Parallel Computing 69 (2017) 78–97

Contents lists available at ScienceDirect

Parallel Computing

journal homepage: www.elsevier.com/locate/parco

Concurrent computation of topological watershed on shared memory parallel machines

Ramzi Mahmoudi[a,*], Mohamed Akil[a], Mohamed Hédi Bedoui[b]

[a] Université Paris-Est, Laboratoire d'Informatique Gaspard-Monge, Equipe A3SI ESIEE Paris - Cité Descartes, BP99, 93162 Noisy Le Grand, France
[b] Université Monastir, Laboratoire Technologie et Imagerie Médicale Faculté de Médecine de Monastir, Rue Ibn Sina 5019 Monastir, Tunisia

ARTICLE INFO

Article history:
Received 2 July 2013
Revised 24 August 2017
Accepted 30 August 2017
Available online 1 September 2017

Keywords:
Watershed transform
Parallel computing
Image processing
Parallelization strategy
Computing methodologies
Enhancement
Parallel algorithms
Mathematical morphology

ABSTRACT

The watershed transform is considered as the most appropriate method for image segmentation in the field of mathematical morphology. In the following paper, we present an adapted topological watershed algorithm suited for a rapid and effective implementation on Shared Memory Parallel Machine (SMPM). The introduced algorithm allows a parallel watershed computing while preserving the given topology. No prior minima extraction is needed, nor the use of any sorting step or hierarchical queue. The strategy that guides the parallel watershed computing, labeled SDM-Strategy (equivalent to Split-Distributes and Merge), is also presented. Experimental analyses such as execution time, performance enhancement, cache consumption, efficiency and scalability are also presented and discussed.

1. Introduction

The watershed was extensively studied during the 19th century by J.C. Maxwell [1] and C. Jordan [2] among others. One hundred years later, the watershed transform was introduced by Beucher and Lantuéjoul [3] for image segmentation, and is now used as a fundamental step in many powerful segmentation procedures [4,5]. Fig. 1 gives a very symbolic description of the mentioned approach. In fact, it shows trends that use watershed transform for image processing.

In order to explain the concept of watershed, let us consider a grayscale image as a topographic surface: the gray level of a pixel becomes the elevation of a point, the basins and valleys of the topographic surface correspond to dark areas, whereas the mountains and crest lines correspond to the light areas. If the topographic relief is flooded by water, watersheds will be the divide lines of the domains of attraction of rain falling over the region [6] or sources of water springing from reliefs' peaks. Another synopsis that has shown consistency is when that topographic surface is immersed in a lake with holes pierced in local minima. Catchment basins will fill up with water starting at these local minima, and, at points where water coming from different basins would meet, dams are built. As a result, the topographic surface is partitioned into different basins separated by dams, called watershed lines.

To simulate these approaches, several techniques are deployed. The oldest one starts by finding basins, then watersheds by taking a set of complements. Other techniques use a boundary detection to rebuild watersheds. But the most innovative

---

[*] Corresponding author.
E-mail addresses: mahmoudr@esiee.fr, mramzi@live.fr (R. Mahmoudi), akilm@esiee.fr (M. Akil), medhedi.bedoui@fmm.rnu.tn (M.H. Bedoui).

http://dx.doi.org/10.1016/j.parco.2017.08.010
0167-8191/© 2017 Elsevier B.V. All rights reserved.

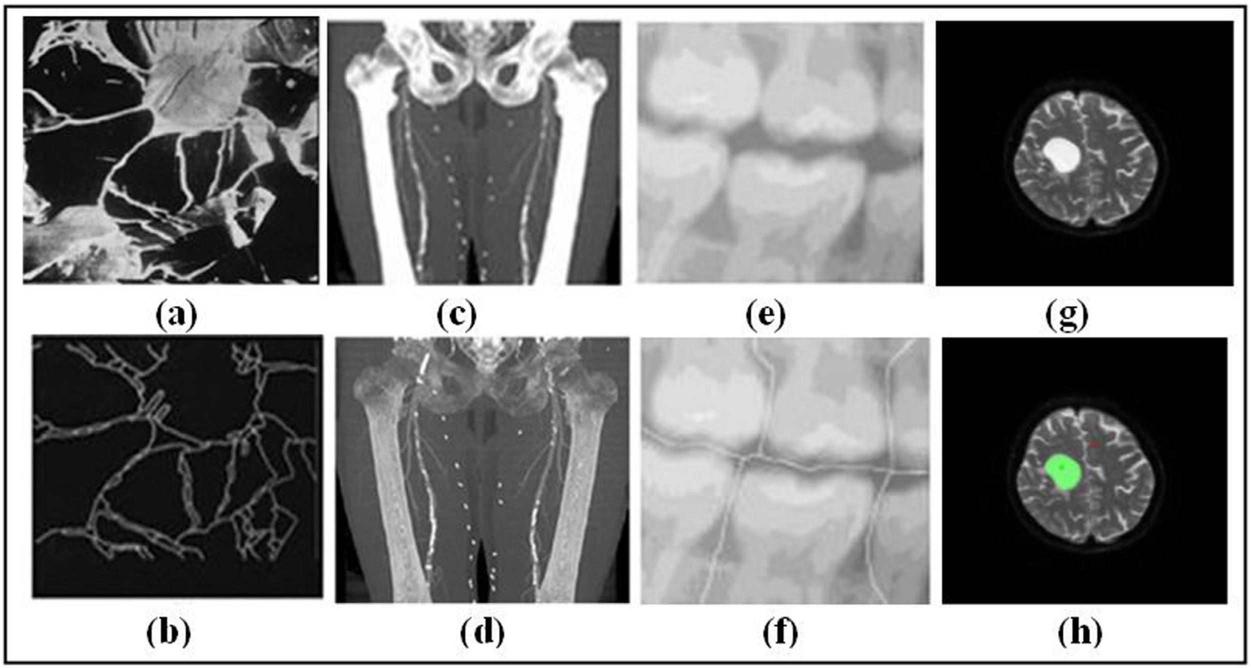

Fig. 1. Watershed applications 1979–2015 *(a)* Cleavage fractures in steel, *(b)* contour of *(a)* the obtained truth watershed definition introduced by Beusher et al. [3] in 1979, *(c)* Maximum intensity projection of the original human lower limb *(d)* Bone tissue removed using mask extended with 3D watershed transform introduced by Straka et al. [12] in 2003, *(e)* original dental X-ray image *(f)* segment line obtained trough watershed transform introduced by Hui Li et al. [13] in 2012, *(g)* Original MRI brain image *(h)* Brain tumor extraction using a marker based Watershed algorithm introduced by Benson et al . [14] in 2015.

technique is proposed by our team [7,8]; it allows to define rigorously the notion of a watershed in a discrete space and to prove important properties that are not guaranteed by most watershed algorithms [9]. This technique consists in lowering the values of the grayscale image - seen as a map - while preserving some topological properties, namely, the number of connected components of each lower cross-section. In this case, the watershed division is the set of points that are not in any regional minimum of the transformed map. It is important to note that our main interest here is for digital images that provide more rigors to define watersheds. Since, there is no unique definition of the path that a drop of water would follow in the discrete case. In general, three large classes of algorithms to compute watershed transform can be figured out. The first one is based on the flooding approach [10], the second is based on the topographical approach [11] and finally a third class is based on the topological approach [7].

An essential difficulty lies in the fact that the watershed transform is not a local concept. The decision whether a pixel belongs to a basin cannot be based on purely local considerations. Some algorithms' results depend also on the order in which pixels are treated during the execution. In the sequential case, this can be resolved by fixing the scanning order, and then a deterministic result is obtained. In a parallel implementation this is no longer true since the outcome depends on the relative time instants at which different processors treat pixels, and this is unpredictable in the case of asynchronous processors. Task becomes even more complicated if one wishes to parallelize algorithms of the topological class. The question is how to preserve the number of connected components of each lower cross-section despite asynchronous nature of the given threads. We must not fail to deal with this problem, which represents the true challenge of this work.

In this paper, we propose an adapted algorithm to compute a watershed. The introduced algorithm that is parallel preserves the topology of the input image. It does not need any prior minima extraction, and does not require any sorting step nor the use of any hierarchical queue. It is also suited for a rapid and effective implementation on Shared Memory Parallel Machine (SMPM).

We also present a tailored parallelization approach, called SD&M (Split Distribute and Merge) strategy that guides the parallel watershed computing. In fact, the splitting step is applied directly on an input graph when selecting sources. Unlike the conventional technique of division such as pixel division, or block division, the source selection is completely random. The associated stream computing is fully parallel (read mode data accesses). Then the distribution depends only on the available processors. This flexibility in data manipulation allowed us to obtain very good results especially in terms of efficiency without using the 'Basic-NPS' scheduler. Finally, the merging step allows the fusion of streams, two by two, to build the watershed.