EFFICIENT IMPLEMENTATION OF MORPHOLOGICAL OPENING AND CLOSING BY RECONSTRUCTION ON MULTI-CORE PARALLEL SYSTEMS

David Valencia and Antonio Plaza

Department of Technology of Computers and Communications
Escuela Politecnica de Cáceres, University of Extremadura
Avda. de la Universidad s/n, E-10071 Cáceres, Spain

E-mail: {davaleco, aplaza}@unex.es

ABSTRACT

We present an efficient parallel implementation of morphological opening and closing by reconstruction operations, which have been used in the past for extracting relevant features prior to classification of remotely sensed hyperspectral images using morphological profiles. The proposed implementation has been developed and tested on various multi-core parallel platforms. These types of multi-processor systems are increasingly being used as a commodity parallel computing platform in different application domains. Our experimental results demonstrate that the proposed parallel codes fully exploit the processing power available in the considered multi-core machines.

Index Terms— Mathematical morphology, morphological reconstruction, parallel processing.

1. INTRODUCTION

Mathematical morphology [1] is a classic nonlinear image processing technique that has been successfully applied to the processing of remotely sensed imagery [2]. Based on set theory, binary morphology was established by introducing fundamental operators applied to two sets [1]. One set is processed by another one having a carefully selected shape and size, and known as the structuring element (SE), which is translated over the image. The SE acts as a probe for extracting or suppressing specific structures of the image objects, checking that each position of the SE fits within the image objects. Morphological operations have extended to grayscale images by viewing these data as an imaginary topographic relief. In practice, set operators directly generalize to gray-tone images.

Morphological filters are characterized by the size and shape of the considered SE. However, if the searched patterns do not have regular properties across the scene, an adaptative scheme is needed to ensure that the correct SE size is considered at each pixel [3]. As reported by various authors [4, 5, 6, 7], this selection can be achieved by plotting the morphological filter output at each pixel against the value of a varying parameter. The resulting plot is called a morphological profile [4], which stems from the concept of *granulometry* [3], and in which the varying parameter is the size of the SE.

Morphological profiles in grayscale imagery are based on opening and closing by reconstruction [8], a special class of morphological filters that have proven to be very successful for multi-scale image processing. These filters do not introduce discontinuities, and therefore preserve the shapes observed in input images. Thus, conventional opening and closing remove the parts of the objects that are smaller than the SE, whereas opening and closing by reconstruction either completely removes the features or retains them as a whole. Although the morphological profile contains information about the spatial content (size, orientation and local contrast), the formulation refers to a single-band image and, therefore, the spectral information contained in multi-band images (such as remotely sensed hyperspectral images) is not considered [9]. A simple approach to deal with this problem is to extract several images that contain parts of the spectral information, and then build the morphological profiles on each of the individual images. In previous work [5, 6], it was suggested that the first principal components (PCs) obtained from the hyperspectral data could be used for this purpose. In this work, we focus on the high computational complexity of morphological reconstruction operations, which often prevents their application to remotely sensed data sets, in particular, when the spatial resolution of such data data sets is very high. In the last few years, the characteristics of massively parallel computers have evolved due to the limitations on the integration level of such systems. Today, multi-core processors have become a common type of parallel system not only available in high performance computing (HPC) platforms, but also in commodity PCs and workstations, which now typically incorporate more than one processor/core for increased processing power via *multi-threaded programming* [10]. Taking advantage of the advent of multi-core processor environments, we develop a parallel version of morphological opening and closing by reconstruction operations which can be easily extended to the case of multicomputers for exploitation on larger-scale parallel platforms such as clusters or heterogeneous networks.

2. MORPHOLOGICAL RECONSTRUCTION

2.1. Basic definitions

In the following, we adopt the notation in [8]. Let us denote an image f as a mapping from a finite rectangular subset D_I of the discrete plane Z^2 into a discrete set $\{1, 2, \cdots, N-1\}$, and the discrete grid $K \subset Z^2 \times Z^2$ provides the neighborhood relationships between pixels: p is a neighbor of q if and only if $(p, q) \in K$, where K is a SE. Similarly, $f(x, y)$ will be defined as a gray-level function or mapping representing an image. Using this terminology, the erosion and dilation of image $f(x, y)$ by SE K can be defined as:

$$\begin{aligned} \varepsilon(x,y) &= (f \otimes K)(x,y) = \inf(f(x,y), K) \\ &= \min_{(s,t)\in K}\{f(x+s, y+t) - k(s,t)\} \end{aligned} \quad (1)$$

$$\begin{aligned} \delta(x,y) &= (f \oplus K)(x,y) = \sup(f(x,y), K) \\ &= \max_{(s,t)\in K}\{f(x-s, y-t) + k(s,t)\} \end{aligned} \quad (2)$$

978-1-4244-4687-2/09/$25.00 ©2009 IEEE

where $k(s,t)$ denotes the weight associated with the different elements of the kernel or SE. The main computational task in dealing with grayscale morphological operations is the query for local maxima or minima in the local search area around each image pixel. This area is determined by the size and shape of the SE denoted by K.

In order to simplify the algorithms developed, we refer only to convex and plain SEs: a special class of SEs that results when $k(s,t) = 0, \forall (s,t) \in K$. This is not a general requirement of morphological operations, and other types of SEs can be used in the future developments of the method. For illustrative purposes, let K be a plain 3×3 SE. If a dilation operation using K is applied over a grayscale image, then the local effect of the operation is the selection of the brightest pixel in a 3×3-pixel search area around the target pixel. The constraints imposed on the kernel definition causes all pixels lying within the kernel to be handled equally, i.e., no weight is associated with the pixels according to their position along the kernel, and the pixel with maximum digital value is selected. The previous operation is repeated with all the pixels of the image, leading to a new image (with the same dimensions as the original).

2.2. Reconstruction operations

Let us assume that g and f are two grayscale images defined on the same domain, taking their values in the discrete set $\{1, 2, \cdots, N-1\}$ and such that $g \leq f$. With the above definition in mind, one can define the elementary erosion $\varepsilon_f^{(1)}(g)(x,y)$ of grayscale image $g \leq f$ as follows:

$$\varepsilon_f^{(1)}(g)(x,y) = (g \otimes K) \vee f, \tag{3}$$

where $\vee$ stands for the pointwise maximum and $(g \otimes K)$ is the erosion of g by the flat SE denoted by K. With this in mind, one can further define the grayscale erosion of size $n \geq 0$ as:

$$\varepsilon_f^{(n)}(g)(x,y) = \underbrace{\varepsilon_f^{(1)} \circ \varepsilon_f^{(1)} \circ \cdots \circ \varepsilon_f^{(n)}(g)(x,y)}_{\text{n times}}, \tag{4}$$

which leads to:

$$\rho_f^*(g)(x,y) = \bigwedge_{n \geq 1} \varepsilon_f^{(n)}(g)(x,y) \tag{5}$$

The expression above implies that the reconstruction by erosion of f from g is obtained by iterating grayscale erosions of g "above" f until stability is reached. Similarly, we can define the reconstruction by dilation $\delta_f^{(1)}(g)(x,y)$ of grayscale image $g \leq f$ as follows:

$$\delta_f^{(1)}(g)(x,y) = (g \oplus K) \wedge f, \tag{6}$$

where $\wedge$ stands for the pointwise minimum and $(g \oplus K)$ is the dilation of g by the flat SE denoted by K. Thus, and similarly to the definition of grayscale erosion of size $n \geq 0$, the grayscale dilation of size $n \geq 0$ is given by:

$$\delta_f^{(n)}(g)(x,y) = \underbrace{\delta_f^{(1)} \circ \delta_f^{(1)} \circ \cdots \circ \delta_f^{(n)}(g)(x,y)}_{\text{n times}}, \tag{7}$$

which leads to:

$$\rho_f(g)(x,y) = \bigvee_{n \geq 1} \delta_f^{(n)}(g)(x,y) \tag{8}$$

The expression above implies that the reconstruction by dilation of f from g is obtained by iterating grayscale dilations of g "under" f until stability is reached.

In both erosion and dilation by reconstruction, the starting image for reconstruction resulting from a morphological operation, i.e. erosion or dilation, is called the ***marker*** image, while the image used to constrain the spread of the flood-filling only to its foreground pixels is called the ***mask*** image.

Table 1. Pseudo-code of the parallel algorithm executed by each *thread* in the proposed parallel multi-core implementation.

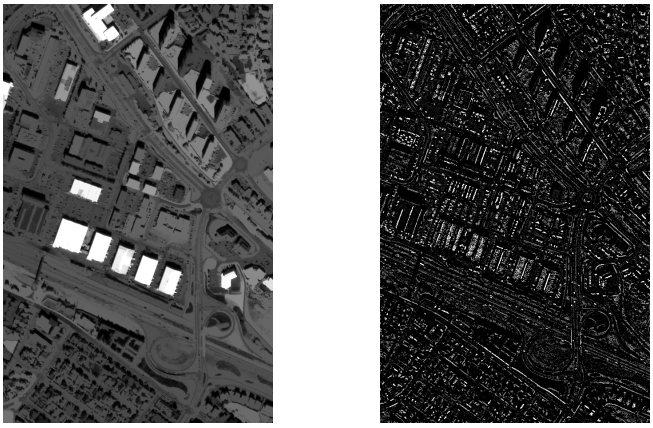
Input: Remotely sensed image $f(x,y)$; maximum number of iterations it; size of SE denoted by K.
Output: Reconstructed image $o(x,y)$.
Img$_{ero} \leftarrow$ Erosion ($\downarrow f(x,y)$,$\downarrow K$);
Changes$_{old} \leftarrow$ -1;
Changes$_{act} \leftarrow$ 0;
Img$_{ope}(0) \leftarrow$ Img$_{ero}$;
iter $\leftarrow$ 0;
while *iter* $<$ *it* **do**
  Img$_{ope}(iter+1) \leftarrow$ Opening_reconstruction($\downarrow$Img$_{ope}(iter)$,$\downarrow$ $f(x,y)$,$\uparrow$Changes$_{act}$);
  if *Changes*$_{act}$ = *Changes*$_{old} \vee$ *iter* = *it* **then**
    End of the execution;
  else
    Changes$_{old} \leftarrow$ Changes$_{act}$;
    iter $\leftarrow$ iter + 1;
  end
  Return of Img$_{ope}$;
end

3. PARALLEL ALGORITHM

In this section we describe the parallel algorithm used for efficient implementation of morphological opening and closing by reconstruction operations. From now on, we focus on the opening by reconstruction operation (closing by reconstruction is exactly the same, but interchanging the erosion operations by dilation operations and *vice-versa*). In general terms, the behavior of the parallel algorithm can be summarized as follows: first, we apply a morphological erosion operation on the original image and then reconstruct the image using as mask the original image and as marker the eroded image. Since the operations are performed on grayscale images, we can establish an absolute distance measure which assures that the implemented method converges in a given number of steps.

Table 1 shows a pseudo-code of the algorithm executed by each *thread* (program running in each core), where each core processes a local partition of the image. It is important to emphasize that the code in Table 1 does not need any communication primitives since the multi-core platform is assumed to be a shared memory architecture. This feature considerably simplifies the design of a parallel multi-core algorithm, as shown by Table 1. In the table, the input and output data are designated by $\downarrow$ and $\uparrow$, respectively. As shown by Table 1, the parallel code has two stopping conditions:

1. The first and most important stopping rule is given by the comparison between changes from the previous iteration to the actual one. This comparison allows us to assess the convergence of the method.
2. A second stopping rule is given by the number of iterations, which is introduced as an input parameter to allow stopping the parallel after a certain number of iterations in the case that we are not using a distance measure that assures convergence of the method.



(a) (b)

Fig. 2. (a) Result of a morphological reconstruction of the Ikonos image using a SE of 15×15 pixels in size. (b) Differences between the original Ikonos image and the reconstructed version of the same image using a SE of 3×3 pixels in size.

4. EXPERIMENTAL RESULTS

In order to illustrate the proposed parallel implementations, we have used a high-resolution panchromatic image obtained by Ikonos over the city of Reykjavik in Iceland (see Fig. 1). This scene is particularly suitable for our experiments thanks to the variability on the size and orientation of the buildings and other urban features. Although experiments with remotely sensed hyperspectral images would also have been possible (e.g. by applying the proposed algorithm to the first principal component obtained after processing a hyperspectral image) we have decided to use a scene with very high spatial resolution in order to better substantiate our findings. Table 2 shows the multi-core systems used in this study to evaluate the performance of the parallel implementation. As shown by Table 2, the considered systems are quite different in nature and architecture, which allows for a detailed study of issues related to the architecture of the parallel systems which can ultimately affect parallel performance. Fig. 2

Fig. 1. Ikonos image of Reykjavik, Iceland.

shows the result of applying a parallel reconstruction operation with a kernel or SE with 15×15 pixels in size, along with the result of selecting the pixels with different value between the original image and the result of a reconstruction of the same image with a SE

Table 2. Description of multi-core systems used.

System	Type of Processor	Number of cores
Escher	Intel Itanium II	16
Frida	AMD Opteron Quad Core (uma)	16
Warhol	Intel Quad Core	8
Gaudi	AMD Opteron Quad Core (numa)	32

with 3×3 pixels in size. As expected, the parallel reconstruction operations behaved in exactly the same way as their corresponding sequential operations. On the other hand, Figs. 3-6 summarize the parallel performance of the proposed implementation by reporting the execution times measured after applying different kernel sizes with increasing number of cores. First of all, we evaluate the results on the Escher machine (Itanium II), reported on Fig. 3. As can be seen from this figure, a reduction of the execution time is generally observed as the number of cores is increased, but the reduction is not as significant as it could be expected. Also, we can observe a local minimum around 8 cores, probably related to the optimal division of the original image, which is around 8 partitions (i.e. the size of the resulting partitions from this size fit into the cache memory of the system). From this value, it is observed in Fig. 3 that the execution time grows due to the small size of the problem in this particular division, which is also affected by the overhead introduced by the control code. Also, we can see in Fig. 3 that, for 2 cores, the execution time is greater than that of 1 processor only, probably due to the bad performance of the libraries for the threads on the Itanium architecture and a poor selection of compiler flags.

On the other hand, the results obtained in the Warhol machine (Intel Quad Core with 8 cores) show much better performance, with an almost linear decrease in execution times as the number of cores is increased, in accordance to what is expected in an optimal parallel implementation. Interestingly, with SEs of size 7×7 and 9×9 pixels, the response times are higher than those measured with SEs of size 11×11, 13×13 and 15×15 pixels, which is probably due to the nature of the algorithm and complexity of the data (morphological reconstruction is a highly non-linear problem). Finally, the results obtained for the AMD Opteron machines (Figs. 5 and 6) are the ones showing the best parallel performance, with a more linear decreasing slope in the reduction of execution times as the number

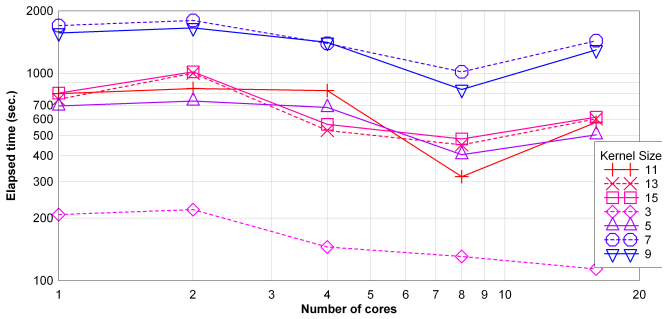


Fig. 3. Execution times on Itanium II (16 Cores).

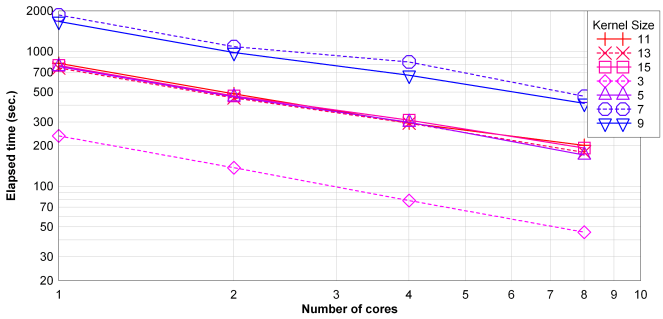


Fig. 4. Execution times on Intel Quad (8 Cores).

of cores is increased, resulting in very low processing times for the reconstruction operations on the considered scene.

5. CONCLUSIONS

We have developed a parallel implementation of morphological opening and closing by reconstruction operations of reconstruction that can be used for the processing of remotely sensed images with high spatial resolution. The proposed parallel implementation can also be used to speed up the construction of morphological profiles (using the first few principal components) for hyperspectral image processing. Our experimental results with a high spatial resolution image are promising, and reveal the benefits that can be obtained after implementing this type of morphological operations in multi-

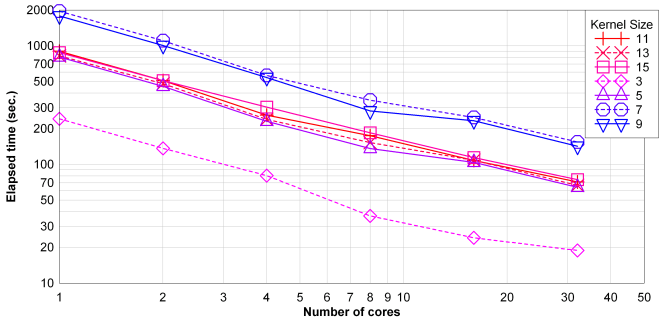


Fig. 5. Execution times on AMD Opteron numa (32 Cores).

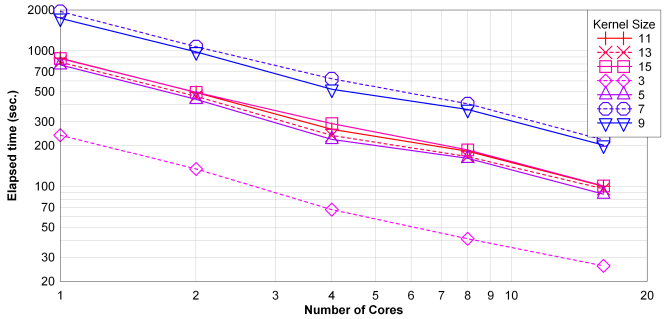


Fig. 6. Execution time on AMD Opteron uma (16 Cores).

core systems, which reduce the problems related with memory restrictions observed in other types of parallel implementations (e.g. cluster computers or distributed networks). Further experiments with additional scenes should be conducted in order to fully validate the preliminary findings reported in this work.

6. REFERENCES

[1] J. Serra, *Image Analysis and Mathematical Morphology*, Academic Press, Inc. Orlando, FL, USA, 1983.

[2] P. Soille and M. Pesaresi, "Advances in mathematical morphology applied to geoscience and remote sensing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 40, pp. 2042–2055, 2002.

[3] P. Soille, *Morphological Image Analysis: Principles and Applications*, Springer-Verlag New York, Inc. Secaucus, NJ, USA, 2003.

[4] M. Pesaresi and J. A. Benediktsson, "A new approach for the morphological segmentation of high resolution satellite imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, pp. 309–320, 2001.

[5] J. A. Benediktsson, J. A. Palmason, and J. R. Sveinsson, "Classification of hyperspectral data from urban areas based on extended morphological profiles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, pp. 480–491, 2005.

[6] M. Fauvel, J. A. Benediktsson, J. Chanussot, and J. R. Sveinsson, "Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, pp. 608–619, 2008.

[7] J. Chanussot, J. A. Benediktsson, and M. Fauvel, "Classification of remote sensing images from urban areas using a fuzzy possibilistic model," *IEEE Geoscience and Remote Sensing Letters*, vol. 3, pp. 40–44, 2006.

[8] L. Vincent, "Morphological grayscale reconstruction in image analysis: applications and efficient algorithms," *IEEE Transactions on Image Processing*, vol. 2, no. 2, pp. 176–201, 1993.

[9] A. Plaza, P. Martinez, R. Perez, and J. Plaza, "Spatial/spectral endmember extraction by multidimensional morphological operations," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 40, pp. 2025–2041, 2002.

[10] B. Lewis and D.J. Berg, *Multithreaded programming with Pthreads*, Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 1998.