

Contour and Detail Detection for Spatially Adaptive Color Median Filtering

F. Robert-Inacio¹, J. Angulo² and É. Dinet³

¹Institut Matériaux Microélectronique et Nanosciences de Provence UMR CNRS 6242
Institut Supérieur de l'Électronique et du Numérique, Toulon, France.

²Centre de Morphologie Mathématique, Ecole des Mines de Paris, Fontainebleau, France

³Université Jean Monnet, Laboratoire d'Informatique Graphique et d'Ingénierie de la Vision, Saint-Étienne, France.

Abstract

In this paper, we present a spatially adaptive filter for color median filtering. Several alternative implementations of such a filter are given, using different ways of gradient computation. Then, several methods of color median filtering are compared in terms of performance on a particular image. Vector median filters are considered as well as median filters inspired by grey-level methods.

Introduction

Median filtering has been widely explored since the last ten years and many methods have been presented in order to filter color images. Some of them are based on vector considerations [1] and we will compare them to ours. First of all, we recall the main principle of the spatially adaptive filtering method [2] before presenting several alternative ways of implementation. The main difference between the alternative implementations lays in the determination of color contours. The processes we are going to consider are based either on color gradient or mathematical morphology. Finally, a comparison of all those filters is set up.

Spatially adaptive median filter

Recalls about the spatially adaptive filtering process

This particular filtering method is based on the determination of two different maps. The first one is supposed to contain high frequencies, in other words, contour information, and the second one is derived from the previous one by computing the chessboard distance between a point of the image and the closest contour[3]. The chessboard distance map gives at each point the half width of the filtering window, this window being chosen as wide as possible so that it does not contain any part of contours. Afterwards the filtered value is determined by using a bit-mixing paradigm [4] in order to sort values in the filtering window. In order to summarize the filtering process, Fig. 2 shows the different stages of the method.

The main points to be discussed and improved are the contour detection method and the determination of the filtered value. The filtering process using the bit-mixing paradigm was compared to vector methods described in [1] and the results were presented in [5]. In the following, we pay more attention to the contour detection. So we are going to set up different ways of determining color contour location in order to create a contour map by thresholding, and we will give the filtering results on a particular image (Fig. 1).

In the original version of the adaptive median filter, the contour detection was approximated by a color Euclidean distance



Figure 1. Original image

map, that enables to locate spots where the color difference between a point and two of its neighbors is the highest [2]. This map is then thresholded in order to keep only significant color differences.

Alternative contour detection

In this paper other family of color contour maps has been explored besides the basic contour map based on the Euclidean distance between colors. More precisely, working in the luminance, saturation and hue representation in norm L_1 [6] (named LSH representation), we have calculated for each color image the following *barycentric* color gradient:

$$\rho(\mathbf{f})(x) = f_S(x) \times \rho^\circ(f_H)(x) + (1 - f_S(x)) \times \rho(f_L) + \rho(f_S). \quad (1)$$

where the L_1 saturation, $f_S(x)$, balances the effects of luminance and hue gradients, $\rho(f_L)$ and $\rho^\circ(f_H)$. Further details about the computation of morphological gradients $\rho(f)$ are given in [6]. The gradient of saturation is also included to include also the contours exclusively associate to changes in the saturation. We have also compared in our study the results associated to the LSH gradient with those for the classical $L^*a^*b^*$ gradient (i.e., the Euclidean distance in the $L^*a^*b^*$ color space). The $L^*a^*b^*$ gradient at any point x is based on computing the Euclidean distance in $L^*a^*b^*$ between the color points of the unitary ball centred at the current point, $K(x)$, in order to obtain the maximal variation of distance which corresponds to the estimate of the modulus of the perceptual color gradient, i.e.,

$$\rho^{L^*a^*b^*}(\mathbf{f})(x) = \vee [d_E^{L^*a^*b^*}(\mathbf{f}(x), \mathbf{f}(y)), y \in K(x)] \quad (2)$$

where $d_E^{L^*a^*b^*}(\mathbf{c}_i, \mathbf{c}_j)$ is the perceptual distance between the colors i and j (with $\mathbf{c}_k = (c_k^{L^*}, c_k^{a^*}, c_k^{b^*})$):

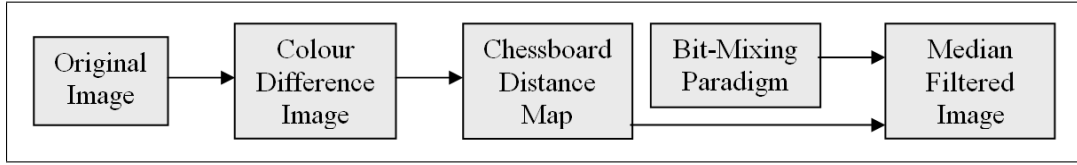


Figure 2. Stages of the filtering process

$$d_E^{L^*a^*b^*}(\mathbf{c}_i, \mathbf{c}_j) = \sqrt{(c_i^{L^*} - c_j^{L^*})^2 + c_i^{a^*} - c_j^{a^*})^2 + c_i^{b^*} - c_j^{b^*})^2}.$$

In addition to the contours between color regions, we consider that a spatially adaptive median filter must take into account also the presence of small, but very contrasted, color details. The median filter should remove the noise without removing these structures. The extraction of the details is achieved by means of the morphological color top-hats [6], defined as follows

- The *chromatic top-hat* is given by

$$\rho_B^C(\mathbf{f}) = [f_S \times \rho_B^O(f_H)] \vee \rho_B^\dagger(f_S), \quad (3)$$

where the product sign “ \times ” means a pointwise multiplication of function values. This operator extracts the fast variations of color regions on a saturated color background (i.e. saturated color peaks on uniform color regions) and the fast variations of saturated color regions on an achromatic (unsaturated) background (i.e. saturated color peaks on achromatic regions).

- The *white-achromatic top-hat* is the absolute value difference between the chromatic top-hat and the global bright top-hat,

$$\rho_B^{A+} = |\rho_B^C - \rho_B^\dagger|, \quad (4)$$

where the *global bright top-hat* is calculated by

$$\rho_B^\dagger(\mathbf{f}) = \rho_B^\dagger(f_L) \vee \rho_B^\dagger(f_S). \quad (5)$$

It characterizes the fast variations of bright regions (i.e. positive peaks of clearness) and the fast variations of achromatic regions on a saturated background (i.e. unsaturated peaks: black, white and grey on color regions).

- The *black-achromatic top-hat* is the difference $\rho_B^{A-} = |\rho_B^C - \rho_B^\dagger|$, where the *global dark top-hat* is obtained by $\rho_B^\dagger(\mathbf{f}) = \rho_B^\dagger(f_L) \vee \rho_B^\dagger(f_S)$. Dually, it tackles the fast variations of dark regions (i.e. negative peaks of clearness) and the fast variations of achromatic regions on a saturated background. The term $\rho_B^\dagger(f_S)$ appears in both ρ_B^\dagger and ρ_B^\dagger to achieve symmetrical definitions.

We have compared the effects of the chromatic and white/black-achromatic top-hats for adapting the median filter. Both the color gradients and color top-hats are thresholded in order to build the corresponding binary maps.

Experimental results and comparison to vector filters

Fig. 3 shows the effects of such a filter when using the previous different methods for contour detection on the original image of Fig. 1. The intermediate contour images have been thresholded at 30 in every case in order to obtain the contour map. We can note the main differences between the filtered images and the degradation of the original image around the letters.

Elements of comparison

Several values are going to be used in order to evaluate the efficiency of the filters under study. First of all, the mean absolute error (MAE) and the mean square error (MSE) are estimated, as well as the normalized color difference (NCD) and the peak signal-to-noise ratio (PSNR) [7][8][1][9].

Experimental results for the spatially adaptive filter

Firstly, let us have a look to experimental results of the spatially adaptive filter.

Results for spatially adaptive median filtering using the bit-mixing paradigm and a color difference map

Threshold	MAE	MSE	NCD	PSNR
5	0.150	0.229	0.003	94.784
15	0.775	2.955	0.010	72.845
25	1.498	8.724	0.015	62.740
35	2.220	16.985	0.020	56.537
45	3.097	32.015	0.025	50.730
55	3.929	49.581	0.029	46.675
65	4.959	77.802	0.034	42.143
75	5.948	109.678	0.039	39.099
85	7.224	163.397	0.044	35.457
95	9.175	264.515	0.053	30.741

Results for spatially adaptive median filtering using the bit-mixing paradigm and achromatic top-hat contours

Threshold	MAE	MSE	NCD	PSNR
5	2.153	39.840	0.015	50.232
10	2.977	53.587	0.021	47.048
15	3.680	66.534	0.025	44.740
20	4.248	77.707	0.029	43.010
25	4.754	89.260	0.032	41.482
30	5.282	103.345	0.035	39.972
35	5.843	121.015	0.030	38.330
40	6.385	140.860	0.041	36.866
45	6.878	160.092	0.043	35.656
50	7.323	177.788	0.045	34.653
55	7.859	201.704	0.047	33.379
60	8.357	226.985	0.049	32.259
65	8.782	248.402	0.051	31.374
70	9.157	267.089	0.053	30.645
75	9.636	291.873	0.055	29.770
80	10.191	322.319	0.050	28.782
85	10.817	358.247	0.061	27.735
90	11.025	369.408	0.062	27.467
95	11.244	381.575	0.062	27.205
100	11.468	394.424	0.063	26.929

Median filter using a bit-mixing paradigm

Directly inspired by the gray scale median filter this approach requires a total order on the set of gray levels for the computation of the median value. Actually, as there exists no such total order for color vectors, a bit-mixing paradigm is used to provide a sequence on any window W [4]. Let us consider a scalar value k_i associated with each color vector x_i of W . For



Figure 3. Spatially adaptive median filter with contour detection by a) color difference b) gradient Lab, c) achromatic top hat and d) gradient LSH.

Results for spatially adaptive median filtering using the bit-mixing paradigm and chromatic top-hat contours

Threshold	MAE	MSE	NCD	PSNR
5	1.249	22.752	0.009	56.333
10	2.132	34.466	0.016	51.759
15	2.871	46.245	.021	48.545
20	3.526	58.102	0.025	46.092
25	4.153	72.418	0.029	43.794
30	4.785	87.844	0.032	41.753
35	5.455	108.415	0.036	39.599
40	6.102	130.596	0.039	37.721
45	6.643	150.682	0.042	36.338
50	7.138	169.926	0.044	35.175
55	7.641	192.842	0.046	34.017
60	8.103	214.554	0.048	32.993
65	8.506	235.010	0.050	32.109
70	8.793	248.381	0.051	31.557
75	9.205	269.811	0.053	30.759
80	9.652	293.814	0.055	29.909
85	10.090	317.236	0.056	29.146
90	10.405	333.055	0.058	28.624
95	10.816	356.203	0.059	27.985
100	11.346	386.813	0.062	27.150

Results for spatially adaptive median filtering using the bit-mixing paradigm and Lab gradient

Threshold	MAE	MSE	NCD	PSNR
5	0.100	0.111	0.002	99.922
10	0.491	1.655	0.006	78.164
15	1.046	5.889	0.010	66.731
20	1.837	16.783	0.015	57.463
25	3.041	42.646	0.021	48.645
30	4.613	88.060	0.028	41.658
35	6.937	182.653	0.040	34.545
40	9.180	288.807	0.051	30.003
45	10.517	359.751	0.056	27.995
50	11.713	427.522	0.061	26.389
55	12.201	452.559	0.064	25.833
60	12.465	466.381	0.065	25.547
65	12.555	471.606	0.066	25.435
70	12.577	472.797	0.066	25.412
75	12.577	472.797	0.066	25.412
80	12.577	472.797	0.066	25.412
85	12.577	472.797	0.066	25.412
90	12.577	472.797	0.066	25.412
95	12.577	472.797	0.066	25.412
100	12.577	472.797	0.066	25.412

example, if x_i is an RGB vector represented by 3 bytes, k_i is a 24-bit integer value as shown in Fig. 4.

The values k_i are arranged in order. Afterwards, vectors x_i are sorted according to values k_i to obtain a set of ordered vectors

y_i . Then the bit-mixing filter value, $BMF(W)$, is given by:

$$BMF(W) = x_M \tag{6}$$

such that $x_M = y_{\frac{N+1}{2}}$.

Results for spatially adaptive median filtering using the bit-mixing paradigm and LSH gradient

Threshold	MAE	MSE	NCD	PSNR
5	0.089	0.082	0.002	102.253
10	0.352	0.827	0.006	84.753
15	0.697	2.706	0.009	74.658
20	1.088	5.778	0.012	67.668
25	1.550	10.898	0.015	61.817
30	2.140	20.701	0.018	56.091
35	2.885	38.295	0.022	50.475
40	3.736	59.401	0.026	46.187
45	4.545	79.362	0.030	43.287
50	5.225	96.455	0.033	41.344
55	5.976	118.486	0.037	39.300
60	6.799	147.918	0.040	37.173
65	7.652	182.861	0.044	35.118
70	8.578	224.178	0.047	33.096
75	9.446	275.836	0.050	31.149
80	10.179	323.139	0.053	29.607
85	10.755	357.586	0.056	28.561
90	11.230	305.095	0.058	27.746
95	12.042	438.291	0.063	26.296
100	12.498	466.932	0.065	25.531

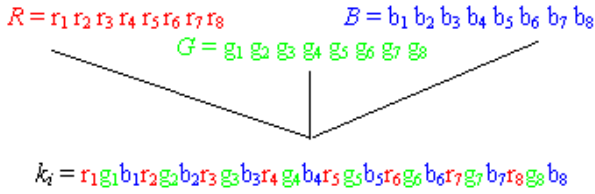


Figure 4. Bit-mixing paradigm.

Results for median filtering using the bit-mixing paradigm

Window size	MAE	MSE	NCD	PSNR
1	2.804	56.102	0.017	47.167
2	5.508	134.226	0.031	38.321
3	8.783	267.689	0.047	31.415
4	11.200	396.631	0.059	27.353
5	13.216	497.212	0.070	24.922

Vector median filter

A distance measure D_i is associated with each vector x_i :

$$D_i = \sum_{j=1}^N \|x_i - x_j\|_L \quad (7)$$

for $i = 1..N$, where $\|x_i - x_j\|_L$ is the distance between x_i and x_j regarding the L -norm. Then the vector median filter value, $VMF(W)$, is given by:

$$VMF(W) = x_M \quad (8)$$

such that $D_M = \min_{i=1..N} D_i$.

Basic vector directional filter

An angular measure α_i is associated with each vector x_i :

$$\alpha_i = \sum_{j=1}^N A(x_i, x_j) \quad (9)$$

Results for vector median filtering for norm L_1

Window size	MAE	MSE	NCD	PSNR
1	2.649	53.844	0.016	47.635
2	5.215	126.877	0.029	38.879
3	8.314	254.103	0.044	31.932
4	10.453	370.853	0.055	28.013
5	12.197	457.836	0.065	25.605

Results for vector median filtering for norm L_2

Window size	MAE	MSE	NCD	PSNR
1	2.426	53.562	0.012	48.066
2	4.877	125.745	0.023	39.530
3	7.886	250.960	0.036	32.574
4	9.957	361.764	0.045	28.837
5	11.584	450.529	0.053	26.556

Results for vector median filtering for norm L_3

Window size	MAE	MSE	NCD	PSNR
1	2.458	54.011	0.012	47.993
2	4.926	127.501	0.024	39.420
3	7.981	255.824	0.036	32.423
4	10.107	372.641	0.045	28.625
5	11.716	458.866	0.052	26.447

for $i = 1..N$, where $A(x_i, x_j)$ is the scalar value representing the angle between x_i and x_j . Then the basic vector directional filter value, $BVDF(W)$, is given by:

$$BVDF(W) = x_M \quad (10)$$

such that $\alpha_M = \min_{i=1..N} \alpha_i$.

Results for vector directional median filtering

Window size	MAE	MSE	NCD	PSNR
1	3.477	90.061	0.015	43.189
2	7.162	237.515	0.031	33.561
3	10.781	444.525	0.045	27.263
4	13.085	589.154	0.055	24.352
5	15.021	732.356	0.064	22.063

Directional distance filter

A scalar value Ω_i , mixing distance and angular measures, is associated with each vector x_i :

$$\Omega_i = D_i^{1-\omega} \cdot \alpha_i^\omega \quad (11)$$

for $i = 1..N$. Thus:

$$\Omega_i = \left(\sum_{j=1}^N \|x_i - x_j\|_L \right)^{1-\omega} \cdot \left(\sum_{j=1}^N A(x_i, x_j) \right)^\omega \quad (12)$$

where ω is a power value ranged from 0 to 1 that modulates the relative influence of D_i and α_i in the calculus. Then the directional distance filter value, $DDF(W)$, is given by:

$$DDF(W) = x_M \quad (13)$$

such that $\Omega_M = \min_{i=1..N} \Omega_i$.

Window size	ω	MAE	MSE	NCD	PSNR	Window size	ω	MAE	MSE	NCD	PSNR
1	0.10	0.126	0.306	0.002	95.740	1	0.10	0.216	0.743	0.003	88.296
2	0.10	0.124	0.245	0.002	97.278	2	0.10	0.233	0.679	0.003	88.766
3	0.10	0.112	0.232	0.002	98.032	3	0.10	0.242	0.810	0.003	87.688
4	0.10	0.093	0.225	0.002	98.922	4	0.10	0.211	0.734	0.003	88.684
5	0.10	0.074	0.191	0.001	100.848	5	0.10	0.178	0.610	0.003	90.360
1	0.20	1.148	15.537	0.007	60.265	1	0.20	1.928	41.835	0.009	50.728
2	0.20	1.075	9.231	0.008	64.701	2	0.20	1.989	27.902	0.012	54.375
3	0.20	1.075	9.758	0.009	64.275	3	0.20	2.247	37.856	0.014	51.445
4	0.20	1.034	9.908	0.009	64.213	4	0.20	2.243	44.833	0.014	49.916
5	0.20	1.077	13.233	0.009	61.668	5	0.20	2.220	48.363	0.014	49.225
1	0.30	2.401	54.646	0.011	47.940	1	0.30	2.396	54.786	0.011	47.932
2	0.30	4.927	127.374	0.023	39.448	2	0.30	4.946	128.299	0.023	39.410
3	0.30	7.834	248.867	0.036	32.716	3	0.30	7.978	254.833	0.036	32.501
4	0.30	9.486	338.853	0.043	29.518	4	0.30	10.060	366.095	0.045	28.791
5	0.30	10.629	406.250	0.049	27.592	5	0.30	11.580	455.053	0.052	26.529
1	0.40	2.409	55.705	0.011	47.779	1	0.40	2.405	55.974	0.011	47.749
2	0.40	4.957	128.536	0.023	39.381	2	0.40	4.968	128.939	0.023	39.371
3	0.40	8.030	257.710	0.036	32.393	3	0.40	8.049	258.954	0.036	32.359
4	0.40	10.172	372.918	0.046	28.611	4	0.40	10.206	374.706	0.046	28.576
5	0.40	11.787	467.987	0.053	26.249	5	0.40	11.825	470.056	0.053	26.219
1	0.50	2.421	56.924	0.011	47.596	1	0.50	2.421	57.165	0.011	47.566
2	0.50	4.981	128.998	0.023	39.361	2	0.50	4.989	129.554	0.023	39.334
3	0.50	8.098	261.523	0.036	32.264	3	0.50	8.127	263.118	0.036	32.216
4	0.50	10.223	376.358	0.046	28.535	4	0.50	10.269	378.591	0.046	28.484
5	0.50	11.877	475.833	0.053	26.100	5	0.50	11.912	477.828	0.053	26.060
1	0.60	2.436	57.451	0.011	47.517	1	0.60	2.441	57.845	0.011	47.457
2	0.60	5.031	131.846	0.023	39.167	2	0.60	5.047	132.786	0.023	39.108
3	0.60	8.174	266.443	0.036	32.100	3	0.60	8.199	268.160	0.036	32.050
4	0.60	10.300	382.094	0.046	28.398	4	0.60	10.356	385.375	0.046	28.323
5	0.60	11.999	487.246	0.053	25.874	5	0.60	12.033	488.982	0.054	25.840
1	0.70	2.483	59.182	0.011	47.236	1	0.70	2.490	59.408	0.012	47.202
2	0.70	5.136	137.020	0.024	38.813	2	0.70	5.154	137.824	0.024	38.765
3	0.70	8.268	273.131	0.037	31.875	3	0.70	8.298	274.854	0.037	31.821
4	0.70	10.412	390.553	0.046	28.198	4	0.70	10.442	392.264	0.046	28.162
5	0.70	12.101	496.856	0.054	25.686	5	0.70	12.132	498.679	0.054	25.648
1	0.80	2.601	62.948	0.012	46.650	1	0.80	2.613	63.297	0.012	46.599
2	0.80	5.325	146.149	0.024	38.210	2	0.80	5.334	146.561	0.024	38.188
3	0.80	8.419	283.375	0.037	31.534	3	0.80	8.446	285.228	0.037	31.474
4	0.80	10.528	401.573	0.047	27.946	4	0.80	10.559	403.850	0.047	27.899
5	0.80	12.218	513.565	0.054	25.370	5	0.80	12.266	516.819	0.054	25.308
1	0.90	2.872	71.338	0.013	45.450	1	0.90	2.883	71.630	0.013	45.411
2	0.90	5.646	161.299	0.025	37.277	2	0.90	5.662	161.718	0.025	37.254
3	0.90	8.709	304.702	0.038	30.857	3	0.90	8.738	306.838	0.038	30.792
4	0.90	10.778	425.866	0.047	27.409	4	0.90	10.804	428.042	0.047	27.362
5	0.90	12.513	543.438	0.055	24.838	5	0.90	12.544	545.675	0.055	24.801

Interpretation

All these error estimations show that the spatially adaptive filter gives better results than vector methods, whatever the contour detection method is. Furthermore, color gradient methods best score on the particular image under study.

For the Lab gradient, we can see that we obtain the same error results for the threshold value varying from 70 to 100. That is because the maximal value on the Lab gradient image is less than 70. In this way the thresholded gradient image is the same for all threshold values greater than 70.

Conclusion

In this paper we have presented several alternative methods to implement a spatially adaptive median filter. The main difference between the alternative methods is the method used to detect color contours. In the primary method a color difference map based on the Euclidean distance between colors was used. In this paper we have taken into account more sophisticated methods such as color gradient or top-hats. Each of these alternative methods have been tested on a particular image. Results are slightly different when comparing error estimations such as mean absolute error, mean square error, normalized color difference or peak signal-to-noise ratio. Nevertheless scores are better

Results for vector directional median filtering for norm L_3

Window size	ω	MAE	MSE	NCD	PSNR
1	0.10	0.258	1.015	0.003	85.554
2	0.10	0.272	0.857	0.004	86.732
3	0.10	0.287	1.027	0.004	85.509
4	0.10	0.262	1.007	0.003	85.865
5	0.10	0.230	0.924	0.003	86.750
1	0.20	2.107	47.457	0.010	49.456
2	0.20	2.370	37.865	0.013	51.487
3	0.20	2.805	55.077	0.016	47.848
4	0.20	2.776	62.807	0.016	46.648
5	0.20	2.745	65.285	0.016	46.292
1	0.30	2.399	55.055	0.011	47.889
2	0.30	4.953	128.578	0.023	39.397
3	0.30	7.997	255.466	0.036	32.487
4	0.30	10.126	369.523	0.045	28.706
5	0.30	11.690	459.443	0.052	26.428
1	0.40	2.402	56.126	0.011	47.732
2	0.40	4.974	129.221	0.023	39.356
3	0.40	8.066	259.629	0.036	32.341
4	0.40	10.227	375.957	0.046	28.546
5	0.40	11.841	470.257	0.053	26.204
1	0.50	2.426	57.326	0.011	47.539
2	0.50	5.002	130.337	0.023	39.282
3	0.50	8.135	263.462	0.036	32.213
4	0.50	10.290	379.703	0.046	28.456
5	0.50	11.931	478.201	0.053	26.040
1	0.60	2.446	58.082	0.011	47.420
2	0.60	5.064	133.379	0.023	39.072
3	0.60	8.207	268.655	0.036	32.036
4	0.60	10.374	386.374	0.046	28.300
5	0.60	12.048	489.480	0.054	25.818
1	0.70	2.495	59.584	0.012	47.175
2	0.70	5.163	138.333	0.024	38.732
3	0.70	8.314	275.574	0.037	31.796
4	0.70	10.454	393.112	0.046	28.144
5	0.70	12.156	500.286	0.054	25.613
1	0.80	2.621	63.556	0.012	46.561
2	0.80	5.340	146.845	0.024	38.170
3	0.80	8.461	286.202	0.037	31.443
4	0.80	10.566	404.549	0.047	27.884
5	0.80	12.277	516.354	0.054	25.312
1	0.90	2.887	71.812	0.013	45.387
2	0.90	5.671	162.199	0.025	37.227
3	0.90	8.755	307.880	0.038	30.761
4	0.90	10.821	429.301	0.047	27.335
5	0.90	12.549	545.601	0.055	24.800

than the ones obtained for vector filtering methods. We have now to validate our algorithm on a large database of images, in order to study links between contour detection and filtering efficiency.

References

- [1] R. Lukac, B. Smolka, K. Martin, K. Plataniotis, and A. Venetsanopoulos, "Vector filtering for color imaging," *IEEE Signal Process. Mag.*, vol. 22, pp. 74–86, 2005.
- [2] F. Robert-Inacio and E. Dinet, "An adaptive median filter for colour image processing," in *Proc. 3rd CGIV*, Leeds, UK, June 2006, pp. 205–210.
- [3] G. Borgefors, "Distance transformations in digital images,"

CVGIP, vol. 34, pp. 344–371, 1986.

- [4] P. Lambert and L. Macaire, "Filtering and segmentation : the specificity of color images," in *Proc. Conference on Color in Graphics and Image Processing*, Saint-Etienne, France, September 2000, pp. 57–64.
- [5] E. Dinet and F. Robert-Inacio, "Color median filtering: a spatially adaptive filter," in *Proc. IVCNZ07*, Hamilton, New Zealand, December 2007.
- [6] J. Angulo and J. Serra, "Modelling and segmentation of colour images in polar representations," *Image Vision and Computing*, vol. 25/4, pp. 475–495, 2007.
- [7] M. Vardavoulia, I. Andreadis, and P. Tsalides, "A new median filter for colour image processing," *Pattern Recognition Letters*, vol. 22, pp. 675–689, 2001.
- [8] R. Lukac, "Adaptive vector median filtering," *Pattern Recognition Letters*, vol. 24, pp. 1889–1899, 2002.
- [9] L. Jin and D. Li, "A switching vector median filter based on the cielab color space for color image restoration," *Signal Processing*, vol. 87, pp. 1345–1354, 2007.

Author Biography

Frédérique Robert-Inacio received her Master degree in applied mathematics in 1988 and her PhD in image processing in 1992 from University Jean Monnet, Saint-Étienne, France. She worked as a researcher at the University of Sherbrooke, Canada, from 1993 to 1994, and since September 1994 at the ISEN. Her present spots of interest concern pattern recognition, color imaging and computer vision, and more accurately image processing furthering the estimation of parameters characterizing geometrical features for shape classification.