Modified Watershed Transform without Gradient

SRIMAN LINGALA, MALEK ADJOUADI, MICHEL MOURAD, AND NAPHTALI RISHE Center for Advanced Technology and Education Department of Electrical & Computer Engineering, Florida International University 10555 W. Flagler Street, Miami FL 33174 U.S.A.

Abstract:-The watershed transform is the popular method of choice for image segmentation of Region of Interest (ROI) in the field of mathematical morphology. However, like other segmentation methods, it has important drawbacks that include sensitivity to noise (poor detection of low signal to noise ratio structures) and over (under)segmentation if an optimal threshold is not found. In addition, most of the times, the Watershed Transform is applied on the gradient estimation of the raw image which gives typical watershed like topography. Most watershed based segmentation is done with the use of gradient estimation, and very little work has been done addressing problems caused by such estimation. The use of gradient estimation worsens the resolution at the output and also adds noise causing over-segmentation. Recently, a graph based approach called Image Foresting Transform (IFT) was developed to address image partition problems from seed pixels into a shortest path forest problem in a graph, whose solution can be obtained in linear time. The watershed transform is best implemented using the IFT algorithm with priority queue data structure. IFT watershed algorithm gives us an option to introduce the different Monotonically Incremental (MI) Application Specific Lexicographic Path Cost Formulation (ASLPCF) in its wave front propagation; otherwise no such introduction was possible with the Classical watershed transform. The premise of this paper is to implement for the first time a variant of the watershed transform as a shortest path forest problem by introducing modified expression of ASLPCF in its wavefront propagation to overcome the low resolution and over-segmentation problems. This is achieved by enabling the use of the watershed transform directly on the raw image, achieving in this way a higher resolution. A comparison is performed between the results obtained from the proposed algorithm and the classical watershed transform, which demonstrates the accuracy of the algorithm in image segmentation.

Key-Words-Image Foresting Transform, Ordered and Priority Queue Data Structures, Application Specific Lexicographic Path Cost Function, Image Segmentation, Watershed Transform

1 Introduction

In gray scale mathematical morphology the watershed transform, originally proposed and improved by Lantuejoul and Beucher [1] in 1979 is the method of choice for image segmentation. Generally defined, image segmentation is the process of separating objects in the image from the background, i.e., partitioning the image into region disioint regions, such that is homogeneous with respect to some property, such as grey value or texture. Due to the number of advantages that it possesses [2]: it is a simple, intuitive method, it is fast and can be parallelized [3] and it produces a complete division of the image in separated regions even if the contrast is poor, thus avoiding the need for any kind of contour joining.

Some important drawbacks also exist, and they have been widely treated in the related literature. Among the most important are the following: over segmentation, poor detection of significant areas with low contrast boundaries, poor detection of thin structures.

In this paper, a variant of watershed formulation with a modified arc weight is applied for 2D grey scale images. This is based on a particular case of the IFT frame work [6] that reflects the behavior of watershed algorithm using ordered queue. The basis for this paper is to show that watershed transform can be made more useful for myriad applications by incorporating various cost functions within the watershed algorithm [7], [5], [8], [2], depending on the problem without using the post and pre - processing methods [4].

Furthermore, for the first time we also have exploited with various adjacency values or relations within the classical watershed segmentation to overcome severe segmentation, contrary to using homotopy modification methods like image reconstruction, distance transform compliment as preprocessing steps.

2 Methods

2.1 Image Foresting Transform

The Image Foresting Transform is a graph-based approach to the design of image processing operators based on connectivity. The proof of correctness of IFT is derived in [6]. But, extensive research is still going on various applications of IFT.

The IFT defines a minimum-cost path forest in a graph, whose nodes are the image pixels and whose arcs are defined by an adjacency relation between pixels. The cost of a path in this graph is determined by an Application Specific Lexicographic Path Cost *Formulation* (ASLPCF), which usually depends on local image properties along the path such as color, gradient, and pixel position. The roots of the forest are drawn from a given set of seed pixels. For suitable path-cost functions, the IFT assigns one minimum-cost path from the seed set to each pixel, in such a way that the union of those paths is an oriented forest, spanning the whole image.

The IFT outputs three attributes for each pixel. They are its predecessor in the optimum path, the cost of that path, and the corresponding root or some label associated with it. All this attributes and their functionalities are defined in the subsequent section.

2.2 Terminology and Definitions-IFT Graphs Image

An image I is a pair (I, I) consisting of a finite set I of pixels (points in Z^2), and a mapping I that assigns to each pixel t in I a pixel value I(t) in some arbitrary value space.

Directed Graphs

A directed graph is a pair (I, \mathcal{A}) , where *I* is a set of nodes and \mathcal{A} is a set of ordered pairs of nodes. The adjacency relation \mathcal{A} , is a binary relation between pixels of *I*, which is usually translation-invariant. Once \mathcal{A} has been fixed, image I can be interpreted as a directed graph, whose nodes are the image pixels in *I* and whose arcs are defined by \mathcal{A} .

Paths

A path π is a sequence of pixels where

$$\begin{array}{l} \displaystyle \pi = <\!\! t1,\!t2,\ldots,\!tk > \\ \displaystyle (t_{i},\!t_{i+1}) \in A \mbox{ for } 1 \leq i \leq k \text{-} 1 \end{array}$$

This is a graphical definition of path π similar to the definition for the path of *steepest descent* as defined previously in the previous section for watershed transforms. A path is trivial if k = 1.

Path Cost Functions

A path-cost function is mapping that assigns to each path π a cost $f(\pi)$, in some ordered set \mathcal{V} of cost values.

A function f is said monotonic-incremental (MI) when

$$f(\langle \mathsf{t} \rangle) = h(t)$$

$$f(\mathbf{T} \cdot \langle \mathsf{s}, \mathsf{t} \rangle) = f(\mathbf{T}) \quad (s,t)$$

where h(t) is a handicap cost value.

$$\begin{array}{l} x' \geq x \rightarrow x' \quad (s,t) \geq x \quad (s,t) \text{ and} \\ x \quad (s,t) \geq x, \text{ for } x, x' \quad \mathcal{V} \left(s,t\right) \quad \mathcal{A} \end{array}$$

Additive cost function

$$f_{sum} () = h(t)$$

$$f_{sum} (\pi .) = f_{sum} (<\pi>) + w(s,t)$$

$$\begin{aligned} Max - arc \ cost \ function \\ f_{max} \ () = h(t) \\ f_{max} \ (=t>) = max \ \{ \ f_{max} \ (<_{\rm I\!I}>), w(s,t) \ \} \end{aligned}$$

where w(s,t) is a fixed arc weight.

Application Specific Lexicographic Path Cost Function (ASLPCF)

A pixel t is said connected to a pixel s if there is a path from s to t in the graph. The cost of a path in the graph is determined by an *application specific lexicographic path cost function* which usually depends on local image properties along the path, such as brightness, gradient, and pixel position. However, this notion of connectivity can be exploited in many different ways [7], [5], [8], [2], [4] to obtain the desired segmentation watershed transform.

Smooth Path-Cost Function

A cost function $f(\pi)$ is smooth if for any node or pixel t I there is an optimum path π ending at t which either is trivial, or has the form as below:

µ.<s,t>

where the three conditions satisfied are

C1. $f(\mu) \le f(\pi)$ C2. μ is optimum, and C3. for any optimum path μ^* ending at $s, f(\mu^* . \langle s, t \rangle) = f(\pi)$

Predecessor Map and Spanning Forest

A predecessor map is a function P that assigns to each node t I either some other node in I, or a distinctive marker nil not a subset of I in which case t is the root of the map.

A spanning forest is a predecessor map which takes every node to nil in a finite number of iterations (i.e. it contains no cycles).

Paths of the forest P

For any node t I, there is a path $P^*(t)$ which is obtained in backward by following the predecessor nodes along the path.

Optimum-Path Forest

An optimum-path forest is a spanning forest P, where $f(P^*(t))$ is minimum for all nodes $t \ I$.

Plateau Problem:

To solve a plateau problem the image is made lower complete. This can be done by a Linear – Time breadth – first algorithm using a FIFO queue to propagate distances. The algorithm for which is presented below. In the case of ordered algorithms, an alternative to lower completion as preprocessing is to use FIFO ordered queues. Various Tie- breaking policies will be explained in the next section.

Tie – breaking

The optimum-path forest may not be unique, because a pixel may be reached from two or more roots at the same minimum cost. This ambiguity requires tie-breaking policies.

FIFO Policy

Any connected set X of pixels with minimum cost with respect to two or more roots tends to be equally partitioned among the respective trees, as in contrast to the LIFO Policy.

2.3 Image Foresting Transform Algorithm

Based on the above mentioned terms and definitions an algorithm will be presented below i.e. the IFT shortest – path forest using an ordered queue to find the catchment basins of the watershed based flooding procedure. The first shortest path algorithm was due to Moore [4]. This algorithm was is very similar to the well known Dijkstra's shortest-path algorithm and is valid for any path cost using a non-decreasing function of the arc-weights. Watershed transform is best implemented using the IFT algorithm using priority queue based data structure.

An ordered, hierarchical, or priority queue, with a FIFO restriction is a data structure very popular in some morphological image processing algorithms such as gray – scale reconstruction and watershed transform.

1. Initialization

- a) flag(p) = TEMP; p in all nodes
- b) $C(p) = \dot{\alpha}; \quad L(p) = 0; \quad p: non marker nodes$
- c) C(p) = 0; Enqueue(p,0); L(p) = Label of markers; p: marker nodes

2. Propagation

While Queue is not empty

- *d*) v = DeQueueMin
- e) flag(v) = DONE
- f) for each p neighbor of v and flag(p) = TEMP
- g) if $Max\{C(v), w(v,p)\} < C(p)$
- h) $C(P) = Max\{C(V), W(v, p)\}; L(p) = L(v);$
- *i) if p is in queue then Dequeue(p);*
- *j)* Enqueue(p, C(p));

A node p, associated with a priority value v, can be inserted (Enqueue(p,c)) in the ordered FIFO queue. When a node is de-queued (DeQueueMin), it selects the oldest from the lowest priority queue. The following algorithm also needs an operation to remove randomly any node *p* from the queue (DeQueue(p)).An important property of this data structure when used in the IFT algorithm above, is to keep the data implicitly sorted following the lexicographic path cost. The First – In – Fist – Out behavior associated with nature of the IFT algorithm to propagate the lower cost paths first (ordered queue) are responsible for the intrinsic lexicographic sorting. The priority queue is essentially implemented using a binary heap data structure. This algorithm runs in linear time.

The watershed transform is implemented with weights w(p,q) in the above algorithm by substituted by corresponding pixel values g(p). However, in this paper we implemented a variant of watershed transform with a modified arc weight (w(p,q) = |f(p) - f(q))| *i.e* absoulute value of difference between pixel values. By this way attaining a higher resolution at the output, when compared to the results obtained from gradient estimation. In this algorithm, C(p) is the cost path from p to its nearest marker; L(p) is the input marker image and also the result of the watershed partitioning with the catchment basins.

The algorithm works with two set of nodes: temporary (TEMP) and permanent (DONE). Initially all nodes are set as temporary (lin2 1a) and as the algorithm evolves, the nodes are transformed in permanent (line 2b). An important property of this algorithm is that once a node is permanent, its path cost is the final optimum shortest-path. For the sake of simplicity we will call simply by path cost this first lexicographic cost component in the description of the algorithm.

In the initialization phase, all nodes are set as temporary, the markers have their cost assigned to zero and all other nodes have costs assigned to infinity. The marker nodes are labeled and nonmarker nodes have label zero. The propagation step works until there is a temporary node. The node with the minimum temporary cost is selected by removing it from the ordered queue and it is transformed in a permanent node. The temporary nodes p which are neighbors of the new permanent node v are processed. If the path cost computed through the permanent node v is smaller than the temporary cost associated with node p, its cost and label are updated. If the node was already in the queue, it is removed. Finally the node is enqueued with the priority of the new path cost.

There are two important differences between the watershed and the IFT.

- 1) In the watershed, node is labeled when entering into the queue whereas in the IFT the node is permanently labeled only when it leaves the queue and while in the queue, its label can change.
- 2) The priority assigned to a node in the queue is the cumulative path cost in the IFT

algorithm as opposed to the value of the pixel associated with the node, in the watershed.

3 Applications

3.1 Cat Image

Initially we applied classical watershed transform to the gradient of input image (Fig: 1.b), which takes region minima as the seed pixel. As we have more than one minimum inside and outside the ROI, we acquired an over segmented at the output (Fig: 1.c). Connectivity between pixels is taken as either 4 – connected or 8 - connected with Euclidean adjacency values a = 1 and 1.5. Next, we used IFT based watershed algorithm with FIFO queue and arc weight value as maximum pixel intensity along the optimum path on the gradient estimation of the input image. In this method we selected a seed inside the ROI with handicap value = 1 and label = 1 and 2 seeds outside ROI with handicap and label equal to zero. In this result (Fig: 1.d) over segmentation is not witnessed, which shows the strength of the IFT algorithm and the precision with which the ROI is segmented is much better when compared to the classical watershed transform. Next we applied variant of IFT watershed algorithm directly on the input image without gradient calculation with our proposed arc weight modified as absolute difference of pixel gray levels: w(p,q) as |f(p) - w(p,q)| = |f(p) - w(p,q)|f(q) in this way attaining a better resolution (Fig:1.e).

Results in (Fig: 1.g) and (Fig: 1.f) are obtained to show the strength of the algorithm in detecting thin structures to overcome the inability of classical watershed transform in detecting thin structures.

3.2 Dog Image

Similarly, classical watershed transform is applied on the gradient of input (Fig: 2.b) and resulting an over segmented image at the output (Fig: 2.c). Later, we first applied IFT algorithm with arc weights as maximum pixel intensity (Fig: 2.d) with one seed inside the ROI (Dog

body: black in color) with handicap value = 0and label =1 and two seeds outside ROI with handicap value = 0 and label = 0. Connectivity between pixels is taken either 8 – connected with adjacency value equals to 1.5. Next we applied variant of IFT watershed algorithm directly on the input image without gradient calculation with our proposed arc weight modified as absolute difference of pixel gray levels: w(p,q) as |f(p) - w(p,q)| = |f(p) - w(p,q)|f(q) with one seed inside the ROI (Dog body: black in color) with handicap value = 0 and label =1 and one seed outside ROI with handicap value = 0 and label = 0. (Fig: 2.e). Connectivity between pixels is taken either 4 – connected with adjacency value equals to 1. In both the cases the results obtained are segmented properly. The ROI is dog body (black in color with a border watershed line red in color). However, resolution at the output with modified algorithm is better.

4 **Results**

The following illustrations exemplify the results obtained contrasting the gradient method, the classical Watershed Transform method and the modified IFT Watershed Transform.



1.a :Cat – original Image



1.c :Cat Eye: Classical Watershed Segmentation



1.b :Cat – Gradient Estimation



1.d :Cat Eye Modified IFT Watershed Segmentation

Figure 1: Results Using the Scene of a Cat Face





2.a :Dog – Original Image 2.b :Gradient Estimation





2.c: Classical watershed segmentation

2.d:Modified IFT Watershed segmentation

Figure 2: Results Using the Scene of a Walking Dog

6 Conclusion

The results obtained using our modified arc weight method has a better resolution when compared to the results obtained from the gradient estimation. However, it is to be observed that a commonly path cost used in many region-growing algorithms, based on the absolute difference between the mean gray-scale value of the last node region and the mean grayscale of all the previous nodes in the path is not a non-decreasing function. This does not necessarily lead to shortest-path forest problem.

It is evident that the proposed method, which was successfully implemented using optimum path forest, provided better results than the generic region growing algorithm. This could prove useful for applications like video segmentation and medical image segmentation, where high resolution segmentation is desired. Similarly, the modified arc function could be applied to multiple object segmentation (like Differential Image Foresting Transform) and multi-dimensional image segmentation.

References:

- S. Beucher and C. Lanteujoul. Use of watersheds in contour detection. In International workshop on Image Processing, Real-Time Edge and Motion Detection/Estimation, Rennes, France, 1979
- [2] V. Grau, A. U. J. Mewes, M. Alcañiz, R. Kikinis, and S. K. Warfield, "Improved Watershed Transform for Medical Image Segmentation Using Prior Information" IEEE transactions on medical imaging, vol. 23, no. 4, April 2004.
- [3] J. B. T. M. Roerdink and A. Meijster, "The watershed transform: Definitions, algorithms and parallelization strategies," *Fundamenta Informaticae*, vol. 41, pp. 187–228, 2000.
- [4] Thierry Geraud, Isabelle Bloch, Henri Matre, "3D Segmentation of Brain Structures from MR Images using Morphological Approaches" Medical Image Analysis pp1–31, Oxford University Press.
- [5] R. A. Lotufo and A. X. Falcão, "The ordered queue and the optimality of the watershed approaches," in *Mathematical Morphology and Its Applications to Image and Signal Processing*. Norwell, MA: Kluwer, June 2000, vol. 18, pp. 341–350.
- [6] A. X. Falcão, J. Stolfi, and R. A. Lotufo, "The image foresting transform: Theory, algorithms and applications," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 26, pp. 19–29, Jan. 2004.
- [7] F. Meyer, "Color image segmentation," in *Proc.* 4th Int. Conf. Image Processing, 1992, pp. 523– 548.
- [8] J. P. Thiran, V.Warscotte, and B. Macq, "A queue-based region growing algorithm for accurate segmentation of multi-dimensional digital images," *Signal Processing*, vol. 60, pp. 1–10, 1997.