Watershed Segmentation Using a Multiscale Ramp Edge Merging Strategy

Padraig Corcoran, Adam Winstanley
National Centre for Geocomputation, Department of Computer Science
NUI Maynooth.
padraigc@cs.nuim.ie

Abstract

For the watershed segmentation algorithm to be successful it must be implemented on a realistic gradient image. In most watershed implementations, gradients are extracted using an operator optimal for ideal step edges. However, image edges are never ideal steps and more closely resemble ramp edges at multiple scales. Therefore this strategy results in an inaccurate measure of image gradients and in turn lessens segmentation performance. In this paper we propose a new multiscale gradient operator for ramp edges. This strategy merges the properties of accurate gradient estimation of a large scale kernel with accurate localization of a small scale kernel by tracking gradients from larger to smaller scales. Quantitative performance evaluation of segmentation results shows this approach to outperform a traditional single small scale gradient operator optimal for step edges.

1. Introduction

Low-level segmentation of a given scene into a set of meaningful objects represents the first step in many object recognition systems. Although research in the area is very active, it is often difficult to produce segmentation of sufficiently high quality that would allow reliably calculation of object specific information like shape. In fact as a result of this failing, the current best approaches to object recognition do not employ low-level segmentation [1, 2]. The goal of this work is to provide an accurate low level segmentation algorithm which could later be used as input to such an object recognition system.

Most segmentation strategies can be classified as a region or boundary based approach. In region based approaches grouping of homogenous areas is performed to produce segmentation. In contrast, boundary based approaches attempt to extract the boundaries between homogenous areas. The watershed transform combines both region and boundary based techniques [3]. Pixels are grouped around the regional minima of a gradient image and boundaries are located along the crest lines of this image. In order to achieve accurate segmentation using the watershed transform accurate boundary gradients must first be extracted using a suitable gradient operator. This paper presents a strategy for computing such gradients accurately for natural scenes.

Most current strategies for extracting gradient images are based on the assumption that the underlying edges can be modelled accurately by a step edge. Canny [4] designed an optimal gradient operator for step edges emersed in Gaussian noise of a single scale in terms of three criteria. Most edges in natural images are immersed in noise or texture of varying scales. To address this issue many multiscale edge detection techniques do exist with all based on the assumption that edges are ideal steps immersed in different scales of noise or texture [5-7]. These techniques basically involve fusing edge information extracted from a Gaussian scale space.

The edges contained in natural images will never match this classic step edge model upon which these techniques are founded. Most edges will consist of a gradual as opposed to sudden change in intensity values. Even if edges have idealized steps edges to begin with, during the process of image capture and digitization these will be converted into ramps. This is due to the fact that any imaging system will have finite bandwidth and therefore it will behave approximately as a low pass filter, blurring the edges [8]. To overcome this obstacle the work of Canny was extended in [8] and [9] to derive the optimal edge detector for ramp edges of a single scale in terms of the same three criteria. The ramp edges contained in natural images will never be of a single scale though. Even if they are of a single scale initially, any imaging system will have a finite depth of field. This results in edges at different distances from the focal plane receiving different amounts of blur.

0-7695-2887-2/07 $25.00 © 2007 IEEE
DOI 10.1109/IMVIP.2007.29

In [10] Laligant defined a strategy for merging detected ramp edges at multiple scales using a classification algorithm. The result of this strategy is a binary edge image with no information regarding relative gradient magnitudes available. Laligant in [11] proposed another algorithm for merging ramp edges at multiple scales in which a local maximum is classified as an edge if it exists at multiple scales in a fine-to-coarse and coarse-to-fine search. Again this approach does not calculate the relative gradient magnitude values returning only a binary edge image. Wang [12] described a method to merge gradients at multiple scales by simply taking a summation of gradients over multiple scales. This technique will respond to ramp edges more strongly than those designed for step edges but gradient values are inaccurate with poor localization. Guimaraes introduced a gradient pile up algorithm in [13] also to enhance gradients corresponding to ramp edges. Again this strategy will respond to ramp edges more strongly than those designed for step edges but these values are inaccurate with imprecise localization.

In the second section of this paper we define our ramp edge model and illustrate the multiscale ramp edge gradient estimation problem. In section 3 we propose a new multiscale ramp edge gradient operator which extracts accurate and precisely localized gradient values for noisy ramp edges at multiple scales. This is followed by a presentation of results in section 4. Finally in section 5 we draw conclusions from this work and propose future research directions.

2. Ramp Edge Model and Properties

In this work we model a ramp edge as the filtering of an ideal step edge with a low pass Gaussian function of a given scale. This is regarded as an accurate ramp edge model and can be approximated by the erf function [10]. An example of this edge model is displayed in figure 1.

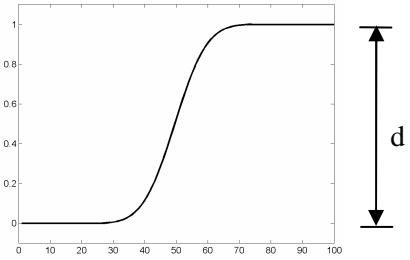

Figure 1. Edge model defined as the smoothing of a step edge with a Gaussian function. The contrast d represents the correct gradient of this ramp edge.

For this ramp edge model the correct gradient is the distance between the two uniform regions on either side of the edge. This is represented by d in figure 1. The gradient of this ramp edge is calculated using the kernel [.5 0 -.5] designed for step edges with the result displayed in figure 2. The correct localization of this edge is the point where the first derivative is maximum and this corresponds to the zero crossing in the second derivative [14]. Although this local maximum offers correct localization, this value does not have the desired gradient magnitude d.

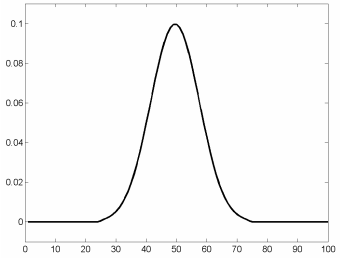

Figure 2. First order derivative of figure 1 calculated using the kernel [1 0 -1]. The location of the maximum value is the correct boundary location.

Applying a gradient operator designed for step edges to such ramp edges does not return a true measure of edge gradient. Figure 3 shows a one dimensional signal containing three ramp edges at various scales. The result of filtering with the kernel designed for step edges which contains the values [-.5 0 .5] and taking the absolute value is displayed in figure 4. From this figure we see that this strategy gives an under-estimation of the gradient for all edges apart from the step edge in the signal.

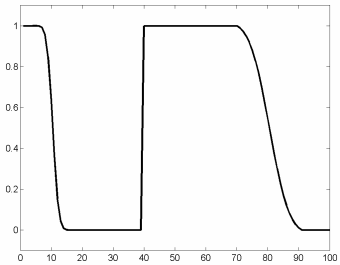

Figure 3. Examples of three 1-D edges, from left to right; a ramp edge, a step edge and a larger scale ramp edge. All edges have an equal contrast d of value 1.

As just shown, applying a gradient operator for step edges to a signal containing ramp edges will result in an under-estimated measure of edge gradient. This will in turn lead to under-segmentation when the marker-controlled watershed transform is applied. To

overcome this difficulty many authors initially run the transform at a very small scale then post-process with a region merging step [15-17]. Segmentation run at a small scale ensures regions are not under-segmented due to under estimated boundary gradients. The region merging step considers the contrast between regions at a larger scale than the original gradient operator therefore reducing the effect of ramp edges. A problem with this approach is that the region merging step assumes that each region is completely homogenous but this is obviously not the case due to the existence of ramp edges at region boundaries.

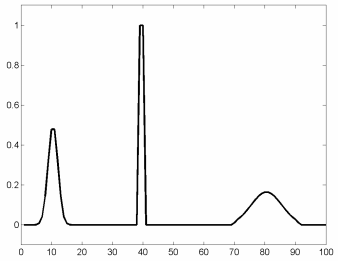

Figure 4. The result of convoluting the gradient operator [1 0 -1] with figure 3 and taking the absolute value is display. This operator only assigns the correct gradient value of 1 to the centre step edge.

3. Multiscale Ramp Edge Gradient Calculation

In this section we propose a new multiscale gradient operator for ramp edges which tracks gradient values from large to small scales. This strategy fuses the benefits of accurate gradient values with accurate localization. In the following sub-section we detail the 1-D implementation of this algorithm. We then discuss how this technique can be extended to two dimensions and made robust to noise.

3.1. Multi-Scale Gradient Tracking

A possible solution to this problem of inaccurate gradients would be to calculate the gradients at scale greater than the greatest ramp edge scale. An example of this procedure applied to figure 3 is displayed in figure 5. The problem with this approach is that although truthful gradient values will be returned for all ramp edges, boundary localization will be lost on all ramp edges having a scale smaller than the scale of the gradient operator.

We now describe a multiscale ramp edge gradient operator which tracks gradient values from larger to smaller scales. This strategy merges the benefits of accurate gradient values of a larger scale gradient operator with the accurate localization of a smaller scale gradient operator.

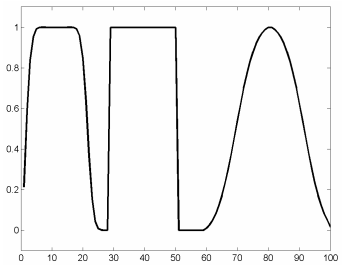

Figure 5. The result of convoluting figure 3 with the gradient operator [1 (23 zeros) -1] and taking the absolute value is displayed. The operator returns a correct boundary gradient measure for all edges, but accurate boundary localization is lost on the left and centre edges.

Our algorithm takes as input a one dimensional signal from which we want to extract gradients, and two parameters LScaleMax and SScaleMin both of which must be odd numbers. LScaleMax represent the largest possible ramp edge scale, and SScaleMin the scale we want to localize edges to. A loop operation is then performed to track gradient values from larger to smaller scales. Each iteration of this loop utilizes two gradient images at different scales. A larger scale gradient image gradLScale at the scale LScale and a smaller scale gradient image gradSScale at the scale SScale. Within this loop the gradient at each location i in gradLScale, referred to as gradLScale(i), is transmitted to gradSScale as follows. Every location gradSScale(j) for which the smaller scale gradient kernel is contained within the larger scale gradient kernel positioned at i is searched for a maximum value. An illustration of this is displayed in figure 6.

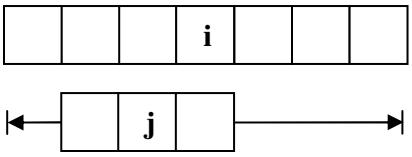

Figure 6. A larger scale gradient kernel is shown above and a smaller scale gradient kernel below. Every location for which the smaller scale kernel is contained within the larger scale kernel is searched for a maximum value.

The location of this maximum value is referred to as trackLoc. Then if the value of gradLScale(i) is greater than gradSScale(trackLoc),

gradSScale(trackLoc) is assigned the value gradLScale(i).

The above procedure tracks the gradient values from a single larger scale to a single smaller scale. This step is repeated, each time the previous SScale becomes the new LScale, and the previous gradSScale becomes the new gradLScale. Finally the process is terminated when LScale reaches the value SScaleMin. The pseudocode for the algorithm is as follows:

```
Input: data, MaxScale, MinScale.
Output: gradSScale.
Algorithm:
LScale = MaxScale.
SScale = MaxScale-2
gradLScale = diff(data, LScale); //Large scale gradients
gradSScale = diff(data, SScale); //Small scale gradients

while SScale >=  MinScale
   for each gradientLScale(j)
      trackLoc = maxContainedSScale(j); // see figure 8
      if gradientLScale(j) > gradientSScale(trackLoc)
         gradSScale(trackLoc) = gradLScale(i);
      end
   end
   gradLScale = gradSScale;
   SScale = SScale - 2;
   gradSScale = diff(data, SScale);
end
return gradSScale;
```

The result of applying this algorithm to the signal displayed in figure 1, with parameters LScaleMax and SScaleMin given the values 25 and 3 respectively is displayed in figure 7. It can be seen from this result that the algorithm combines the properties of accurate localization of a smaller scale kernel (see figure 2 for an example of this) with accurate gradient values of a larger scale kernel (see figure 5 for an example of this).

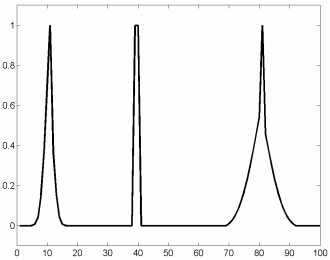

Figure 7. The result of applying the proposed 1D algorithm to figure 3. Accurate localization of a smaller scale kernel is merged with accurate gradient values of a larger scale kernel.

3.2. Extension to 2D and Dealing with Noise

In the previous sub-section we described a method for accurately measuring and localizing the gradients of multiscale ramp edges in 1-D. We now show how this strategy can be extended to 2-D and made robust to noise.

A 1-D filter can be extended to 2-D by applying the filter perpendicular to the edge and a projection function parallel to the edge. The projection function averages along the edge reducing noise. This implementation is almost impossible without prior knowledge regarding edge orientation. Therefore a low-pass projection function is generally utilized [18]. Then making use of the fact that the slope of a surface in any direction can be determined from the slope in two orthogonal directions, we apply 2-D separable filters in x and y directions. In previous work where a Gaussian noise model was assumed the projection function derived closely resembled a Gaussian function [8]. We therefore decided to use a Gaussian as our projection function. When deriving our multiscale ramp edge gradient operator we assumed the data was noise free which is not the case in most real data. Therefore we also smooth the data in a direction parallel to the filter direction.

Smoothing a noisy ramp edge with a Gaussian filter will cause to edge to approach a noise free ramp edge but of a larger scale. Figure 8 displays the sample 1-D ramp edges of figure 3 with added Gaussian noise. These edges are smoothed with a Gaussian function with the outcome displayed in figure 9. These edges now match the desired ramp edge model. The result of applying our multiscale ramp edge gradient operator to these smoothed edges is displayed in figure 10. We can see these ramp edge gradients are of a correct magnitude and localized accurately.

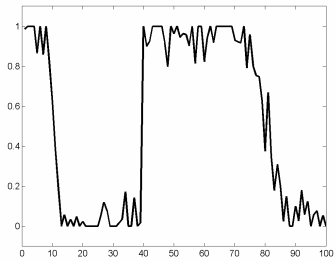

Figure 8. Result of adding Gaussian noise of mean 0 and variance 0.01 to figure 3.

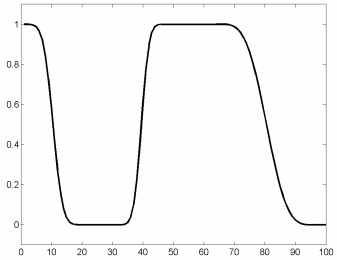

Figure 9. Smoothing of noisy edges in figure 8 causes them to approach the desired edge model.

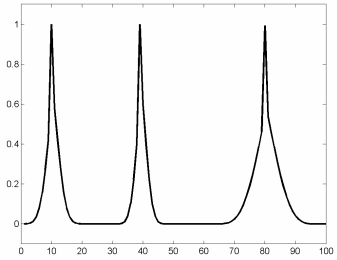

Figure 10. The result of applying the proposed gradient operator to figure 9. Accurate gradient values and localization is obtained.

An example of a synthetic 2-D dataset containing two ramp edges of different scales with added Gaussian noise is displayed in figure 11. The contrasts d of both edges contained in this image are equal.

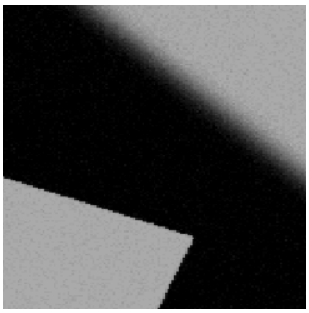

Figure 11. Synthetic image containing two ramp edges of different scales immersed in Gaussian noise of mean 0 and variance 0.001. The contrast (d from figure 3) of both edges is equal.

Smoothing with a Gaussian function of sigma 1.5 is performed to remove noise. This is followed by the application of the single small scale kernel [.5 0 -.5] in x and y directions. Gradient magnitudes are calculating from the resulting values and this is displayed in figure 12. Although both edges have equal contrast, using this single small scale kernel results in very different gradient magnitude values for each edge. Also these responses are not very localized. We applied our proposed multiscale ramp edge gradient operator to the smoothed image in x and y directions followed by calculation of gradient magnitudes. From the result displayed in figure 13, we can see that this strategy returns similar gradient magnitude values which are localized to a fine scale for both ramp edges.

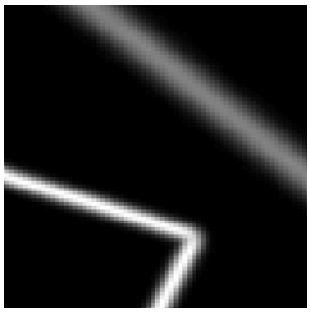

Figure 12. Gradient magnitude values for figure 11 are calculated with a single small scale kernel. Although each ramp edge has equal contrast, the use of this small scale gradient operator results in very different gradient magnitude values for each edge.

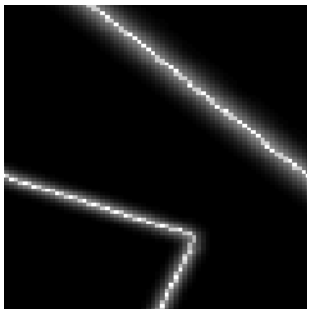

Figure 13. Gradients for figure 11 are tracked from a large scale kernel of size 23 to a small scale kernel of size 3. This is followed by the calculation of gradient magnitudes. Both ramp edges receive similar gradient magnitudes values which are localized to the scale of the smaller kernel.

To perform segmentation the marker-controlled watershed transform was used. The watershed transform combines region growing and edge detection techniques. Pixels are grouped around the regional minima of a gradient image and boundaries are located along the crest lines of this image. In practice, direct computation of the watershed algorithm results in over-segmentation due to the presence of spurious minima. To overcome this, the gradient image is first filtered using a marker function; in this case the H-minima transform, to remove all irrelevant minima [3]. The H-minima transform takes one parameter H which specifies the scale of minima to be suppressed and therefore the resulting scale of segmentation. From figure 13 we can see that the boundaries defined by our

proposed gradient operator are very narrow. Therefore it is important to use 4- not 8-connectivity to prevent one region flowing into a neighbour resulting in under-segmentation.

4. Results

Figure 14 displays an image taken from the Berkeley segmentation dataset [19]. Smoothing with a Gaussian of sigma 1.5 to reduce noise is performed. This is followed by the application of a small scale gradient operator and our proposed multiscale gradient operator with the results displayed in figures 15 and 16 respectively. To allow a closer examination of these images a magnified region of figures 15 and 16 is displayed in figures 17 and 18 respectively. Gradient magnitude values along the arm are more uniform when extracted with the proposed multiscale gradient fusion strategy than when extracted with a single small scale kernel.

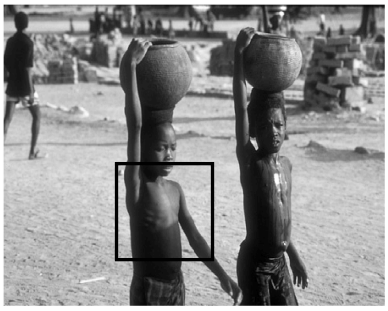

Figure 14. sample image taken from the Berkeley segmentation dataset. The black square represents the area which is magnified in figures 17 and 18.

Figure 15. Gradient magnitudes extracted from figure 14 with a single small scale kernel of size 3.

Figure 16. Gradients tracked from a kernel of size 9 to a kernel of size 3 using the proposed multiscale gradient tracking strategy.

Figure 17. Magnified section of figure 15.

Figure 18. Magnified section of figure 16.

As mentioned in the introduction the goal of our algorithm is to segment a given scene into a set of useful objects. To evaluate the calibre of our proposed multiscale gradient operator against a single small scale gradient operator at extracting gradients for use in the marker-controlled watershed transform, the Berkeley segmentation dataset [19] was used.

To quantitatively measure the accuracy of segmentation results, the Normalized Probabilistic Rand (NPR) Index was utilized [20]. Given a segmentation and corresponding set of ground-truths, the NPR index quantifies the agreement of segmentation with ground-truths. Greater agreement results in higher index values. Three hundred images were taken from the Berkeley segmentation dataset for evaluation. This was divided into one hundred training

images to optimize the segmentation scale or H parameter and two hundred test images. All images were smoothed with a Gaussian of sigma 1.5 before any gradients were calculated. Single small scale gradient images were calculated using the same procedure described in section 3.2. Within the proposed multiscale gradient operator gradients were tracked from a kernel of size 9 to a kernel of size 3. On the test dataset our proposed multiscale gradient operator achieved an average NPR index value of 0.37. This result outperformed the single scale gradient operator which accomplished an average NPR index value of 0.33 on the same data.

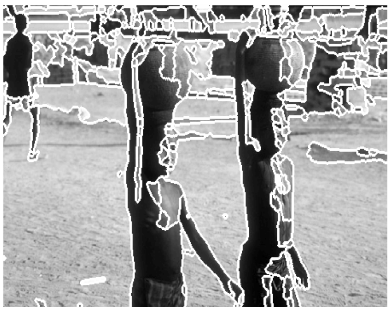

Figure 19. Watershed segmentation result with watershed lines represented by white.

Figure 20. Watershed segmentation result with watershed lines represented by white.

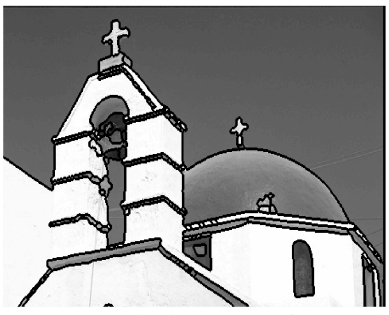

Figure 20. Watershed segmentation result with watershed lines represented by black. In this non-textured image, segmentation quality is high will edges localized accurately to a small scale.

Figure 22. Watershed segmentation result with watershed lines represented by white. Significant over-segmentation is evident due to texture.

A selection of segmentations achieved using the proposed multiscale gradient operator are displayed in figures 19, 20, 21 and 22. These segmentations are run at the scale which optimized performance on the training dataset. From these figures we see that the segmentation quality is high in non-textured regions. However in textured regions the segmentation quality is poor with significant over-segmentation evident.

5. Conclusions

Most natural edges do not match the ideal step edge model upon which the majority of existing gradient estimation techniques are based. In fact they more closely resemble ramp edges of varying scales. The main contribution of this paper is the introduction of a new multiscale ramp edge gradient operator. It was shown that this technique can extract accurate gradient values for ramp edges of varying scales while maintaining accurate boundary localization. Qualitatively performance evaluation of segmentation results demonstrates that this strategy outperforms a single small scale gradient operator optimal for step edges.

Visual inspection of these results shows significant over-segmentation to be evident in textured regions. To tackle this problem texture information needs to be integrated into the segmentation process. Prior to calculation of image gradients all locations were smoothed with a Gaussian of equal scale. Different regions will contain varying amounts of texture or noise and consequently require different amounts of smoothing. Utilizing a locally adaptive smoothing process which could apply the correct amount of smoothing to each location would therefore also reduce over-segmentation.

Colour is another important cue used by the visual system to define boundaries. In this work we have ignored its presence and concentrated on the visual cue

of brightness. Incorporating colour information would almost certainty improve segmentation performance.

Addressing the above issues will be the focus of future work.

6. Acknowledgments

The authors would like to acknowledge Oliver Laligant for the provision of some test images.

7. References

[1] Roth, V. and B. Ommer (2006), *Exploiting low-level image segmentation for object recognition.* Pattern Recognition (Symposium of the DAGM), LNCS , Springer, Vol. 4174, No. pp. 11-20.

[2] Agarwal, S., A. Awan, and D. Roth (2004), *Learning to detect objects in images via a sparse, part-based representation.* IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 26, No. 11, pp. 1475-1490.

[3] Soille, P. (2002), *Morphological image analysis: principles and applications*: Springer-Verlag Berlin and Heidelberg GmbH & Co. K.

[4] Canny, J.F. (1986), *A computational approach to edge detection.* IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 8, No. 6, pp. 679-698.

[5] Basu, M. (2002), *Gaussian-based edge-detection methods-a survey.* Systems, Man and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, Vol. 32, No. 3, pp. 252-260.

[6] Sumengen, B. and B.S. Manjunath. (2005), *Multi-scale edge detection and image segmentation*, in *European signal processing conference (EUSIPCO)*.

[7] Midoh, Y., K. Nakamae, and H. Fujioka (2007), *Object size measurement method from noisy SEM images by utilizing scale space.* Measurement science and technology, Vol. 18, No. 3, pp. 579-591.

[8] Petrou, M. and J. Kittler (1991), *Optimal edge detectors for ramp edges.* IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 13, No. 5, pp. 483-491.

[9] Wang, Z., K. Raghunath Rao, and J. Ben-Arie (1996), *Optimal ramp edge detection using expansion matching.* IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, No. 11, pp. 1092-1097.

[10] Laligant, O., et al. (2005), *Merging system for multiscale edge detection.* Optical Engineering, Vol. 44, No. 3.

[11] Laligant, O., F. Truchetet, and A. Dupasquier (2005), *Edge enhancement by local deconvolution.* Pattern Recognition, Vol. 38, No. 5, pp. 661-672.

[12] Wang, D. (1997), *A multiscale gradient algorithm for image segmentation using watersheds.* Pattern Recognition, Vol. 30, No. 12, pp. 2043-2052.

[13] Guimaraes, L., et al. (2004), *Gradient pile up algorithm for edge enhancement and detection*, in *Lecture notes in computer science*, Springer Berlin / Heidelberg. pp. 187-194.

[14] Jain, R., R. Kasturi, and B.G. Schunck (1995), *Machine Vision*: McGraw-Hill.

[15] Castilla, G., A. Lobo, and J. Solana (2004), *Size-constrained region merging (SCRM): a new segmentation method to derive a baseline partition for object-oriented classification.* Proceedings of SPIE, Vol. 5239, pp. 472-482.

[16] O'Callaghan, R.J. and D.R. Bull (2005), *Combined morphological-spectral unsupervised image segmentation.* IEEE Transactions on Image Processing, Vol. 14, No. 1, pp. 49-62.

[17] Jung, C.R. (2007), *Unsupervised multiscale segmentation of color images.* Pattern Recognition Letters, Vol. 28, No. 4, pp. 523-533.

[18] Sarkar, S. and K.L. Boyer (1991), *On optimal infinite impulse response edge detection filters.* IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 13, No. 11, pp. 1154-1171.

[19] Martin, D., et al. (2001), *A database of human segmented natural images and its applications to evaluating segmentation algorithms and measuring ecological statistics*, in *International conference on computer vision*, Vancouver, pp. 416-423.

[20] Unnikrishnan, R., C. Pantofaru, and M. Hebert. (2005), *A measure for objective evaluation of image segmentation algorithms*, in *IEEE conference on computer vision and pattern recognition, workshop on empirical evaluation methods in computer vision*, San Diego, pp. 34-41.