Watershed Pyramids for Edge Detection

Anthony S. Wright and Scott T. Acton

The Oklahoma Imaging Laboratory
School of Electrical and Computer Engineering, Oklahoma State University
Stillwater, OK 74078

ABSTRACT

In this paper, we present a multiresolution implementation of the watershed segmentation algorithm. Our approach uses the morphological pyramid to form a scale space representation, offering a significant reduction in computational cost. In addition to increased efficiency, the multiresolution approach avoids the over-segmentation problem of traditional fixed scale watershed algorithms. As shown in the examples, the watershed pyramids produce edge maps corresponding to the desired scale without sacrificing accuracy in edge location.

I. INTRODUCTION

Edge detection is one of the classic problems in image processing, with applications ranging from biomedical systems to manufacturing. Although a variety of edge detection schemes have been demonstrated, no one method has proven to be ideal. One technique that has garnered interest in recent years is the watershed algorithm. The concept of the watershed is borrowed from topography. The image is considered as a topographic surface, where the low valued areas represent catchment basins and the ridges are watershed boundaries. By applying the watershed approach to the gradient magnitude of the image, it is possible to find edges. In this paper we introduce a method of determining watershed boundaries in a coarse-to-fine manner. The application of a pyramid proves to not only be more computationally efficient, but also results in a more effective segmentation of the image.

One advantage of using the watershed for edge detection is that the resulting edges are continuous and of single-pixel width, since the edges are defined as boundaries between connected regions called watersheds. The shortcomings of the watershed approach are the prohibitive computational expense and the over-segmentation of the image. Here, we describe a multiresolution watershed algorithm. We demonstrate that the watershed pyramid algorithm both increases the computational efficiency of the watershed and is tunable to the desired feature scale.

The notion of using a multi-scale representation of the watershed to avoid over-segmentation is not new. Gauch and Pizer used Gaussian blurring to represent images at different scales [1], and more recently Jackway applied a morphological pyramid to the watershed [2]. Our approach differs from the previous approach in that the watershed is applied at a coarse level of the pyramid then the information is propagated through the finer levels of the pyramid to the original image. This coarse-to-fine hierarchy provides a dramatic reduction in computational expense.

II. THEORY

The watershed - In order to understand the watershed, it is necessary consider the image as a surface, where high pixel values correspond to peaks and low pixel values correspond to valleys. Just as with actual watersheds, if a drop of water were to fall on any point of the contour it would find its way to lower ground until it reached a local minimum. These local minima are referred to as catchment basins, and all points that drain into the same catchment basin are referred to as members of the same watershed [3].

The first step in performing edge detection using the watershed approach is to calculate the local gradients within the image; these gradient magnitude values are also blurred to insure that each magnitude value is locally unique [1]. Given an original image **I** and a low variance Gaussian kernel **G**, the blurred gradient **B** becomes

This work was supported by the National Aeronautics and Space Administration under EPSCOR grant NCC5-171.

0-8186-8183-7/97 $10.00 © 1997 IEEE

$$\mathbf{B} = \mathbf{G} * \nabla \mathbf{I} \quad (1)$$

where $\nabla \mathbf{I}$ is defined such that

$$\nabla I(i,j) = [I(i,j+1) - I(i,j)]^2 + [I(i,j) - I(i+1,j)]^2 .$$

The next step is to find the local minima of **B**. An element $m(i,j)$ is said to be a local minimum if $m(i,j)<p(i_0,j_0)$ $\forall (i_0,j_0)\in \mathbf{N}(i,j)$ where $\mathbf{N}(i,j)$ represents spatial neighborhood of the element at row i and column j in eight connectivity. These local minima m_i represent the catchment basins of the image and are each assigned a unique label greater than zero. The final step is to follow each element in **B** toward its lowest valued neighbor until it merges into one of the catchment basins, m_i. Once an element merges with a catchment basin m_i it assumes the label of that basin. We will define the above function to be represented as WS so the watershed **W** is defined as

$$\mathbf{W} = \mathrm{WS}(\mathbf{G} * \nabla \mathbf{I}) \quad (2)$$

where $W(i,j)$ is the label of watershed in which element (i,j) is a member. After all of the watersheds have been labeled, the resulting edge map **E** is defined as

$$E(i,j) = \begin{cases} 1 & \nabla W(i,j)>0 \\ 0 & otw. \end{cases} \quad (3)$$

Unfortunately, even with prefiltering, the above process results in over-segmented images. There are a number of methods to merge watersheds and remedy the problem of over-segmentation in the literature, and the technique employed depends on the nature of the application. For the examples shown in section III, a simple statistical similarity approach was used to perform region combinations.

The first step is to assign each watershed label a pixel intensity $X(d)$ representative of the original image. This is accomplished by

$$X(d) = \frac{\sum_{(i,j)\in \mathbf{WS}_d} I(i,j)}{|\mathbf{WS}_d|} \quad (4)$$

where $\mathbf{WS}_d \equiv \{(i,j) \Rightarrow W(i,j) = d\}$. The next step is to merge watersheds in **W** based on the pixel intensities **X**, and a predetermined threshold T (based on the standard deviation of the pixel intensities) forming a modified watershed **W′**.

$$W'(i,j) = \begin{cases} d_i & \{X[W(i,j)] - X(d_i)\} < T \\ W(i,j) & otw. \end{cases} \quad (5)$$

for all $d_i \in \mathbf{X}$.

Watershed pyramids - The concept behind the pyramid is to create a scale space where only the most prominent features appear at the coarsest representation. In a morphological pyramid, the original image is used to create a series of images at coarser scales by morphologically filtering each level and subsampling. Pyramid level L is defined by

$$\mathbf{I}_L = \left[\left(\mathbf{I}_{(L-1)} \circ \mathbf{K} \right) \bullet \mathbf{K} \right]_{\downarrow 2} \quad L = 0,1,\ldots,n \quad (6)$$

where $\mathbf{I}_0$ is the original image, $[\cdot]_{\downarrow 2}$ represents a down sampling by a factor of two in each spatial dimension (along rows and columns), $(\mathbf{I} \circ \mathbf{K})$ represents the morphological opening of the image **I** with structuring element **K**, and $(\mathbf{I} \bullet \mathbf{K})$ represents the morphological closing of the image **I** with structuring element **K**. The parameter n is largest integer such that for an MxN image $\left\{ M/2^n, N/2^n \right\} \geq 1$. In the pyramid, every element in level L has one "parent" in level $L+1$ and four "children" in level $L-1$ since there is a 4 to 1 pixel reduction with every ascending level. The open-close operation was chosen because it introduces less distortion and is less biased than an individual open or close operation [4].

Multiresolution watersheds have been demonstrated by both Gauch and Pizer [1] and Jackway [2], but our multiresolution algorithm introduces a method where the watershed algorithm is applied at a coarse level, and the edges are propagated back to finer representations without performing the watershed algorithm at each level of the pyramid. Once the watershed has been applied at a coarse level L, each element of $\mathbf{W}_{L-1}$ must be linked to an element in $\mathbf{W}_L$. This is accomplished by

$$W_{L-1}(i_0,j_0) = \begin{cases} W_L(i,j) & \nabla W_L(i,j) = 0 \\ 0 & otw. \end{cases} \quad (7)$$

for all $(i_0,j_0)\in \mathbf{C}(i,j,L)$ where $\mathbf{C}(i,j,L)$ represents the children of element (i,j) at level L. In (7), if $\nabla W_L(i,j)=0$, signifying that no change in watershed label exists in that neighborhood, we say that the label for the children of $W_L(i,j)$ is known to be equal to the label of $W_L(i,j)$. If $\nabla W_L(i,j)\neq 0$ then the label of the children in question is uncertain. The final step in linking level $L-1$ to level L is to perform the watershed on the elements of $W_{L-1}(i,j)$ with undefined labels.

$$W_{L-1}(i_0,j_0) = \mathrm{WS}\left[B_{L-1}(i_0,j_0) \right] \quad (8)$$

for all $(i_0,j_0)\in \{W_{L-1}(i,j)=0\}$, where **B** is defined in (1). Since the pyramid structure insures the causality of watersheds, no watershed can appear in level $L-1$ that did not exist in level L. For this reason any new watershed which formed while evaluating the regions of uncertainty must be flooded into its nearest neighbor by equation (5) with the condition that any watershed formed in level $L-1$ must merge with one and only one watershed which existed in level L. This linking continues until $L-1=0$, and finally, edge detection is performed on level 0 using equation (3).

In practice the multiresolution algorithm decreases the computational complexity of the watershed segmentation by an order of magnitude. Making the assumption that a comparison between elements is equivalent to an addition operation, the cost of performing the full resolution watershed on an NxN image is: $3N^2$ adds and $2N^2$ multiplies for the ∇ operation, $8(G{\cdot}N)^2$ adds and $9(G{\cdot}N)^2$ multiplies for convolution with a GxG Gaussian, and $10N^2$ adds to perform the WS operation. This means that the full resolution watershed algorithm requires $(13+9G^2)N^2$ addition operations and $(2+9G^2)N^2$ multiplication operations, not including the computational cost of prefiltering. For the multiresolution algorithm, the watershed is applied at a level R that is of size $N/2^R \times N/2^R$, then linked to the finer levels of the pyramid. The cost of constructing the pyramid using an open-close filter with a kernel of size KxK is $4K^2\sum_{L=0}^{R-1}\left(N/2^L\right)^2$ adds. Assuming there are E_R elements in level R which represent watershed boundaries, the watershed must be performed on $4E_R$ elements to link level R to level $R-1$. This linking will produce $E_{R-1} \approx 2E_R$ elements in level $R-1$ since connectivity is maintained. The resulting computational cost of linking the multiresolution watershed is found to be $\left(13+9G^2\right)\cdot 4\sum_{L=0}^{R-1} 2^L E_R$ adds. An example is given in section III.

III. RESULTS

An example of the results from the watershed pyramid is shown in figure 1. Fig. 1(b) shows the resulting edge map from the single resolution watershed algorithm, and Fig. 1(c) shows the resulting edge map using a watershed pyramid. Edge maps from each of the coarser levels of the pyramid are shown in Figs. 1(d)-(f). The multiresolution watershed algorithm yielded an accurate edge detection while filtering out edges corresponding to insignificant detail.

In computing the full resolution watershed, the 256x256 image was prefiltered with a 9x9 open-close filter. The prefilter required $12.8\text{x}10^6$ adds, and the watershed algorithm required $5.4\text{x}10^6$ multiplies and $6.2\text{x}10^6$ adds. In computing the watershed pyramid, a 3x3 open-close filter was used to form the pyramid and the watershed was applied to level 3 (size 32x32). The number of elements representing watershed boundaries in level 3 was 152 (see fig. 1(f)). The computational expense included $3.1\text{x}10^6$ adds to construct the pyramid, $8.5\text{x}10^4$ multiplies and $9.6\text{x}10^4$ adds to perform the watershed segmentation, and approximately $40.0\text{x}10^4$ adds for linking. The total computational cost was $5.4\text{x}10^6$ multiplication operations and $19.0\text{x}10^6$ addition operations for the single resolution algorithm, and $8.5\text{x}10^4$ multiplication operations and $3.2\text{x}10^6$ addition operations for the watershed pyramid approach. This is a decrease in the number of addition operations by a factor of 6, and a decrease in the number of multiplication operations by a factor of 64.

In conclusion, we have shown that the watershed pyramid edge detector is not only more computationally efficient, but it also offers a solution to the problem of over-segmentation. Our linking algorithm allows for the watershed boundary detection to be performed once at a coarse resolution, and the information to be propagated back to the original resolution without sacrificing localization in the edge map. Future research will focus on a robust method of watershed region combination for the pyramidal approach.

IV. REFERENCES

1. J. M. Gauch and S. M. Pizer, "Multiresolution Analysis of Ridges and Valleys in Grey-Scale Images", *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, no. 6, pp. 635-646, 1993.
2. P. T. Jackway, "Gradient Watersheds in Morphological Scale-Space", *IEEE Trans. Image Processing*, vol. 5, no. 6, pp. 913-921, 1996.
3. L. Vincent, "Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations", *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, no. 6, pp. 583-598, 1991.
4. A. Morales, R. Acharya, and S. Ko, "Morphological Pyramids with Alternating Sequential Filters", *IEEE Trans. Image Processing*, vol. 4, no. 7, pp. 965-977, 1996.

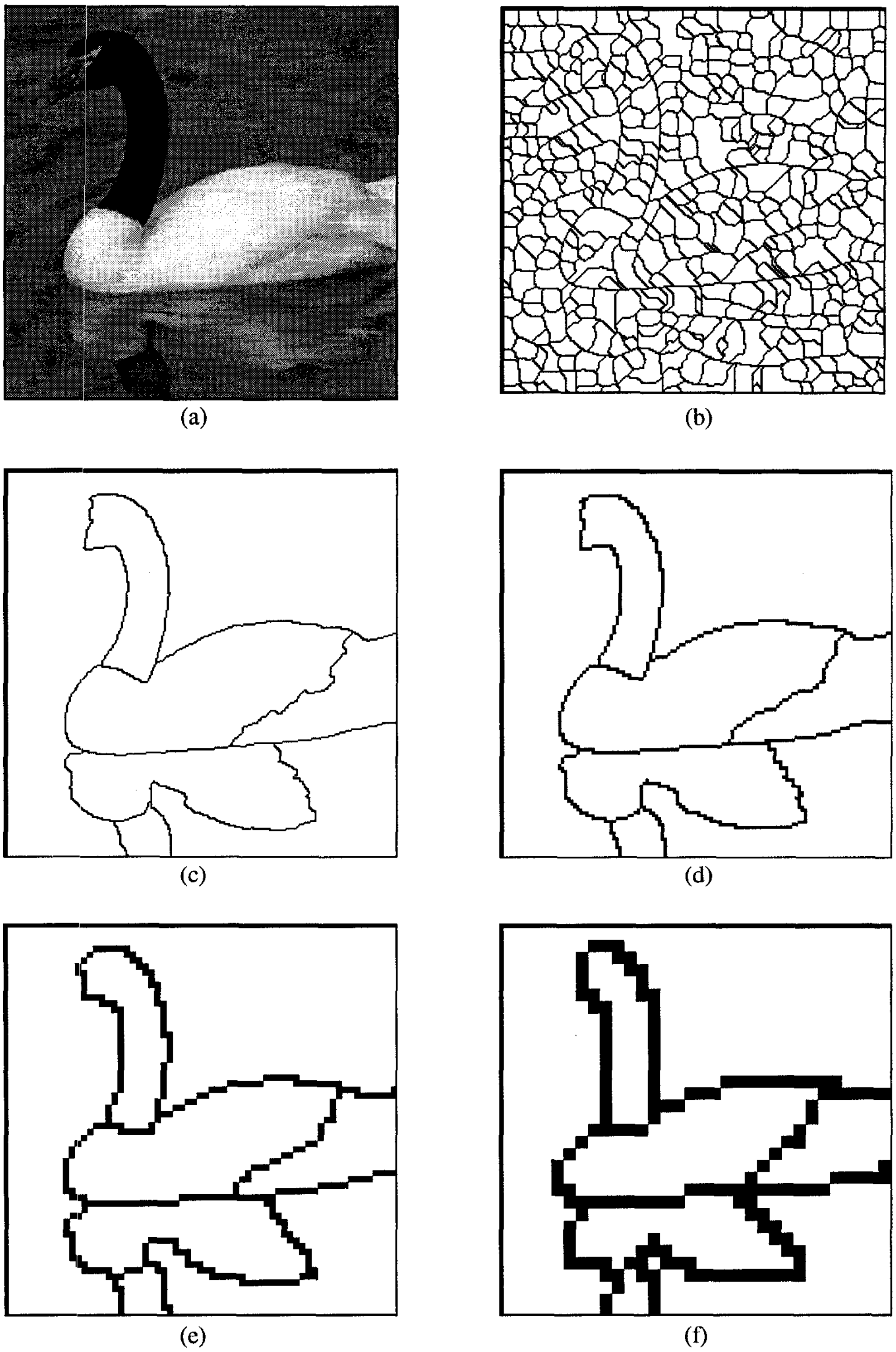

FIGURE 1: (a) Original image – 256x256. (b) Watershed on full resolution image after applying a 9x9 open-close filter. (c) Edge map of the base pyramid level using a 3x3 open-close morphological pyramid. (d) Edge map of the first pyramid level – 128x128. (e) Edge map of the second pyramid level – 64x64. (f) Edge map of the third level of the edge map pyramid – 32x32.