



HAL
open science

Automatic underwater image pre-processing

Stéphane Bazeille, Isabelle Quidu, Luc Jaulin, Jean-Philippe Malkasse

► **To cite this version:**

Stéphane Bazeille, Isabelle Quidu, Luc Jaulin, Jean-Philippe Malkasse. Automatic underwater image pre-processing. CMM'06, Oct 2006, Brest, France. hal-00504893

HAL Id: hal-00504893

<https://hal.science/hal-00504893v1>

Submitted on 22 Jul 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Automatic Underwater Image Pre-Processing

Stéphane Bazeille⁽¹⁾, Isabelle Quidu⁽¹⁾, Luc Jaulin⁽¹⁾, Jean-Phillipe Malkasse⁽²⁾

⁽¹⁾ Laboratoire E3I2 EA-3876, ENSIETA, 2 rue François Verny, 29806 BREST Cedex 9 FRANCE

⁽²⁾ Thales Underwater Systems S.A S. Route de Sainte Anne du Portzic Site Amiral Nomy CS 43814 29238 BREST Cedex 3 FRANCE

SHORT ABSTRACT: *A novel pre-processing filter is proposed for underwater image restoration. Because of specific transmission properties of light in the water, underwater image suffers from limited range, non uniform lighting, low contrast, color diminished, important blur... Today pre-processing methods typically only concentrates on non uniform lighting or color correction and often require additional knowledge of the environment. The algorithm proposed in this paper is an automatic algorithm to pre-process underwater images. It reduces underwater perturbations, and improves image quality. It is composed of several successive independent processing steps which correct non uniform illumination, suppress noise, enhance contrast and adjust colors. Performances of filtering will be assessed using an edge detection robustness criterion.*

Keywords: *Image processing, contrast enhancement, denoising, color correction.*

RÉSUMÉ COURT: *L'obstacle majeur dans le traitement des images sous marines résulte des phénomènes d'absorption et de diffusion dus aux propriétés optiques particulières de la lumière dans l'eau. Ces deux phénomènes auxquels s'ajoute le problème de turbidité, impose de travailler sur des images très bruitées, avec souvent une illumination non uniforme, des contrastes faibles, des couleurs atténuées... Cet article présente une nouvelle méthode de prétraitement des images sous marines. L'algorithme proposé qui ne nécessite ni paramétrage manuel ni information a priori, permet d'atténuer les défauts précédemment cités et d'améliorer de façon significative la qualité des images. L'approche utilisée est basée sur le rehaussement, l'éclairage, le bruit, les contrastes puis les couleurs sont corrigés séquentiellement.*

Mots-clés: *Traitement d'image, rehaussement de contraste, débruitage, compensation colorimétrique.*

1 INTRODUCTION

Underwater vehicles are used to survey the ocean floor, much often with acoustic sensors for their capability of remote sensing. Optical sensors have been introduced into these vehicles and the use of video is well integrated by the underwater community for short range operations. However, these vehicles are usually remotely operated by human operators : the automated processing and analysis of video data is only emerging and first suffers from a poor quality of the images due to specific propagation properties of the light in the water. To summarize underwater images suffer from limited range, non uniform lighting, low contrast, diminished colors, important blur... Moreover many parameters can modify the optical properties of the water and underwater images show large temporal and spatial variations. So, it is necessary to pre-process those images before using usual image processing methods. Today pre-processing methods typically only concentrate on non uniform lighting or color correction and often require additional knowledge of the environment: as depth, distance object/camera or water quality [6][7]. The algorithm proposed in this paper is a parameter-free algorithm which reduces underwater perturbations, and improves image quality

without using any knowledge and without any human parameter adjustment. It is composed of several successive independent processing steps which respectively correct non uniform illumination, suppress noise, enhance contrast and adjust colors [3][4][5][8]. The pre-processing step occurs before the segmentation. In most cases, a great improvement is observed while filtering, as it is showed by the edge detection criterion.

The remaining of the paper is organized as follows: Section II details underwater characteristic perturbations. Section III describes the complete filter bank composed of five different processes : homomorphic filtering to reduce illumination problems and to enhance the contrast, wavelet denoising and anisotropic filtering to cancel out the noise and enhance edges, contrast adjustment and, color compensation to suppress the predominant color. Section IV then details one by one those algorithms and explains our choices, Section V presents results on real underwater images. Finally Section VI shows qualitative improvements of our filter for the following step of segmentation.

2 UNDERWATER DEGRADATION

A major difficulty to process underwater images comes from light attenuation. Light attenuation limits the visibility distance, at about twenty meters in clear water and five meters or less in turbid water. The light attenuation process is caused by the absorption (which removes light energy) and scattering (which changes the direction of light path). Absorption and scattering effects are due to the water itself and to other components such as dissolved organic matter or small observable floating particles. Dealing with this difficulty, underwater imaging faces to many problems [1][6]: first the rapid attenuation of light requires attaching a light source to the vehicle providing the necessary lighting. Unfortunately, artificial lights tend to illuminate the scene in a non uniform fashion producing a bright spot in the center of the image and poorly illuminated area surrounding. Then the distance between the camera and the scene usually induced prominent blue or green color (the wavelength corresponding to the red color disappears in only few meters). Then, the floating particles highly variable in kind and concentration, increase absorption and scattering effects: they blur image features (forward scattering), modify colors and produce bright artifacts known as “marine snow”. At last the non stability of the underwater vehicle affects once again image contrast.

Our preprocessing filter has been assessed on natural underwater images with and without additional synthetic underwater degradations as proposed in [1]. Underwater perturbations we added are typical perturbations observed and they have been tested with varying degrees of severity. We simulate blur and unequal illumination using Jaffe and McGlamery’s model [14][16], gaussian and particles noise as additive contributions to the images and finally reduced color range by histogram operation.

3 ALGORITHM DESCRIPTION

The algorithm proposed corrects each underwater perturbations sequentially.

1. **Removing potential *moiré* effect.** A *moiré* effect has the appearance of a wavy repetitive pattern on the image. It is not an underwater perturbation, and it is often considered as aliasing phenomena. Sampling *moiré* mainly occurs in the analog to digital conversion process. *Moiré* pattern is removed via spectral analysis by detecting peaks in the Fourier transform and deleting them assuming that they represent the *moiré* effect [13]. Only few images suffer from *moiré* degradation but removing it is important because the following processes enhance contrast so enhance the *moiré* effect and consequently highly degrade results.
2. **Resizing and extending symmetrically the image to get a squared image whose size is a power of two.** Symmetric extension prevents from potential border effects and resizing to squared image speeds up the following process by enabling to use fast Fourier transform and fast wavelet transform algorithms.
3. **Converting color space from RGB to YCbCr (Luminance Chrominance).** This color space conversion allows us to work only on one channel instead of processing the three RGB channels. In YCbCr color space we process only the luminance channel (Y) corresponding to intensity component (gray scale image). The two other components correspond

to chroma color-difference. This step speeds up again all the following processings avoiding to process each time each RGB channels.

4. **Homomorphic filtering.** The homomorphic filtering is used to correct non uniform illumination and to enhance contrasts in the image. It's a frequency filtering, preferred to others techniques [4][8] because it corrects non uniform lighting and sharpens the edges at the same time.
5. **Wavelet denoising.** As explained in the previous part gaussian noise (i.e noise acquisition) is always present in natural images. This noise currently important is further amplified by homomorphic filtering. A step of denoising is so necessary to suppress it. This wavelet denoising method was preferred to many others algorithms [15] because of it performances of speed in comparison of its denoising quality.
6. **Anisotropic filtering.** Anisotropic filtering allows us to simplify image features to improve image segmentation. This filter smooths the image in homogeneous area but preserves edges and enhance them. It is used to smooth textures and reduce artifacts by deleting small edges amplified by homomorphic filtering.
7. **Adjusting image intensity.** This step increases contrast by adjusting image intensity values. It suppresses eventually outliers pixels to improve contrast stretching. It then stretches contrast to use the whole range of intensity channel and if necessary it saturates some low or high values.
8. **Converting from YCbCr to RGB and reverse symmetric extension.** After this step luminance channel has been preprocessed, so to regain colors we convert back the image the RGB space, and cut out the symmetric extension part of the image to recover the image with original size.
9. **Equalizing color mean.** In underwater imaging color channels are rarely balanced correctly. This step enables to suppress predominant color by equalizing RGB channels means. It is rather used to produce a more pleasant image than to better segmentation. Because segmentation is in general performed on gray image and color equalization does really not change the gray image.

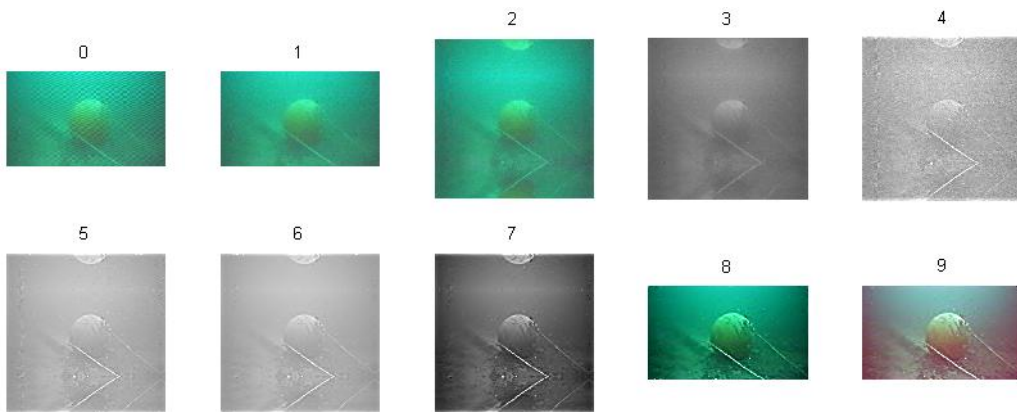


Fig. 1: The pre-processing filter step by step.

4 DETAILED ALGORITHMS

In this section we will detail the major algorithms used and justify our choices for those algorithms and for their parameters.

4.1 Homomorphic filtering

Considering the illumination-reflectance model, we assume that an image is a function of the product of the illumination and the reflectance as described by equation (Eq.1)

$$f(x, y) = i(x, y).r(x, y) \quad (1)$$

where $f(x, y)$ is the image sensed by the camera, $i(x, y)$ the illumination multiplicative factor, and $r(x, y)$ the reflectance function. If we take into account this model, we can assume that the illumination factor changes slowly through the view field, therefore it represents low frequencies in the Fourier transform of the image. On the contrary reflectance is associated with high frequency components. By multiplying these components by a high-pass filter we can then suppress the low frequencies i.e the non uniform illumination in the image. The algorithm can be decomposed as follows :

- Separation of the illumination and reflectance components by taking the logarithm of the image (Eq.2). The logarithm converts the multiplicative effect into an additive one.

$$g(x, y) = \ln (f(x, y)) = \ln (i(x, y).r(x, y)) = \ln (i(x, y)) + \ln (r(x, y)) \quad (2)$$

- Computation of the Fourier transform of the log-image (Eq.3)

$$G(w_x, w_y) = I(w_x, w_y) + R(w_x, w_y) \quad (3)$$

- High-pass filtering. The filter applied to the Fourier transform decreases the contribution of low frequencies (illumination) and also amplifies the contribution of mid and high frequencies (reflectance), sharpening the edges of the objects in the image (Eq.5).

$$S(w_x, w_y) = H(w_x, w_y).I(w_x, w_y) + H(w_x, w_y).R(w_x, w_y) \quad (4)$$

$$\text{with, } H(w_x, w_y) = (r_H - r_L).(1 - \exp(-(\frac{w_x^2 + w_y^2}{2\delta_w^2}))) + r_L$$

where $r_H = 2.5$ and $r_L = 0.5$ are the maximum and minimum coefficients values and δ_w a factor which controls the cutoff frequency. These parameters are selected empirically.

- Computation of the inverse Fourier transform to come back in the spatial domain and then taking the exponent to obtain the filtered image.

4.2 Wavelet denoising

Multiresolution decompositions have shown significant advantages in image denoising. For this denoising filter we choose a nearly symmetric orthogonal wavelet bases with a bivariate shrinkage exploiting interscale dependency [3]. We prefer this filter to the Kovesi [9] used by [8] because even if visual appearance seems better it is very slow and often poor than the use of both wavelet filter and anisotropic filter. This wavelet denoising gives very good results compared to other denoising methods because, unlike other methods, it does not assume that the coefficients are independent. Indeed wavelet coefficients in natural image have significant dependencies. Moreover the computation time is very short. The algorithm can be decomposed as follows :

For the next and similarly to [3] we defined the $(g)_+$ function as :

$$(g)_+ = \begin{cases} 0 & \text{if } g < 0 \\ g & \text{otherwise} \end{cases}$$

- Multiscale decomposition of the image corrupted by gaussian noise using wavelet transform. We use Farras wavelet base (nearly symmetric filters for orthogonal 2-channel perfect reconstruction filter bank).
- Estimation of noise variance σ_n^2 using (Eq.5).

$$\sigma_n^2 = \text{median}(|y_i|)/0.6745, \quad y_i \in \text{subband HH.} \quad (5)$$

- For each subband of each level except for the lowpass residual : Computation of the signal variance using (Eq.6) and modification of the noisy wavelet coefficient according to the (Eq.7).

$$\sigma = \sqrt{(\sigma_y^2 - \sigma_n^2)_+}, \quad \text{where } \sigma_y^2 = \frac{1}{M} \sum_{y_i \in N(k)} y_i^2, \quad M \text{ the size of the neighborhood } N(k) \quad (6)$$

$$y_i = \frac{(\sqrt{y_i^2 + y_{i+1}^2} - \frac{\sqrt{3}\sigma_n}{\sigma})_+}{\sqrt{y_i^2 + y_{i+1}^2}} \cdot y_i, \quad \text{where } y_i \text{ is the child and } y_{i+1} \text{ its parent.} \quad (7)$$

- Inversion of the multiscale decomposition to reconstruct the filtered image.

4.3 Anisotropic filtering

This filter removes or attenuates unwanted artifacts and remaining noise. The anisotropic diffusion algorithm is used to reduce noise and prepare the segmentation step. It allows to smooth image in homogeneous areas but it preserves and even enhances the edges in the image. We follow the algorithm proposed by Perona and Malik [5]. This algorithm is automatic so it uses constant parameters selected manually. The previous step of denoising is very important to obtain good results with anisotropic filtering. It is the association of wavelet denoising and anisotropic filtering which gives such results. Anisotropic algorithm is usually used as long as result is not satisfactory. In our case we loop only few times set to constant value, to preserve a short computation time. One loop of the algorithm can be decomposed as follows : For each pixel

- Computation of the nearest-neighbor differences and computation of the diffusion coefficient in the four directions North, South, East, West. Many possibilities exist for this calculation, the easiest way is as follows:

$$\begin{aligned} \nabla_N I_{i,j} &= I_{i-1,j} - I_{i,j}, & c_{N_{i,j}} &= g(|\nabla_N I_{i,j}|) \\ \nabla_S I_{i,j} &= I_{i+1,j} - I_{i,j}, & c_{S_{i,j}} &= g(|\nabla_S I_{i,j}|) \\ \nabla_E I_{i,j} &= I_{i,j+1} - I_{i,j}, & c_{E_{i,j}} &= g(|\nabla_E I_{i,j}|) \\ \nabla_W I_{i,j} &= I_{i,j-1} - I_{i,j}, & c_{W_{i,j}} &= g(|\nabla_W I_{i,j}|) \end{aligned} \quad (8)$$

Where the function g is defined as : $g(\nabla I) = e^{-(\|\frac{\nabla I}{K}\|)^2}$ and with K set to 0.1. This diffusion function favors high contrast edges over low contrast ones.

- Modification of the pixel value using (Eq.9)

$$I_{i,j} = I_{i,j} + \lambda [c_N \cdot \nabla_N I + c_S \cdot \nabla_S I + c_E \cdot \nabla_E I + c_W \cdot \nabla_W I]_{i,j} \quad \text{with } 0 \leq \lambda \leq 1/4 \quad (9)$$

4.4 Contrast stretching and color correction

- Contrast stretching (often called normalization) is a simple image enhancement technique that attempts to improve the contrast in an image by ‘stretching’ the range of intensity values it contains to a desired range of values, e.g. the full range of pixel values that the image concerned allows as Eq.10.

$$I_{i,j} = \begin{cases} \frac{I_{i,j} - \min_I}{\max_I - \min_I} & \text{if } 0 < I_{i,j} < 1 \\ 0 & \text{if } 0 > I_{i,j} \\ 1 & \text{if } 1 < I_{i,j} \end{cases} \quad (10)$$

- Color correction is performed by equalizing each color means. In underwater image colors are rarely balanced correctly, this processing step suppresses prominent blue or green color without taking into account absorption phenomena (results using absorption’s law are better but a priori knowledge is required [7]). This algorithm is a linear translation of the histogram. We add to each pixel the difference between desired mean value and the mean of the channel. We do that for each RGB channel. Values can’t overrun the interval of color. Our method is a compromise that produces at the output a more pleasant image without any knowledge on the environment or any calculation.

5 RESULTS

The computation time to pre-process a color image 512×512 is about 1.5 seconds on pentium 4, 3Ghz using Matlab 7.0. The prefilter is studied to be very fast and is optimized for Matlab.



Fig. 2: Pairs of images before (left) and after preprocessing (right), the first four images come from the web, and others are images extracted from TOPVISION videos*.



Fig. 3: Images with additional underwater noise (average blur, gaussian white noise, spot effect and color range reduced) on the left and the same images after pre-processing on the right.

* The information contained in this publication are derived from data property of the French State that have been provided by the GESMA (Groupe d'Etudes Sous-Marines de l'Atlantique) within TOPVISION project coordinated by Thales Underwater Systems SAS. This project is related to Techno-Vision Programme launched by french Ministry of Research and french Ministry of Defense.

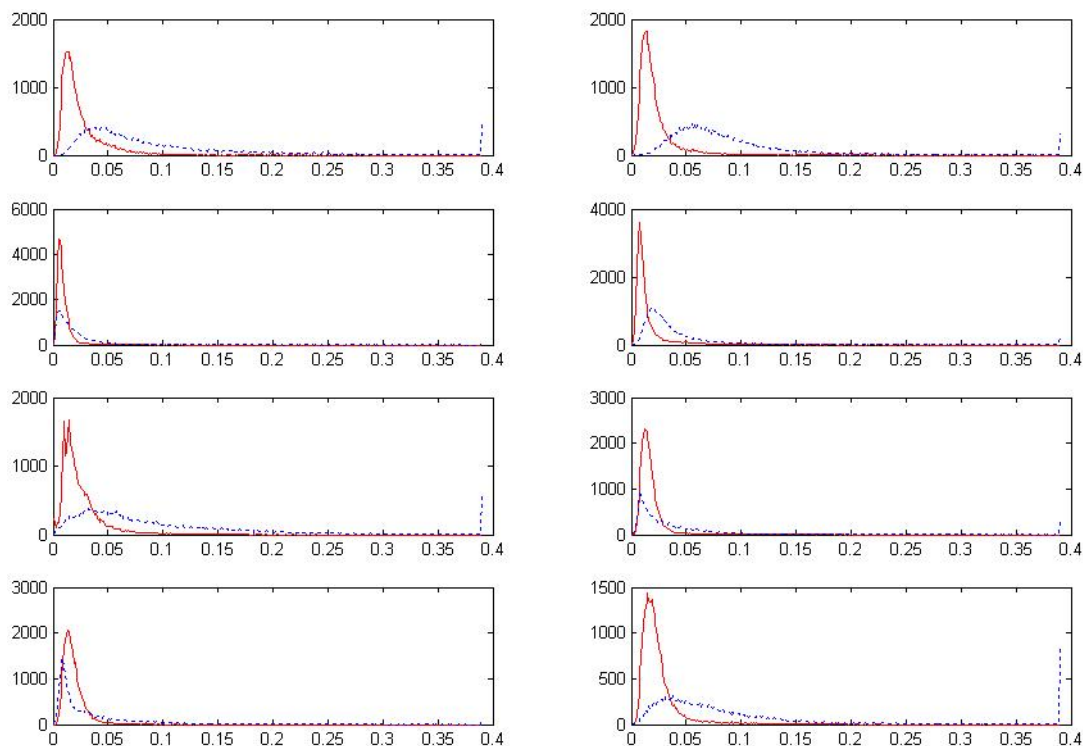


Fig. 4: Gradient magnitude histogram of the eight previous images (Fig. 2) before pre-processing (solid line) and after pre-processing (dotted line).

6 ROBUSTNESS

This pre-processing algorithm is the preliminary step of a feature point extraction or an edge detection. In order to illustrate our results we use gradient magnitude histogram Fig.4 and Fig. 5 (gradient histograms are plotted between 0 and 0.4 so each value greater than 0.4 is assigned to 0.4).

These histograms show that gradient values are larger after pre-processing. Moreover the peak of small gradient is greatly attenuated. Consequently the edges are enhanced and well separated from noise, and it is easier for thresholding. Also we have assess quality of our restoration procedure using the robustness criterion of [11]. This criterion assumes that a well contrasted and noise-free image has a distribution of the gradient magnitude histogram close to exponential, it attributes a mark from zero to one. Following results presented in the table are the mean values of the criterion on the previous eight images before and after pre-processing. Each value corresponds to a curve Fig.5.

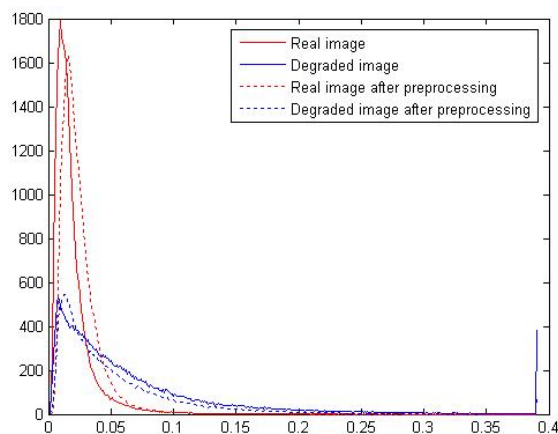


Fig. 5: Mean of gradient magnitude histogram on the eight previous images with and without additional synthetic degradations.

	Before	After
Without additional synthetic degradations	0.398	0.539
With additional synthetic degradations	0.313	0.538

7 CONCLUSION AND FUTURE WORK.

In this paper we present a novel underwater pre-processing algorithm. This algorithm is automatic and requires no parameter adjustment and no a priori knowledge of the acquisition conditions. This is because functions evaluate their parameters or use pre-adjusted default values. This algorithm is fast and can be improved with a translation in C language. We have shown that this filter greatly enhances edge detection and also often increase image visual quality. We have illustrated those enhancements on edge detection comparing gradient magnitude histograms and using a robustness criterion.

Many adjustments can still be done to improve the whole pre-processing algorithms. We will be able to investigate notably curvelets-based methods for contrast enhancement and image denoising which seems to give very good results [10], and also deconvolution methods. Inverse filtering gives good results but generally requires a priori knowledge on the environment [12]. Our filtering needs no parameters adjustment so it can be used systematically on underwater images before every pre-processing algorithms.

References

- [1] M. Arredondo and K. Lebart, "A methodology for the systematic assessment of underwater video processing algorithms," *IEEE OCEANS 05 Europe Conference*. June 2005.
- [2] A. Farras Abdelmour, Ivan W. Selesnick, "Symmetric Nearly Orthogonal, and Orthogonal Nearly Symmetric Wavelets," *Research Report* February 2003.
- [3] L. Sendur and I. W. Selesnick, "Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency," *IEEE Trans on Signal Processing*, November 2002.
- [4] R. Garcia, T. Nicosevici and X. Cufi, "On the way to solve lighting problems in underwater imaging," *IEEE OCEAN 02*, 1018-1024, October 2002.
- [5] P. Perona and J. Malik, "Scale space and edge detection using anisotropic diffusion," *IEEE Trans on Pattern Analysis and Machine Intelligence*, 629-639, July 1990.
- [6] A. O. Olmos Antillon, "Detecting underwater man-made objects in unconstrained video image," *Thesis*, Department of Computing and Electrical Engineering, December 2002
- [7] J. Ahlen, "Color correction of underwater images using spectral data," *Thesis*, Uppsala University, Centre for Image Analysis, 2005.
- [8] A. Arnold-Bos, J. P. Malkasse and Gilles Kervern, "Towards a model-free denoising of underwater optical image," *IEEE OCEANS 05 EUROPE*, June 2005.
- [9] P. Kovesi, "Phase preserving denoising of images," *Proceeding of the Australian Pattern Recognition Society Conference*, December 1999.
- [10] J. L. Starck, Fionn Murtagh, Emmanuel J. Candés and David L. Donoho, "The curvelet transform for image denoising," *IEEE Transactions on image processing*, June 2002.
- [11] A. Arnold-Bos, J. P. Malkasse and Gilles Kervern, "A preprocessing framework for automatic underwater images denoising," *European Conference on Propagation and Systems*, March 2005.
- [12] Z. Liu, Y. Yu, K. Zhang and H. Huang, "Underwater image transmission and blurred image restoration," *SPIE Journal of optical Engineering*, June 2001.
- [13] D. N. Sidorov and Anil C. Kokaram, "Suppression of moiré patterns via spectral analysis," *Proceedings of SPIE in Visual Communications and Image Processing*, January 2002.
- [14] J.S Jaffe, "Computer modeling and the design of optimal underwater imaging systems," *IEEE Journal of oceanic Eng.*, April 1990.
- [15] A. Buades, B. Coll, J.M Morel, "A review of image denoising algorithms, with a new one," *Multiscale Modeling and Simulation (SIAM interdisciplinary journal)*, 2005.
- [16] B.L McGlamery, "A computer model for underwater camera systems," *SPIE Ocean Optics*, 1979.