Available online at www.sciencedirect.com
SCIENCE DIRECT
Image and Vision Computing 25 (2007) 24–33
image AND vision COMPUTING
www.elsevier.com/locate/imavis

Combining wavelets and watersheds for robust multiscale image segmentation

Cláudio Rosito Jung *

UNISINOS—Universidade do Vale do Rio dos Sinos, PIPCA - Graduate School on Applied Computing, Av. UNISINOS, 950. CEP 93022-000, São Leopoldo, RS, Brazil

Received 11 May 2004; received in revised form 2 August 2005; accepted 7 January 2006

Abstract

This paper proposes a new segmentation technique that combines multiresolution wavelet decompositions with the watershed transform. The wavelet transform is applied to the intensity image, producing detail and approximation coefficients. Gradient magnitudes of the approximation image at the coarsest resolution are computed, and an adaptive threshold is used to remove small gradient magnitudes. The watershed transform is then applied, and the segmented image is projected up to higher resolutions using inverse wavelet transforms. Typically, if a low resolution is chosen for the initial segmentation, large relevant objects will be captured; on the other hand, a higher initial resolution will lead to smaller (and more detailed) segmented objects. The low-pass filtering involved in the wavelet decomposition provides robust segmentation results for noisy images, even when the amount of noise is very large.

Keywords: Segmentation; Watersheds; Wavelets; Multiresolution; Denoising; Region merging

1. Introduction

Image segmentation is a fundamental problem in image analysis. It provides a partitioning of the image, where each region should represent a different object. With this information, it is possible to accomplish quantitative object measurements, such as size, shape and position, with many practical applications (for example, measurement of the size and shape of tumors in medical images).

A powerful tool for image processing based on mathematical morphology is the watershed transform [1,2]. For example, images can be segmented into visually sensible regions by finding the watershed regions in a gradient magnitude image [1]. However, small fluctuations in the image gray values (usually due to noise) produce spurious gradients, which cause oversegmentation. To overcome this problem, many techniques based on watersheds have been proposed [3–7]. For example, Meyer [6] introduced the *levelings* approach, which consists of applying morphological filters to reduce small details in the image. Haris et al. [3] proposed an edge-preserving statistical noise reduction approach as a pre-processing for the watershed transform, and a hierarchical merging process as a post-processing stage. Weickert [7] proposed partial differential equations for image denoising or edge enhancement (as pre-processing), combined with watersheds segmentation and region merging. Nguyen and collaborators [8] approached the problem of watershed segmentation as the minimization of an energy functional, integrating smoothness constraints of *snakes* [9] with the watershed transform.

A particular class of segmentation techniques relies on multiscale representations of images. Such methods are typically based on image transformations that change image resolution, being able to segment objects of different sizes. In general, small details are detected in higher resolution images, while larger objects are segmented in coarser images.

The multiscale behavior of image features have been analyzed in different ways. Tracking of intensity extrema along scales [10] and multiscale behavior of graylevel blobs (defined relative to intensity extrema) [11] were used for image segmentation. Since edge information was not explicitly included, the performance of both techniques had deficiencies. Jackway [12], Gauch [13] and Mukhopadhyay and Chanda [14] used morphological scale-spaces for image segmentation. Vanhamel et al. [15] explored multiscale image representations (based on diffusion schemes) and watersheds, focused on color

* Tel.: +55 51 3591 1122 ext.1626; fax: +55 51 3590 8162
E-mail address: crjung@unisinos.br

0262-8856/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2006.01.002

image segmentation. Kim and Kim [16,17] proposed a multiresolution wavelet-based watershed image segmentation technique, using markers and a region merging procedure to reduce oversegmentation. Jung and Scharcanski [18] used a combined image denoising/enhancement technique based on a redundant wavelet transform for multiscale image segmentation. Ma and Manjunath [19,20] proposed the *Edge Flow* segmentation technique, which consists of computing and updating changes in color and texture in a pre-defined scale. Deng and Manjunath [21] proposed the *JSEG* method for multiscale segmentation of color and texture, based on color quantization and region growing. Wy et al. [22] proposed a multiscale wavelet-based directional image force as external forces for *snakes* segmentation. Comaniciu and Meer [23] used a kernel in the joint spatial-range domain to filter image pixels and a clustering method to retrieve segmented regions.

It should be noticed that several others segmentation techniques have been proposed in the past years, such as the graph-based approach in [24], and statistical and fuzzy techniques designed to reduce oversegmentation produced by watersheds [25,26].

In this paper, a new multiscale segmentation technique based on wavelets and watersheds is presented. The first step of this technique is to describe an image in multiple resolutions using an orthogonal wavelet decomposition. A certain resolution 2^J is chosen, and gradient magnitudes at that resolution are estimated by applying the Prewitt edge detector [27]. An adaptive threshold is used to remove small magnitudes, and the watershed transform is applied. This initial segmentation is projected to higher resolutions using the inverse wavelet transform, until the full resolution segmented image is obtained. A variation of the proposed technique relies on region merging to further reduce oversegmentation produced by watersheds. The proposed technique is particularly efficient for segmenting noisy images, as it will be discussed in Section 6.

The remaining structure of this paper is arranged as follows. In Section 2, a brief description of the wavelet transform is given. Section 3 presents an adaptive threshold and the initial segmentation at the coarsest resolution using watersheds. Section 4 describes the projection of the initial segmentation to the full resolution image. An alternative to further reduce oversegmentation by region merging is given in Section 5. Section 6 contains several experimental results of the proposed technique, and the conclusions are presented in Section 7.

2. The wavelet transform

The wavelet transform is a mathematical tool that can be used to describe images in multiple resolutions. The wavelet decomposition is a complete representation, since it allows a perfect reconstruction of the original image [28]. Also, since a low-pass filter is involved, noise suppression is inherent to this transform.

According to Mallat's pyramid algorithm [28], the input image is convolved with low-pass and high-pass filters associated with a *mother wavelet*, and downsampled afterwards. Four images (each one with half the size of the original image) are produced, corresponding to high frequencies in the horizontal direction and low frequencies in the vertical direction (HL), low frequencies in the horizontal direction and high frequencies in the vertical direction (LH), high frequencies in both directions (HH) and low frequencies in both directions (LL). This last image is a low-pass version of the original image, and will be called the approximation image. This procedure is repeated for the approximation image at each resolution 2^j (please note that dyadic scales are used). The four images HL, LH, HH and LL are denoted, respectively, by $W^h_{2^j}$, $W^v_{2^j}$ $W^d_{2^j}$ and A_{2^j}. If the wavelet transform is applied up to the scale 2^J, the original image can be reconstructed using images A_{2^J} and $\{W^h_{2^j}, W^v_{2^j}, W^d_{2^j}\}_{\{j=1,2,\ldots,J\}}$.

In this paper, the Haar wavelet [29] was chosen because of its orthogonality and, more important, its small support (the importance of using small support basis will become clear in Section 4.2). Also, it requires small computational complexity (linear with respect to the size of the input image) to compute the wavelet decomposition with the Haar wavelet. The expressions for the low-pass $h[n]$ and high-pass $g[n]$ filters for the Haar wavelets are provided in Eq. (1).

$$h[n] = \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right], \qquad g[n] = \left[\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right] \tag{1}$$

3. Image segmentation at the coarsest resolution

The first stage of the proposed segmentation method is to choose the initial resolution 2^J, and compute the wavelet representation up to that resolution. The Prewitt edge detector is applied to the coarsest resolution image A_{2^J}, and the gradient magnitude image M is obtained. Some magnitudes of M are related to actual edges, but others are related to noise (or small fluctuations of the grayvalues). Even though noise was reduced when the approximation image was obtained (because of low-pass filtering), some residual noise still remains. To remove these small magnitudes (and reduce oversegmentation produced by the watershed transform), an adaptive threshold is applied to image M.

3.1. Adaptive thresholding and watershed segmentation

Scharcanski et al. [30] used wavelet shrinkage for image denoising, based on a redundant wavelet transform. The shrinkage function was a posterior probability based on gradient magnitudes, that should return '1' at edge positions and '0' in homogeneous (and/or noisy) regions. In this work, such shrinkage function was adapted for magnitude thresholding using the Prewitt operator, and is described next.

If the original image is constituted solely of additive Gaussian noise, the corresponding detail wavelet coefficients W^{h} and W^{v} at a certain resolution can be considered Gaussian [31], with standard deviation σ_{noise}. As a consequence, the magnitude image (estimated using detail coefficients of the wavelet transform) $M = \sqrt{(W^{\mathrm{h}})^2 + (W^{\mathrm{v}})^2}$ will be a Rayleigh process [32] with probability density function given by:

$$p(r|\text{noise}) = \frac{r}{[\sigma_{\text{noise}}]^2} e^{-r^2/2[\sigma_{\text{noise}}]^2}. \quad (2)$$

On the other hand, noise-free images typically consist of homogeneous regions and not many edges. In general, homogeneous regions contribute with a sharp peak around zero for the histograms of W^h and W^v, and the edges contribute to the tail of the distribution. This distribution presents a sharper peak than a Gaussian [33], and therefore, the Gaussian model is not the most appropriate model. However, the distribution of wavelet coefficients W^v and W^h related exclusively to edges (and not related to homogeneous regions) can be approximated by a Gaussian function [30]. The normal model for edge-related coefficients is assumed because it leads to a simple model (Rayleigh) to approximate the corresponding edge-related gradient magnitudes. Therefore, edge-related magnitudes are also approximated by a Rayleigh process:

$$p(r|\text{edge}) = \frac{r}{[\sigma_{\text{edge}}]^2} e^{-r^2/2[\sigma_{\text{edge}}]^2}. \quad (3)$$

The overall gradient magnitude distribution can be expressed as a mixture of two Rayleigh functions: one representing noise (and also fairly homogeneous regions), and the other one representing actual edges. Hence,

$$p(r) = w_{\text{noise}} p(r|\text{noise}) + (1 - w_{\text{noise}}) p(r|\text{edge}). \quad (4)$$

where w_{noise} is the *a priori* probability for the noise-related gradient magnitude distribution (and, consequently, $1 - w_{\text{noise}}$ is the a priori probability for edge-related gradient magnitudes).

Parameters σ_{noise}, σ_{edge} and w_{noise} can be estimated through a maximum likelihood approach [30], and the posterior probability function $p(\text{edge}|r)$ is calculated using Bayes theorem as follows:

$$p(\text{edge}|r) = \frac{(1 - w_{\text{noise}}) p(r|\text{edge})}{(1 - w_{\text{noise}}) p(r|\text{edge}) + w_{\text{noise}} p(r|\text{noise})}. \quad (5)$$

Given a coefficient with gradient magnitude r, the value $p(\text{edge}|r)$ represents the likelihood of such coefficient being related to an actual edge (i.e. it should be close to one near edges, and close to zero in homogeneous and/or noisy regions).

In this work, a normalized threshold P (i.e. $0 \le P \le 1$) is defined such that all magnitudes r satisfying $p(\text{edge}|r) < P$ are set to zero, and the remaining magnitudes are multiplied by the shrinkage function $p(\text{edge}|r)$. Then, for each pixel $[n, m]$, the thresholded magnitude image M_{thresh} is given by:

$$M_{\text{thresh}}[n,m] = \begin{cases} 0, & \text{if } p(\text{edge}|M[n,m] < P \\ M[n,m] p(\text{edge}|M[n,m]), & \text{if } p(\text{edge}|M[n,m] \ge P \end{cases}. \quad (6)$$

This means that only magnitudes with likelihood of being edges greater than P will be kept after the thresholding process. Small values of P result in less noise removal (hence, more segmented regions). As P gets larger, more magnitudes are removed, and less regions are retrieved (however, some small contrast edges may be removed). A suggested default value is $P = 0.5$, so that magnitudes with probability less than 50% of being actual edges are removed, in accordance with the adaptive thresholding procedure adopted by Henstock and Chelberg [34].

(a) (b)

Fig. 1. (a) Original *peppers* image. (b) Noisy *peppers* image (PSNR = 16.58 dB).

In [30], horizontal (W^h) and vertical (W^v) details of a redundant wavelet transform were used to estimate gradient magnitudes. Such redundant wavelet transform was based on a mother wavelet similar to the derivative of a Gaussian, which is very efficient for edge detection. In this work, a decimated wavelet transform based on Haar wavelets is employed, and it produces three detail images per scale (horizontal, vertical and diagonal). These detail images are not very adequate for edge detection, since the small support of Haar wavelets tends to produce broken contours. Hence, instead of using wavelet coefficients for estimating gradient mangitudes, we chose to use horizontal and vertical image differences based on the Prewitt operator. It should be noticed that Gaussian assumptions for horizontal and vertical differences of convolution-based edge detectors (such as Prewitt) are acceptable for edges and nonedges [34]. Hence, the mixture of Rayleigh functions provided in Eq. (4) can also be used for gradient magnitudes computed through the Prewitt operator.

Finally, the watershed transform of M_{thresh} is computed, and the initial segmentation at the coarsest resolution is obtained. The adopted gradient magnitude thresholding scheme and the initial watershed segmentation for a 'clean' image and its noisy version are illustrated in the next figures. Fig. 1(a) shows the *peppers* image (256×256 pixels), and Fig. 1(b) shows its noisy version (PSNR[1] = 16.58 dB). Fig. 2(a) and (e) shows the respective approximation images A_{2^2} (with 64×64 pixels), corresponding to the resolution 2^2, and Fig. 2(b) and (f) shows the respective thresholded magnitude images (using $P = 0.5$). The corresponding shrinkage functions $p(\text{edge}|r)$ are shown in Fig. 2(c) and (g). As expected, the adaptive threshold is higher for the noisy image (in noisy images, gradients related with noise have larger magnitudes). Finally, the initial watershed segmentation at resolution 2^2 is shown in Fig. 2(d) and (h). For

[1] The peak-to-peak signal-to-noise ratio is defined as $\text{PSNR} = -20 \log_{10}\left(\frac{255}{\sigma_{\text{noise}}}\right)$, where σ_{noise} is the standard deviation of the noise corrupting the image.

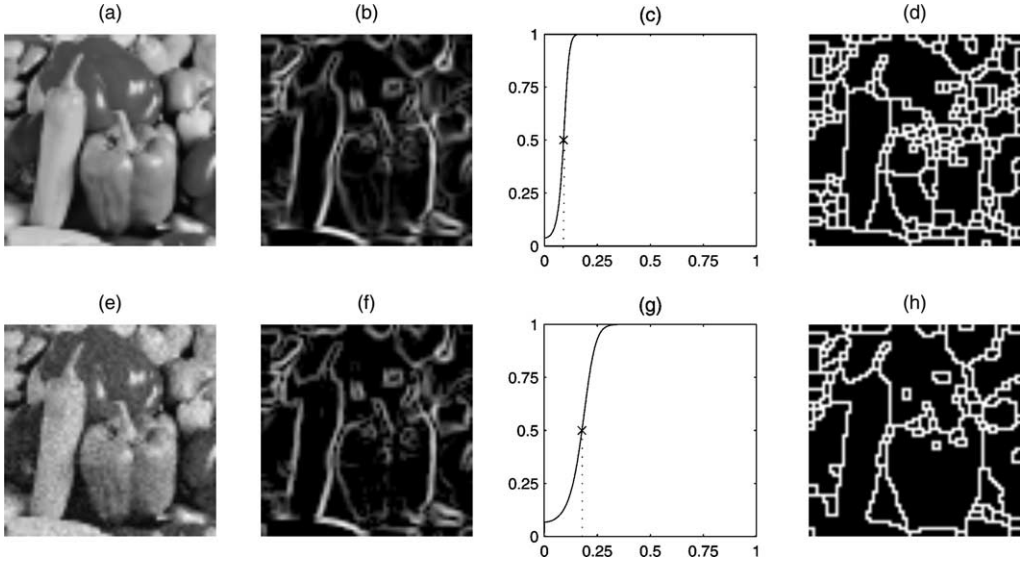

Fig. 2. (a) Approximation coefficient A_{2^2} related to the original *peppers* image, (b) thresholded magnitude image, (c) the corresponding function $p(\text{edge}|r)$ used to determine the threshold, (d) initial watershed segmentation. (e) Approximation coefficient A_{2^2} for the noisy *peppers* image, (f) thresholded magnitude image, (g) the corresponding function $p(\text{edge}|r)$ used to determine the threshold, (h) initial watershed segmentation.

the original *peppers* image, 150 regions were segmented; for the noisy version, 80 regions were obtained (it can be noticed that texture of the objects was partially destroyed by noise corruption, leading to less segmented regions).

4. Projection of the segmented image to higher resolutions

The result of applying the watershed transform to the edge map of approximation image A_{2^J} is a segmentation at resolution 2^J, with each segmented region having a different label. It must be noticed that the segmented image has about 2^{-J} of the size of the original image, due to the downsampling used in the wavelet transform. Direct projection of the simplified image by pixel duplication (upsampling) in both horizontal and vertical directions offers very poor results (producing a blocking effect), as noticed by Kim and Kim [17]. This effect happens because high frequencies that allow edge definition are contained in the detail coefficients $W^h_{2^j}$, $W^v_{2^j}$ and $W^d_{2^j}$, which are not taken into consideration in the pixel duplication process, but are considered when computing the inverse wavelet transform (IWT). In [17], the IWT is computed to project the initial segmentation to finer scales, and a combination of watershed transforms and connection of segmented regions is applied at each new projected scale. In this work, a simple adjustment of object boundaries is performed at each projected scale through the IWT, as explained next.

4.1. Computing the simplified image

The first step to project the initial segmentation to finer resolutions is to obtain a simplified version S_{2^J} of the approximation image A_{2^J}. Such image S_{2^J} is obtained by replacing each label of the initial watershed segmentation by the average grayvalue of the corresponding region in image A_{2^J} (at this point, S_{2^J} is piece-wise constant, separated by watershed lines). Then, watershed lines are removed by setting $S_{2^J} = A_{2^J}$ at all regions borders (i.e. the watershed lines are replaced by the corresponding grayvalues of the approximation image). Fig. 3 illustrates this procedure. The simplified images S_{2^2} for the original and noisy *peppers* images at the resolution 2^2 are shown in Fig. 3(a) and (b), respectively.

In order to project S_{2^J} to resolution 2^{J-1}, detail coefficients $W^h_{2^J}$, $W^v_{2^J}$ and $W^d_{2^J}$ are needed. However, only some detail coefficients (those related to region borders) are used to prevent noise from being introduced back into the upsampled image. The updated detail coefficients $NW^h_{2^J}$, $NW^v_{2^J}$ and $NW^d_{2^J}$ are defined as:

$$NW^l_{2^J}[n,m] = \begin{cases} W^l_{2^J}[n,m], & \text{if } [n,m] \text{ belongs to a region border} \\ 0, & \text{otherwise} \end{cases}, \tag{7}$$

where $l \in \{h,v,d\}$. Eq. (7) implies that all detail coefficients are set to zero, except in the positions corresponding to watershed lines of S_{2^J}.

4.2. Applying the inverse wavelet transform

The inverse wavelet transform is applied using S_{2^J} as the approximation image and the updated detail coefficients $NW^h_{2^J}$, $NW^v_{2^J}$ and $NW^d_{2^J}$, resulting is a higher resolution image $S_{2^{J-1}}$. Since detail coefficients are non-zero only at region borders, the interior of homogeneous regions of S_{2^J} are kept homogeneous when computing $S_{2^{J-1}}$. However, such non-zero detail coefficients at region borders introduce fluctuations in the grayvalues close to region borders of $S_{2^{J-1}}$.

The result of the projection algorithm is illustrated in Fig. 4. Fig. 4(a) and (c) shows the projection of the initial segmentation

Fig. 3. (a) Simplified image S_{2^2} for the original *peppers* image at resolution 2^2. (b) Simplified image S_{2^2} for the noisy *peppers* image at resolution 2^2.

(performed at scale 2^2) onto scale 2^1 for the original and noisy *peppers* images, respectively. It can be observed that projected images are formed mostly by homogeneous plateaus, but there are some pixels close to region boundaries that are not assigned to any homogeneous region (such pixels are denoted *lost pixels*). Due to the very small support of Haar wavelets, we can determine the exact position of those lost pixels. More precisely, each detail coefficient will influence a 2×2 neighborhood in the upsampled image, so that all lost pixels must lie in the positions of the upsampled watershed lines. It should be noticed that other wavelets with larger support could be used instead of Haar wavelets, but they would produce a larger number of lost pixels (hence, causing more distortions to pre-existing homogeneous regions).

4.3. Assigning lost pixels

The next step is to scan all lost pixels in $S_{2^{J-1}}$ and assign them to an existing adjacent homogeneous region, so that a piece-wise constant image representation is achieved. For that purpose, let us consider a lost pixel $[n, m]$ and its 8-connectedness neighborhood. Some of these neighboring pixels belong to existing homogeneous regions, while others may be also lost pixels. Let $N_h \leq 8$ denote the number of adjacent pixels that effectively belong to existing homogeneous regions, and let $[n_k, m_k]$ denote the positions of such pixels, for $k=1,\ldots,N_h$. The intensity differences d_k between lost pixel $[n, m]$ and its neighbors $[n_k, m_k]$ are computed through:

$$d_k = |S_{2^{J-1}}[n,m] - S_{2^{J-1}}[n_k,m_k]|, \qquad k = 1,\ldots,N_h, \tag{8}$$

and $[n, m]$ is assigned to the region for which the value d_k is the smallest. In other words, the value of $S_{2^{J-1}}$ at position $[n, m]$ is redefined as:

$$S_{2^{J-1}}[n,m] = S_{2^{J-1}}[n_l,m_l], \qquad \text{where} \quad l = \underset{k}{\operatorname{argmin}}\, d_k. \tag{9}$$

This procedure for assigning lost pixels may be interpreted as competitive region growing, where each existing homogeneous region tries to capture neighboring lost pixels. When a lost pixel is connected to two or more homogeneous regions, the winner region is the one for which the intensity difference is the smallest.

At the end of this process, an updated simplified image $S_{2^{J-1}}$ is obtained at the resolution 2^{J-1}, consisting of homogeneous regions with constant intensity. Fig. 4(b) and (d) illustrate such procedure. They show, respectively, the results of assigning lost pixels for Fig. 4(a) and (c). As it can be observed, resulting images after correcting lost pixels are indeed piece-wise constant.

The next step consists of projecting $S_{2^{J-1}}$ to scale 2^{J-2}. For that purpose, object boundaries of $S_{2^{J-1}}$ are computed (since $S_{2^{J-1}}$ is piece-wise constant, it is trivial to obtain region boundaries). Then, the values of $S_{2^{J-1}}$ are replaced by the respective values of $A_{2^{J-1}}$ at all boundary positions, and the procedure described above for computing the IWT and assigning lost pixels to valid regions is applied. This process is repeated until the full resolution image is obtained (i.e. until resolution 2^0 is reached). It should be noticed that the process of assigning lost pixels, recomputing object boundaries and applying the IWT is necessary to refine object contours as finer resolutions are reached.

Fig. 5 shows the projection of the simplified images S_{2^1} related to original and noisy *peppers* images from resolution 2^1 to 2^0 (which corresponds to full resolution), and the respective object boundaries. There are 156 segmented objects in Fig. 5(a), and 82 objects in Fig. 5(b). It is expected that fewer regions will be obtained for noisy images, because noise corruption destroys small textured regions (and the adaptive threshold described in Section 3 tends to be larger). Also, it can be noticed that contours of segmented objects are clear and well-defined in the full resolution simplified image. However, for the noisy image, edges are a little jagged. Such jaggedness originates in the assignment of lost pixels, because noise produces large intensity variations close to objects borders. To

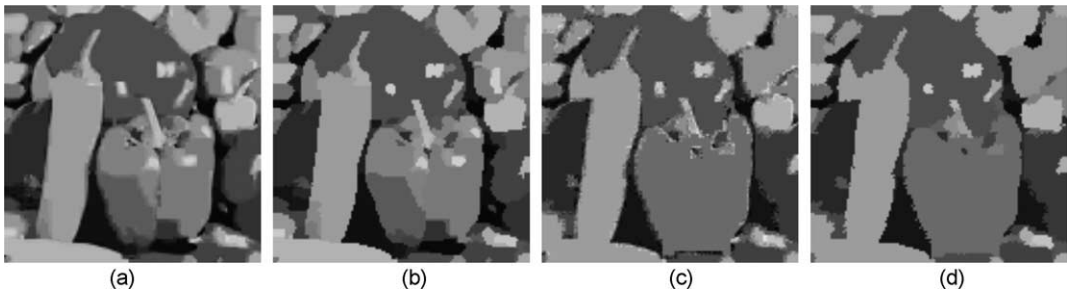

Fig. 4. Simplified image S_{2^1} for the original *peppers* image at resolution 2^1 before (a) and after (b) assigning lost pixels. Simplified image S_{2^1} for the noisy *peppers* image at resolution 2^1 before (c) and after (d) assigning lost pixels.

obtain a smoother contour for a specific object, one could use the boundary of the segmented region as the initialization step for active contour methods, such as *snakes* or one of its many variations.

5. Region merging

In order to further reduce oversegmentation during the initial segmentation of A_{2^J}, region merging can be applied. There are several approaches for merging watershed regions to obtain larger segments corresponding to objects of interest [7,16,17,35,36]. Since this work deals with image simplifications based on intensity averages of segmented regions, a merging criterion based on contrast difference (similar to the approach adopted by Weickert [7]) seems appropriate. In the present approach, two adjacent regions will be merged if their contrast[2] is smaller than a threshold T_m. Although this merging procedure indeed reduces the number of segmented regions, it is not crucial to the proposed technique. In general, region merging becomes more significative when a finer resolution 2^J is chosen as starting point, where many irrelevant details may be segmented.

Fig. 6 shows the segmentation result for the original and noisy *peppers* images, starting at resolution 2^2, after region merging. Fig. 6(a) and (b) shows, respectively, the segmentation of the original *peppers* image using $T_m = 10$ (126 regions) and $T_m = 30$ (58 regions). Fig. 6(c) and (d) shows analogous results for the noisy *peppers* image, and the number of segmented regions is, respectively, 80 and 38. As expected, the number of segmented objects was reduced after the merging process, as regions with small contrast were combined into larger ones. As the threshold T_m increases, the number of regions decreases.

6. Experimental results

In this Section, the influence of the initial resolution 2^J over the final segmentation is investigated. The performance of the proposed method under the presence of intense noise is also analyzed, and the results are compared with other segmentation methods. All results concerning the proposed technique were obtained without region merging, and default value $P = 0.5$ was used to obtain the adaptive threshold. Since segmentation results for most natural images are very application dependent (hence, it is difficult to obtain reliable ground truth for segmented images), a qualitative comparison through visual inspection was performed. The number of segmented regions for each example is also provided as a quantitative measure.

Fig. 7 shows the segmentation of an image corrupted by additive Gaussian noise (PSNR = 24.60 dB). The 256×256 noisy *house* image is shown in Fig. 7(a), and the segmentation results starting at resolutions 2^1, 2^2 and 2^3 are shown, respectively, in Fig. 7(b)–(d). In this order, the number of segmented regions is 127, 62 and 40. As expected, when a finer initial resolution is chosen, more details are segmented. On the other hand, as the initial resolution gets coarser, the proposed technique returns larger (more relevant) objects. As expected, there is a little loss of contour definition when larger values of J

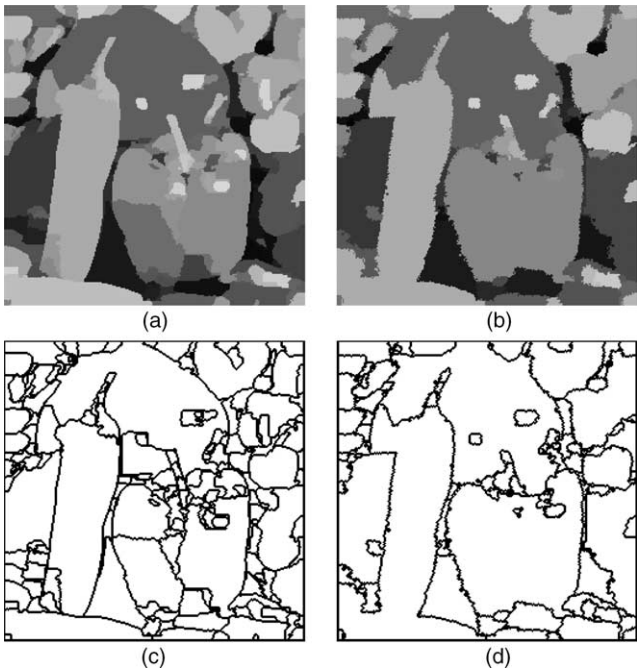

Fig. 5. (a) Simplified image S_{2^0} for the original *peppers* image at full resolution. (b) Simplified image S_{2^0} for the noisy *peppers* image at full resolution. (c)–(d) Region boundaries of (a)–(b).

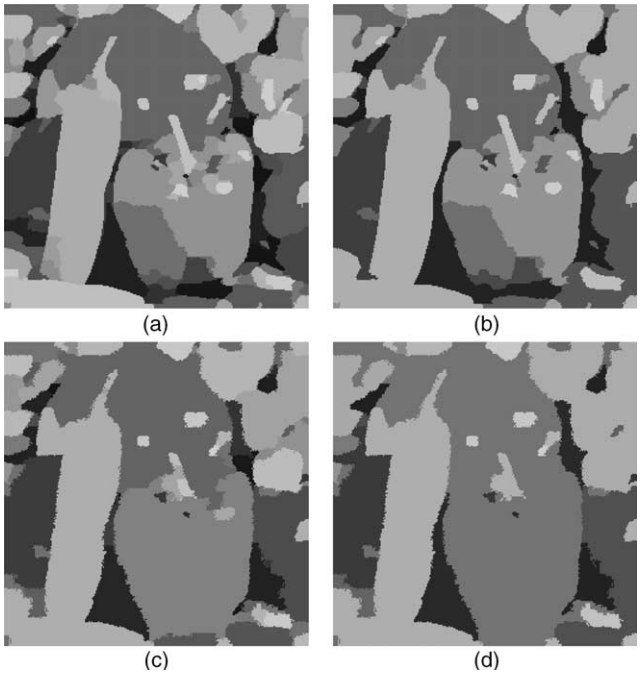

Fig. 6. Final segmentation of original and noisy *peppers* image starting at resolution 2^2 after region merging using. (a) original, $T_m = 10$; (b) original, $T_m = 30$; (c) noisy, $T_m = 10$; (d) noisy, $T_m = 30$.

[2] By contrast, we mean the absolute value of intensity differences.

are used, mostly due to noise influence when the initial segmentation is projected to finer resolutions.

An example of even larger contamination with Gaussian noise is given in Fig. 8. The 256×256 noisy *cameraman* image (PSNR = 20.57 dB) is shown in Fig. 8(a), and the segmentation results starting at resolutions 2^1, 2^2 and 2^3 are shown, respectively, in Fig. 8(b)–(d). The number of segmented objects is, respectively, 125, 75 and 27. The influence of noise was drastically reduced in the magnitude thresholding step, and oversegmentation was prevented. Also, the level of detail gets worse as J increases, as it can be observed in the face of the cameraman.

The proposed method was also tested for images containing inherent noise. Fig. 9 shows the segmentation a 288×352 webcam image, with starting resolutions 2^1, 2^2 and 2^3. The number of segmented regions associated with these resolutions is 109, 51 and 7. In Fig. 9(b) and (c), a finer segmentation was achieved (clouds and details in buildings were detected). In Fig. 9(d), a rough description of the scene was obtained (basically, sky and buildings).

The present method was compared with the multiscale *Edge Flow* segmentation technique [19]. The results of such technique for the noisy *peppers*, *house*, *cameraman* and *webcam* images are shown in Fig. 10 (the number of segmented regions are, respectively, 16, 9, 9 and 14). The offset parameter (which is related to the scale factor of segmentation) was adjusted manually to obtain a good visual result, trying to minimize both false positives and false negatives in the segmentation. The offset values used to produce Fig. 10(a)–(d) were, respectively, 10, 17, 16 and 14. A visual qualitative comparison shows that the proposed technique produces better

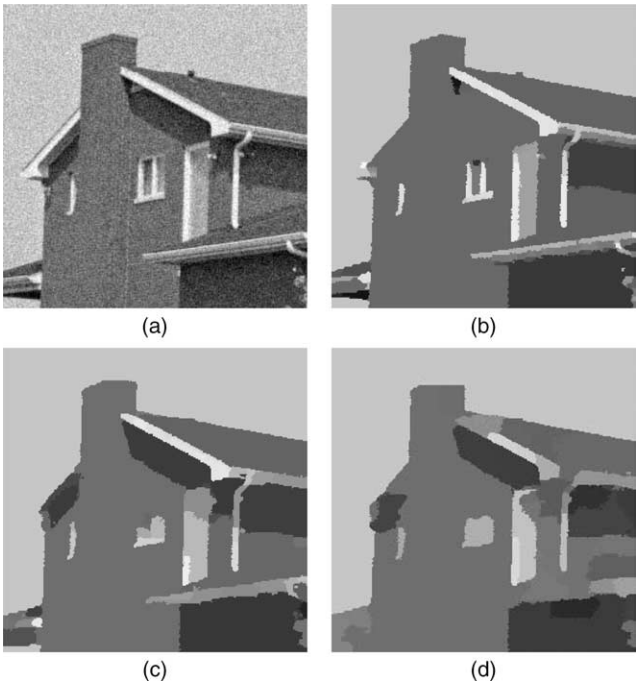

Fig. 7. (a) Noisy *house* image (PSNR = 24.60 dB). (b)–(d) Final segmented image starting at resolutions 2^1, 2^2 and 2^3.

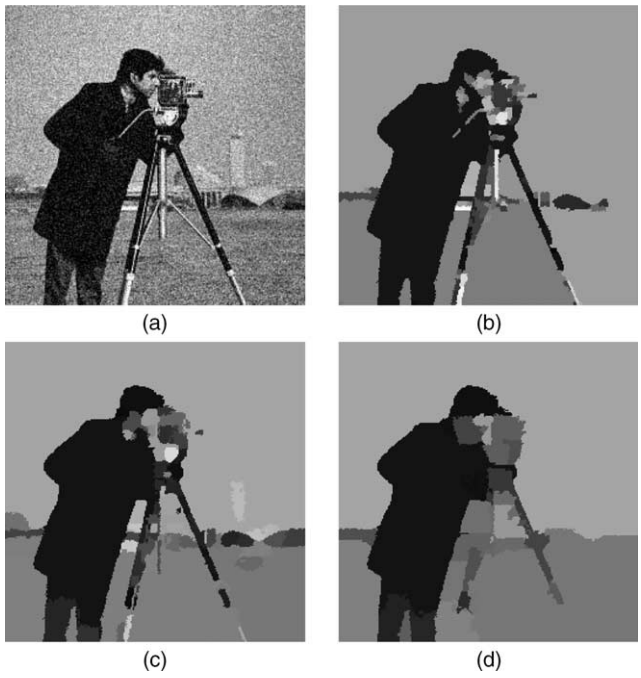

Fig. 8. (a) Noisy *cameraman* image (PSNR = 20.57 dB). (b)–(d) Final segmented image starting at resolutions 2^1, 2^2 and 2^3.

results. For example, the window of the house was not segmented in Fig. 10(b), and all clouds were gathered in the same segmented region in Fig. 10(d).

The same images were also segmented using the *JSEG* technique [21], and the results are shown in Fig. 11 (the number of segmented regions are, respectively, 6, 23, 13 and 34). The scale parameter for this method was also chosen to obtain good visual results. For images noisy *peppers, house* and *cameraman*, the chosen scale was 3, while for the *webcam* image the scale was 2. Visually, the results produced by the new method are better than those

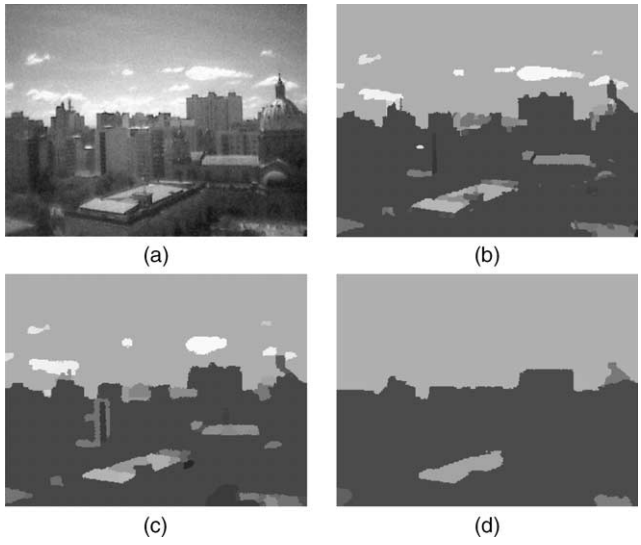

Fig. 9. (a) *Webcam* image. (b)–(d) Final segmented image starting at resolutions 2^1, 2^2 and 2^3.

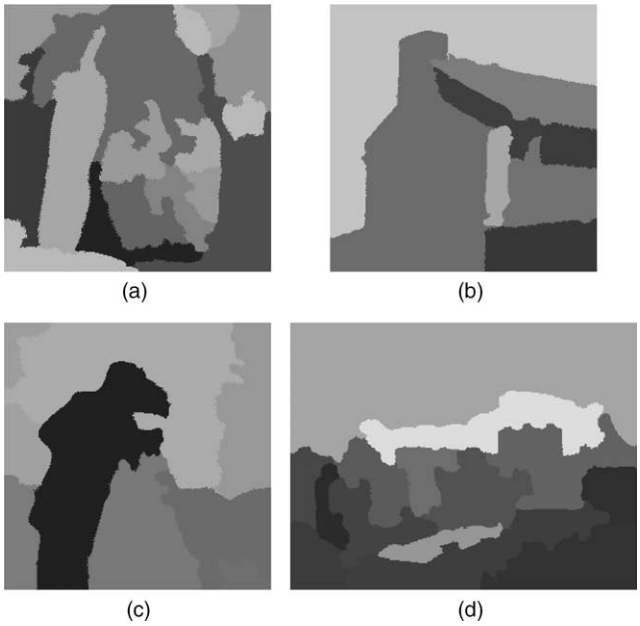

Fig. 10. (a)–(d) Segmentation of the noisy *peppers, cameraman, peppers* and *webcam* images using *Edge Flow*.

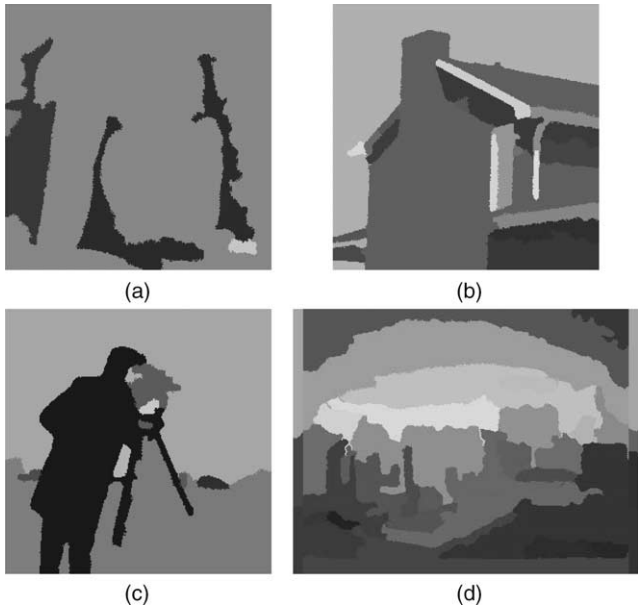

Fig. 11. (a)–(d) Segmentation of the *noisy peppers, cameraman, peppers* and *webcam* images using the *JSEG* method.

obtained using *JSEG*. For instance, the result shown in Fig. 11(a) is very poor (all vegetables were segmented in a single region). Also, the window of the house was not captured in Fig. 11(b).

A more traditional approach for watershed segmentation was also used for comparison, consisting of pre-processing, gradient estimation and watersheds. For pre-processing, the morphological noise-reduction OCCO filter [37] was chosen, and the image gradient was computed through the morphological gradient [2]. Gradient magnitudes smaller than a fraction T of the largest magnitude were removed to reduce oversegmentation. The selection of parameter T is tricky: larger values of T result in less segmented regions, but weak edges tend to be removed, and relevant objects may not be segmented; on the other hand, smaller values of T result in more segmented regions, but several small spurious regions tend to appear. In this work, T was manually chosen aiming to balance the number of segmented regions and the visual quality of the segmented image. The result of such procedure can be seen in Fig. 12 (the number of segmented regions are, respectively, 312, 86, 212 and 82). As expected, segmentation results for 'cleaner images' (*house* and *webcam*) are acceptable, but several small objects we retrieved for noisier images (*peppers* and *cameraman*). In particular, small variations of T lead to drastic changes in segmentation results for noisier images. For example, using $T=0.1$ results in 548 regions for the *peppers* image, while $T=0.125$ results in 312 regions.

The adaptive threshold explained in Section 3 was developed assuming Gaussian noise. However, the proposed technique was also tested for images containing non-Gaussian noise, with promising results. Fig. 13(a) shows a *Satellite Aperture Radar* (SAR) image, which is contaminated by speckle noise [38]. The segmentation results starting at resolutions 2^2, 2^3 and 2^4 are shown, respectively, in Fig. 13(b)–(d). In this order, the number of segmented regions is 80, 64 and 27. Larger initial resolutions were chosen for this example because the original image is larger (558×558 pixels) and noise corruption is very intense. Most of rice crops (brighter regions) are effectively segmented from the background, specially in Fig. 13(c) and (d).

Fig. 12. (a)–(d) Segmentation of the noisy *peppers, cameraman, peppers* and *webcam* images using morphological filtering and watersheds.

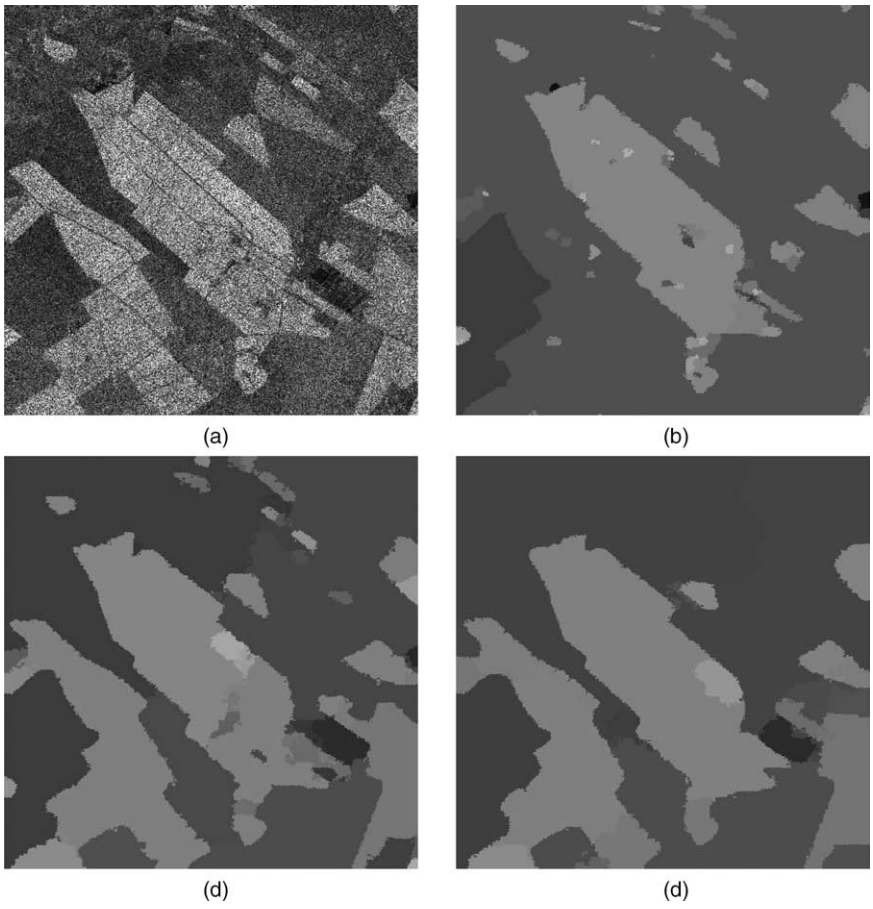

Fig. 13. (a) SAR image. (b)–(d) Final segmented image starting at resolutions 2^2, 2^3 and 2^4.

The technique described in this paper was implemented in MATLAB 6.1, and typical running time for segmenting a 256×256 image starting at resolution 2^2 is about 1.2 s[3] in a Pentium IV 1.6 MHz personal computer. The implementation of *Edge Flow* (provided in [19]) running on the same machine requires approximately 12.6 s to segment a 256×256 image, and *JSEG* (provided in [21]) requires about 8.3 s.

7. Conclusions

In this paper, a new multiresolution technique for image segmentation based on wavelet decompositions and watersheds was proposed. It allows the selection of the initial resolution for applying the watershed transform, and projects the segmentation result back into the original full resolution, without significant loss of edge definition. The initial resolution must be chosen in accordance with the resolution of the image, the size of the desired objects and the amount of noise contamination. Typically, more and smaller objects are segmented when finer initial resolutions are selected, while less and larger objects are obtained for coarser resolutions. A region merging process was also suggested to further reduce oversegmentation.

Experimental results indicate that the proposed technique performs well for both inherent and artificial noise contamination. In particular, it is robust in the presence of intense noise.

The proposed method is completely adaptive: after the selection of the initial resolution 2^J, it does not require any user defined parameter (it should be noticed that default value $P=0.5$ was used in all experiments). Other state-of-the art segmentation techniques [7,16,17] require a manual selection of thresholds (in the region merging stage), what may influence the result of the final segmentation. A qualitative comparison with other segmentation techniques that require only the selection of the scale [19,21] indicates that the proposed method produces equivalent or superior results in less time.

Future work will concentrate on combining the segmentation result obtained with different starting resolutions, extending the method to color image segmentation, and combining this approach with energy-based segmentation techniques (to improve the smoothness of the segmented contours in the presence of noise).

Acknowledgements

This work was developed in collaboration with HP Brazil R&D and Brazilian research agency CNPq. The author would also like to thank anonymous reviewers, for their valuable comments.

[3] An efficient implementation on a compiled language is expected to improve significantly the execution time.

References

[1] F. Meyer, S. Beucher, Morphological segmentation, Journal of Visual Communication and Image Representation 1 (1990) 21–46.

[2] L. Vincent, P. Soille, Watersheds in digital spaces: an efficient algorithm based on immersion simulations, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (6) (1991) 583–598.

[3] K. Haris, S.N. Efstratiadis, N. Maglaveras, A.K. Katsaggelos, Hybrid image segmentation using watersheds and fast region merging, IEEE Transactions on Image Processing 7 (12) (1998) 1684–1699.

[4] A. Bieniek, A. Moga, An efficient watershed algorithm based on connected components, Pattern Recognition 33 (6) (2000) 907–916.

[5] A. Bleau, L. Leon, Watershed-based segmentation and region merging, Computer Vision and Image Understanding 77 (3) (2000) 317–370.

[6] F. Meyer, Levelings and morphological segmentation, in: Proceedings of SIBGRAPI, Rio de Janeiro, Brazil, 1998, pp. 28–35.

[7] J. Weickert, Efficient image segmentation using partial differential equations and morphology, Pattern Recognition 34 (9) (2001) 1813–1824.

[8] H.T. Nguyen, M. Worring, R.V.D. Boomgaard, Watersnakes: energy-driven watershed segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (3) (2003) 330–342.

[9] M. Kass, A. Witkin, D. Terzopoulos, Snakes: active contour models, International Journal of Computer Vision 1 (1987) 321–331.

[10] L.M. Lifshitz, Image segmentation using global knowledge and a priori information, PhD thesis, University of North Carolina, Chapel Hill, 1987.

[11] T. Lindeberg, Discrete scale-space theory and the scale space primal sketch, PhD thesis, Royal Institute of Technology, Stockholm, Sweden, May 1991.

[12] P. Jackway, Morphological multiscale gradient watershed image analysis, in: Nineth Scandinavian Conference on Image Analysis, 1995.

[13] J.M. Gauch, Image segmentation and analysis via multiscale gradient watershed hierarchies, IEEE Transaction on Image Processing 8 (1) (1999) 69–79.

[14] S. Mukhopadhyay, B. Chanda, Multiscale morphological segmentation of gray-scale images, IEEE Transactions on Image Processing 12 (5) (2003) 533–549.

[15] I. Vanhamel, I. Pratikakis, H. Sahli, Multiscale gradient watersheds of color images, IEEE Transactions on Image Processing 12 (6) (2003) 617–626.

[16] J.B. Kim, H.J. Kim, A wavelet-based watershed image segmentation for vop generation, in: IEEE International Conference on Pattern Recognition, Québec City, Canada, 2002, pp. 505–508.

[17] J.B. Kim, H.J. Kim, Multiresolution-based watersheds for efficient image segmentation, Pattern Recognition Letters 24 (2003) 473–488.

[18] C. Jung, J. Scharcanski, Robust watershed segmentation using wavelets, Image and Vision Computing 23 (7) (2005) 661–669.

[19] W.Y. Ma, B.S. Manjunath, Edge flow: a framework of boundary detection and image segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, 1997, pp. 744–749, software available for download at http://vision.ece.ucsb.edu/segmentation/edgeflow/software/index.htm.

[20] W.Y. Ma, B.S. Manjunath, Edge flow: a technique for boundary detection and image segmentation, IEEE Transactions on Image Processing 9 (8) (2000) 1375–1388.

[21] Y. Deng, B.S. Manjunath, Unsupervised segmentation of color-texture regions in images and video IEEE Transactions on Pattern Analysis and Machine Intelligence. 23(8) (2001) 800–10software available for download at: http://vision.ece.ucsb.edu/segmentation/jseg/software/index.htm

[22] H. Wu, J. Liu, C. Chui, A wavelet-frame based image force model for active contouring algorithms, IEEE Transactions on Image Processing 9 (11) (2000) 1983–1987.

[23] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis IEEE Transactions on Pattern Analysis and Machine Intelligence. 24(5) (2002) 603–19software available for download at: http://www.caip.rutgers.edu/riul/research/code/EDISON/

[24] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 888–905.

[25] V. Gies, T. Bernard, Statistical solution to watershed over-segmentation, in: International Conference on Image Processing, 2004, pp. III: 1863–1866.

[26] L. Patino, Fuzzy relations applied to minimize over segmentation in watershed algorithms, Pattern Recognition Letters 26 (6) (2005) 819–828.

[27] W.K. Pratt, Digital Image Processing, Wiley, New York, 1991.

[28] G. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 11 (7) (1989) 674–693.

[29] G. Strang, T. Nguyen, Wavelets and Filter Banks, Wellesley-Cambridge Press, Cambridge, 1996.

[30] J. Scharcanski, C.R. Jung, R.T. Clarke, Adaptive image denoising using scale and space consistency, IEEE Transactions on Image Processing 11 (9) (2002) 1092–1101.

[31] D.L. Donoho, Wavelet shrinkage and w.v.d.: a 10-minute tour, Progress in Wavelet Analysis and Applications.

[32] H.J. Larson, B.O. Shubert, Probabilistic Models in Engineering Sciences, vol. I, Wiley, London, 1979.

[33] E.P. Simoncelli, E. Adelson, Noise removal via bayesian wavelet coring, in: Proceedings of IEEE International Conference on Image Processing, Lausanne, Switzerland, 1996, pp. 279–382.

[34] P. Henstock, D.M. Chelberg, Automatic gradient threshold determination for edge detection, IEEE Transactions on Image Processing 5 (5) (1996) 784–787.

[35] M.W. Hansen, W.E. Higgins, Watershed-driven relaxation labeling for image segmentation, in: IEEE International Conference on Image Processing, Austin, TX, 1994, pp. 460–464.

[36] B. Ogor, V. Haese-Coat, K. Kpalma, Cooperation of mathematical morphology and region growing for remote sensing region growing, Image Signal Processing and Remote Sensing II 2579 (1995) 375–386.

[37] R.A. Peters, A new algorithm for noise reduction using mathematical morphology, IEEE Transactions on Image Processing 4 (3) (1995) 554–568.

[38] A.K. Jain, Fundamentals of Digital Image Processing, Prentice-Hall, Englewood Cliffs, NJ, 1989.