Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 53(67), Fascicola 2, 2008

Image Filtering and Segmentation Using Kernel Density Estimation

Vasile Gui[1], Jyrki Laitinen[2], Florin Alexa[1]

Abstract – Kernel density estimation and mode finding techniques play an active role in solving contemporary computer vision problems, like edge preserving smoothing, segmentation, registration, motion estimation and tracking. The mean shift algorithm is a popular approach to locate density modes. Recently we proposed the multiscale mode filter, a generalization of the mean shift filter, which is able to avoid spurious modes while minimizing outlier sensitivity. In this paper we evaluate the effectiveness of the multiscale mode filter in edge preserving smoothing and image segmentation.

Keywords: edge preserving smoothing, multiscale, mode location, mean shift, segmentation.

I. INTRODUCTION

Meaningful image segmentation and salient feature extraction are difficult low level image processing tasks. Reliable detection of image features and grouping pixels with similar features into image segments can be done based on uniformity or, conversely, based on non-uniformity detection. Preprocessing filters have to be able to discriminate between natural image variability within a real world object and variability between different adjacent objects of a scene. Image smoothing is an important tool for reducing the first kind of image variability. Since edges are important image features for proper region finding, smoothing filters used in computer vision are required to smooth the image differences within different object areas, while preserving edges separating objects. Such conflicting demands are best addressed within a nonlinear image processing framework. Currently, major research directions in edge preserving smoothing are anisotropic diffusion [1], [2], [3], bilateral filtering [4], [5], mean shift filtering [6], and mode filtering [7]. A unified framework for these approaches has been formulated in [8]. This paper concentrates on the bilateral and mean shift filter paradigm, theoretically founded on kernel density estimation and mode finding.

Kernel density estimation methods use a continuous and convex kernel function to generate a continuous density estimate from a finite and usually small set of data samples. The kernel function is defined based on a set of scale or bandwidth parameters, controlling the amount of smoothing. Scale selection is a critical, yet not completely solved issue in kernel density estimation. In the context of computer vision applications, relevant papers addressing the problem lately are [9][10]. No matter what performance criterion is used, the optimal scale finds the desired trade-off between maximum use of inlaying data samples on one hand and outlier data rejection on the other hand. Inspired from multiscale approaches, in our previous work [11] we proposed a generalization of both bilateral and mean shift filters, called multiscale mode filter, as a means of obtaining a better compromise in this trade-off and we found some examples where the concept leads to performance improvement. Inherently, mode freedom in design means also more possibilities to explore and compare. In this paper we report results of our more recent work in exploring the potential of the MSMF approach in image filtering and segmentation.

The remaining of the paper is organized as follows. In Section II, we give a brief review of the kernel density estimation methods, the bilateral and mean shift filters and show how the mean shift algorithm can be used for robust clustering. The MSMF is introduced in Section III, while the new experiments are included in Section IV. Some concluding remarks and proposal of future work are left for the last section.

II. MODE FINDING FOR IMAGE FILTERING AND SEGMENTATION

A common feature of the bilateral and mean shift filters is the use of an extended analysis space, joining both spatial data, that is, pixel coordinates - in the case of static images - and range data, like pixel color vectors. Denote spatial coordinates by a vector with index "*s*":

$$\mathbf{x}_s = \begin{bmatrix} x \\ y \end{bmatrix} \tag{1}$$

[1] Facultatea de Electronică şi Telecomunicaţii, Departamentul Comunicaţii Bd. V. Pârvan Nr. 2, 300223 Timişoara, e-mail vasile.gui@etc.upt.ro
[2] University of Applied Sciences, School of Engineering, Kotkaantie 1, FI-90250 Oulu, e-mail jyrkila@oamk.fi

For colour images using RGB space, the range vector will be indexed by „r" and expressed as:

$$\mathbf{x}_r = \mathbf{f}(\mathbf{x}_s) = \begin{bmatrix} r(\mathbf{x}_s) \\ g(\mathbf{x}_s) \\ b(\mathbf{x}_s) \end{bmatrix} \quad (2)$$

A joint domain pixel data is then:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_s \\ \mathbf{f}(\mathbf{x}_s) \end{bmatrix} = \begin{bmatrix} x \\ y \\ r(x,y) \\ g(x,y) \\ b(x,y) \end{bmatrix}. \quad (3)$$

Since generally space and range data have different scales, data is supposed to be conveniently normalized prior to joint domain representation. A 2D image consisting of N pixels is represented by the set of 5D vectors: $\{\mathbf{x}_i\}$, $i = 1,2,\ldots,N$. Let $H(\mathbf{x}_1,\mathbf{x}_2)$ be a function measuring the similarity of two vectors, $\mathbf{x}_1$, $\mathbf{x}_2$. The bilateral filter response at pixel $\mathbf{x}_c$, is defined by an equation of the form:

$$\mathbf{y}_{cr} = \mathbf{f}(\mathbf{y}_{cs}) = \frac{\sum_{i=1}^{N} \mathbf{x}_{ir} H(\mathbf{x}_c, \mathbf{x}_i)}{\sum_{i=1}^{N} H(\mathbf{x}_c, \mathbf{x}_i)}. \quad (4)$$

Clearly, $\mathbf{y}_{cr}$, the output range vector at location $\mathbf{x}_{cs} = \mathbf{y}_{cs}$ is a weighted sum of input image range vectors, $\mathbf{x}_{ir}$, with weights defined by the similarity to the currently processed input data, measured by the function $H()$. The sum at the denominator is a normalization factor, needed to make weights add up to 1 in order to preserve the mean of each component of the range data. In the paper of Tomasi and Manduchi, the similarity function is the product of two functions, defining spatial similarity and range (colour) similarity with the pixel currently being processed:

$$H(\mathbf{x}_c, \mathbf{x}_i) = H_s(\mathbf{x}_{cs}, \mathbf{x}_{is}; h_s) H_r(\mathbf{x}_{cr}, \mathbf{x}_{ir}; h_r) \quad (5)$$

In the equation above, $\mathbf{x}_{cs}$ and $\mathbf{x}_{cr}$ are the spatial and respectively the range components of the currently filtered data vector, $\mathbf{x}_c$, while $\mathbf{x}_{is}$ $\mathbf{x}_{ir}$, denote the same components of another data vector, $\mathbf{x}_i$. Two parameters, h_s and h_r are used to scale the similarity functions. With proper data normalization, a single scale parameter may be used. To have a high influence on the currently computed output image, an input image pixel needs to be similar in both *location* and *value*. This is why bilateral filters are able to effectively smooth the image without blurring edges. Even details with high contrast are preserved, despite low spatial extent, since only pixels similar to the currently processed pixel wil be given high weights. An often used similarity function is the Gaussian function:

$$H(\mathbf{x}_1, \mathbf{x}_2) = \exp\left(-\frac{1}{2} \left\| \frac{\mathbf{x}_1 - \mathbf{x}_2}{h} \right\|^2 \right). \quad (6)$$

A mean shift filter response at an input pixel, $\mathbf{x}_c$, is defined as the convergence point of the mean shift algorithm initialized with $\mathbf{x}_c$. The mean shift algorithm proposed by Fukunaga and Hostetler [12] is a Newton type optimization method [13], used to find local modes of the probability density in a data set. Therefore, the mean shift filter finds a nearby density mode in the feature space. Starting from the kernel density estimate at a data point, $\mathbf{x}$, with scale h and profile k,

$$\hat{p}_{k,h}(\mathbf{x}) = \frac{c_{k,h}}{N} \sum_{i=1}^{N} k\left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h} \right\|^2 \right), \quad (7)$$

the algorithm successively moves in the direction of the estimated density gradient. In the equation above, $k()$ is the kernel profile and $c_{k,h}$ a normalization constant. It can be shown [6] that the density gradient estimated with the kernel profile $k()$ is proportional to the mean shift vector, given by

$$\mathbf{m}(\mathbf{x}) = \frac{\sum_{i=1}^{n} \mathbf{x}_i g\left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h} \right\|^2 \right)}{\sum_{i=1}^{n} g\left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h} \right\|^2 \right)} - \mathbf{x}, \quad (8)$$

where $g(x) = -k'(x)$. The mean shift algorithm can be summarized as follows:

1. Set the current result as the current input pixel: $\mathbf{y}_0 = \mathbf{x}_c$.
2. Compute the next value of the current result:

$$\mathbf{y}_{j+1} = \frac{\sum_{i=1}^{N} \mathbf{x}_i g\left(\left\| \frac{\mathbf{y}_j - \mathbf{x}_i}{h} \right\|^2 \right)}{\sum_{i=1}^{N} g\left(\left\| \frac{\mathbf{y}_j - \mathbf{x}_i}{h} \right\|^2 \right)}, \quad (9)$$

$$j = 1,2,\ldots$$

until convergence has been reached, that is the mean shift vector norm drops under a small threshold:

$$\left\| \mathbf{y}_{j+1} - \mathbf{y}_j \right\| = \left\| \mathbf{m} \right\| < \varepsilon .$$

The result of filtering $\mathbf{x}_c$ is $\mathbf{y}_{j+1}$. Only the range information of the result is stored in the output image. The similarity of the equation (8) iterated by the mean shift algorithm with the bilateral filter difned by equation (4) is obvious. Making $H() = g()$ the bilateral filter is equivalent to one step of the mean shift filter.

The mean shift filtering algorithm can be easily used to obtain image segmentation by clustering all pixels converging to the same mode[14],[15]. What is additionally needed after the filter is a simple pixel linking process. Optionally, close modes and small regions may be merged with nearest neighbour clusters.

III. MULTISCALE MODE FILTER (MSMF)

In contrast with the mean shift filter, the MSMF works on a different scale at each iteration and the number of iterations, J, is predefined. For an anisotropic kernel, the MSMF is defined in Fig. 1

For each image pixel, $\mathbf{x}_c$, do:
1. Set the current result as the current input pixel:

$$\mathbf{y}_0 = \mathbf{x}_c.$$

2. for j = 0 to J-1 compute

$$\mathbf{y}_{j+1} = \frac{\sum_{i=1}^{N} \mathbf{x}_i g\left(\left\| \frac{\mathbf{y}_j - \mathbf{x}_i}{h_j} \right\|^2\right)}{\sum_{i=1}^{N} g\left(\left\| \frac{\mathbf{y}_j - \mathbf{x}_i}{h_j} \right\|^2\right)}, \qquad (10)$$

3. Set the final result as $\mathbf{f}(\mathbf{y}_c) = \mathbf{f}(\mathbf{y}_J)$.

Fig. 1. Multiscale mode filter algorithm

Note the similarity of equation (10) with the variable bandwidth mean shift filter [10]. The important difference is that here the scale parameter changes with the iteration index, j, not with the data sample, index, i. According to the value of J and the choice of scales h_j, several multiscale mode filters can be designed. The case $J = 1$ corresponds to a conventional bilateral filter. The case $h_j = h_0$, for all j, with J sufficiently high, corresponds to the conventional mean shift filter. A monotonically decreasing set of scales $h_{j+1} < h_j$ for any $j < J$ is the main case motivating the proposed generalization. The highest scale, h_1, defines the degree of smoothing of the density field needed to clean out spurious local maxima, while the final scale of analysis, h_J, is supposed to be obtained by one of the many existing techniques described in the literature [9], [10].

Our purpose in using a decreasing set of scales is twofold. On one hand, we want to reduce the excessive influence of the range data of the processed pixel on the result. When images are corrupted by heavy noise, the current pixel may be an outlier and the result may be severely offset. If the first iterations are done at a larger scale, the influence of the other pixels is increased. This effect can be augmented by using smaller windows and space scale parameters in the first part of the filtering scenario. On the other hand, we want to reduce the chances of the algorithm to be trapped into spurious local maxima of the density. As experiments with real data have shown, given a desired scale of analysis, the probability density function often has several local maxima close to one another. This is particularly true for ramp edges, with uniform densities, corrupted by noise. Since the mean shift is a gradient ascent type of algorithm, it may be trapped in such a spurious density maximum point. The event is more likely to happen at low signal to noise ratios. If a large scale is used to smooth the estimated pdf, spurious local modes can be removed at the expense of shifting the locations of the maxima, when the distributions are not symmetrical. This effect is illustrated in Fig. 2.

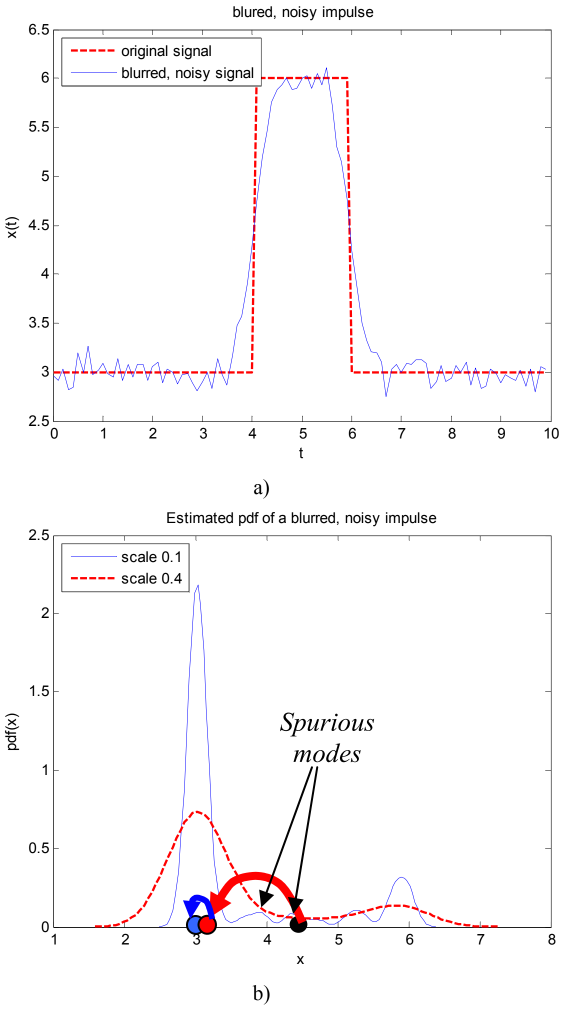


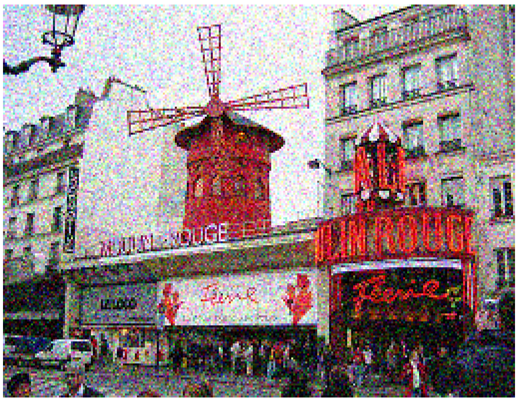
Fig. 2. a) 1D blurred edge with added noise;
b) Estimated pdf of the signal at two different scales and multiscale mode finding with starting point x.

A 1D blurred impulse signal with added noise is shown in Fig. 2a). The probability density function, pdf, estimated at two different scales and the multiscale mode seeking are illustrated in Fig. 2b). The algorithm starts the mode seeking on the large scale (red dotted line) and the large scale mode is found first. Then the search is continued on the finer scale (blue, cntinuous line) and mode location (the real mode is x = 3) is adjusted more precisely in the second step of the MSMF. A closer, spurious mode would have been reached by searching directly at the final scale. The conventional mean shift filter working on the final scale (0.1) would generate an impulse with jagged edges, as proved in our previous experiments [11]. Run at the lager scale, the conventional mean shift filter generates a rectangular response, but with the amplitude underestimated, as a result of excessive influence of the data samples in the tails of the distribution. Again, this has been demonstrated in our previous paper [11], along with experiments on colour images. In the next section, we describe new experiments with image filtering and segmentation tasks.

IV. EXPERIMENTS

In the first experiment, we show that the MSMF can be used effectively to clean noise from images. In Fig. 3, we show comparative results of the MSMF with Gaussian range kernel parameters $h_1 = 45$, $h_J = 20$, $J = 8$ and equal steps. The space kernel scale parameter was kept fixed to 12 and the space half window was incremented at each step, from 1 to 8. The mean shift filter was run with range parameter h = 20, space scale 12 and halfwindow 8. The original image was corrupted with white zero mean Gaussian noise. The original image is shwon in Fig. 3a), while the noisy input image with a PSNR of 15.9481dB is shown in Fig. 3b). In Fig. 3c), we reproduce the result of the mean shift filter on the noisy image. This image has a PSNR of 16.7541dB and the visual quality is not too much changed. In fact, the conventional mean shift filter is not particularly effective at cleaning heavy noise. The result of the MSMF is shown in Fig. 3d. The PSNR of this image is 17.0791 dB. Although the gain in PSNR is not impressive the visual effect is more convincing.

a)



b)

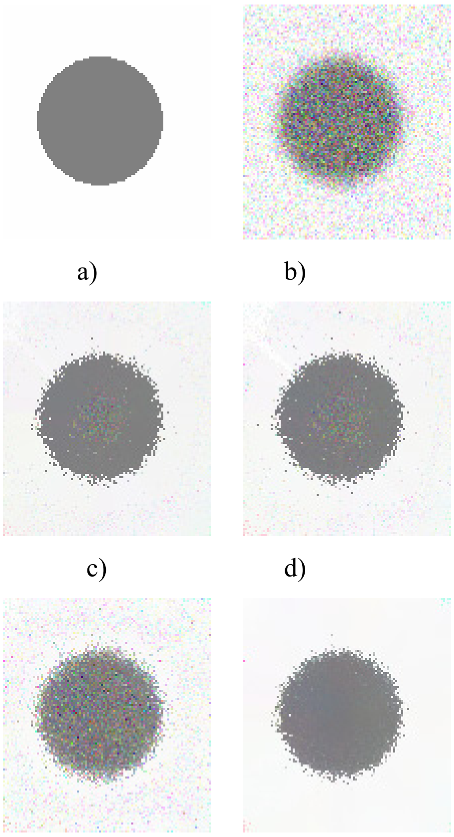
c)

d)

Figure 3 – Comparative results on a noisy image. a) original image; b) noisy image; c) result of the mean shift filter with scale h = 20; d) result of the MSMF with $h_1 = 45$, $h_J = 20$, $J = 8$.

In the MSMF processed image from Fig. 8d), the noise is almost completely cleaned. However the improvement in PSNR is less dramatic, as the additional smoothing, which is actually useful (for example for subsequent segmentation), is penalized as error in the PSNR criterion.

In a second experiment, we used the artificial test image, representing a circular shaped constant grey spot on a constant white background, shown in Fig. 4a). The image was blurred by a Gaussian filter with scale sigma=4 and a 21×21 window, then zero mean white, Gaussian noise was added. The resulting blurred, noisy image is shown in Fig. 4b). This image was filtered by both the conventional mean shift filter and the MSMF, with the same parameters as in the first test. The conventional filter was run at three different range scales: the highest scale used by the MSMF, the average scale of the MSMF and the lowest, final scale of the MSMF. The results of the conventional mean shift filters are shown in Fig. 4c), Fig. 4d) and Fig. 4e). The results of the MSMF are shown in Fig. 4f).



a) b) c) d) e) f)

Fig. 4. a) Original test image; b) blurred noisy test image; c),d),e) mean shift restored image with scales 45, 32 and 20; f) MSMF restored image, with scales $h_1 = 45$, $h_J = 20$, $J = 8$.

The PSNR of the images in Fig. 4b) to 4f) are given in Table 1.

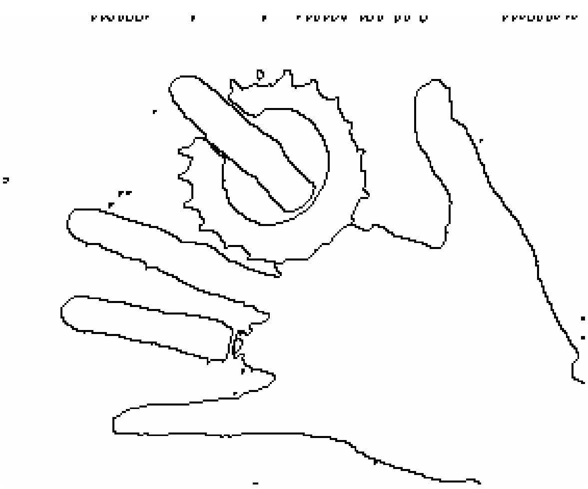
Table1.

	b)	c)	d)	e)	f)
PSNR	33.43	46.77	46.50	42.25	53.57

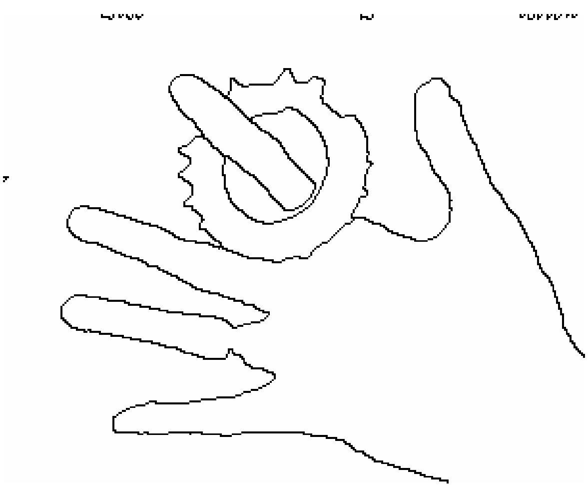
Clearly, the MSMF (f) outperformed the conventional mean shift filter at all scales.

In a second group of experiments, we tested the effectiveness of the MSMF in image segmentation. The MSMF and the conventional mean shift filters were run with the same parameters as in the first experiment on two images with rather different features. Filtering was followed by a simple pixel linking process with the linking threshold set equal to the final filtering scale for both the MSMF and the conventional mean shift. The results are shown in Fig. 5 and in Fig. 6. In both cases, the MSMF segmentation generates a simpler and more meaningful result. MSMF borders are somewhat smoother and less small regions result in this case.

a)



b)

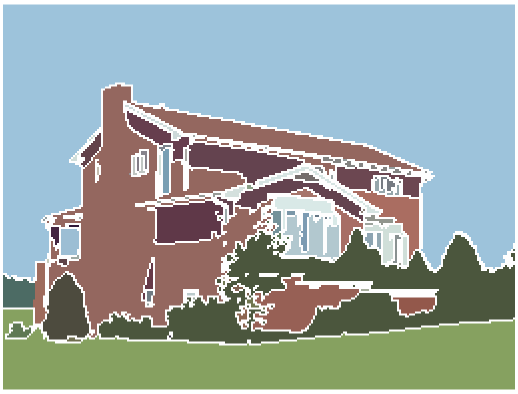


c)

Fig. 5. a) Original hand image; b) mean shift segmentation borders; c) MSMF segmentation borders.

a)

b)



c)

Fig. 6. a) Original house image; b) Mean shift segmentation with highlighted borders; c) MSMF segmentation with highlighted borders;

V. DISSCUSSION AND CONCLUSIONS

The experiments reported in this paper further confirm that the MSMF can outperform the conventional mean shift filter in both image filtering and segmentation tasks. Since the MSMF is more general, the result should not be considered surprising. However, the best design of the MSMF remains an open question. Like the design of the mean shift filter, in fact. If optimal scale selection in kernel density estimation is still a matter of research, after several decades of work, the best filtering scenario for the MSMF leaves more freedom, hence more problems to solve. By no means did we prove that the MSMF can always outperform a well designed mean shift filter. It is expected to be the case when the feature space has complex distributions with spurious local maxima that cannot be avoided by the conventional mean shift filter without using excessive smoothing, which is causing biased results. By gradually reducing the kernel scales, after avoiding the spurious maxima, the MSMF can obtain better location estimation. Additionally, using an increasing space scale is increasing the weights of nearby samples in the early filtering steps, causing more smoothing. In the final stages, the larger spatial scale combined with a narrower range scale allows higher resolution in estimating the mode location. We call this filtering strategy twisting and this is just one of the many capabilities of the MSMF, waiting to be explored in the future.

REFERENCES

[1] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion", *IEEE Trans., PAMI-12(7),*1990, pp. 629–639.

[2] G. Sapiro and D. L. Ringach. "Anisotropic diffusion of color images", *Proc. SPIE*, 2657, 1996, pp. 471–382.

[3] S. Osher, L. Rudin, "Feature-oriented image enhancement with shock filters", *SIAM J. Numerical Analysis*, 1990, pp. 919-940.

[4] C. Tomasi, R. Manduchi, "Bilateral filtering for gray and color images", *Int'l Conf. Comp. Vis., Bombay,* India, pp. 839-846, 1998.

[5] M. Elad, "On the Bilateral Filter andWays to Improve It," *IEEE Transactions on Image Processing*, Vol. 11, No. 10, p.1141, 2002.

[6] D.Comaniciu, P.Meer, "Mean shift: a robust approach toward feature space analysis", *IEEE Trans. Pattern Recognition & Machine Intelligence,Vol.24,No.5*,pp.603-619, 2002.

[7] J.Van de Weijer and R. van den Boomgaard, "Local mode filtering", in *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR), Vol. 2*, pp.428-433, 2001.

[8] D.Barash, "A Fundamental relationship between bilateral filtering, adaptive smoothing and the nonlinear diffusion equation", *IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.24, No.6*, pp.844-847, 2002.

[9] D. Comaniciu, "An Algorithm for Data-Driven Bandwidth Selection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2003, Vol. 25, No. 2, 2003.

[10] D. Comaniciu, V. Ramesh, P. Meer, " The variable bandwidth mean shift and data-driven scale selection", *IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.25, No.2*, pp.281-288, 2003.

[11] V. Gui, "Edge preserving smoothing by multiscale mode filtering". Accepted paper at the IEEE *EuropeeanConference on Signal Processing, EUSIPCO'08, Lausanne, August 25-29,* 2008.

[12] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition", *IEEE Trans. Info. Theory*, IT-21, pp32-40, 1975.

[13] C. Yang, R. Duraiswami, D. DeMenthon and L. Davis. Mean-Shift Analysis Using Quasi-Newton Methods. In *IEEE International Conference on Image Processing*, pages 447 - 450, vol.3, 2003.

[14] I. Gu, V. Gui, „Joint Space-Time-Range Mean Shift Based Image and Video Segmentation", Invited Paper in Y-J.Zhang, Ed., "Advances in Image and Video Segmentation" IRM Press, Hershey, London, Melbourne, Singapore, 2006, pp 113-140

[15] I. Gu, V. Gui, „Video Segmentation Using Joint Space-Time-Range Adaptive Mean Shift", Yueting Zhuang, Shiqiang Yang, Yong Rui, Qinming He (Eds.): Advances in Multimedia Information Processing - PCM 2006, 7th Pacific Rim Conference on Multimedia, Hangzhou, China, November 2-4, 2006, Proceedings. Lecture Notes in Computer Science 4261 Springer 2006, ISBN 3-540-48766-2, pp 740-748.