A simpler approach to waterfall

Chris Nicholls, Irina Voiculescu and Stuart Golodetz

Oxford University Computing Laboratory

This project is motivated by the need to *segment* greyscale images originating from computerised tomography (CT) scanners. That is, we would like to fit contours around organ tissue featured in each image. A pair of algorithms called the watershed and waterfall algorithms have been shown to be effective for this purpose [1], although other approaches exist. This paper presents a new, simplified, implementation of the *waterfall* algorithm.

The watershed algorithm, introduced by Beucher and Lantuejoul [2], produces a segmentation of an image, grouping together regions of pixels deemed to be similar. Usually, 'similar' refers to similarity in the greyscale values, though other approaches are possible. A typical problem of the watershed algorithm is that it over-segments images significantly, leading to far more regions than can be handled sensibly, as illustrated in Figure 1.
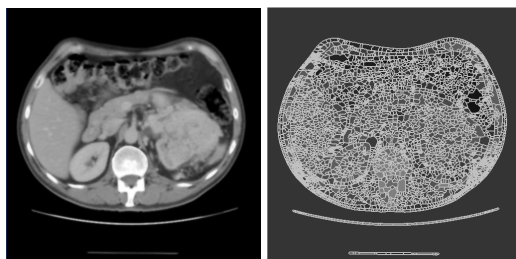

Fig. 1. Example of over-segmentation, output by applying the watershed algorithm to an axial slice of a CT volume. The individual regions are small and do not correspond to any anatomic features.

The waterfall algorithm is an iterative process which can extract further structure from an initial watershed segmentation. The waterfall yields a partition forest hierarchy, which is a comprehensive data structure which can subsequently be used for feature identification. Figure 2 illustrates the various layers that result from applying the waterfall algorithm to the segmentation shown in Figure 1. Each iteration of the algorithm yields a higher-level grouping of the regions in the previous layer.
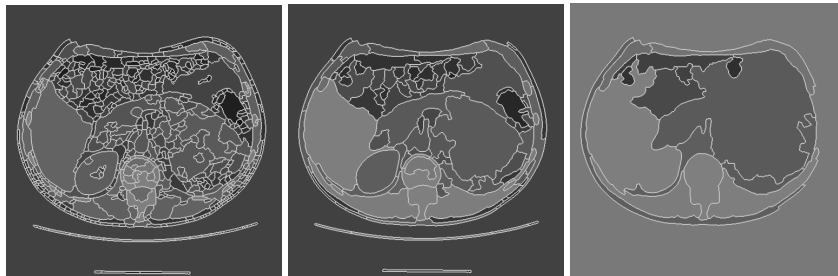

Fig. 2. Hierarchy of segmentations produced by applying the waterfall algorithm to the output of the watershed illustrated in Figure 1, showing regions merging successively (left to right).

Both the watershed and the waterfall algorithms are based on a geographical metaphor. The image is regarded as a landscape, with each grey value at representing the height of the terrain at given (x, y) coordinates. The valleys are in the darker areas, whereas the lighter areas are regarded as peaks.

The waterfall algorithm [3, 4] can then be imagined as a flooding process. The water falls into (low) catchment basins and gradually fills them up to the nearest boundaries, sometimes spilling into adjacent regions. This process continues until the whole image becomes a single basin. The intermediate stages of the process can be regarded as intermediate segmentations of the image, with each basin representing a region.

An implementation of this algorithm, proposed by Marcotegui and Beucher [4], involves the construction of a Minimum Spanning Tree (MST) and the gradual elision of some of its edges. Its nodes are initially the regions of the watershed output and its edges are the lowest pass points on the boundaries between these regions; the nodes and edges in subsequent layers are derived from these initial ones through a merging process.

A regional minimum edge of a graph G is part of a connected subgraph of G whose edges have the same weight as each other, and whose adjacent edges in G have strictly higher weights. The waterfall algorithm relies heavily on finding these regional minimum edges, eliding them and rebuilding the MST – a process which not only requires careful implementation of the MST but, more importantly, is relatively complex and hard to implement.

In this paper we present a new data structure for the waterfall algorithm that simplifies the process and improves efficiency compared to current implementations. It is based on a recursive-tree data structure and a recursive relation on the nodes rather than the conventional iterative transformations.

The main advantage of our approach to the waterfall problem is that the algorithm uses a single loop to walk the MST and is therefore simpler to implement. For each iteration, it walks the MST bottom-up in a single pass and merges regions that belong together. The waterfall algorithm, thus improved, produces the same layers of segmented images, combined in a hierarchical structure that can be processed for feature identification.

A further advantage of our approach is that the algorithm can can be written in pure functional style. In particular, we have implemented it in Haskell. For this reason, the memory requirements are not directly comparable to existing imperative implementations, but we are about to integrate this new approach into an existing C++ code base.

We are also in the process of constructing a formal proof of correctness, which we hope to present at a later date. We have tested both algorithms on a number of small, measurable test cases and found that they produce the same output. Empirical tests indicate that this is also true of larger test cases, such as axial slices of CT volumes.

Production of partition forests in this manner is independent of this application and has many applications outside of the field of medical imaging.

References

1. Stuart Golodetz, Irina Voiculescu and Stephen Cameron. *Region Analysis of Abdominal CT Scans using Image Partition Forests*. In Proceedings of CSTST 2008 pages 432-7. October 2008.
2. S. Beucher, C. Lantuejoul. *Use of watersheds in contour detection*. International Workshop on image processing, real-time edge and motion detection/estimation, Rennes, France, Sept. 1979.
3. Serge Beucher. *Watershed, hierarchical segmentation and waterfall algorithm*. In Mathematical Morphology and its Applications to Image Processing, Proc. ISMM 94, pages 69-76, Fontainebleau, France, 1994. Kluwer Ac. Publ.

4. Beatriz Marcotegui and Serge Beucher, *Fast Implementation of Waterfall Based on Graphs*. In Mathematical Morphology: 40 Years On, Springer Netherlands, 2005.