manuscript No.
(will be inserted by the editor)

RBG ordering based on image metrics

José Luis Vázquez Noguera · Christian E. Schaerer · Jacques Facon · Horacio Legal Ayala

Received: date / Revised: date

Resumen In color ordering literature, the lexicographical ordering and its variants are the most used methods and an usual problem is the "a priori" definition of the most important color component **(NE...)** This paper proposes a new ordering criteria, which assign a weight to every component in accordance with the metrics applied to it, with this criteria we pursue to avoid arbitrary definitions and build the ordering based on image-based information. **... Obtener los terminos correctos de fuentes oficiales (filtrado de imagenes, mejora de constraste...)** Applications used for validation of the proposal are: image filtering, contrast improvement and textures characterization for later classification. The results using the proposed ordering in different applications are better in most cases compared to different ordering methods of the state of the art.

1. Introduction

Digital image processing on color images resemble human vision [?], which is **NE...**, on the other hand grayscale images and binary images contribute less information because of their use of single dimentional intensities and binary values, white and black. At its inception, digital image processing algorithms where only develop for grayscale and binary images because of the limited computing power available at the time and their high cost, these conditions demands to reduce the visual information into a single plane. **ver citas en inglés...**

Valuable information can be obtain from grayscale images, for example an object boundaries could been detected from sudden changes in intensities. By calculating the gradient **on every pixel (Added)** we could extract every object from the image, however unexpected reflections could produce errors on the boundaries detections. Reflections, lighting effects and the lack of chromatic information limit the efficiency of many grayscale algorithms [?]. Considering these ideas and the new improvements on computational resources, as image processing specialized processors **review**, a lot of grayscale images are been generalized into color images [?].

Color spaces are formalisms **(alternative)** that allow the definition of color and their properties for proper manipulation [? ?].

The most used color space on screens is the RGB color space, which it is based on the tristimulus model and aditive syntesis of color **review from cite**. In the RBG color space a color, where the colors used are red, green and blue, is defined as 3-tuples where the scalar value on every component measures its influence on the mix [?]. In the CMY color space, cyan, magenta and yellow also known as secondary colors, represents the substractive syntesis of color [?]. The CMYK color space, used in printers[?], extends the previous color space adding the K component which represents the maximum value between the 3 secondary values[?] **NE**.

The XYZ color space has been introduced because there are some colors that can only be represented by a negative value of stimuli **review** and it is a linear transformation over the RGB color space[?]. This color space

José Luis Vázquez Noguera · Horacio Legal Ayala · Christian E. Schaerer
Polytechnic Faculty, National University of Asuncion - San Lorenzo, Paraguay
E-mail: {jlvazquez,hlegal,cschaer}@pol.una.py

Jacques Facon
PPGIa - PUCPR-Pontifícia Universidade Catolica do Paraná - Curitiba - Pr, Brazil
E-mail: facon@ppgia.pucpr.br

is used when the color representation is independent of the hardware.

The L*a*b* is a 3 component color space where the L* represents the luminosity from black to white, the a* measures from red to green and the b* from yellow to blue[**?**] **review from cites**.

The color spaces HSI, HLS, HSV and their variants are the color spaces that resembles the human vision the most because they are based on luminosity, saturation and hue perseptions[**?**].

A lot of aplications needs color ordering, as noise reduction, contrast enhancement, borders detection and segmentation of color images**review terminos**[**?**]. Because of *multi*-dimentional nature of color representation their is no natural order and **NE**.

This work presents a new ordering for the RBG color space based on the metrics associated to each component. The proposed ordering is compared with the state of art orderings on noise reduction, contrast enhancement and texture characterization. **IMP REVIEW**

The paper is organized as follows. The second section presents the current state of the art on the matter. The third section presents the fundamentas of color image filtering as main conceps of image filtering, ordering and mathematics morphology. The fourth section presents the proposed ordering. The fifth section presents the experimental results of the comparisons with the current state of the art on aplications as noise reduction, contrast enhancement and texture characterization. Finally, the sixth and last section presents our conclutions for future works.**review**.

2. State of the art methods.??

The extension of the filter order to color images requires on the one hand select the color space in which the image is processed and in the other establish an order in this color space. To establish an arrangement they have worked in different color spaces, among which we can mention, L*a*b* color spaces [**?**], HLS [**?**], CIELAB [**?**], HSI [**?**], HSV [**?**] and the RGB color space [**? ? ?**].

....

Mathematical morphology borns in 1964 from the colaboration of Georges Matheron and Jean Serra at École de Mines, Paris [**?**] and currently ir reach is as broad as image processing itself. A few examples of aplications are image segmentation, restoration, border detections, contrast enhacement, texture analysis, compression, etc. [**?**]. Erosion and dilation are the most basic operations where **NE reticulo?**[**?**]. Erosion is the minimum and dilation is the maximum of a window **review for better terminology** called structural element and from this two all other operations are builded. To generalize the mathematical morphology to color images is required an ordering between colors to be able to determine a minimum and a maximum inside the structural element.

Recent publications presented generalizations of mathematical morphology [**? ? ? ? ? ? ? ? ? ? ? ? ?**]. In RGB the interlace bit ordering **review...!!** has been proved to be efficient on color image filtering [**?**]. For a more in depth analysis on mathematical morphology methods, we suggest Aptoula and Lefevre [**?**].

Generally, that is for many color spaces, the lexicographical arrangement is one of the most used in the literature [**? ?**], since it has desired theoretical properties and can easily customize the way you are going to compare the components of the image.

Louverdis et. al. [**?**] and Vardavoulia et. al. [**?**] presents a lexicographic ordering in HSV, while Louverdis et. al. [**?**] uses the same ordering and color space to develop a novel morphologic method for shape and size analisys on granular images. Angulo and Serra [**?**] discuss over lexicographic ordering on RGB and HLS for JPEG image compression. Ortiz et. al. [**?**] uses the I→H→S ($Href = 0$) lexicographical ordering for gaussian noise elimination.

The lexicographical ordering suffers from a serious inconvinient. More precisely, the final result of most lexicographical ordering are highly biased towards the first components where the last components are virtually ignored [**?**] **over reaction?**.

In order to improve the tuning degree of influence of each component of the vector in the comparison result they were proposed variations lexicographic ordering. A group of variants is based on the use of an additional component for comparison.

Angulo [**?**] and Sartor et. al. [**?**] located in the first position of the lexicographical cascade a distance measurement to a reference vector. Comer et. al. [**?**] used a Euclidean norm as a method of sorting pixels, i.e. the reference pixel color is the black (0,0,0) in RGB.Two RGB colors can be visually practically identical but differ in value standard, or distance to a reference color, as well as two different colors have the same norm so it is not recommended to use this strategy. In the L*a*b space there is a defined distance measure, about the origin $L = 0$, $a = 0$ and $b = 0$ which it is widely used for the evaluation of the quality of reproduction in color, or in techniques of understanding of color images. [**?**].

Other types of arrangement seeking the extension of lexicographic ordering is to use a parameter α defined by the user so as to modify the degree of influence of the first componente [**? ?**]. Even with the changes in the lexicographic ordering, the criteria for choosing which component will have higher priority in the comparison, and the value α are usually arbitrary. Gao et. al. [?] tries to solve this problem by presenting an approach of adaptive lexicographic ordering. In order to avoid the most subjective user intervention, it would be of great importance that the arbitrary criteria of lexicographic order and its variants can be eliminated or reduced.

Bouchet et. to the. [?] uses fuzzy logic so that the three color components have the same weighting in the sorting, although it is desirable that the priority of the components of the vector representing the image are audited self-image information, not being exactly the same in all cases. Benavent et. al. [?] presents a sorting method that is dependent of the image and ordering the colors according to the probability density of the appearance of colors in the image.

The main difference of this proposal, with those presented in the state of art, is the extraction of information for each component of RGB color in a specific domain image. This information is extracted by a vector of weights which are previously calculated by a function applied to each of the RGB components of the color.

3. Fundamentals of color image filtering

This section presents the formal formulations of the theoretical concepts behind ordering filters and their extenssions for color images, vector arrangement and morphological mathematics.

3.1. RGB images

In general, an image is a function $f : \mathbb{Z}^2 \to \mathbb{Z}^n$. Each pair $(u, v) \in \mathbb{Z}^2$ is a pixel, and $f(u, v) \in \mathbb{Z}^n$ is the color image in the pixel (u, v). In particular, a RGB image (red, green and blue) with a color depth of k bits is, $f(u, v) = (R, G, B)$, where $R, G, B \in \{0, 1, ..., 2^k - 1\}$ is the intensity of each component and $f(u, v)$ is the color resulting from mixing these components in the pixel (u, v). The image f can be digitally presented as an array $M \times N \times 3$, where each pixel (u, v) has as its value a triplet (R, G, B) [?]. An RGB image can be seen as a "stack" of three images in grayscale (see Figure 1) that, when fed to the red, green and blue entries of the color monitor, produces a color image on the screen [?].

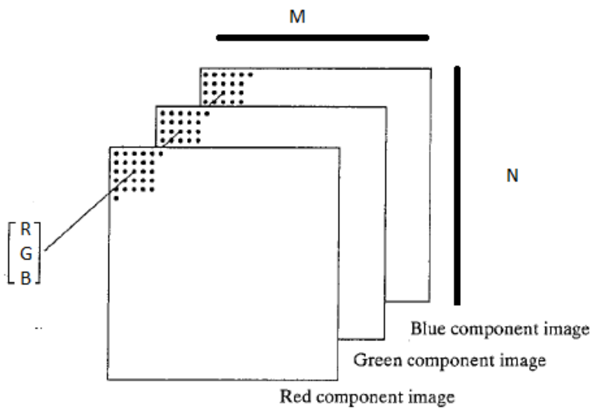


Figura 1: RGB Image

3.2. Image filtering

Image filtering covers all techniques in image processing, that from an input image, another image is obtained in which are removed, emphasized or highlighted some features of the input image. A filter F of a digital image color f can be expressed as:

$$g(u, v) = F\{f(u, v)\} \quad (1)$$

where $f(u, v)$ is a color of the input image, $g(u, v)$ is a color of the image output and F is the filter defined over a window of the pixel (u, v).

Ordering filters are nonlinear neighborhood operations, where the function return value for each pixel is calculated from its neighborhood. The idea is to move a window centered on the pixel, either a rectangle (usually a rectangle of odd sides) or other shape on an given image.(Figure 2).

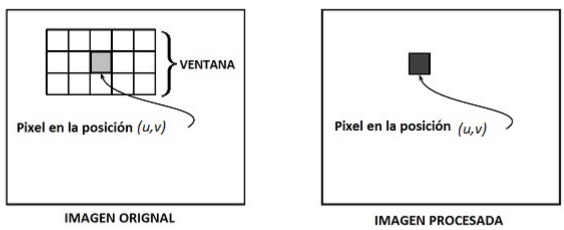


Figura 2: Digital image filtering.

For example, a pixel of the new image can be a result of obtaining the median, minimum or maximum of the colors ordered in the window of the processed image. The combination of the window and the function is called filter.

3.3. Ordering

The concept of order plays a important role in the use of a ordering filter, or to define the basic operations of mathematical morphology. A detailed study of the theory of order can be found at [?].

According to the article [?], vector ordering techniques can be classified into the following groups:

- Marginal Ordering (M Ordering): The marginal ordering compares each color component independently.
- Conditional Ordering (C Ordering): The vectors are sorted by some marginal component selected sequentially according to different conditions. The lexicographic ordering is a well-known example of C Sorting employing all available components of the given vectors.
- Partial Ordering (P Ordering): This ordering is based on the partition of the vectors in groups of equivalence such that between groups there exist an order. In this case, "partial" is an abuse of terminology, as there are total ordering that belong to this particular class.
- Reduced Ordering (R Ordering): The vectors are first reduced to scalar values and then classified according to their naturally scalar order. For example, a R sorting in $\mathbb{Z}^n$ could be first define a transformation $T : \mathbb{Z}^n \to \mathbb{R}$ and then sort the colors with respect to the scalar order of his projection in $\mathbb{Z}^n$ by T.

In practice there are two general methods for processing color images: marginal and vectorial.

The marginal processing consists in the independent processing of each component of the image. Despite its simplicity, the marginal processing has two disadvantages [?]:

- The correlation between components is completely ignored.
- Creates false colors after processing.

The use of marginal processing is inadequate for images with highly correlated components (eg RGB color images) [?]. Therefore this work focuses on the vectorial processing that will be explained below.

Vectorial methods processes all available components, globally and simultaneously. Since vectors (way of representing a color) are considered as the new processing units, the correlation between the different components is no longer ignored. However, its most significant drawback is mainly the need to adapt existing algorithms in order to accommodate vectorial data [?].

The vectorial processing can have two approaches:

- Based on the preorder relationship .
- Based on the order relationship

The approach based on preorder relationship, is the set of approaches that do not satisfy the antisymmetric property. Thus different colors can eventually become equivalent. So to solve existing ambiguities, additional actions are necessary. The main method of sorting of this approach is based on the Reduced Sorting (R Sorting), where colors are reduced to scalar values corresponding to their norm, or their distance to a some reference color.

The approach based on order relationship, in the same way can be partial or total. If the relationship is partial, there will be colors that can not be compared.

The total order relation has two main advantages. First, all colors are comparable, and second, there are no distinct colors that can be equivalent. Because of this, most works are based on approaches of total order relationship. [?]. In particular, the lexicographical order (C Sorting), along with its variants is among the most widely deployed options.

3.4. Morphological mathematics

Mathematical morphological operations are based on two basic operators: dilatation and erosion. Both operators are filters that can be defined from the minimum and maximum within a window called structuring element [?]. From erosion and dilation can extend all the morphological mathematics. Morphological operators must comply certain theoretically properties, such as anti-extensive or extensive, idempotent, homotopic and growing [?].

Given a digital image f and B window, called structuring element. The erosion (ε) and the dilation (δ) of the image f by B can be expressed as:

$$\varepsilon(f,B)(u,v) = \min_{(s,t)\in B}\{f(u-s,v-t)+B(s,t)\} \qquad (2)$$

$$\delta(f,B)(u,v) = \max_{(s,t)\in B}\{f(u+s,v+t)-B(s,t)\} \qquad (3)$$

We called $\delta(f,B)$ and $\varepsilon(f,B)$ as dilation and erosion respectively for all pixels (u,v) of the image f. The combination of erosion and dilation produces other operators such as opening and closing. The opening softens the bright regions of the image. The closure softens the dark areas of the image. The opening $\circ$ and the closing $\bullet$ of f by B are defined based on dilation and erosion as follows:

$$f \circ B = \delta(\varepsilon(f,B),B), \tag{4}$$

$$f \bullet B = \varepsilon(\delta(f,B),B). \tag{5}$$

Based on the opening and closing is defined the top-hat transform. The clear top-hat transform (WTH) could extract bright regions of the image and the dark top-hat transform (BTH) could extract dark areas . The transformed WTH and BTH are defined for an image f as follows:

$$WTH(f) = f - f \circ B, \tag{6}$$

$$BTH(f) = f \bullet B - f. \tag{7}$$

The extent of the mathematical morphological color images is still an open problem [?], mainly by the drawback that not exist a natural order between vectors, and that colors can be represented in different ways (forming different color spaces). In the absence of a natural order between colors it is not easy to define the basic operators of erosion and dilation.

The following section presented an ordering strategy of RGB colors, given metric extracted from each color component, so as to establish weights to components from own image information.

4. Proposed Sorting

A histogram function is defined from the RGB image, which corresponds to the frequency distribution of the values that can take a picture f, either in a plane or in three dimensions (R, G, B). The histogram of the j-th component of the color image f (R, G o B) is a discrete function $h^D_{f_j}$ defined as:

$$h^D_{f_j}(i) = n_i, \tag{8}$$

where i represents the $i - esimo$ intensity level in the range $\{0, 1, ..., 2^k - 1\}$ of the component j, and n_i is the number of pixels in the image f whose intensity level is i in the component j within the domain D (subset of pixels (u, v) inside the image f).

The probability of occurrence $p^D_{f_j}(i)$ of each level of intensity i in the component j of the image f in the domain D is defined as:

$$p^D_{f_j}(i) = \frac{h^D_{f_j}(i)}{n}, \tag{9}$$

Donde $n = n_0 + n_1 + ... + n_{255}$, es decir la cantidad total de pixeles de la imagen f dentro del dominio D.

So to avoid giving the highest priority to a component of the vector representing the color, a new value (R, G, B) in the first position of the corresponding lexicographical cascade transformation obtained from metric associated to each component is placed. RGB colors are reduced to a scalar value. For this purpose, is first defined a transformation $T : \mathbb{Z}^3 \to \mathbb{R}$ and then ordered te colors with respect to the order of his scalar projection in $\mathbb{Z}^3$ by T. A $C = (C_1, C_2, C_3)$ color reduction is achieved through of the color inner product C with a $w = (w_1, w_2, w_3)$ weights vector, that is to say:

$$T(C) = \sum_{l=1}^{3}(w_l \times C_l) \tag{10}$$

wheree l is the color component index C and $w_l \in \mathbb{R}$.

Two colors, $C = (C_1, C_2, C_3)$ and $C' = (C_1^{'}, C_2^{'}, C_3^{'})$, with $C \neq C'$, may have the same transformation, that is to say $T(C) = T(C')$. Therefore, the transformation is used as the first component of the lexicographical order:

$$C \leq C' \Leftrightarrow [T(C), C_1, C_2, C_3] \leq_L [T(C'), C_1^{'}, C_2^{'}, C_3^{'}] \tag{11}$$

wheree $\leq_L$ shows the relationship $\leq$ according to the lexicographical order.

Opportunely, could vary the order of priority of the color components after the transformation. Vector values of w are obtained by applying a function $\phi \in \mathbb{R}$ on the histogram of each component in a domain D of the image f, that is to say $w_1 = \phi(h^D_{f_1})$, $w_2 = \phi(h^D_{f_2})$, $w_3 = \phi(h^D_{f_3})$, with $f_1 =$ component R, $f_2 =$ component G y $f_3 =$ component B.

The function ϕ can be obtained from applying any metric (statistical, for example) to the histogram of each component (R, G, B), so as to give greater weight to that component whose metric has greater value in a specific domain D (can be the entire image or part of it).

5. Experimental results

This section will hold a series of comparative tests, in order to measure the relative performance of different ordering methods of the state of the art together with the proposed ordering, in three image processing applications. The selected applications were noise removal, contrast stretching and characterization textures for subsequent classification. More precisely, the ordering methods involving in the tests were: the classic lexicographical ordering, the ordering α-lexicographical [?], the ordering α-module lexicographical [?], I→H→S lexicographical ordering, $(Href = 0)$ [?], the euclidean distance to color $(0, 0, 0)$ in the color space L*a*b* and RGB [?], and the bit interlaced [?]. All images used on different tests were of 8 bits.

The function ϕ applied to the histogram of each component j of the image f in all tests are:

- Average (Me): Is the sum of all intensity levels i listed in the domain D divided the total amount n of pixels that are in D:

$$Me(h_{f_j}^D) = \sum_{i=0}^{255} \frac{i \times h_{f_j}^D(i)}{n}, \tag{12}$$

Where $n = n_0 + n_1 + ... + n_{255}$.

- Minimum (Min): is the lowest level of intensity i in the domain D:

$$Min(h_{f_j}^D) = \text{mín}\{i | h_{f_j}^D(i) > 0\} \tag{13}$$

- Maximum(Max): is the highest level of intensity i in the domain D:

$$Max(h_{f_j}^D) = \text{máx}\{i | h_{f_j}^D(i) > 0\} \tag{14}$$

- Moda Minimum $(minM_o)$: is the lowest level of intensity i that appears more times in the domain D, that is the lowest level of intensity i which has greater $p_{f_j}^D(i)$:

$$minM_o(h_{f_j}^D) = \text{mín}\{i | h_{f_j}^D(i) \geq h_{f_j}^D(i'), \forall i \neq i'\} \tag{15}$$

- Moda Maximum $(maxM_o)$: is the highest level of intensity i that appears more times in the domain D, that is the highest level of intensity i which has greater $p_{f_j}^D(i)$:

$$maxM_o(h_{f_j}^D) = \text{máx}\{i | h_{f_j}^D(i) \geq h_{f_j}^D(i'), \forall i \neq i'\} \tag{16}$$

- Variance (Var): is the variance of intensity levels i in the domain D:

$$Var(h_{f_j}^D) = \sum_{i=0}^{255} \frac{h_{f_j}^D(i) \times (i - Me(h_{f_j}^D))^2}{n} \tag{17}$$

- Smoothness(R): Measure of relative softness intensity in the domain D:

$$R(h_{f_j}^D) = 1 - \frac{1}{1 + Var(h_{f_j}^D)} \tag{18}$$

A parameter to be defined is the domain to be considered for the calculation of the weights w_l. Domain distributions were used for different applications are discussed below.

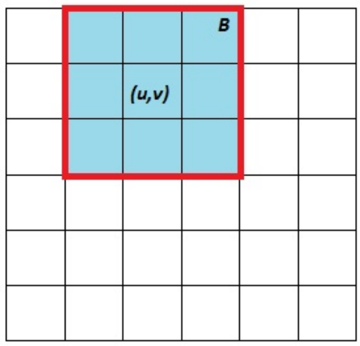


Figura 3: B Neighborhood of size 3×3 centered in the pixel (u, v)

5.1. Neighborhood as Domain

The domain D where the function ϕ is applied (applied to the histogram of each component $h_{f_j}^D$) is the window itself B (called structuring element for morphological mathematics) where the operation of the nonlinear filter is applied. In the figure 3 can see a domain D corresponding to a neighborhood B of 3×3 size centered in the (u, v) pixel .

5.2. Division of the image into subregions

The image f is divided into subregions $W_1, W_2, \ldots W_x$, so as to obtain local information from the image. Let B a window or structuring element, the domain D corresponding to the window B centered on (u, v) is the set of subregions $W_{\{1,2,\ldots,x\}}$,that touch some pixel of B.

In the figure 4 the image is divided into 4 subregions: W_1, W_2, W_3, W_4. The region delimited by the window B is shaded. As you can see, the domain D on which w_l weights are calculated for B will be the area corresponding to the W_1 sub-region .

Note that the filter window does not have to to be of equalsized to sub-regions, as in this case. In the Figure 5 can be seen as the domain D of which shall be calculated weights belong to the union of the sub-regions W_1 and W_2, as the window B touches both. This is done to avoid that when comparing two identical colors may have different values, affected by the weights that come from two different subregions.

In the case the user choose to have only one subregion , that is not divide the image f, the weights will be calculated considering the whole picture f as domain D.

In our tests, the input images of size $M \times N$ pixels, are divided into $W_{\{1,2,\ldots,x\}}$ subregions of $\lfloor \frac{M}{M'} \rfloor$ rows and $\lfloor \frac{N}{N'} \rfloor$ columns, where $\lfloor . \rfloor$ denotes the floor function.

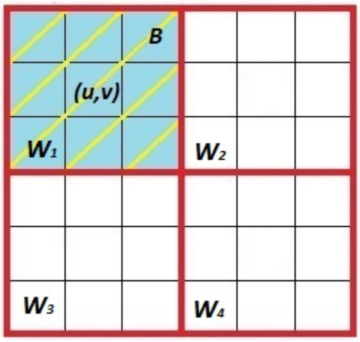


Figura 4: Domain when the window touches one subregion

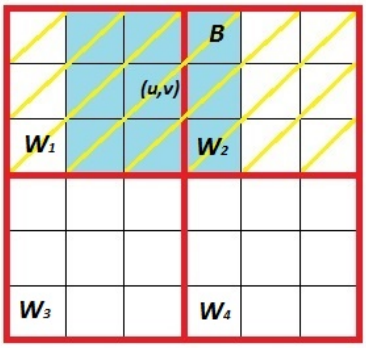


Figura 5: Domain when the window touches more than one subregion.

Thus, we have a new matrix of M' rows y N' columns, whose element is a subregion W_l.

5.3. Application 1: Noise Removal

Noise is a term used to refer to unwanted changes that may suffer a signal of any kind of nature during his capture, storage, transmission, processing or conversion [?].

Noise in images is an undesirable product that adds misinformation. Noise occurs in digital images in the form of random variations in brightness or color information. Several mathematical models have been developed to simulate the generation of different types of existing noise.

5.3.1. Used noises

Given an input image f, the image f' resulting from contaminating f with some kind of noise and a vector $z = (z_1, z_2, z_3)$ in each element z_l corresponds to random variable; is defined the main noise models as follows:

- Gaussian noise: It is an additive statistical noise with a gaussian density function probability [?]. Gaussian noise is expressed as follows:

$$f'(x, y) = f(x, y) + z \tag{19}$$

where each component z_l is a random variable with normal distribution, μ average, σ^2 variance and it represents the value of noise added.

- Speckle noise: is a multiplicative noise with a uniform density function of probability, defined as follows:

$$f'(x, y) = f(x, y) + z * f(x, y) \tag{20}$$

where the operator $*$ symbolizes the Hadamard product or element by element. Each item z_l is a random variable uniformly distributed with average μ and variance σ^2.

- Salt and pepper noise: this noise, unlike the Gaussian and Speckle noise, is not additive and multiplicative respect to the values of the original image. In the images affected with salt and pepper noise original values are replaced by bright values (salt) or dark values (pepper), that correspond to pulses inside to the signal.
The salt pixels have the minimum possible value (zero) and the pixel values pepper the maximum value possible ($2^k - 1$, where k is the number of bits used to represent the intensity of each color component). The salt and pepper noise does not affect all pixels within an image, as with Gaussian and Speckle noises. The number of pixels in an image that are affected by salt and pepper noise parameter depends on the probability of noise p, which it is in the range $[0, 1]$.
The salt and pepper noise is modeled as follows:

$$f'(x, y) = \begin{cases} s & \text{, with a probability } p/2 \\ r & \text{, with a probability } p/2 \\ f(x, y) & \text{, with a probability } 1 - p \end{cases} \tag{21}$$

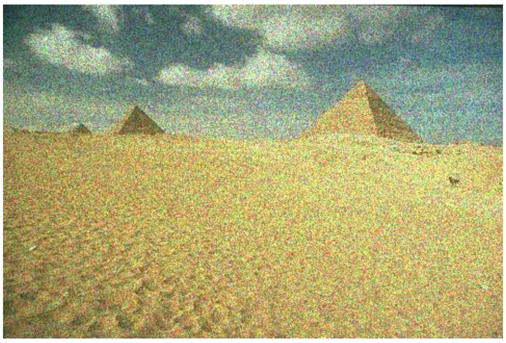
where:
$s = (0, 0, 0)$ represents the salt noise,
$r = (2^k - 1, 2^k - 1, 2^k - 1)$ represents the pepper noise.

In the Figure 6(a) can see an image that is contaminated with Gaussian noise (6(b)), speckle noise (6(c)), with salt and pepper noise (6(d)).

So to evaluate the filter with the different types of ordering, is proposed to use a statistical metric used to measure how close they are forecasts or predictions of real results [?].

(a) Imagen original

(b) Image with gaussian noise ($\mu = 0; \sigma^2 = 0{,}05$)



(c) Image with speckle noise ($\mu = 0; \sigma^2 = 0{,}05$)

(d) Image with salt and pepper noise ($p = 0{,}05$)

Figura 6: Image with different kinds of noises.

Given an image f and g filtered image of $M \times N$ dimensions, the average absolute error of the filtered image is given by:

$$MAE(f,g) = \frac{1}{3 \times M \times N} \sum_{j=1}^{3} d_j \tag{22}$$

where:

$$d_j = \sum_{\substack{u \in \{1,\ldots,M\} \\ v \in \{1,\ldots,N\}}} |[f(u,v)]_j - [g(u,v)]_j| \tag{23}$$

5.3.2. Results

Below are listed the codes used to abbreviate the names of the sorting methods that were the subject of experimentation:

- ED: RGB Euclidean distance is used as a method of sorting colors [?].
- BM: RGB bit interleaving method is used as a ordering of colors [?].
- LEX: RGB lexicographical ordering is used to sort the colors .
- ALEX: The RGB α -lexicographic ordering is used [?] for ordering colors.
- AMLEX: The RGB lexicographical α-Module is used [?] for ordering colors.
- HLEX, I→H→S lexicographical ordering is used to sort colors.
- DLAB, distance in L* a*b* is used as sorting method [?].
- MIN: the 10 equation is used to add this transformation as the first component of the RGB lexicographical cascade , where: $w = (Min(h^D_{f_1}), Min(h^D_{f_2}), Min(h^D_{f_3}))$.
- MAX: the 10 equation is used to add this transformation as the first component of the RGB lexicographical cascade, where: $w = (Max(h^D_{f_1}), Max(h^D_{f_2}), Max(h^D_{f_3}))$.
- MO1: the 10 equation is used to add this transformation as the first component of the RGB lexicographical cascade, where: $w = (minM_o(h^D_{f_1}), minM_o(h^D_{f_2}), minM_o(h^D_{f_3}))$.
- MO2: the 10 equation is used to add this transformation as the first component of the RGB lexicographical cascade, where: $w = (maxM_o(h^D_{f_1}), maxM_o(h^D_{f_2}), maxM_o(h^D_{f_3}))$.
- SMO: the 10 equation is used to add this transformation as the first component of the RGB lexicographical cascade, where: $w = (R(h^D_{f_1}), R(h^D_{f_2}), R(h^D_{f_3}))$.

Figura 7: Gaussian noise ($\mu = 0; \sigma^2 = 0{,}105$)

- MEAN: the 10 equation is used to add this transformation as the first component of the RGB lexicographical cascade, where:
  $w = (Me(h^D_{f_1}), Me(h^D_{f_2}), Me(h^D_{f_3}))$.
- VAR: the 10 equation is used to add this transformation as the first component of the RGB lexicographical cascade, where:
  $w = (Var(h^D_{f_1}), Var(h^D_{f_2}), Var(h^D_{f_3}))$.

The filter used to eliminate different types of noise was the median. This filter consists to sort the colors within the filter window, and selecting the middle value to replace in the output image. So to avoid having a false color, the size of the filter window is usually odd (3×3 in our tests). The tests were made with 100 different images ([?] test images), polluted with noise: gaussiano, speckle, salt and pepper. In the case of Gaussian and Speckle noise, the parameter μ is set to 0 and the σ^2 parameter was varied between 0,005 and 0,165, with 0,02 increases. In the case of salt and pepper noise, the probability parameter p was varied with the same values of the parameter σ^2 of Gaussian and Speckle noise.

The Figure 7 corresponds to the original image of Figure 6 with $\sigma^2 = 0{,}105$ and $\mu = 0$. The result of applying the proposed order filters and other filters evaluated on the contaminated image shown in the Figure 8.

Should be noted that filtered images with the proposed order and with different weights are better visually to the state of the art, but they are perceptually very similar to each other, the difference between them is evident in the numerical results presented later in this section.

For every noise and measuring a results table and graph trend curve of each filter with respect to the variation of noise parameter (σ^2 to Gaussian and speckle noise, and p for salt and pepper noise). Each point represents the average of metric obtained by the filter to a noise parameter value (σ^2 MAE) or (p, MAE). The corresponding curve of a filter is obtained by joining each pair of successive points of the filter with the line (straight) passing through both points. This is done so as to be able to view the trend as a continuous function. The results tables order the filters as the total sum of all points on the graph curves. The filters shown in the top of the tables are those with the lowest values of the total sum of MAE obtained by each filter. In some cases, the state of the art filters perform better for less noise parameter values but are overcome by the filters proposed for higher noise parameter values.

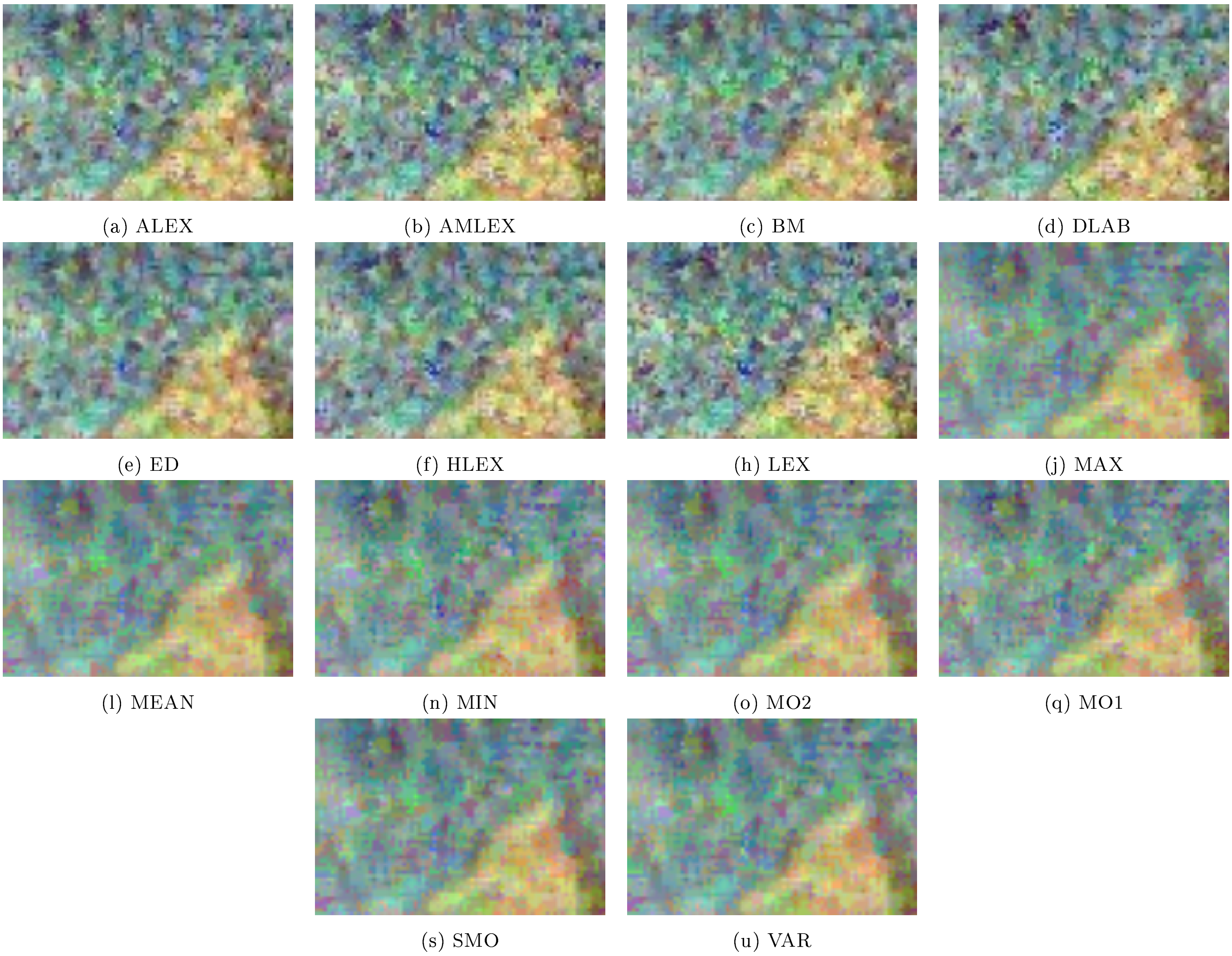
The results of this section differ the order filters of each weight according to his domain settings. For reference, are added the suffix "WX"to the proposed filters codes, where X is a number representing the number of sub-regions in which the image was divided, with M'= $\sqrt{X}$ and N'= $\sqrt{X}$. When the neighborhood (marked by the filter window) is the domain, suffix "Bïs used.

Cuadro 1: Ruido gaussiano. Sumatoria de MAE por σ^2

Filtro	Suma
SMOW9	204,1807
MAXW9	204,182
VARW9	204,2071
MEANB	204,2333
MEANW9	204,2333
MAXB	204,3576
SMOB	204,5693
ED	204,7305
VARB	204,9878
MO2B	205,3976
BM	208,4018
MINB	208,5482
MO1B	208,5818
HLEX	211,6824
AMLEX	212,1728
MO2W9	212,1857
MO1W9	212,196
ALEX	212,9069
LEX	214,813
MINW9	215,2299
DLAB	226,3062

In the Figure 9 the trend curves of the filters applied to images contaminated with gaussian noise are observed . As can be seen, for gaussian noise, the best filter was the proposed using SMO, MAX, VAR and MEAN for calculating weight vector. Table 1 contains the total sum of the points on the trend curve by sorting method.

Figure 10 shows the trend curves of the filters applied to images contaminated with salt and pepper noise. In this case the filter to obtained the lowest overall average and whose curve is placed below all others was ED. The MAX, SMO, VAR and MEAN filters showed better



(a) ALEX (b) AMLEX (c) BM (d) DLAB
(e) ED (f) HLEX (h) LEX (j) MAX
(l) MEAN (n) MIN (o) MO2 (q) MO1
(s) SMO (u) VAR

Figura 8: Results of applying different filters evaluated on the image in Figure 7. The image was divided into 5 x 5 subregions .

performances after ED. Table 2 contains the total sum of the points on the curve trend by sorting method.

In the Figure 11 there are the trend curves of the filters applied to images contaminated with speckle noise .The SMO, MAX and VAR filters obtained the best results. The filter SW9 obtained the best results for all points of the curve. The ED filter resulted the best of the state of the art. Table 3 contains the total sum of the points in the trend curve by sorting method in ascending sort.

The following two applications use morphological mathematical. Operators are not purely morphological, since it can not guarantee its theoretical properties, as idempotency for opening and closing. In the Figure 12 you can see a counter-example, where the opening operator is not idempotent ($f \circ B \neq (f \circ B) \circ B$). To the synthetic image of the Figure 12(a) is applied an opening with 3 x 3 structural element, where the domain D of the image is the neighborhood of the structuring element itself, and the function applied to each color component is the average inside D, resulting image of the Figure 12(b). To image of the Figure 12(b) is applied again the operator of opening with the same structuring element and where the domain D is also the neighborhood of the structuring element itself, resulting the image of the Figure 12(c). The resulting image is not equal to the previous one, this can be visualized on the image 12(c), which is the result of performing the difference between the two images. Thus it is demonstrated that the opening operator is not idempotent , the same occurs for the closing operator.

This is because a color can be more or less than other color in a domain D, but not in another domain, because the extracted information in the form of weights

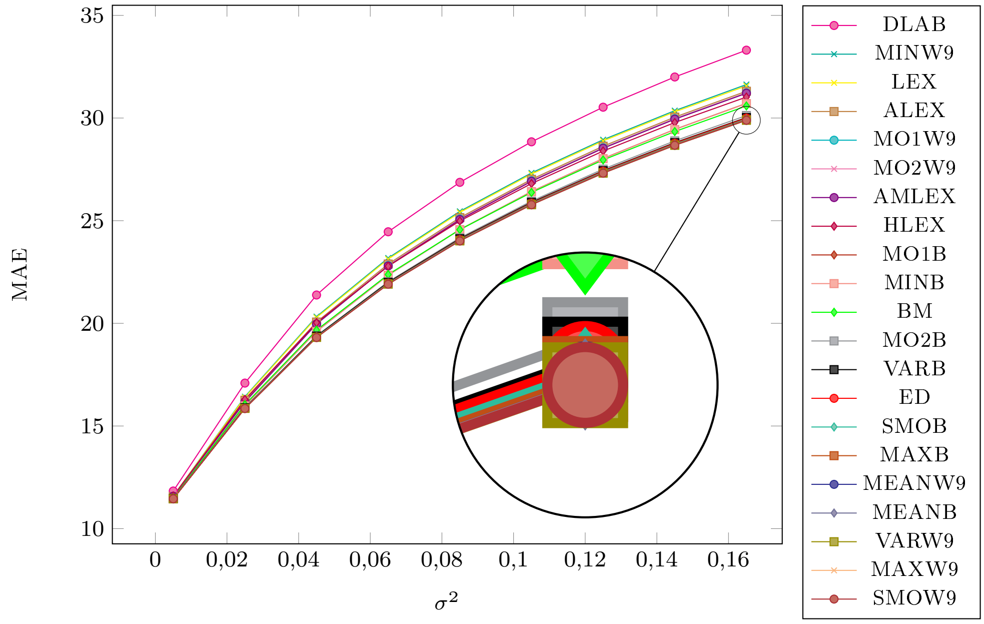


Figura 9: Ruido gaussiano. MAE por σ^2

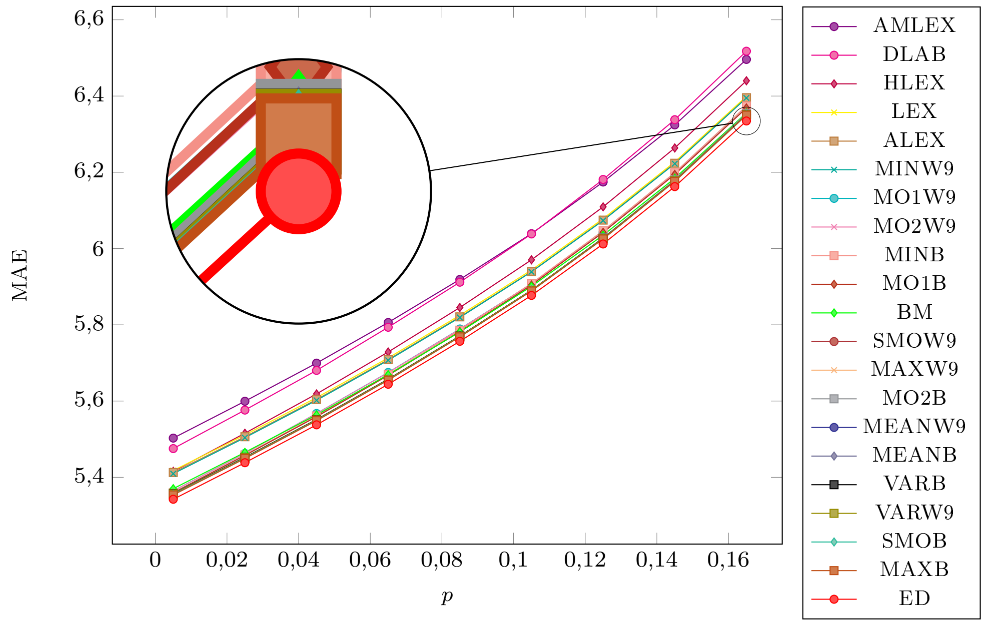


Figura 10: Salt and pepper noise. MAE by p

(result of applying a function) may be different. Even in the same domain but in the next iteration (product of reapplying the same operator) they can vary the weights, since the extracted information also varies from one iteration to another. In the literature these operators are called pseudo-operators [? ? ? ? ?].

Top-hat transform is widely used in different applications [? ? ? ? ? ?]. As we mentioned earlier white top-hat transform extracts the bright regions of the image and top-hat transform the dark areas of the image.

5.4. Application 2: Contrast improving

A basic idea of improving contrast of image f is to add the bright regions of the image f and subtract the dark regions of the image f as follows [?]:

$$Contrast(f) = f + WTH(f) - BTH(f) \tag{24}$$

The effectiveness of the application of contrast improvement it is determined using the method called Color Enhancement Factor (CEF) which quantifies the lev-

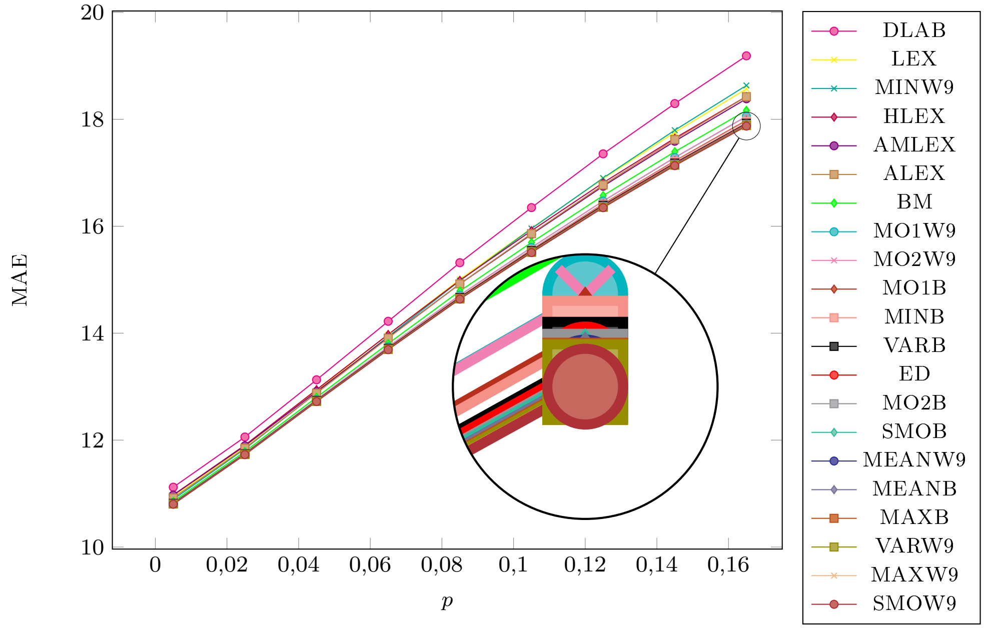
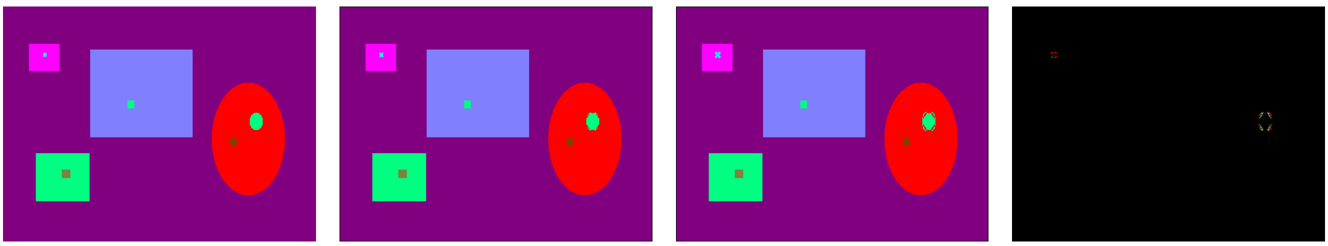


Figura 11: Speckle noise. MAE by σ^2



(a) Synthetic image (b) Aperture of the image (a) (c) Aperture of the image (b) (d) Difference between (b) and (c)

Figura 12: A counter-example shows that openness is not idempotent with the ordering proposed.

el of contrast enhancement of an image as mentioned in [?]. This method applied to the image f is based on the average and standard deviation of two axes of a simple contrary color representation with $\gamma = f_1 - f_2$ y $\beta = \frac{1}{2}(f_1 + f_2) - f_3$. The equation 25 represents the level of contrast enhancement of the image f as follows:

$$CM(f) = \sqrt{\sigma_\gamma^2 + \sigma_\beta^2} + \sqrt{\mu_\gamma^2 + \mu_\beta^2} \tag{25}$$

Where σ_γ and σ_β correspond to the standard deviation of γ and β respectively. Similarly, μ_γ and μ_β corresponds to the respectively mean.

Then, the CEF is calculated by the ratio of the f' image and f original image:

$$CEF = \frac{CM(f^{'})}{CM(f)} \tag{26}$$

Where $CM(f^{'})$ is the value obtained from the contrasted image f' product of applying the equation 25 and $CM(f)$ represents the result of applying the equation 25 to the original image f. If the result is > 1 then the metric of the equation 26 indicates an improvement in the contrast, otherwise, metric indicates no contrast enhancement.

5.4.1. Results

The tests were made with 100 test images of [?] and were used the same abbreviations as in the previous experiment to differentiate the sorting methods with different domain decomposition. In the 4 can see the results of the various iterations (iter) of contrast enhancement algorithm (applied several times the equation 24 to the same image). Is observed that SMOB has better results in all the iterations, followed by the variance VARB and then follow the other methods. The

Cuadro 2: Salt and pepper noise. Summation of MAE by p

Filtro	Suma
ED	52,1052
MAXB	52,2144
SMOB	52,2207
VARW9	52,2314
VARB	52,234
MEANB	52,235
MEANW9	52,235
MO2B	52,2349
MAXW9	52,2361
SMOW9	52,2365
BM	52,3259
MO1B	52,336
MINB	52,3643
MO2W9	52,3724
MO1W9	52,3742
MINW9	52,6715
ALEX	52,6865
LEX	52,725
HLEX	52,9079
DLAB	53,5125
AMLEX	53,5607

Cuadro 3: Ruido speckle. Sumatoria de MAE por σ^2

Filtro	Suma
SMOW9	204,1807
MAXW9	204,182
VARW9	204,2071
MEANB	204,2333
MEANW9	204,2333
MAXB	204,3576
SMOB	204,5693
ED	204,7305
VARB	204,9878
MO2B	205,3976
BM	208,4018
MINB	208,5482
MO1B	208,5818
HLEX	211,6824
AMLEX	212,1728
MO2W9	212,1857
MO1W9	212,196
ALEX	212,9069
LEX	214,813
MINW9	215,2299
DLAB	226,3062

improvement using SMOB as it grows the amount of iterations is approximately 3 % and the difference with the second and third is of 0.30 % and 2.25 % on average in each of the iterations. It can be seen that as the number of iterations increases also improves the contrast according to the mentioned metric , being also important the domain and sorting method used.

In the Figure 13 can see an example of improving contrast of an image, to which was applied four iterations of equation 24 with the proposed sorting method

Cuadro 4: Contrast improvement

METODO	iter1	iter2	iter3	iter4
SMOB	**1,03482**	**1,03482**	**1,06084**	**1,08889**
VARB	**1,0329**	**1,0329**	**1,05798**	**1,0852**
MO2W9	1,02047	1,02047	1,03668	1,05426
MO1W9	1,02045	1,02045	1,03664	1,05421
SMOW9	1,0203	1,0203	1,03639	1,05369
MAXW9	1,02023	1,02022	1,03629	1,05364
BM	1,01997	1,01997	1,03576	1,05273
MINW9	1,01992	1,01991	1,03573	1,05295
ED	1,01989	1,01989	1,0356	1,05265
AMLEX	1,01988	1,01987	1,0352	1,05159
MEANW9	1,01986	1,01986	1,03567	1,05277
MO1B	1,01937	1,01937	1,03497	1,05197
MAXB	1,01928	1,01928	1,03493	1,05199
MO2B	1,01922	1,01922	1,03481	1,05182
MEANB	1,01912	1,01912	1,03461	1,0514
LEX	1,01909	1,01909	1,03369	1,04908
ALEX	1,01909	1,01909	1,03369	1,04908
MINB	1,01908	1,01908	1,03448	1,05128
DLAB	1,01756	1,01756	1,03049	1,04426
HLEX	1,00743	1,00743	1,0128	1,01863
VARW3	0,99727	0,99725	0,98268	0,98489

using SMO for calculating weights.The resulting image is clearly much more contrasted than the original image.

5.5. Application 3: Classification of textures

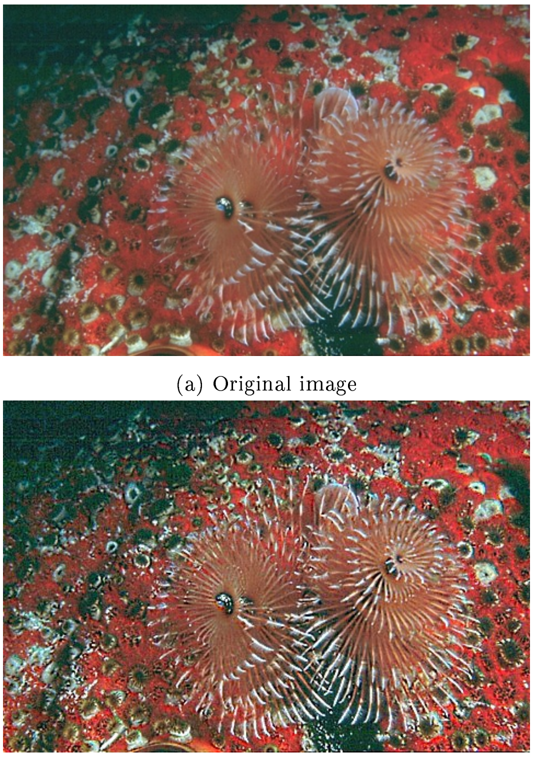
The problem of texture characterization and classification consists of two steps. In the first instance, image characteristics that allow numerically describe their textural properties using a feature vector or descriptor are calculated. Later it is assigned a class of texture according criteria of similarity between descriptors [**?**].

The granulometry and morphological covariance are the main morphological tools of texture characterization, both used intensity distributions to describe the properties of the textures [**?**].

The way the color and texture information is incorporated into the descriptor is studied in [**? ?**]. In this work the morphological tools use the integrative approach where color and texture information are processed together.

The granulometry was proposed in [**?**] and is applied in feature extraction and estimation of size [**? ?**]. Consists of a $f \circ \lambda B$ openings family of $n+1$ elements including the input image. It is parameterized by the growing λ size of the structural element ($0 \leq \lambda \leq n$). The values are collected by an evaluation measure that is usually the volume (Vol):

$$G_j^n(f,\lambda) = \mathrm{Vol}([f \circ \lambda B]_j) \ / \ \mathrm{Vol}(f_j) \tag{27}$$



(a) Original image

(b) Improved image

Figura 13: (a) Image contrast improvement applying 4 times the equation 24 with the proposed sorting method

Where j is the j-th component of the color image f and the volume is defined as:

$$\mathrm{Vol}(f_j) = \sum_{\substack{u \in \{1,\dots,M\} \\ v \in \{1,\dots,N\}}} [f(u,v)]_j \tag{28}$$

The morphological covariance proposal at [? ?] denoted by K of f image, is defined as the volume of the image f, after applying ε erosion from a pair of pixels (u,v) and (u',v') separated by a vector $\mathbf{v}$ denoted by $P_{2,\mathbf{v}}$.
In practice K is calculated by applying ε erosion to the original image f with the structuring element $P_{2,\mathbf{v}}$ varying orientations and lengths of $\mathbf{v}$, where n is the number of variations of $\mathbf{v}$. Its normalized version is given by:

$$K_j^n(f, P_{2,\mathbf{v}}) = \mathrm{Vol}([\varepsilon(f, P_{2,\mathbf{v}})]_j)/\mathrm{Vol}(f_j) \tag{29}$$

Allows to obtain a distribution of orientation and distance from a texture image [?].

The sorting method used to support the morphological characterization tools of texture, affects the percentages of classification. This is because the intensity distributions used as texture descriptors, vary according to the intermediate images. These intermediate images are the result of applying a morphological filter.

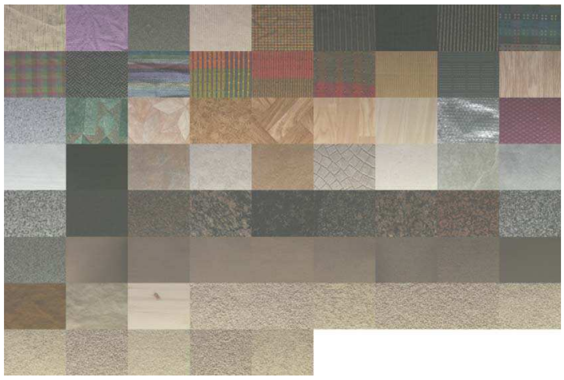


Figura 14: Texture samples OutexTC13

5.5.1. Results

The tests were made with OutexTC13 data base consisting of 1360 images of size 128 $\times$ 128 pixels, with 68 kinds of surface textures (Figura 14 with 20 samples of each class, where the 50 % of each class is the training set. Totaling 680 training and test images respectively [?]. The classifier used was k-nn (k-nearest neighbors) using Euclidean distance with k = 1.

The purpose of the experiment was the classification percentages obtained using the sorting methods exposed. Several statistical parameters were evaluated in the order strategy proposed in this work. In granulometry tests they have been used structural elements of square shape of size λ and $2\lambda+1$ side pixels, varying λ from 1 to 15. For each element of the series, 15 values for each channel are calculated which are then concatenated. The choice of simple increase of λ, is based on the smallest increases provide better classification results [?]. With respect to the configuration of the parameters of the proposal, each texture sample was divided in 2×2 subregions and denoted by the suffix W4. This division allows each texture sample is divided equally and perceptually similar. Is denoted by the suffix B when the image domain is the structuring element itself.

Morphological covariance requires varying the direction and distance between the pair of points that composing the structural elemen. The addresses used were 0°, 45°, 90°, 135°, in practice only these addresses are important and significantly recognizable [?]. The separation distances were used for each direction were from 1 a 20 píxeles. Using these 4 directions and 20 distances have generated 80 values for each channel, finally these values are concatenated to obtain the feature vector of the texture sample.

In the Table 5 the three best results of each method were marked in bold. Covariance Morphological with HLEX y DLAB orderings have higher performances

Cuadro 5: Classification results by Ordering Methods

	% Correctly classified	
Sort	**Covariance**	**Granulometry**
BM	79,56	77,79
ED	81,91	**84,11**
LEX	79,74	80,15
ALEX	76,32	69,12
AMLEX	81,62	81,30
HLEX	**83,97**	77,35
DLAB	**84,26**	72,35
MEANW4	81,47	78,97
SMOW4	82,50	**85,44**
MO1W4	81,76	82,65
MO2W4	80,44	80,01
MAXW4	81,62	81,32
MINW4	81,47	83,38
VAR	81,76	83,97
MEANB	81,91	78,82
SMOB	**83,82**	**84,71**
MO1B	82,01	83,23
MO2B	81,21	81,36
MAXB	81,32	81,06
MINB	81,05	83,12
VARB	83,09	82,21

of $\approx 1\,\%$ with respect to SMOW4 that presented the best performance in RGB space. The results are consistent with experiments in [?] where best results are obtained in the L*a*b space compared to RGB space using structuring elements of variant length. This is because in those color spaces the chromatic information is separated of the brightness or intensity. In experiments mentioned in [?] about the perception of texture they indicate that the texture and color are perceived independently. With the granulometry, the RGB space has better results with SMOW4 and SMOB orderings, both higher than the ED ordering at ($\approx 1,33\,\%$). Moreover, the ED ordering provides far superior results to the BM, LEX, ALEX and AMLEX orderings.

The SMOW4 y SMOB orderings (85,44 %, 84,71 %) show the best results of classification (with granulometry) with respect to the ordering methods implemented of the state of the art. The division in 2×2 sub-regions (W4) leads to better results than using B.

6. Conclusions and Future Work

In this work a new strategy of RGB color ordering that is dependent on the image we are presented. The ordering is performed by extracting histogram information of each color component in a certain domain image. Two strategies of domain decomposition are presented to extract this information, one is extracting information from the same window filter, and another consists in dividing the image into sub-regions of same size, taking the union of them when the filter window takes more than one sub-region. Tests were performed for three image processing applications: Noise Removal, contrast stretching and textures characterization for further classification. Is used morphologythis mathematical for this two latest applications. Morphological operators in this case are pseudo-operators as it can not be guaranteed certain theoretical properties, such as idempotency. The median filter was used to eliminate noise, achieving better results with some extracted weights of different methods of the state of the art, both gaussian noise such as speckle. For salt and pepper noise the euclidean distance to the origin in the RGB color space in the MAE metric gave better results than the proposed method with the different extracted information for each component. This is because the salt noise is expressed by the minimum value in each color component, and pepper noise is expressed as the maximum value in each component. Thus, if the colors of the pixels are sorted by the euclidean distance, it is quite unlikely that the median filter select a pixel to be noise. For the application of contrast improvement the proposed method was more efficient according to the CEF metric with different weights extracted from image information. For the characterization of textures using Covariance Morphological with lexicographic ordering I→H→S [?] and the euclidean distance to the color $(0, 0, 0)$ in the L*a*b* color space [?] has better performance to the proposal, demonstrated in its best ranking. The proposed method using the softness as weight for each component achieves better results using the granulometry as characterization of textures. Is proposed as future work analyze the importance of domain decomposition to extract information from each color component. More experiments could be done in different applications such as fusion or segmentation of color image. Could be extract other information for each component as Entropy and Energy.