Graph-based Image Sharpening Filter Applied to Image Denoising

Guillaume Noel, Karim Djouani, and Yskandar Hamam
French South African Institute of Technology,
Tshwane University of Technology
noelg, djouanik, hamama@tut.ac.za

Abstract

The present paper focuses on a new class of mesh filter for grayscale images, called grid smoothing filter. The framework presented considers an image as a sampling grid associated to a set of gray levels. Furthermore, the sampling grid is seen as mesh composed by vertices and edges, the number of vertices being equal to the number of pixels in the image. Embedding the mesh in a 2D Euclidian space, each vertex has two spatial coordinates and one attribute, the value of the gray level. Starting from the classical formulation of Laplacian mesh filtering, a novel objective function is introduced. The minimization of the objective function leads to new spatial coordinates for the vertices in the mesh. A reconstruction mechanism is then applied to the non-uniform mesh to reconstruct a grayscale image. Whereas the Laplacian mesh filter aims at smoothing an image, the grid smoothing tends at sharpening the edges of the image. The grid smoothing framework is applied to image enhancement in this paper.

Keywords: *mesh filtering, grid smoothing, image enhancement.*

1. Introduction

Image enhancement techniques seek at improving the appearance of an image without referring to a specific model for the degradation process, while *image restoration* relies on the knowledge of a degradation model [1]. The framework presented in the paper belongs to the *image enhancement* domain. The two main forms of image quality degradation are blur (loss of sharpness) and noise. Methods have been developed and address both types of degradation, either in a pixel-representation of the image or a mesh-representation. The pixel-representation considers an image to be a matrix of pixels. The edge enhancement methods modify the gray level of the pixels to improve the quality of the image. In a mesh representation, the image is represented by nodes (or vertices) and edges. Using the pixel-representation of an image, adaptive bilateral filters and quadratic-weighted median filters were applied with success for edge enhancement [1], [2]. These methods are based on filtering and may induce overshooting of the edges. Moreover, the parameters of the filters, in the case of the adaptive bilateral filter, are tuned according to a training dataset, narrowing the scalability to various applications. In any case, the performance of these methods is bounded by the use of the pixel-representation of the image. In low resolution images, the shape of the object does not systematically match the matrix and may lead to severe distortion of the original shape. For example, a clear straight line whose orientation is *45* degrees is represented by a staircase-like line. Image enhancement techniques, such as super-resolution

[3], [4], [5] tackle the issue of the misrepresentation of an image by enhancing the resolution of the image. In the number of pixels is to stay the same, the paradigm image and matrix may be overcome by the use of a mesh or graph-based representation of an image. The mesh defined on the image may be feature-sensitive or insensitive. In the feature-insensitive mesh, the definition of the mesh does not take into account the gray levels. The edge enhancement relies on the filtering of the gray levels in the image [6] which may lead to the same issues as in the pixel-representation (overshooting,...). In the feature-sensitive mesh, the graph is defined according to the distribution of the gray levels in the image and the position of the mesh is adjusted to the objects present in the image to enhance the image [7]. The usual problems of this class of techniques are the computation cost (interpolation) and the large number of edges in the graph. Mesh creation techniques may be found in [8], [9] and [10]. The present paper presents a novel combination of feature-insensitive and feature-sensitive mesh approaches, which reduces the complexity of the definition of the mesh and improves the representation of the information in the image.

In the presented framework, an image is represented by a graph in which the nodes represent the pixels and the edges reflect the connectivity. The original graph (or grid) is a uniform grid composed by squares or triangles, depending on the connectivity chosen. The grid smoothing process modifies the coordinates of the nodes in the (x, y) plane while keeping the gray scale levels associated to the node unchanged. The grid smoothing relies on the minimisation of a cost function leading to a compression of the grid in the regions with large gradient values and a relaxation in the other regions. As a consequence, the new grid fits the objects in the image. The grid smoothing enhances the edges in the original image and does not modify the number of nodes. Noise removal techniques may be applied before the grid smoothing, depending on the properties of the original images. A new type of mesh smoothing is being used [11]. The mesh smoothing approach in [11] modifies the gray levels of the image while preserving the (x, y)coordinates of the nodes.

Section 2 of this paper presents the graph-based representation of an image while section 3 exposes the mathematical framework of the grid smoothing as well as the convergence properties. Section 4 focuses on the reconstruction mechanism and shows results of the grid smoothing on blurry images. When dealing with noisy images, smoothing of the image is required before the grid smoothing is applied. The complete image denoising chain is presented in section 5 as another application of grid smoothing. Conclusion and recommendations are underlined in section 6.

2. Graph-based image representation

Our input data is a graph *G=(V,E),* embedded in the 3D Euclidian space. Each edge *e* in *E* is an ordered pair (s, r) of vertices, where *s* (resp. *r*) is the sending (resp. receiving) end vertex of *e* [11]. To each vertex *v* is associated a triplet of real coordinatesx_v, y_v, z_v.
Let C_{ve} be the node-edge incidence matrix of the graph G, defined as:

$$C_{ve} = \begin{cases} 1 \text{ if } v \text{ is the sending end of edge } e \\ -1 \text{ if } v \text{ is the receiving end of edge } e \\ 0 \text{ otherwise} \end{cases}$$

In the rest of the paper, the node-edge matrix C_{ve} is also denoted C.

Table 1. Number of connections according to the connectivity

Number of nodes	Number of connections Connectivity 4	Number of connections Connectivity 8
2500	4900	9702
10000	19800	39402
90000	179400	358202

Table 2. Computation time according to the connectivity ($\theta = 0.005$)

	Computation time (s)			
	Connectivity 4		Connectivity 8	
	Per edge	Image	Per edge	Image
2500 nodes	1.2×10^{-5}	5.9×10^{-2}	1.5×10^{-5}	1.5×10^{-1}
10000 nodes	1.6×10^{-5}	3.3×10^{-1}	2.2×10^{-5}	8.9×10^{-1}
90000 nodes	2.8×10^{-5}	5.0	3.3×10^{-5}	1.2×10^{1}

Considering an image with *M* pixels, *X*, *Y* and *Z* respectively represent$[x_1, \ldots, x_M]^t$, $[y_1, \ldots, y_M]^t$, and $[z_1, \ldots, z_M]^t$. *X* and *Y* are at first uniformly distributed (coordinates of the pixels in the plane), while *Z* represents the gray level of the pixels. Each pixel in the image is numbered according to its column and then its rows. We define *L* as the number of edges in the graph. *C* is consequently a matrix with *L* rows and *M* columns.

The number of edges depends on the choice of the connectivity for the pixel. If the connectivity is equal to four, each pixel is connected to its four closest pixels. The initial grid is then composed of squares. If the connectivity is equal to eight, each pixel is connected to its eight closest pixels. The initial grid is then composed of triangles. The choice of the connectivity is important as it increases the size of the matrix C and consequently the computation time required for the grid smoothing (Tab.1, 2). An evaluation of L may be derived using the dimensions of the image. If L_x and L_y represent respectively the number of pixels along the x-axis and y-axis, we have $M = L_x \times L_y$. For a connectivity equals to 4, $L = 2M - L_x - L_y$ and $L = 4M - 3L_x - 2L_y - 2$ if the connectivity equals 8. Using the notation introduced before, it may be observed that the complexity of the algorithm is $L \times \log(L)$ (Fig.1). It may be explained by the complexity of the conjugate gradient with a stopping criterion ε in which the maximal number of iteration is bounded by $\alpha \times \log(L/\varepsilon)$, α being a constant. When using the high connectivity, the number of connections doubles as well as the computation time. The choice should be made according to the applications and the characteristics of the images. If an image includes thin lines which have to be preserved, the high connectivity should be used. For the other cases, the low connectivity gives satisfactory results.

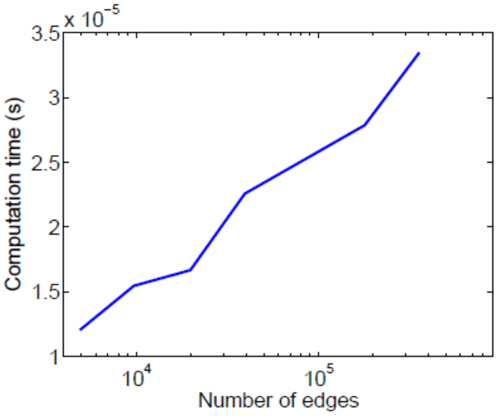


Figure 1. Computation time per edge in seconds

3. Optimization-based approach to grid smoothing

The present section introduces the framework for the grid smoothing. An extensive study of the convergence of the method as well as its application to satellite images may be found in [12], [14]. As depicted in Fig.2, the input of the grid smoothing (left) is a grayscale image and its output is a non-uniform mesh (right). Each vertex in the new mesh is associated with the same grayscale value as its corresponding pixel.

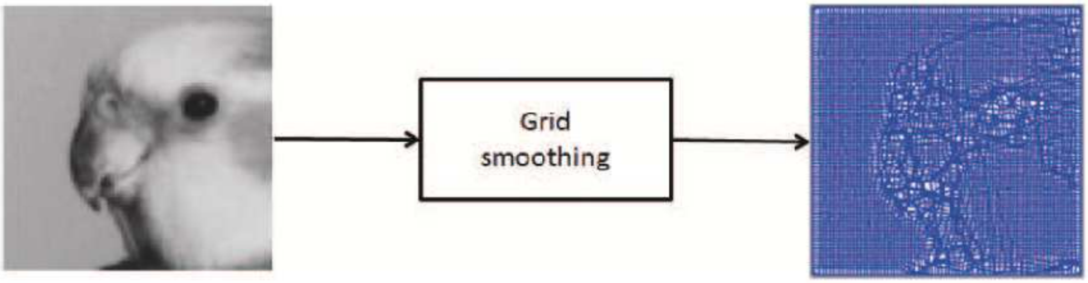


Figure 2. Synoptic of the grid smoothing

3.1. General framework

A cost function is introduced to fit the content of the image with the grid. The main idea is that the regions where the variance is small (low gradient) require fewer points than the regions with a large variance (large gradient). The grid smoothing techniques will move the points of the grid from small variance regions to large variance regions. To achieve this goal, a cost function J is defined as follows:

$$J = J_X + J_Y$$

where

$$J_X = \frac{1}{2}[(X-\hat{X})^t Q(X-\hat{X}) + \theta(X^t AX)]$$

and

$$J_Y = \frac{1}{2}[(Y-\hat{Y})^t Q(Y-\hat{Y}) + \theta(Y^t AY)]$$

with $A = C^t \Omega C$ and $\hat{X}$ (resp. $\hat{Y}$) represents the initial coordinates of X (resp. Y)**.** Ω is a diagonal matrix. The first term in the expression of the cost function is called the *attachment* as it penalises the value of the cost function if the coordinates are too far from the original values. It is introduced to avoid large movement in the grid [11]. θ is a real number and acts as a weighing factor between the terms of the cost function.

Using the notations introduced before, the diagonal elements of the matrix Ω are defined as follows:

$$\Omega_e = (z_s - z_r)^2$$

The matrix Ω represents the square of the difference in graylevels of each edge. We may remark that Ω and Q are square diagonal matrices which dimensions are respectively $L \times L$ and $M \times M$. If we consider the triangles defined in Fig.3, the minimization of J leads to the reduction of the areas of the triangle formed by two connected nodes (filled circles) and the projection of one of the point on the Z-axis (empty circle). We have indeed:

$$X^t AX = \sum_{l=1}^{L} (x_s - x_r)^2 (z_s - z_r)^2 = \sum_{l=1}^{L} B_x^2 h^2$$

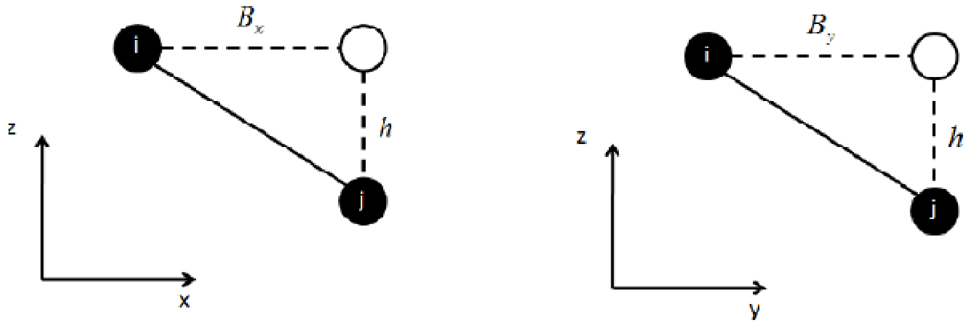


Figure 3. Geometric interpretation of the grid smoothing

As it may be seen in Fig.4 where the arrows represents the effect of the grid smoothing, by minimizing the areas of the triangles, the edges in the image act as attractors for the points in the grid. As a consequence, the edges are better defined in terms of location and steepness in the smoothed grid.

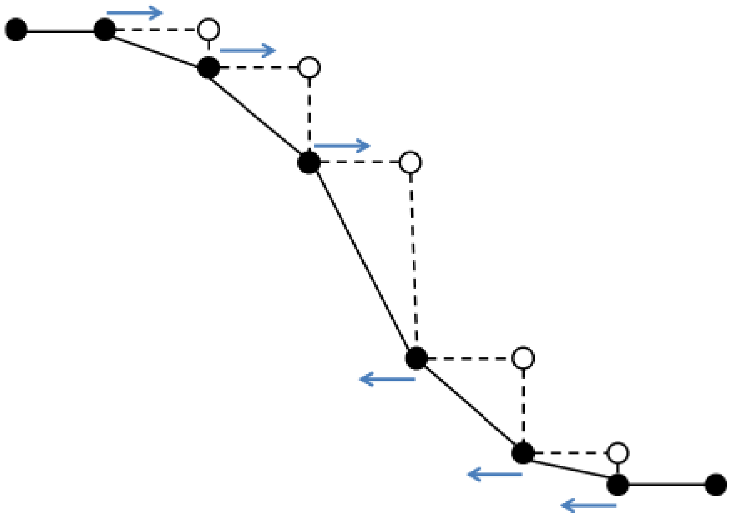


Figure 4. Effect of the grid smoothing

3.2. Convergence of the cost function with fixed points and attachment

The cost function with attachment results in a grid whose size might differ from the original grid size. A solution to conserve the original size is to fix the coordinates of the outer points of the grid.

Let the X coordinates be partitioned into two parts, variable coordinates 'x' and fixed coordinates 'a' giving

$$X = [x\ a]^t$$

Then the first order cost function without attachment is

$$J_x = \frac{1}{2}\left([(x-\hat{x})^t\ 0]Q\begin{bmatrix}(x-\hat{x})\\0\end{bmatrix} + \theta[x^t\ a^t\ ]\begin{bmatrix}C_x^t\\C_a^t\end{bmatrix}\Omega[C_x\ C_a]\begin{bmatrix}x\\a\end{bmatrix}\right)$$

Expanding the above equation gives

$$J_x = \frac{1}{2}[(x-\hat{x})^t Q_x(x-\hat{x}) + \theta x^t C_x^t \Omega C_x x + 2\theta x^t C_x^t \Omega C_a a + \theta a^t C_a^t \Omega C_a a]$$

The gradient of J_x with respect to x is

$$\nabla_x J_x = Q_x(x-\hat{x}) + \theta C_x^t \Omega C_x x + \theta x^t C_x^t \Omega C_a a$$

Setting the gradient to zero gives

$$x = -[Q_x + \theta C_x^t \Omega C_x]^{-1}[Q_x \hat{x} - \theta C_x^t \Omega C_a a]$$

This gives the exact solution for the coordinates x.
Let x_{n+1} and x_n be x at iterations $n+1$ and n then

$$x_{n+1} = x_n - \alpha_n \nabla_x J_x$$

The gradient of J_x at the point x_{n+1} is equal to

$$\nabla_x J_{x_{n+1}} = \nabla_x J_{x_n} - \alpha_n Q_x \nabla_x J_{x_n} - \alpha_n C_x^t \Omega C_x \nabla_x J_{x_n}$$

The optimal step condition may by expressed by $\nabla_x {J_{x_n}}^t . \nabla_x J_{x_{n+1}} = 0$. It leads to:

$$\alpha_n = \frac{\nabla J^t \nabla J}{\nabla J^t (\mathrm{Q_x} + \theta C_x^t \Omega C_x) \nabla J}$$

The experience shows that the convergence is quicker using the conjugate gradient descent with optimal step. A quadratic function may be expressed by:

$$J(x) = \frac{1}{2} x^t A x + b^t x + c$$

where *A* is a definite positive matrix [13].
At each iteration, $x_{n+1} = x_n - \alpha_n d_n$, where α_n is the step and d_n, is the direction of descent. The direction and the step are calculated at each iteration. By assimilation with the cost function with fixed points and attachment, we have $A = Q_x + \theta C_x^t \Omega C_x$ and $b = \hat{x}^t Q_x - \theta a^t C_a^t \Omega \mathrm{C_x}$. The step at the iteration *n* may be computed by:

$$\alpha_n = \frac{(b - Q_x \hat{x})^t (b - Q_x \hat{x})}{d_n^t Q_x d_n}$$

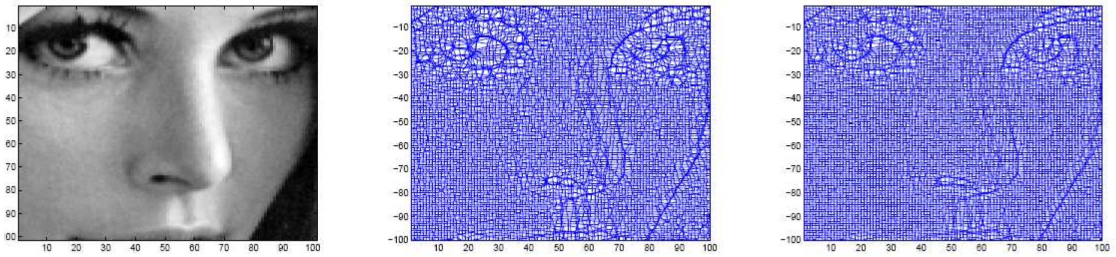
and the direction at iteration *n+1* is equal to:

$$d_{n+1} = e_{n+1} + \frac{{e_{n+1}}^t e_{n+1}}{{e_n}^t e_n} d_n$$

where $e_{n+1} = e_n - \alpha_n Q_x d_n$ and $e_1 = b - Q_x \hat{x}$.

3.3. Stopping criterion

As mentioned earlier, for large scale problem, the minimisation uses a gradient descent algorithm as it is computationally expensive to inverse very large matrices. Three gradient methods are used for the simulation, namely the steepest descent gradient with fixed step, the steepest descent gradient with optimal step and the conjugate gradient with optimal step. The descent gradient methods are iterative process and require a stopping criterion ε to stop the iterations. The chosen criterion is the simulation is the norm of the gradient ∇J. The iterative process continues while $\nabla J^t \nabla J \geq \varepsilon$. When it is possible, the comparison between the exact coordinates given by the inversion of the matrix and the result of the gradient descent algorithm is small and is of the order of ε. For example, if $\varepsilon = 10^{-3}$, the difference between the exact coordinates (matrix inversion) and the coordinates obtained through the gradient descent is 10^{-3} of the width of a pixel. The conjugate gradient descent is faster for any ε.



a) *Lena* (details) b) Grid ($\theta = 1$) c) Grid ($\theta = 0.05$)

Figure 5. Results of the grid smoothing according to θ

4. Image reconstruction

4.1. General Framework

The output of the grid smoothing is a non-uniform mesh denoted $\mathcal{M}_g$ in which the number of vertices is equal to the initial number of pixel in the image. The purpose of this section is to devise a mechanism to map the gray values associated to $\mathcal{M}_g$ into a set of gray values associated to a uniform grid $\mathcal{M}_s$. The reconstruction may be seen as an application $\mathcal{R}$ defined by:

$$\mathcal{R}: \big(x, y, I(x, y)\big) \rightarrow \big(i, j, I(i, j)\big)$$

Fig.5 represents $\mathcal{M}_g$ in red and $\mathcal{M}_s$ in blue for an image representing a white square (pixel 5 to 15) over a black background. It may be seen that $\mathcal{M}_g$ and $\mathcal{M}_s$ overlaps for most of the mesh. It is then straightforward to assign gray values in this case. However, at the edge proximity (rows/columns 5 and 15), the two grid does not overlap. The image reconstruction mechanism uses two rules to assign gray value to these vertices.

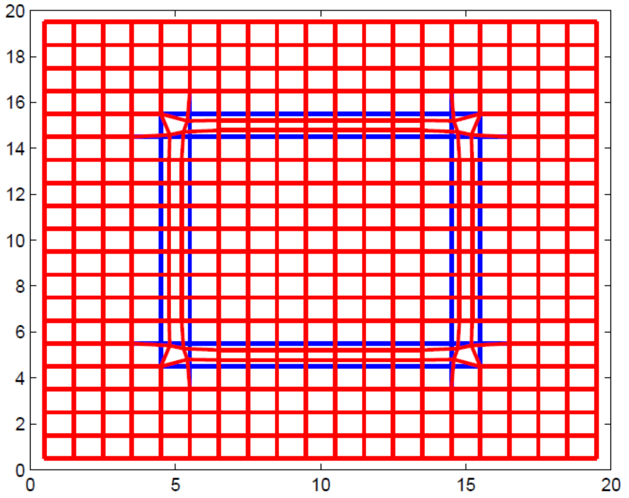


Figure 6. Output of the grid smoothing (red) and resampling grid (blue)

For each vertex v in $\mathcal{M}_s$ it is possible to find the three spatially closest vertices denoted i, j, and k in $\mathcal{M}_g$. From this point, two options are available: either we map the gray value of the closest node in $\mathcal{M}_g$ onto v or we map the average of the gray values. The first option is named *Nearest neighbor mapping* while the second option is called *Facet mapping*.

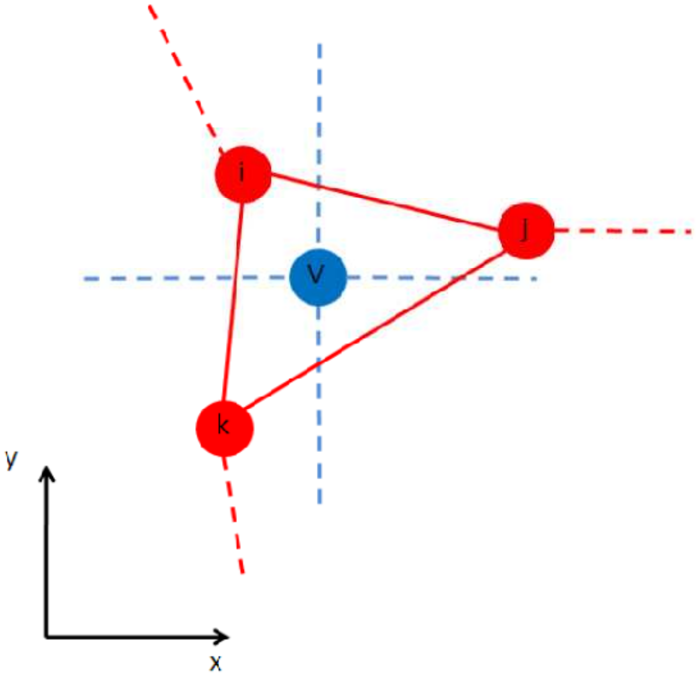


Figure 7. Detail of figure 6 ($\mathcal{M}_g$ in red and $\mathcal{M}_s$ in blue).

Considering the example in Fig.6, a nearest neighbor mapping will allocate the gray level $I(v)$to vertex v as

$$I(v) = I(i)$$

as node i is the closest vertex to node v.

In the same example, the facet mapping will give

$$I(v) = \frac{1}{3}\big(I(i) + I(j) + I(k)\big)$$

4.2. Simulations

The simulations were performed using a standard laptop (1.87 GHz processor, 2GB RAM and *Windows Vista SP1* as operating system) and *Matlab R14 Service Pack 2*. Fig.5.b and 5.c shows the results of the grid smoothing process on a detail of the original image (Fig.5.a Fig.5.b shows the results of the grid smoothing when $\theta = 1$ while Fig.5.c presents the result when $\theta = 0.005$. The results were obtained with a conjugate gradient descent. The regions with high variations in the gray levels exposes more points than the other regions leading to a distortion of the original grid. The distortions present in the two grids are similar. However, a higher θ leads to a greater fitting of the shapes, while being more sensitive to noise.

Fig.8 presents the results of the edge enhancement, on a blurry image. The size of the original image (Fig.8.a is 256×256 pixels. It may be observed that the level of noise is low but that the edges are not well defined (blurriness). The enhanced image (Fig.8.b exposes a good restoration on the edges while not compromising the quality of the image. The edges are smooth and continuous (the pixels which may be seen are due to the *pdf* compression of the image and are not present in the original simulation results). The texture of the bird is recovered while the dimensions are slightly altered. A closer look at the improvement may be found in the details presented in Fig.9.

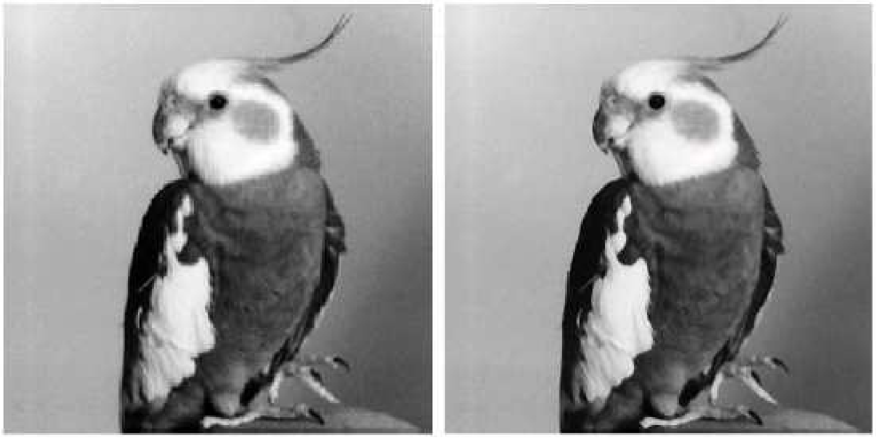


Figure 8. Test image *bird* 256×256 pixels (left) and the enhanced image using grid smoothing (right)

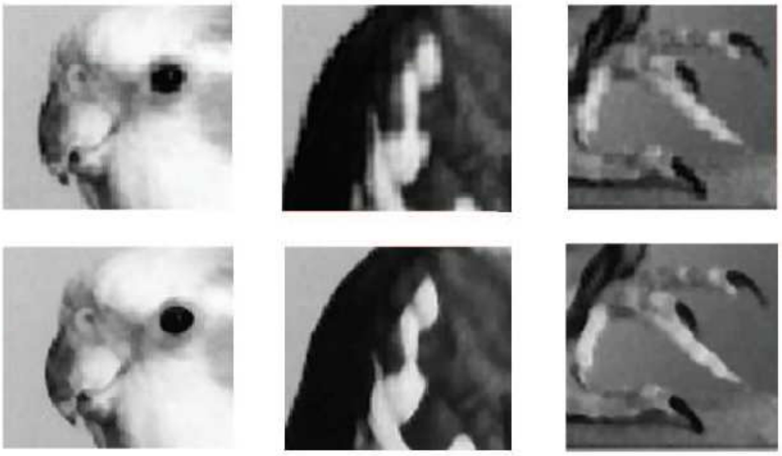


Figure 9. Details of the test image *bird* (top row) and the enhanced image using grid smoothing (bottom row).

5. Application: Edge enhancement of natural scene smoothed image

5.1. Image smoothing

Smoothing is often used to reduce noise within an image and most smoothing methods are based on low pass filters. The objective of these techniques is to separate the useful information from the noise. The range of image smoothing techniques may be divided into linear and non-linear methods. The linear methods are based on the convolution of a *box* of coefficient with the image and are usually named box filter. The coefficients are chosen in a way that a low pass filtering of the image is achieved. The average filter and the weighted (or binomial) average filter fall into this class. These types of filter are computationally inexpensive and easy to implement. They are however seldom used because they induce blur in the image and do not conserve the crispness of it. Much better performances may be obtained using non-linear smoothing filters. This class of smoothing filters includes the very well-known median filter as well as bilateral filters and Wiener filter.

5.2. Graph-based image smoothing

The graph-based technique used in this paper is a generalization of the Laplacian mesh smoothing. In the mesh smoothing framework, the image is represented using the matrix C_{ve} introduced in second section [11]The mesh smoothing techniques rely on the minimisation of a cost function J_Z, Z being the gray levels of the vertices. The result is a new vector Z_{opt}containing the filtered gray levels of the image. The general form of the cost function is:

$$J = \frac{1}{2}\left[(Z-\hat{Z})^t Q (Z-\hat{Z}) + \theta_0 Z^t Z + \theta_1 Z^t \bar{A} Z + \theta_2 Z^t \bar{A}^2 Z\right]$$

where

- Q is a symmetric positive definite weighting matrix,
- θ_0, θ_1, and θ_2 are weighting scalars,
- $\bar{A} = C^t \Omega C$, and Ω is a diagonal matrix of weights associated to each edge,
- C is the node-edge matrix of the image,
- Z and $\hat{Z}$ are respectively the value of the pixels and their initial value.

It may be shown that the previous cost function is of quadratic form and accept an unique solution.

Various techniques may be devised from this general framework. The following list summarizes the ones used in this report:

- First-order grid smoothing (FOWA)
  Cost function:

  $$J = \frac{1}{2}\left[(Z-\hat{Z})^t (Z-\hat{Z}) + \theta Z^t A Z\right]$$

  where $= C^t C$.

  Solution:

$$Z_{opt} = (I + \theta A)^{-1}\hat{Z}$$

- Second-order grid smoothing (SOWA)
  Cost function:

$$J = \frac{1}{2}\left[\left(Z - \hat{Z}\right)^t\left(Z - \hat{Z}\right) + \theta Z^t A^2 Z\right]$$

Solution:

$$Z_{opt} = (I + \theta A^2)^{-1}\hat{Z}$$

- Contextual first-order grid smoothing (CFOWA)
  Cost function:

$$J = \frac{1}{2}\left[\left(Z - \hat{Z}\right)^t Q\left(Z - \hat{Z}\right) + \theta Z^t A Z\right]$$

where Q is a diagonal matrix chosen equal to

$$Q = diag(\sigma_1, \dots, \sigma_M)$$

σ_i, being the local standard deviation evaluated around the node *i*.

Solution:

$$Z_{opt} = (Q + \theta A)^{-1}\hat{Z}$$

- Contextual second-order grid smoothing (CSOWA)
  Cost function:

$$J = \frac{1}{2}\left[\left(Z - \hat{Z}\right)^t Q\left(Z - \hat{Z}\right) + \theta Z^t A^2 Z\right]$$

Solution:

$$Z_{opt} = (Q + \theta A^2)^{-1}\hat{Z}$$

5.3. Simulations

Tab.3, 4, 5 and 6 presents the results of the mesh smoothing applied to a series of test images (extracted from the *Waterloo* database). In these tables, various smoothing techniques (Median filter, Wiener filter and bilateral filter) are used to benchmark the mesh smoothing.

Tab.3 presents the results in terms of Peak Signal to Noise Ratio (PSNR) of the smoothing of the image when corrupted by AWGN of mean equals to 0 and variance to 0.01. For each image, the corruption of the image and its smoothing were applied 15 times to each image and the mean PSNR are displayed in the table. It may be observed that the mesh smoothing filters are very competitive compared to other techniques and outperform them in most cases.

Tab.4 highlights the effect of the characteristic of the image in the image denoising process. Good results are achieved when the image include simple shapes and a low level of details (*bird*, *lena*), while results on complex images like *bridge* are usually poor.

Table 3. Performance analysis of the smoothing on various test images

	Initial	Median	Wiener	Bilateral	FOWA	SOWA	CFOWA	CSOWA
Bird	21.02	27.83	28.39	26.28	30.10	30.98	30.37	**31.07**
Bridge	21.11	23.84	25.24	24.16	25.45	25.64	25.67	**25.81**
Camera	21.38	24.62	**26.95**	25.90	25.68	25.95	26.67	26.65
Goldhill	21.02	24.94	26.33	24.88	26.68	26.70	**26.98**	26.84
Lena	21.20	26.23	27.43	25.35	26.95	27.35	27.47	**27.77**

Table 4. Global PSNR improvement over the dataset

	FOWA	SOWA	CFOWA	CSOWA
Bird	8.16	8.97	8.65	**9.30**
Bridge	3.6	3.78	3.91	**3.99**
Camera	3.6	3.90	**4.84**	**4.84**
Goldhill	4.72	4.74	**5.11**	4.99
Lena	4.93	5.35	5.68	**5.91**
Average	5.00	5.35	5.64	**5.81**

Table 5. PSNR improvement according to the power of the noise

σ^2	FOWA	SOWA	CFOWA	CSOWA
0.002	3.23	3.57	4.19	**4.29**
0.004	4.43	4.78	5.17	**5.31**
0.006	5.25	5.57	5.84	**6.03**
0.008	5.83	6.18	6.30	**6.50**
0.01	6.28	6.61	6.69	**6.91**

Tab.5 shows a comparison between the mesh smoothing techniques. The images were corrupted by 5 levels of noise ranging from 0.002 to 0.01 in variance. It appears that the CSOWA performs globally better over the dataset that the other techniques. However, the computational cost of CSOWA is greater than FOWA for example. From a general point of view, the second order techniques give better results than the first order ones.

Finally, Tab.6 presents the optimal value of θ for the mesh smoothing. The larger the power of the noise is, the greater θ should be. It may be explained by the fact that the image requires greater smoothing in this case. When working with contextual mesh smoothing, the value of θ increases by a factor 10 to 20. The reason is that, as the local standard deviation values ranges from 0 to 100 in our dataset, the attachment term is strengthen. As a consequence, θ should be increased to achieve similar smoothing as the one obtained in non-contextual smoothing.

Table 6. Optimal θ according to the power of the noise

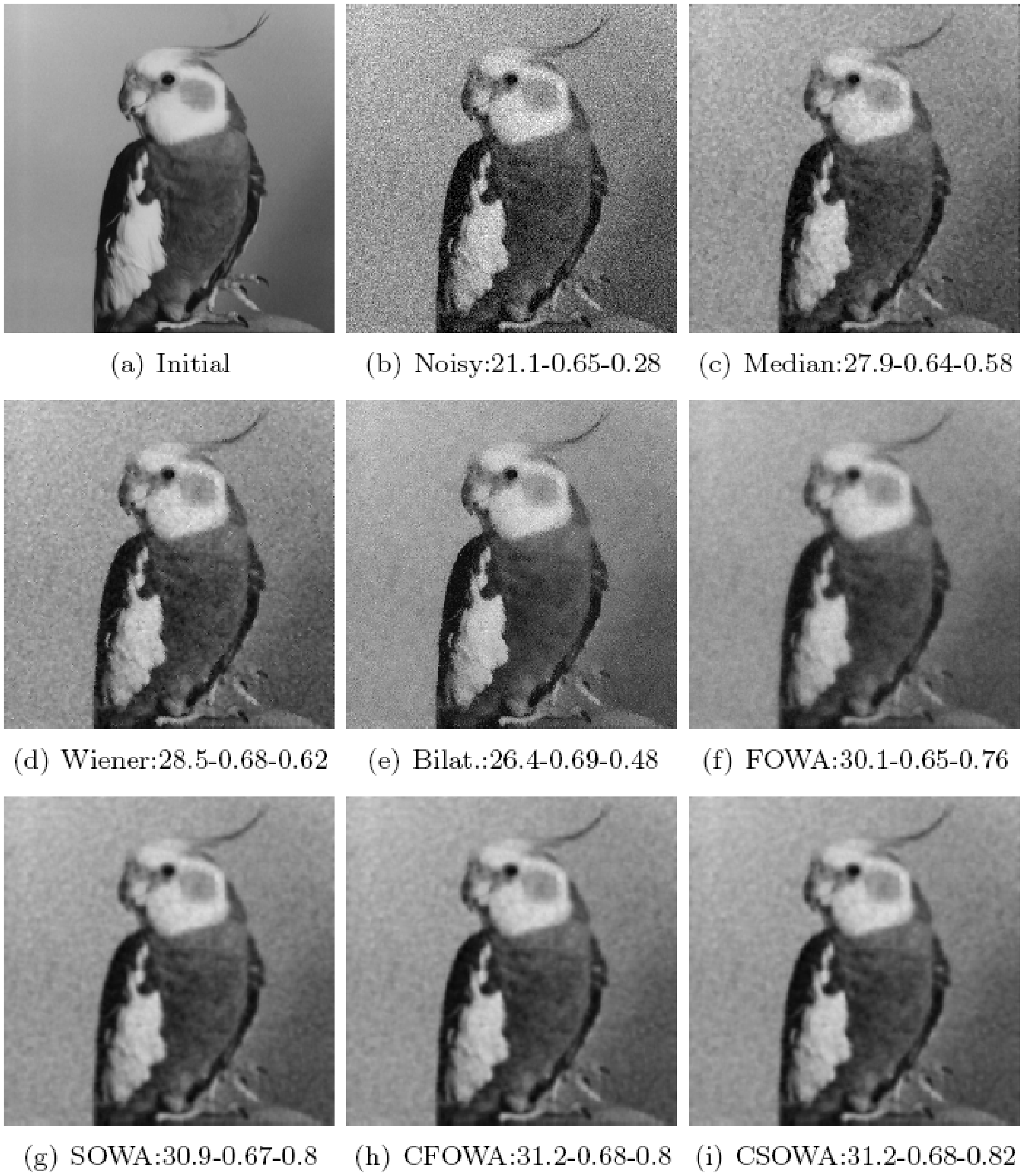
σ^2	FOWA	SOWA	CFOWA	CSOWA
0.002	0.32	0.15	6.10	2.90
0.004	0.50	0.29	11.40	6.68
0.006	0.66	0.43	16.60	10.80
0.008	0.80	0.58	22.00	16.24
0.1	0.93	0.72	27.20	19.90

Fig. 10 exposes the results of the mesh smoothing when the power of the AWGN is 0.008. Under the image, the PSNR, VIF and MSSIM index are mentioned. By opposition to the PSNR which is an absolute value of the error, the VIF (visual information factor [15]) and MSSIM (multi-scale structure similarity index [16]) index oriented to the perception by the human eye of the quality of an image. The VIF and MSSIM are numbers between 0 and 1. A index close to 1 means that the test image and reference image are very close in quality. It may be seen that the mesh smoothing methods outperforms the other ones for the *bird* test image. However, the denoised image becomes blurry.

Fig.11 shows the results of the grid smoothing on the test images. The corrupted images are fist smoothed using the mesh smoothing approach (FOWA) and then enhanced using the grid smoothing filter. The reconstruction of the image uses the nearest neighbour mapping. It may be clearly observed that the edges in the enhanced image are sharper and of better visual quality.

7. Conclusion

In conclusion, a new framework to enhanced images without a model for the degradation is presented in the paper. The method relies on the smoothing of the coordinates of the pixels in the image. The results of the image enhancement are promising. Combined with the mesh smoothing approach, the method performs well on noisy images. The output of the process is not a pixel-representation of the image. It leads to the main drawback of the technique which is the computational cost of the display of the facets. To overcome these limitations, further studies will involve redefinition of the connections in the grid to limit the number of facets. Another direction for future research will be to combine the mesh and the grid smoothing in a single operation, the objective being to define a single cost function performing the same operations.



(a) Initial (b) Noisy:21.1-0.65-0.28 (c) Median:27.9-0.64-0.58

(d) Wiener:28.5-0.68-0.62 (e) Bilat.:26.4-0.69-0.48 (f) FOWA:30.1-0.65-0.76

(g) SOWA:30.9-0.67-0.8 (h) CFOWA:31.2-0.68-0.8 (i) CSOWA:31.2-0.68-0.82

Figure 10. Performance evaluation on the *bird* image with AWGN of $\sigma^2 = 0.008$ (PSNR, VIF-MSSIM).

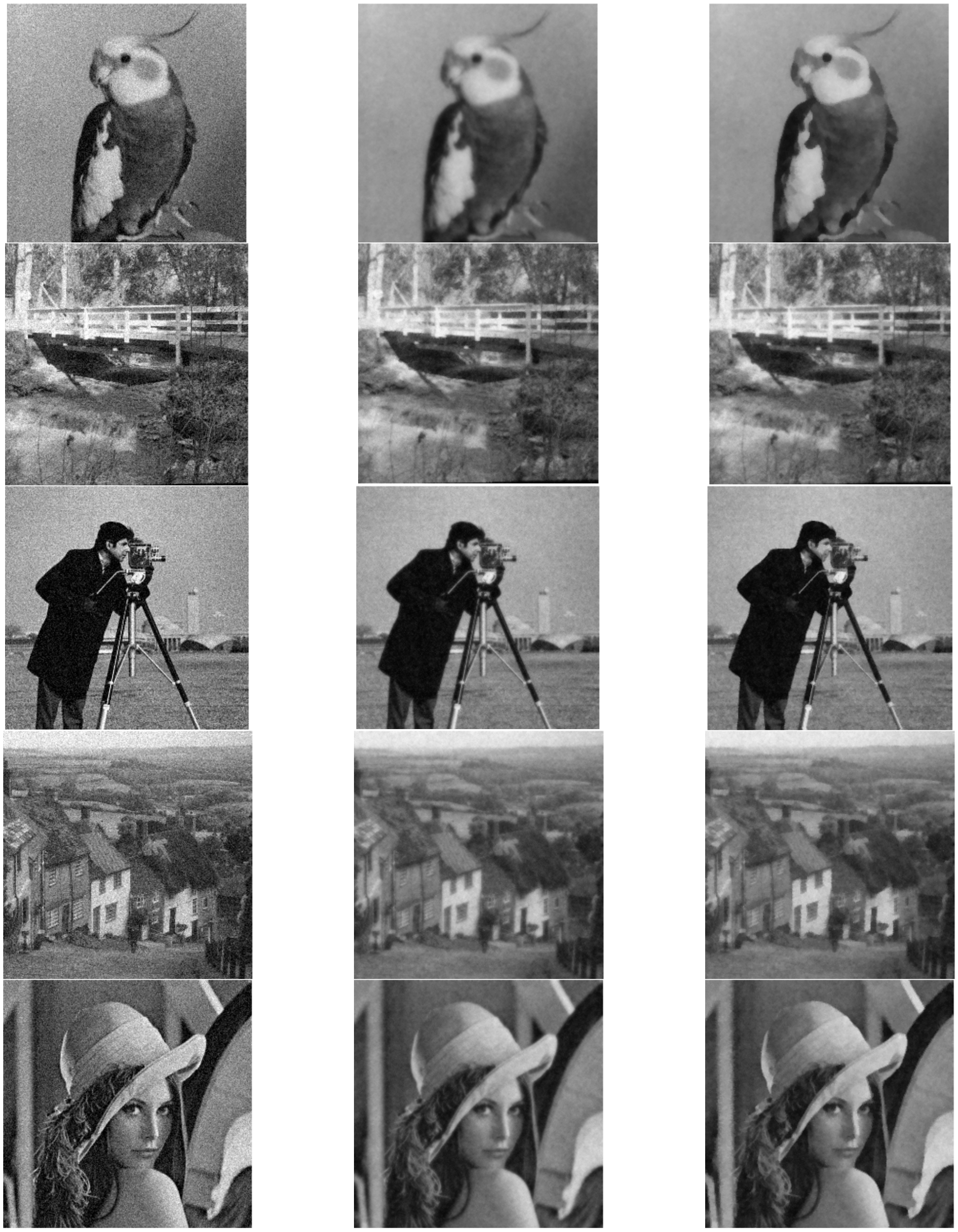


Figure 11. Image denoising results (Noisy images in the left column, smoothed image in the middle column and enhanced image in the right column).

References

[1] Zhang B. and Allebach J.P., “Adaptive bilateral filter for sharpness enhancement and noise removal”, IEEE Trans. on Image Processing, vol.17, no.5, pp.664-678, May 2008.

[2] Aysal T.C. and Barner K.E., “Quadratic weighted median filters for edge enhancement of noisy images”, IEEE Trans. on Image Processing, Vol. 13, No. 5, pp.825-938, Sep./Oct. 2007.

[3] Liyakathunisa Kumar C.N.R. and Ananthashayana V.K., “Super Resolution Reconstruction of Compressed Low Resolution Images Using Wavelet Lifting Schemes”, Computer and Electrical Engineering, 2009. ICCEE '09. Second International Conference on,Volume 2 pp629 -633, 2009.

[4] Caramelo F.J., Almeida G., Mendes L., and Ferreira N.C., “Study of an iterative super-resolution algorithm and its feasibility in high-resolution animal imaging with low-resolution SPECT cameras”, Nuclear Science Symposium Conference Record, 2007. NSS '07. IEEE , vol.6, no., pp.4452-4456, Oct. 26 2007-Nov. 3 2007.

[5] Toyran M. and Kayran A.H. “Super resolution image reconstruction from low resolution aliased images”, Signal Processing, Communication and Applications Conference, 2008. SIU 2008. IEEE 16th , vol., no., pp.1-5, 20-22 April 2008.

[6] Wang C.C.L., “Bilateral recovering of sharp edges on feature-insensitive sampled meshes”, IEEE Trans. on Visualization and Computer Graphics, vol.12, no. 4, pp.629-639, Jul./Aug. 2006.

[7] Xu D. and Adams M.D.:, “An improved normal-meshed-based image coder”, Can. J. Elect. Comput. Eng., Vol. 33, No. 1, Winter 2008.

[8] Feijun J. and Shi B.E., “The memristive grid outperforms the resistive grid for edge preserving smoothing”, Circuit Theory and Design, 2009. ECCTD 2009.181 -184.

[9] Shuhui B., Shiina T., Yamakawa M., and Takizawa H., “Adaptive dynamic grid interpolation: A robust, high-performance displacement smoothing filter for myocardial strain imaging”, Ultrasonics Symposium, 2008. IUS 2008. IEEE , vol., no., pp.753-756, 2-5 Nov. 2008.

[10]Huang C.L.and Chao-Yuen Hsu C.Y.:, “A new motion compensation method for image sequence coding using hierarchical grid interpolation”, Circuits and Systems for Video Technology, IEEE Transactions on , vol.4, no.1, pp.42-52, Feb 1994.

[11] Hamam Y. and Couprie M., “An Optimisation-Based Approach to Mesh Smoothing: Reformulation and Extension”, Lecture Notes in Computer Science, Springer, Volume 5534-2009 pp31-41, 2009.

[12]Noel G., Djouani, K., and Hamam,.Y, “Optimisation-based Image Grid Smoothing for Sea Surface Temperature Images”, Advanced Concepts for Intelligent Vision Systems, ACIVS 2010, Sydney, Australia.

[13] Fletcher, R., and Reeves, CM., “Function Minimization by Conjugate Gradient”, The Computer Journal, British Computer Society, 1964.

[14]Noel G., Djouani, K., and Hamam,.Y, “Grid Smoothing: A graph-based Approach”, 15th Iberoamerican Congress on Pattern Recognition, CIARP 2010, Sao Paulo, Brasil.

[15]Sheikh, H.R. and Bovik, A.C.,” Image information and visual quality”, Image processing, IEEE Transactions on, vol.15, no.2, pp. 430-444, Feb. 2006.

[16] Z. Wang, E. P. Simoncelli and A. C. Bovik, "Multi-scale structural similarity for image quality assessment," *IEEE Asilomar Conference Signals, Systems and Computers*, Nov. 2003.

Authors

Guillaume Noel received his MSc degree in information and communication technologies from the Ecole Telecom ParisTech ENST of Paris, France in 2003. Since April 2008 he is seconded by the French Ministry of Foreign Affairs to the French South African Institute of Technology (FŚATI) at Tshwane University of Technology (TUT), Pretoria, South Africa. He is currently working towards the doctorate of technology in electrical engineering. His research interests are focused on image analysis and processing, computer vision, machine intelligence, and wireless cellular networks.

Karim Djouani is professor, scientist and technical group supervisor of soft computing, complex systems and optimization. Since July 2008 he is seconded by the French Ministry of Higher Education to the French South African Institute of Technology (FŚATI) at Tshwane University of Technology (TUT), Pretoria, South Africa. He is also with the SCTIC team of the LISSI lab, University Paris Est. He was also national and European projects manager at the LISSI Lab. His current works focus on the development of novel and highly efficient algorithms for reasoning systems with uncertainty as well as optimization, for distributed systems, networked control systems, wireless ad-hoc network, wireless and mobile communication, and wireless sensors networks as well as Robotics. He has authored/co-authored over 100 articles in archival journals and conference proceedings as well as five chapters in edited books. Prof. Djouani is a Member of IEEE ComSoc and CS societies and several National Research task Group (GDR-MACS, GDR-ISIS, etc.).

Prof. Yskandar HAMAM graduated as a Bachelor of Electrical Engineering from the American University of Beirut (AUB) in 1966. He then obtained his M.Sc. in 1970 and Ph.D. in 1972 from the University of Manchester Institute of Science and Technology. He also obtained his "Diplôme d'Habilitation à Diriger des Recherches" (equivalent to D.Sc.) from the « Université des Sciences et Technologies de Lille » in 1998. He conducted research activities and lectured in England, Brazil, Lebanon, Belgium and France. He was the head of the Control department and dean of faculty at ESIEE, France. He was an active member in modeling and simulation societies and was the president of EUROSIM. He is presently the Scientific Director of the French South African Institute of Technology (F'SATI) at the Tshwane University of Technology in South Africa. He is presently Emeritus Professor at ESIEE and is a member of the Informatics Centre of ESIEE-Paris and a member of the LISV laboratory of the "Université de Versailles Saint Quentin en Yvelines". He is senior member of the IEEE.