and D.P. Mukherjee, "Scale space classification using area morphology," accepted for publication, *IEEE Transactions on Image Processing*.

[3] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, pp. 679-698, 1986.

[4] D. Marr and E. Hildreth, "Theory of edge detection," *Proc. R. Soc., London*, vol. B207, pp. 187-217, 1980.

[5] D.P. Mukherjee and S.T. Acton, "Document page segmentation using multiscale clustering," *Proc. IEEE Int. Conf. on Image Processing*, Kobe, Japan, Oct. 25-29, 1999.

[6] P. Salembier and J. Serra, “Flat zones filtering, connected operators, and filters by reconstruction,” *IEEE Trans. Image Processing*, vol. 4, pp.1153-1160, 1995.

[7] L. Vincent, “Morphological gray scale reconstruction in image analysis: applications and efficient algorithms,” *IEEE Trans. Image Processing*, vol. 2, pp. 176-201, 1993.