Locally Adaptive Smoothing Method Based on B-splines

G. Vidal-Cassanya
Physics and Applied Mathematics Deparment, University of Navarra, Pamplona, Spain
A. Muñoz-Barrutia
Oncology Division, Center for Applied Medical Research, Pamplona, Spain
Electrical, Electronics and Automative Department, University of Navarra, San Sebastián, Spain
M. Unser
Biomedical Imaging Group, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland

This paper presents a novel method for the edge-preserving smoothing of biomedical images. It is based on the convolution of the image with scaled B-splines. The size of the spline convolution kernel at each image position is adaptive and matched to the underlying image characteristics; i.e., wide splines for smooth regions and narrow ones for pixels belonging to edges. Consequently, the algorithm reduces image noise in homogeneous areas while, at the same time, preserving image structures such as edges or corners. We argue that the proposed adaptive filtering strategy provides a good balance between the improvement in the Signal to Noise Ratio (SNR) and perceptual quality. Our algorithm takes advantage of the unique convolution and factorization properties of B-splines. Specifically, the input signal is expressed in a B-spline basis; the inner product with a B-spline of arbitrary size is then computed by using an adequate combination of 1D integrations (preprocessing) and rescaled finite differences. The method is computationally efficient with a cost per pixel that is fixed and independent upon the scaling factor.
Keywords: B-spline, scale-variant smoothing, perceptual metrics.

1 INTRODUCTION

Biomedical images are often degraded by noise. This tends to produce artifacts that affect many image processing tasks such as segmentation, registration, visual rendition and feature extraction. Noise reduction is therefore of considerable interest in many applications.

The simplest and most widely-used method for denoising is to low-pass filter the image; for example, by moving averaging, Gaussian filtering or, possibly, Wiener filtering. These classical scale-invariant restoration techniques improve the SNR, but they have the drawback of introducing a significant amount of blur.

To overcome this limitation, a variety of local image feature-dependent adaptive filtering strategies have been developed during the past two decades. One of the first methods that appeared in the literature is the gradient inverse weighted smoothing described in (1).

In this paper, we present a generalization of Wang's method that use B-splines as smoothing kernels. The key component of the algorithm is the convolution at each location with a smoothing B-spline whose scale depends on the local characteristics of the image in the pixel neighborhood. In our case, the masks storing the scale values are calculated from the inverse image gradient and the more sophisticated Noise Visibility Function (NVF) (2).

We choose local convolution kernels that are rescaled (normalized) B- splines (with a unit integral); by tuning the spline degree, we are able to switch from a moving average to a weighted Gaussian-like smoothing. We also adopt a continuous domain formulation by interpolating the input image and expressing it into a B-spline basis. Thanks to the convolution properties of B-splines, we are then able to derive an exact and efficient scale-variant filter implementation; it uses a combination of moving sums and size-adjustable finite differences that are implemented efficiently by means of a look-up table.

2 ADAPTIVE FILTERING ALGORITHM

The continuous input signal $h(x)$ is represented by its spline interpolant which is in a one-to-one relation with the discrete input samples $h(k)$. Thus, we have $h(x) = c * \beta^{n_1}(x) = \sum_{k \in Z} c[k]\beta^{n_1}(x-k)$ where the sequence of interpolation coefficients $c[k]$ is calculated as shown in (3).

The output smoothed signal $f(x)$ at position x is calculated as the convolution of the input signal $h(x)$ with a B-spline kernel at scale a denoted by $\beta\left(\frac{x}{a}\right)$ which is given by

$$f(x) = c * \beta^{n_1} * \beta^{n_2}\left(\frac{x}{a}\right). \tag{1}$$

For details about the derivation of the algorithm, see (4).

The first step of the algorithm that provides the exact evaluation of the convolution given in equation (1) is to calculate the (n_2+1)-fold integral of the interpolation coefficients c_k

$$g = \Delta^{-(n_2+1)} * c, \tag{2}$$

where Δ^{-1} is the inverse finite-differences operator defined as $\Delta^{-1}(x) = \sum_{n \geq 0} \delta(x-n)$. The second step is to compute the inner products

$$f(b) = \sum_{k=0}^{n_2+1} \sum_{p=0}^{n_1+n_2+1} g(p+p_0) w_a(k,p) \tag{3}$$

where

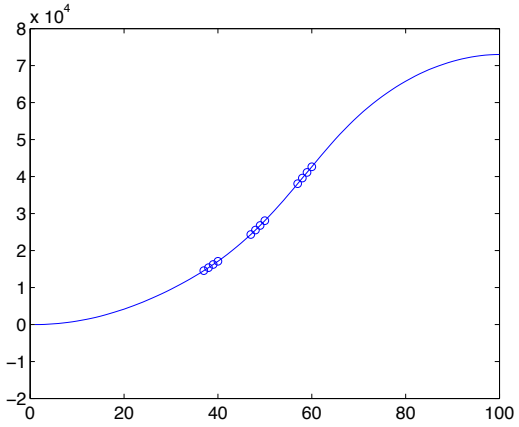


Figure 1: Spatial structure of the filter w_a to calculate $f(b)$ with $b = 50$. The filter weights form $(n_2+2) = 3$ 'clusters' of length $(n_1+n_2+2) = 4$, with an inter-cluster distance of $a = 10$. ('-' integral function g and 'o' coefficients of g ponderated by the filter weights).

- $w_a(k,p) = \frac{1}{a^{n_2}} q(k) \beta^{n_1+n_2+1}(\tau - ak - p - p_0)$ is a filter mask which we can store in a look-up table;
- $q(k) = \binom{n_2+1}{k}(-1)^k$ is the kth weighting coefficient of the (n_2+1)-finite difference at scale a;
- $p_0 = \left\lceil \frac{(a-2)(n_2+1)}{2} - ak - \frac{n_1+1}{2} \right\rceil + b$ is the first meaninful index in the sum over p that is computed using the compact-support property of B-splines;
- $\tau = \frac{(a-1)(n_2+1)}{2}$ is a time shift.

The operation described by equation (3) is equivalent to a discrete convolution with a modified 'a trous' filter. The scale-variant smoothing computation at each position b consists in filtering the coefficients $g(p)$ with (n_2+2) 'clusters' of length (n_1+n_2+2), each 'cluster' being separated from its neighbors by a distance a as shown in Figure 1.

A box diagram for a fast implementation of this scale-variant smoothing algorithm is shown in Figure 2. For each of the N scales in which we choose to quantize the mask, we compute the weights w_a and store them in a 3D look-up-take of dimensions $(n_2+2) \times (n_1+n_2+2) \times N$. In the initialization step, the B-spline expansion coefficients c_k of the sampled signal $h(x)$ are calculated and the running sum operator Δ^{-1} is applied (n_2+1)-times. The intermediate result $g(p)$ does not depend on the scale a. The filtered output $f(b)$ at position b is calculated with the modified 'a trous' filter whose coefficients are stored in the ith plane w_a of the 3D look-up-table. The values w_a and the intercluster distance for the filtering depend on a, but the computational complexity is constant and does not depend on a. Not taking into account the initialization step, we need $(n_2+2)(n_1+n_2+2)$ multiplications and $(n_2+1)(n_1+n_2+1)$ additions per point, corresponding to the filtering of g with the weights w_a.

We obtain a higher dimensional scheme by successive 1D processing along the various dimensions of the data. For the rest of the paper, we will restrict ourselves to 2D processing.

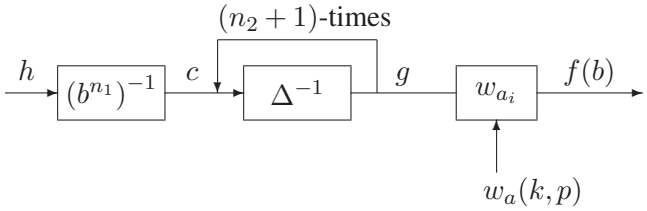


Figure 2: Schematic representation of the locally adaptive smoothing algorithm. $(b^{n_1})^{-1}$: Computation of the interpolation coefficients c. $\Delta^{-(n_2+1)}$: Calculation of the (n_2+1)-fold integral of c. $w_{a_i}(k,p)$: Look-up table calculation where $k \in [0, n_2+1]$, $p \in [0, n_1+n_2+1]$ and $a_i \in [a_1, a_N]$ with N the number of scales. w_{a_i}: Filtering with the weights calculated for each scale. $f(b)$: Smoothing output f at position b.

3 EXPERIMENTAL RESULTS

We propose now to realize a quantitative comparison of the performance of the scale-invariant denoising filtering using different B-spline kernels. We add Gaussian noise to Lena as to have a SNR $= 15$ dB (see Figure 8 (a)). In order to be able to evaluate the scale-invariant denoising with respect to a ground truth, i.e., original Lena image (see Figure 3). We perform the evaluation in terms of noise reduction (using the Signal to Noise Ratio (SNR) measure) and perceptual quality (using the Average Edge Width (AEW) and the Michelson contrast (MC) measure). The metrics above are defined as:

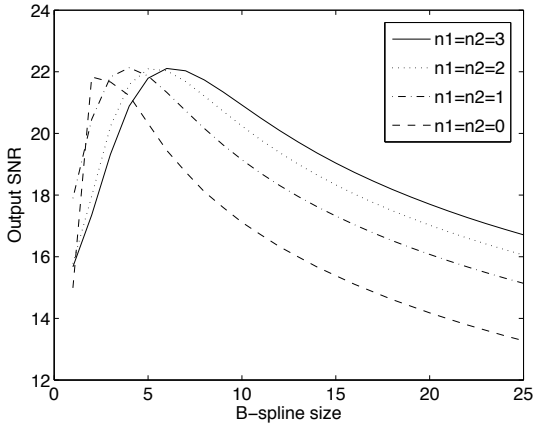
- SNR $= 10 \cdot \log 10 \left(\sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \frac{in(i,j)^2}{(in(i,j) - out(i,j))^2} \right)$ where in is the input image, out the denoised image and n_x and n_y the x- and y-dimensions.
- AEW $= \frac{\text{TotBM}}{\text{NbEdges}}$ where TotBM is the total number of pixels in the image that belongs to a edge and NbEdges is the total number of edges in the image. For more information in how to compute this quantity we refer to (5).
- MC $= \frac{\max(out) - \min(out)}{\max(out) + \min(out)}$ where $\max(out)$ ($\min(out)$) is the maximum (minimum) value over all the pixels of the image out.

The SNR of the scale-invariant smoothed image versus the B-spline size is shown in Figure 4. For the correspondence between scale factor and B-spline size note that the support of a B-spline of degree n and scaled by a factor a is $a(n+1)$. The peak SNR obtained for the best compromise between denoising and edge blurring is 22 dB independly of the B-spline degree but it is reached for a higher B-spline size as the degree increases. From the slower falling of the cubic B-spline SNR curve with respect to the rest we can deduce that cubic B-splines offers a good compromise between noise reduction and boundary sharpness conservation for a scale-invariant smoothing with a kernel of a given size.

Figure 3: Original Lena image.

The AEW of the scale-invariant smoothed image versus the B-spline size is shown in Figure 5 (a). We observe a linear increment of the AEW measure with respect to the B-spline size for all the B-spline kernels. The cubic B-spline curve has a smaller slope that the rest of the considered kernels. Thus, cubic B-splines results in less blurring than their counterparts for a given B-spline size.

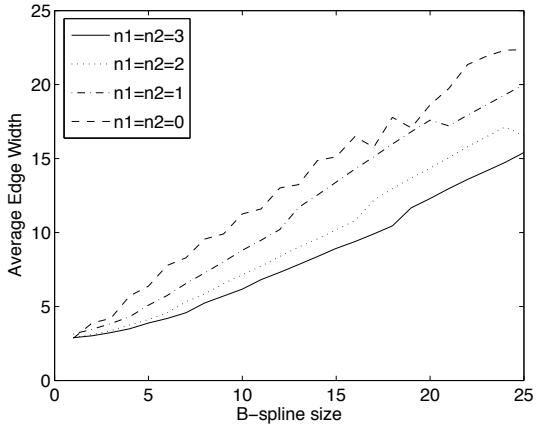
The MC of the scale-invariant smoothed image versus the B-spline size is shown in Figure 5 (b). The contrast decreases steeply for the convolution with a B-spline of a smaller size than the corresponding to the peak SNR output. It seems that the pronounced contrast loss is due to the dramatic noise reduction caused by the convolution with incremental size kernels. The linear decrement in contrast observed after that point seems to be caused by the linear increment of the blurring with the B-spline size. Moreover, the cubic B-spline curve keeps itself over the rest for all the sizes. So, we observe from another point of view how the cubic B-splines results in images of higher quality for a given B-spline size.



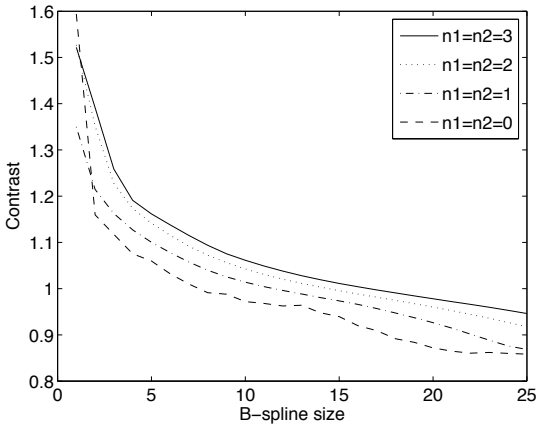
(a)

Figure 4: For a noisy Lena with SNR $= 15$ dB: SNR vs. B-spline size.

As the three quality measures agree in indicating that the use of cubic B-splines results in images of higher quality for a given B-spline size, we decide to use cubic B-splines for the interpolation and convolution in the rest of the experiments.



(a)



(b)

Figure 5: For a noisy Lena with SNR $= 15$ dB: (a) AEW vs. B-spline size. (b) MC vs. B-spline size.

Next, we will compare the performance of the scale-independent and scale-dependent smoothing implemented as described in Section 2. We have chosen to use the following masks: the binarized inverted gradient and the binarized inverted NVF function. For a detailed description of the NVF function consult (2). The inverted gradient and NVF function are binarized using the unimodal background symmetry method. The masks computed for the noisy Lena are shown in Figure 8 (b) and (c). We can observe from them how the size of the spline convolution kernel adapt somehow to the underlying image characteristics. Wider splines are applied to smooth regions and narrower ones to edges.

We show in Table 1 the quantitative comparison of the denoising of Lena with SNR $= 15$ dB (see Figure 8 (a)) using the scale-invariant filter with a scale 2.0 and the gradient and NVF masks with scales $[0.5, 2.0]$. We observe that the scale-variant filtering approach (for both masks) have a gain of 0.2 dB in SNR, 0.345 in MC and a AEW 0.5 pixels smaller that the corresponding scale-invariant filtering. Figure 8 shows the visual result. We observe that the scale-dependent smoothing reduces image noise in homogeneous areas, while preserving edges much better than the scale-invariant smoothing.

	Original	Uniform 2.0	Gradient $[0.5, 2.0]$	NVF $[0.5, 2.0]$
SNR (dBs)	21.737	14.987	21.949	21.843
AEW (pixels)	5.245	2.857	4.868	4.708
MC	1.594	1.094	1.354	1.354

Table 1: SNR, AEW and MC measures for the smoothed noisy Lena with SNR $= 15$ dB.

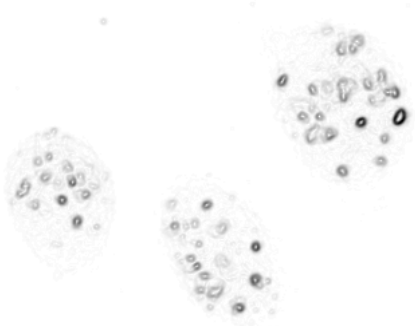
	Original	Uniform 3.0	Gradient $[0.5, 3.0]$	NVF $[0.5, 3.0]$	NVF $[0.5, 1.0, 3.0]$
SNR (dBs)	11.912	Inf	17.098	24.544	19.868
AEW (pixels)	12.061	6.286	8.658	6.660	7.333
MC	0.98443	0.98096	0.98377	0.98624	0.98432

Table 2: SNR, AEW and MC measures for the smoothed noisy centromeres image.

To shown an example of the performance of the approach on a real biomedical image, we have applied our algorithm on a fluorescent microscopy image of chromosome centromeres (see Figure 9). The quantitative comparison between the three approaches (scale-invariant filter with a scale 3.0, and the gradient and NVF filter with scales $[0.5, 3.0]$) is shown in Table 2. In this case, we measure the SNR with respect to the original noisy centromeres image. It seems from the data collected in the table that the NVF method outperforms the other two: a higher SNR and MC, and a lower AEW value. The truth is that the NVF mask separates the background from the cells but does not distinguish the spots. In consequence, the NVF filtering does not clean very much from the noise in the inter-spot regions inside the nucleus. On the other hand, the gradient mask clearly segments the spots. So, the gradient method gives a better compromise between denoising and spot conservation. As expected, the scale-invariant filtering introduces much more blurring than the other two methods.

The key point here is that the masks we have implemented indicate where the contours and the fine structures are in the image but they do not give any clue on what is the size of the smooth regions. High values in the magnitude of the gradient refers mainly to the contours of the image. High values in the NVF function correspond not only to the contours but also to areas of the images with high to medium activity. The differences can be observed in Figure 6. We use this fact to point out the interest to introduce an estimation of local scale (equivalently, spatial frequency) into the method. For example, we have constructed a NVF mask that separates the three differentiated scale regions in the centromeres images: the background, the nucleus and the spots (see Figure 7 (a)). We have computed the corresponding NVF denoised centromeres image with scales $[0.5, 1.0, 3.0]$ which is included in Figure 7 (b) to facilitate a visual comparison with the other methods. We observe sharp spots and clean smooth regions. The quantitative comparison of the denoising with the tri-scaled NVF with respect to the binarized gradient method shows a gain of 1.8 dB in SNR, 0.0025 in MC, and a AEW 1.3 pixels smaller as collected in Table 3.

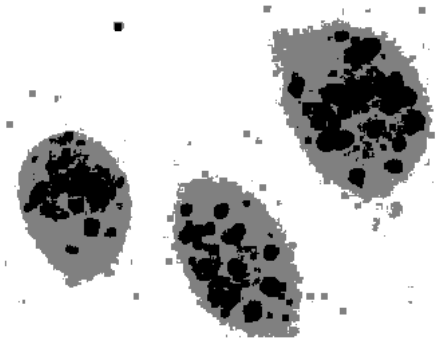
In consequence, we observe the convenience to introduce a measure of local scale into the method and incorporate such an information to the mask to be able to smartly control the degree of smoothing that is done in different regions of the image. Another weak point of the method is that the separable filtering we propose although fast is not quite equivalent to dilating the B-spline in 2D (only if the x- dilation is the same over the region covered by the y-dilation). For that reason we plan to extend the method to work for a non-separable tensor spline kernels.



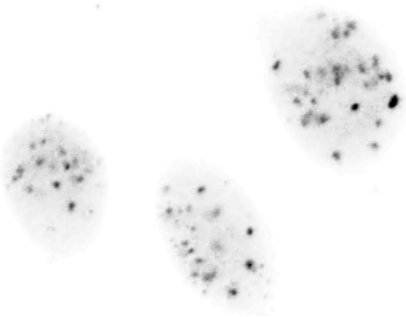
(a)

(b)

Figure 6: For the centromeres image: (a) Inverted gradient magnitude. (b) Inverted NVF function.



(a)



(b)

Figure 7: For the centromeres image: (a) Inverted tri-scale NVF mask. (b) Inverted NVF smoothed image with $[0.5, 1.0, 3.0]$. We have use cubic B-spline interpolation and convolution.

4 CONCLUSIONS

In summary, we have presented a novel adaptive B-spline based smoothing algorithm that is capable of reducing noise without degrading edges. The proposed scheme is general and flexible. It works for all spline degrees which allows one to easily switch from moving average to Gaussian-like filters of arbitrary sizes. It is also extendable for more sophisticated masks that are linear combination of B-splines and that can be rescaled in a signal-adaptive fashion. The overall operation count only depends on the degrees of the chosen B-splines but it is independent of the value of the local scale. The results we show are promising and a further improvement of the method could be obtained in a future work by the introduction of a measure of local scale and by its extension to use 2D non-separable spline kernels.

ACKNOWLEDGEMENTS

A. Muñoz-Barrutia is funded by a Ramon y Cajal fellowship of the Spanish Ministry of Science and Technology.

REFERENCES

[1] D. C. C. Wang, A. H. Vagnucci, and C. C. Li, "Gradient Inverse Weighted Smoothing Scheme and the Evaluation of its Performance", in *Computer Graphics Image Proc.*, vol. 15, pp. 167-181, 1981.

[2] S. Voloshynovskiy, A. Herrigel, N. Baumgaertner, and T. Pun, "A Stochastic Approach to Content Adaptive Digital Image Watermarking", *in Int. Workshop on Inf. Hiding, Lecture Notes in Computer Science*, vol. 1768, pp. 212-236, Oct. 1999.

[3] M. Unser, "Splines, A Perfect Fit for Signal and Image Processing", in *IEEE Signal Proc. Mag.*, vol. 16, no. 6, pp. 22-38, Nov. 1999.

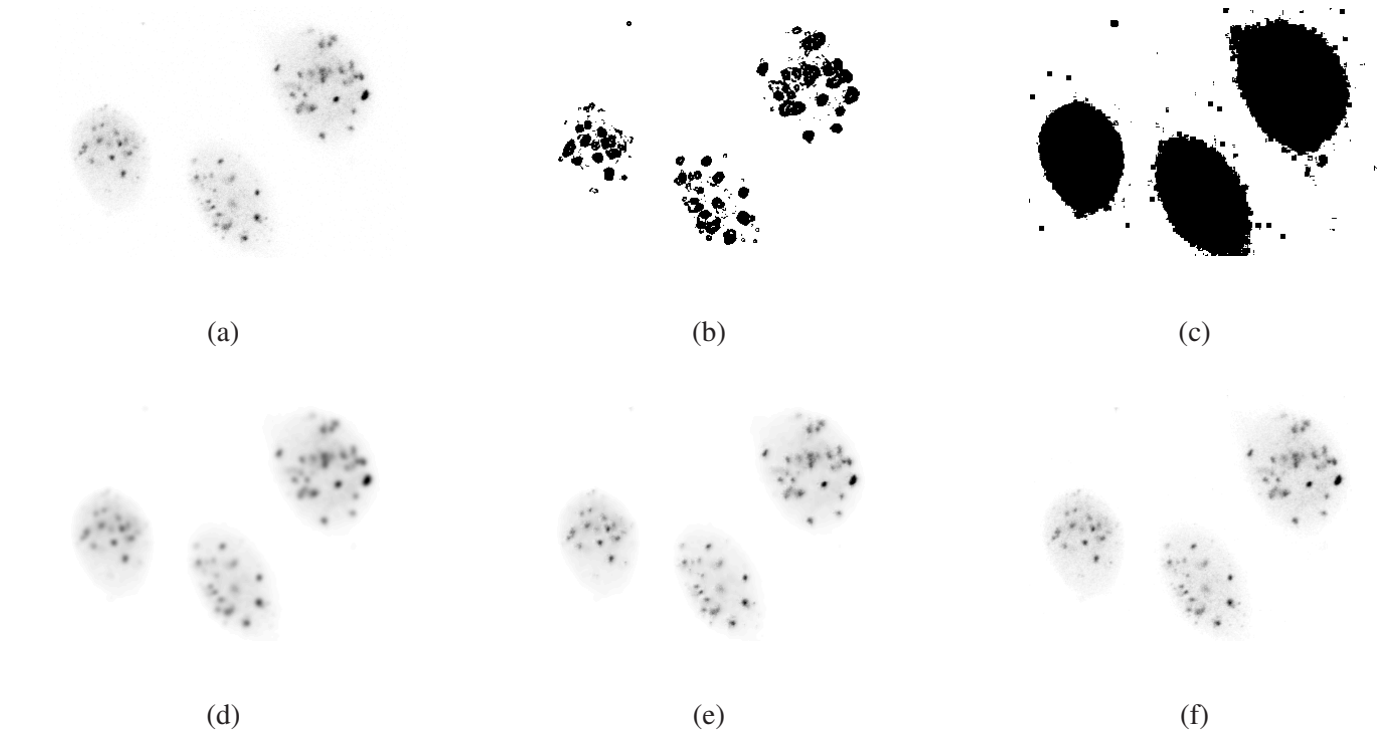
[4] A. Muñoz-Barrutia, R. Ertlé, M. Unser, "Continuous Wavelet Transform with Arbitrary Scales and $O(N)$ Complexity", *Signal Proc.*, vol. 82, no. 5, pp 749-757, May 2002.

[5] P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi, "A No-reference Perceptual Blur Metric", *in Proc. of the Int. Conf. on Image Proc.*, vol. 3, pp 57-60, Sept. 22-25, 2002.

(a) (b) (c)

(d) (e) (f)

Figure 8: (a) Noisy Lena with a SNR $= 15$ dB. (b) Inverted gradient mask. (c) Inverted NVF mask. (d) Space independent smoothed image for $a_2 = 2.0$. (e) Inverted gradient smoothed image with $[a_1, a_2] = [0.5, 2.0]$. (f) NVF smoothed image with $[0.5, 2.0]$. We have use cubic B-spline interpolation and convolution.



(a) (b) (c)

(d) (e) (f)

Figure 9: (a) Inverted original noisy centromeres image. (b) Inverted gradient mask. (c) Inverted NVF mask. (d) Inverted space independent smoothed image for $a_2 = 3.0$. (e) Inverted gradient smoothed image with $[a_1, a_2] = [0.5, 3.0]$. (f) Inverted NVF smoothed image with $[0.5, 3.0]$. We have use cubic B-spline interpolation and convolution.