

Consensus for image noise removal with unknown noise

Luis González Jaime^{a,*}, Etienne E. Kerre^a, Mike Nachtgael^a, Humberto Bustince^b

^a*Applied Mathematics and Computer Science, Ghent University, Krijgslaan 281 - S9, 9000 Ghent, Belgium*

^b*Departamento de Automática y Computación, Universidad Pública de Navarra, Edificio Los Pinos, 1^a planta, Campus Arrosadía, 31006 Pamplona, Spain*

Abstract

Noise removal has been, and it is nowadays, an important task in computer vision. Usually, it is a previous task preceding other tasks, as segmentation or reconstruction. However, for most existing denoising algorithms the noise model has to be known in advance. In this paper, we introduce a new approach based on consensus to deal with unknown noise models. To do this, different filtered images are obtained, then combined using multi-fuzzy sets and averaging aggregation functions. The final decision is made by using a penalty function to deliver the compromised image. Results show that this approach is consistent and provides a good compromise between filters.

Keywords: consensus, image noise removal, unknown noise, penalty function, aggregation function, OWA operator

1. Introduction

The degradation of an image is unavoidable during acquisition. The restoration of degraded images is an important task widely studied in computer vision (Rudin et al. (1992); You et al. (1996); Molina et al. (2001); Buades et al. (2005); La Riviere and Billmire (2005); Srinivasan and Ebenezer (2007); Aja-Fernandez et al. (2008); Goossens et al. (2008); Borsdorf et al.

*Corresponding author

Email addresses: luis.gonzalez@ugent.be (Luis González Jaime),
etienne.kerre@ugent.be (Etienne E. Kerre), mike.nachtgael@ugent.be (Mike Nachtgael), bustince@unavarra.es (Humberto Bustince)

Preprint submitted to Elsevier

September 26, 2013

(2008); Skiadopoulos et al. (2009); Deledalle et al. (2010)). It always received a lot of attention from many researchers of different fields. Denoising is one of the most fundamental image restoration techniques (Molina et al. (2001); Buades et al. (2005); Srinivasan and Ebenezer (2007); Aja-Fernandez et al. (2008); Goossens et al. (2008); Skiadopoulos et al. (2009); Deledalle et al. (2010)), due to random distortions which make it difficult to perform any required image processing. The desired goals of a denoising algorithm are to completely remove noise, while effective information (edge, corner, texture and contrast...) is preserved, at the same time that artifacts do not appear.

In order to find an ideal image denoising algorithm, researchers have proposed hundreds of algorithms. The most popular noise assumption is the additive Gaussian noise (Rudin et al. (1992); You et al. (1996); Buades et al. (2005)). However a Gaussian noise assumption is too simplistic for most applications, specifically for medical and astronomical images (Molina et al. (2001)). In the particular case of medical images, in computer tomography (CT), the decay of the signal is better modelled with a Poisson distribution (Suzuki (1985); Hsieh (1998); La Riviere and Billmire (2005)). Other medical images, as single-photon emission computed tomography (SPECT) or positron emission tomography (PET), can also be well modelled with a Poisson distribution (Rosenthal et al. (1995); Skiadopoulos et al. (2009)). In the case of magnetic resonance images (MRI), a Rice distribution better models the abnormalities in the image for a single-coil (Bernstein et al. (1989); Aja-Fernandez et al. (2008)).

Despite different approaches that exist in order to reduce noise, all of them fail in their performance with images owning a noise distribution for which these algorithms are not optimal. It would be desirable to have a denoising algorithm being able to deal with any noise distribution. However this is a complex issue due to the different nature of the images (e.g. CT capturing process is different from the digital camera). Therefore, this work is focused on the fusion of a set of filtered images, through a multifuzzy set, previously filtered from a noisy image with unknown noise distribution. We select filters existing in the literature that are optimal for a concrete noise. In particular, filters for impulse, Poisson, Gaussian and Rician noise are applied. Then, the fusion is carried out using consensus via penalty functions on a cartesian product of lattices, where the penalty function chooses the value that minimizes the error for each pixel in accordance to the different options.

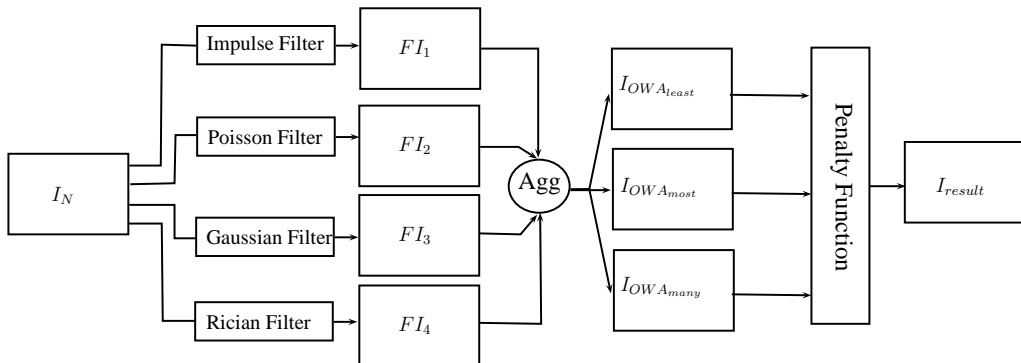


Figure 1: Schema consensus algorithm

Figure 1 shows graphically the proposed schema. Starting from the noisy image I_N , the first step is to build a multifuzzy set from the filtered images, in our case (FI_1, FI_2, FI_3, FI_4) , so each pixel (i, j) is represented by several values (each value corresponds to pixel (i, j) of each filtered image). But, we need to obtain a fused image, I_{result} , with only one value for each pixel. For this reason we continue by using averaging aggregation functions. However, we do not know which is the best function to use. To solve this problem, we select a set of functions. In this paper we decide to use OWA operators. In particular, OWA operators constructed from fuzzy quantifiers, since they provide a more flexible knowledge representation than classical logic, that it is restricted to the use of only two quantifiers, *there exists* and *for all* (Chiclana et al. (1998)). We select three different OWA operators, namely ‘at least half’, ‘most of them’ and ‘as many as possible’ because of their good performance. We apply these operators to each pixel, so we obtain three new possible values for each pixel $(I_{OWA_{least}}, I_{OWA_{most}}, I_{OWA_{many}})$. In order to decide the best aggregated value among them, we use penalty functions that take the value that minimizes the error with respect to the filtered images, and thus, the best fused image is obtained, I_{result} . Our aim is to obtain consistent and stable results, regardless of the image nature (e.g. CT, MRI, digital image). One of the applications of this work is with MRI, because they present a more sophisticated noise model than a simple Gaussian noise, it however can be applied to other images with different nature as it is also shown.

The paper is composed as follows: Section 2.1 introduces the different noise models and filters. In Section 2.2, multifuzzy sets are explained. Then, Section 3 presents the idempotent functions, their properties and a

specific case: the OWAs operators, a family of idempotent averaging functions. Penalty functions and the consensus algorithm are explained in Section 4. Finally, in Sections 5 and 6 specific results and a final conclusion are exposed.

2. Construction of multifuzzy sets from a set of filtered images

Given an unknown noisy image, our first step consists in associating a multifuzzy set composed by several images. Each one of these images will be obtained by applying some filter optimized for a certain type of noise.

2.1. Noise models and filters

Many digital image devices often produce a degradation in the image quality. This noise is mainly introduced during the image capturing (sensors, amplifiers), the transmission or the recording (Morris (2003)), although in some modalities, as CT or MRI, it can also be introduced in the reconstruction algorithm (Aja-Fernandez et al. (2009)). This can e.g. be caused by dust sitting on the lens, by a dissipation in the electronic components or by electromagnetic distortions during transmission. Digital imaging techniques must deal with the degradations present in the images.

Each element involved in the pipeline used to obtain the final (reconstructed) image (sensors, lens, A/D converter, enhancement algorithm, reconstruction algorithm...), influences the noise characteristics. Several approaches exist that deal with Gaussian or impulse noise (Rudin et al. (1992); You et al. (1996); Buades et al. (2005); Srinivasan and Ebenezer (2007)), although in some cases these are simple approximations compared to the real noise that is presented. For instance, MRI, specifically MR magnitude image, are mainly characterized by Rician noise, although this noise is dependent on the number of coils or the reconstruction method (Aja-Fernandez et al. (2009)). Furthermore CT, PET, SPECT or astronomical images are identified by Poisson noise (Suzuki (1985); Rosenthal et al. (1995); Hsieh (1998); Molina et al. (2001); La Riviere and Billmire (2005); Skiadopoulos et al. (2009)).

Different filters are applied in this work with the aim to prove the effectiveness of consensus, and how it can help to obtain a good performance. The selected filters cover different approaches to the image denoising problem, as well as they perform better for a specific noise distribution. We give an overview of the characteristics of these filters.

The first approach tackles the problem of impulse noise, and uses the DBAIN filter proposed by Srinivasan and Ebenezer (2007). The algorithm, in a first step detects if a processed pixel is noisy or noise-free depending on its occurrence in a corresponding window. If the pixel is determined as corrupted, then the pixel is replaced by the median value of the window. Although, in case the median is considered corrupted, instead of the median, it is replaced by the value of neighborhood pixels. This method does not require any parameter for its performance.

Additive white Gaussian noise (AWGN) has generally been found to be a reasonable model for noise originating from electronic amplifiers. The considered filter to deal with white Gaussian noise has been the approach proposed by Goossens et al. (2008). This filter is based on the non-local means (NLmeans) filter proposed by Buades et al. (2005). This version of NLmeans improves the original version, dealing with noise in non-repetitive areas with a post-processing step and presenting a new acceleration technique that computes the Euclidean distance by a recursive moving average filter. Moreover, they introduce an extension that can deal with correlated noise. However, its performance depends on a previous configuration. The standard deviation estimation, the searching window or the block size needs to be defined previously. We use the configuration from the original paper for our experiments.

The approach used to estimate Rician noise, the probability density function that mainly characterizes MRI in single-coil systems (Bernstein et al. (1989); Aja-Fernandez et al. (2009)), is proposed by Aja-Fernandez et al. (2008). This filter adapts the linear minimum mean square error (LMMSE) to Rician distributed images. Moreover, noise estimation can be automatically calculated based on local statistics. Although the version used in our experiments is the approach in which the standard deviation is given as an input.

Finally, for Poisson noise, an extension of the NLmeans is proposed for images damaged by Poisson noise. Deledalle et al. (2010) propose to adapt the similarity criteria of NLmeans algorithm to Poisson distribution data. For this filter, a previous configuration is required. For our experiments, the used parameters are those suggested in the original article, as the algorithm is tuned to obtain good results.

2.2. Multifuzzy sets

Once the set of filtered images is obtained, we represent them by means of multifuzzy sets, in which each element is given by a set of n memberships,

taking n as the number of filters. A unique multifuzzy set is built with all the elements of the images.

Definition 1. (González-Jaime et al. (2012)) A multifuzzy set of dimension $n \geq 2$ over a finite universe U is defined by a mapping

$$A : U \rightarrow [0, 1]^n$$

given by

$$A(u) = (A_1(u), \dots, A_n(u))$$

where each of the A_j for $j = 1, \dots, n$ is a fuzzy set $A_j : U \rightarrow [0, 1]$.

Notice that the previous definition is equivalent to the following. Take a family of $n \geq 2$ fuzzy sets Q_1, \dots, Q_n on the same referential set U . Then an n -dimensional multifuzzy set on U is just the ordered combination of these n fuzzy sets as follows:

$$A = \{(u, A(u)) | u \in U\} \text{ given by } A(u) = (Q_1(u), \dots, Q_n(u))$$

In this sense, the space of all multifuzzy sets inherits the order from the usual fuzzy sets, which endows it with a partial, bounded order.

In this work, we will deal with two finite referential sets $X = \{0, 1, \dots, N - 1\}$ and $Y = \{0, 1, \dots, M - 1\}$, where N and M are the number of rows and columns of the image, respectively. We will consider multifuzzy sets defined on the Cartesian product $X \times Y$.

Notice that an n -dimensional multifuzzy set can also be understood as a type n fuzzy set, as well as an L -fuzzy set with $L = [0, 1]^n$ (Klir and Yuan (1995)).

3. Idempotent functions: building a fuzzy set from multifuzzy sets

After the noisy image is filtered, we get a set of filtered images that composes the multifuzzy set. So, each pixel (i, j) is represented by n values, as many as filters used. This multifuzzy set needs to be fused in one single image, a fuzzy set. Therefore, we need functions that take n inputs and obtain one single value satisfying one condition: if all the values are the same, the value remains the same. For this reason we decide to use idempotent functions.

Definition 2. An n -dimensional idempotent function is a mapping $\gamma : [0, 1]^n \rightarrow [0, 1]$ such that

$$\gamma(x, \dots, x) = x$$

for every $x \in [0, 1]$.

Example 1. Some of the idempotent functions are:

1. The mode, that is the value that occurs most frequently in a data set or a probability distribution.
2. Smallest idempotent function

$$\gamma_{\text{smallest}}(x_1, \dots, x_n) = \begin{cases} 0 & \text{if } \exists i, j \in \{1, \dots, n\} \text{ such that } x_i \neq x_j; \\ x_1 & \text{in other case,} \end{cases}$$

3. Largest idempotent function

$$\gamma_{\text{largest}}(x_1, \dots, x_n) = \begin{cases} 1 & \text{if } \exists i, j \in \{1, \dots, n\} \text{ such that } x_i \neq x_j; \\ x_1 & \text{in other case,} \end{cases}$$

Remark 1. Notice, that neither of the functions from Example 1 are monotone.

3.1. Construction of idempotent functions

In Proposition 1 we present a method for constructing idempotent functions.

Proposition 1. (González-Jaime et al. (2012)) The mapping $\gamma : [0, 1]^n \rightarrow [0, 1]$ is an n -dimensional idempotent function if and only if there exist $f, g : [0, 1]^n \rightarrow [0, 1]$ such that

- (i) $\gamma(x, \dots, x) \neq 0$ for every $x \in [0, 1]$;
- (ii) $f(x, \dots, x) = \frac{x}{1-x}g(x, \dots, x)$ for $x \in [0, 1[$, $f(1, \dots, 1) = 1$ and $g(1, \dots, 1) = 0$;
- (iii) $\gamma(x_1, \dots, x_n) = \frac{f(x_1, \dots, x_n)}{f(x_1, \dots, x_n) + g(x_1, \dots, x_n)}$

Proof. Assume that γ is an n -dimensional idempotent function. Take $f = \gamma$ and $g = 1 - \gamma$. Then

- (i) $g(x, \dots, x) = 1 - \gamma(x, \dots, x) = 1 - x \neq 0$ for every $x \in [0, 1[$.
- (ii) $\frac{x}{1-x}g(x, \dots, x) = \frac{x}{1-x}(1-x) = x = \gamma(x, \dots, x) = f(x, \dots, x)$ and $f(1, \dots, 1) = \gamma(1, \dots, 1) = 1$ and $g(1, \dots, 1) = 0$.
- (iii) $\frac{f(x_1, \dots, x_n)}{f(x_1, \dots, x_n) + g(x_1, \dots, x_n)} = \gamma(x_1, \dots, x_n)$.

To see the converse, we only need to check the idempotency. But if γ is defined as in the statement of the proposition, we have that $\gamma(x, \dots, x) = \frac{f(x, \dots, x)}{f(x, \dots, x) + g(x, \dots, x)} = \frac{\frac{x}{1-x}g(x, \dots, x)}{\frac{x}{1-x}g(x, \dots, x) + g(x, \dots, x)}$ which is equal to x for every $x \in [0, 1[$. Finally, if $x = 1$ then clearly $\gamma(1, \dots, 1) = 1$. \square

Example 2.

- Taking $f(x_1, \dots, x_n) = \frac{1}{n} \sum_{i=1}^n x_i$ and $g(x_1, \dots, x_n) = \frac{1}{n} \sum_{i=1}^n (1 - x_i)$ we obtain as idempotent function the arithmetic mean,
 $\gamma_{\text{mean}}(x_1, \dots, x_n) = \frac{1}{n} \sum_{i=1}^n x_i$
- Taking $f(x_1, \dots, x_n) = \sqrt[n]{x_1 \cdot x_2 \cdot \dots \cdot x_n}$ and $g(x_1, \dots, x_n) = \max(1 - x_1, \dots, 1 - x_n)$ we get $\gamma_{\text{root}}(x_1, \dots, x_n) = \frac{\sqrt[n]{x_1 \cdot x_2 \cdot \dots \cdot x_n}}{\sqrt[n]{x_1 \cdot x_2 \cdot \dots \cdot x_n} + \max(1 - x_1, \dots, 1 - x_n)}$

Notice that the definition of an idempotent function is a very general one. In fact, we have directly the following construction result, that allows us to obtain a kind of inductive process.

Proposition 2. *Let $f : [0, 1]^{n-1} \rightarrow [0, 1]$ be an $(n-1)$ -dimensional idempotent function. Then the mapping:*

$$\gamma(x_1, \dots, x_n) = \sqrt{x_n \cdot f(x_1, \dots, x_{n-1})}$$

is an n -dimensional idempotent function.

Proof. Straightforward. \square

Regarding the structure of the space of n -dimensional idempotent functions, we also have the following.

Proposition 3. *Let $\gamma_1, \gamma_2 : [0, 1]^n \rightarrow [0, 1]$ be two n -dimensional idempotent functions. Then:*

1. $\frac{1}{2}(\gamma_1 + \gamma_2)$ is also an n -dimensional idempotent function;

2. $\sqrt{\gamma_1\gamma_2}$ is also an n -dimensional idempotent function.

Proof. Straightforward. \square .

It is known that a $[0, 1]$ -automorphism φ is a continuous and strictly increasing function, such that $\varphi(0) = 0$ and $\varphi(1) = 1$. We can use this concept to build n -dimensional idempotent functions.

Theorem 1. (Pahl and Damrath (2001)) Let $\gamma_1, \gamma_2 : [0, 1]^n \rightarrow [0, 1]$ be two n -dimensional idempotent functions. Let $F : [0, 1]^2 \rightarrow [0, 1]$ be a mapping such that $d(x) = F(x, x)$ is an automorphism of $[0, 1]$. Then the mapping:

$$\gamma = d^{-1}(F(\gamma_1, \gamma_2))$$

is also an n -dimensional idempotent function.

3.2. Some interesting properties

Homogeneity, shift-invariance and migrativity are properties quite used in image processing. In fact, it is desirable that these properties are satisfied.

Definition 3. A mapping $f : [0, 1]^n \rightarrow [0, 1]$ is called homogeneous of order $k \geq 0$ if for every $x_1, \dots, x_n, \alpha \in [0, 1]$ the identity

$$f(\alpha x_1, \dots, \alpha x_n) = \alpha^k f(x_1, \dots, x_n)$$

holds.

Example 3. The mapping $f : [0, 1]^2 \rightarrow [0, 1]$, defined by $f(x_1, x_2) = \max(x_1, x_2)$ is homogeneous of order 1.

Proposition 4. Every homogeneous idempotent operator is homogeneous of order 1.

Proof. Straightforward. \square .

Corollary 1. There are no idempotent operators homogeneous of order $k \neq 1$.

Definition 4. A mapping $f : [0, 1]^n \rightarrow [0, 1]$ is called shift-invariant if for every $x_1, \dots, x_n \in [0, 1]$ and for every $\lambda \in [-1, 1]$ such that $x_1 + \lambda, \dots, x_n + \lambda \in [0, 1]$ the identity

$$f(x_1 + \lambda, \dots, x_n + \lambda) = f(x_1, \dots, x_n) + \lambda$$

holds.

Proposition 5. *An idempotent operator γ is shift-invariant over the diagonal; that is, $\gamma(x + \lambda, \dots, x + \lambda) = \gamma(x, \dots, x) + \lambda$, for every $\lambda \in [0, 1]$.*

Proof. Straightforward. \square .

Definition 5. *A mapping $f : [0, 1]^n \rightarrow [0, 1]$ is called migrative if for every $x_1, \dots, x_n, \alpha \in [0, 1]$ the identity*

$$f(x_1, \dots, \alpha x_i, \dots, x_j, \dots, x_n) = f(x_1, \dots, x_i, \dots, \alpha x_j, \dots, x_n)$$

holds for every $i, j \in \{1, \dots, n\}$.

Example 4. *The mapping $f : [0, 1]^2 \rightarrow [0, 1]$, defined by $f(x_1, x_2) = x_1 \cdot x_2$, is migrative.*

An idempotent operator is not necessarily migrative, homogeneous and shift-invariant. We have the following result.

Proposition 6. *Let γ be an idempotent migrative operator. Then $\gamma(x, 0, \dots, 0) = 0$ for every $x \in [0, 1]$.*

Proof. Just observe that $\gamma(x, 0, \dots, 0) = \gamma(x, 0 \cdot 0, \dots, 0) = \gamma(0, 0, \dots, 0) = 0$ from the migrativity with $\alpha = 0$ and the homogeneity. \square

Proposition 7. *Let γ be an idempotent migrative operator. Then $\gamma(x^n, 1, \dots, 1) = x$ for every $x \in [0, 1]$.*

Proof. Straightforward. \square .

3.3. Idempotent aggregation functions: averaging functions

We have studied only the use of idempotent functions in order to transform a multifuzzy set into a fuzzy set. Now we study *monotonic non-decreasing idempotent functions*, that are a special case of aggregation functions called *averaging functions*. With these functions we have not only idempotence, but also the value of the function will be bounded by the minimum and maximum of the input arguments.

Definition 6. *An aggregation function of dimension n (n -ary aggregation function) is a non-decreasing mapping $f : [0, 1]^n \rightarrow [0, 1]$ such that $f(0, \dots, 0) = 0$ and $f(1, \dots, 1) = 1$.*

Definition 7. An aggregation function $f : [0, 1]^n \rightarrow [0, 1]$ is called averaging or a mean aggregation function if

$$\min(x_1, \dots, x_n) \leq f(x_1, \dots, x_n) \leq \max(x_1, \dots, x_n).$$

Proposition 8. (Fodor and Roubens (1994)) Idempotent monotonic non-decreasing functions and idempotent averaging functions are the same.

Example 5. Some averaging aggregation functions are:

1. The arithmetic mean.

$$\gamma_{\text{mean}}(x_1, \dots, x_n) = \frac{1}{n} \sum_{i=1}^n x_i \quad (1)$$

2. The median.

$$\gamma_{\text{median}}(x_1, \dots, x_n) = \begin{cases} \frac{1}{2}(x_k + x_{k+1}) & \text{if } n = 2k \\ x_k & \text{if } n = 2k - 1 \end{cases} \quad (2)$$

3. The min operator.
4. The max operator.

3.4. Specific case: OWA operators and fuzzy quantifiers

Introduced by Yager (1988), Ordered Weighted Averaging operators, commonly called OWA operators, are a parameterized family of idempotent averaging aggregation functions. They fill the gap between the operators *min* and *max*. The *min*, *max*, *arithmetic mean* or *median* are particular cases of this family.

Definition 8. (Yager (1988)) A mapping $F : [0, 1]^n \rightarrow [0, 1]$ is called an OWA operator of dimension n if there exists a weighting vector W ,

$W = (w_1, \dots, w_n) \in [0, 1]^n$ with $\sum_{i=1}^n w_i = 1$ and such that $F(a_1, \dots, a_n) = \sum_{j=1}^n w_j b_j$ with b_j the j -th largest of the a_i .

A natural question in the definition of the OWA operators is how to obtain the associated weighting vector. Our idea is to calculate the weights for the aggregation operators using linguistic quantifiers, e.g, *about 5*, *almost all*, *a few*, *many*, *most*, *as many as possible*, *nearly half*, *least half*. The

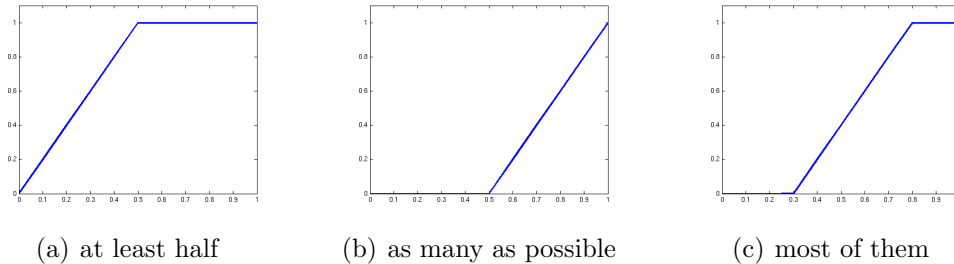


Figure 2: Proportional increasing quantifiers used to construct OWA operators

concept of a fuzzy quantifier was introduced by Zadeh (1983), offering a more flexible tool for knowledge representation.

Zadeh also suggested that the semantic of a fuzzy quantifier can be captured by using fuzzy sets for its representation. Two types of fuzzy quantifiers are distinguished, *absolute* and *proportional*. Absolute quantifiers are those used to represent amounts that are absolute in nature, such as *about 2* or *more than 5*. While proportional quantifiers, such as *most*, *at least half*, can be represented by fuzzy sets of the unit interval. For any $r \in [0, 1]$, $Q(r)$ indicates the degree to which the proportion r is compatible with the meaning of the quantifier it represents.

Fuzzy quantifiers are usually of one of three types, *increasing*, *decreasing* and *unimodal*. For instance, some increasing quantifiers are ‘at least half’, ‘as many as possible’ and ‘most of them’. Decreasing quantifiers are ‘few of them’ and ‘as much k’. Yager (1988) suggested an interesting way to compute the weights of the OWA aggregation operator using fuzzy quantifiers, which, in the case of an increasing quantifier Q , is given by the expression:

$$Q(r) = \begin{cases} 0 & \text{if } r < a \\ \frac{r-a}{b-a} & \text{if } a \leq r \leq b \\ 1 & \text{if } r > b \end{cases} \quad w_i = Q\left(\frac{i}{n}\right) - Q\left(\frac{i-1}{n}\right)$$

Figure 2 shows examples of proportional increasing quantifiers, ‘at least half’, ‘as many as possible’ and ‘most of them’, where the parameters (a, b) are $(0, 0.5)$, $(0.5, 1)$ and $(0.3, 0.8)$, respectively.

The OWA operators can also be studied by their properties. We use two known measures: $Disp(F)$ (Definition 9) and $orness(F)$ (Definition 10). The dispersion (or entropy) measures the degree to which we use all the aggregates equally. While orness measures its behaviour or optimism. An OWA operator F with much of the weights near the top will be an ‘orlike’ operator,

$orness(F) \geq 0.5$. At the other extreme, when the weights are non-zero near the bottom the OWA operator F will be ‘andlike’, $orness(F) \leq 0.5$.

Definition 9. (Yager (1993)) Let F be an OWA operator and W its weighting vector. The dispersion measure of F is defined as

$$Disp(F) = \sum_{i=1}^n w_i \ln(w_i).$$

Definition 10. (Yager (1993)) Let F be an OWA operator and W its weighting vector. The orness measure of F is defined as

$$orness(F) = \frac{1}{(n-1)} \sum_{i=1}^n (n-i)w_i.$$

In our case, we use the OWA operators over four images, $n = 4$. It can be seen that the OWA ‘at least half’ is an ‘orlike’ operator with an $orness(F_{least}) = 0.833$ and $Disp(F_{least}) = 0.6931$. OWA ‘most of them’, presents an ‘andlike’ behaviour with an $orness(F_{most}) = 0.1667$ and $Disp(F_{most}) = 0.693$, is the same as for the OWA ‘at least half’, because they aggregate with the same degree, although the weights are distributed differently. In our last case, for OWA ‘as many as possible’, the $orness(F_{many}) = 0.433$, treated as ‘orlike’ operator. Its dispersion is $Disp(F_{many}) = 0.9433$, contemplating almost all the weights.

4. Penalty functions and the consensus methodology

The concept of a *penalty function* P shown in Figure 3, is used in order to measure the disagreement or dissimilarity between n candidates, x_1, \dots, x_n , and one output of an aggregation function. Then, we rely on P to define the *Penalty-based function* for a set of q aggregation functions, $\{y_1, \dots, y_q\}$ (Definition 11). It is the function that returns the least dissimilar output y_i from the n candidates in the sense of the penalty function P used (see Calvo and Beliakov (2010)). We use this concept because we do not know before hand which aggregation function is better to use, so our aim is to find the one that minimizes the overall error with respect to the set of filtered images.

Definition 11. (Bustince et al. (2011)) A penalty function is a mapping $P : [0, 1]^{n+1} \rightarrow [0, \infty]$ such that:

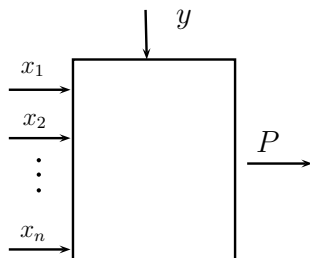


Figure 3: Schematic representation of a penalty function

1. $P(\mathbf{x}, y) = 0$ if $x_i = y$ for all $i = \{1, \dots, n\}$;
2. $P(\mathbf{x}, y)$ is quasi-convex in y for any \mathbf{x} ; that is $P(\mathbf{x}, \lambda \cdot y_1 + (1 - \lambda) \cdot y_2) \leq \max(P(\mathbf{x}, y_1), P(\mathbf{x}, y_2))$ for any $\lambda, y_1, y_2 \in [0, 1]$.

Let P be a penalty function. We call penalty-based function the mapping:

$$f(\mathbf{x}) = \arg \min_y P(\mathbf{x}, y),$$

If there exists only one point y in which $P(\mathbf{x}, \cdot)$ has a minimum, then $f(\mathbf{x}) = y$. Whereas if $P(\mathbf{x}, \cdot)$ has more than one minimum point, then from the quasi-convexity it follows that $P(\mathbf{x}, \cdot)$ attains its minimum on the whole interval of $[a, b]$ and we define $f(x) = \frac{a+b}{2}$. The quasi-convexity enforces that one and only one of these two possibilities happens. So, the minimum always exists, and either it is at a single point or at a whole interval.

Suppose we apply a *penalty-based function* considering the aggregated values for a pixel (i, j) using the three OWA operators, $(OWA_{least}, OWA_{most}, OWA_{many})$, on the set of the filtered images $\{FI_1(i, j), FI_2(i, j), FI_3(i, j), FI_4(i, j)\}$. Therefore, three different penalties $P(\mathbf{FI}(i, j), \cdot)$ are computed, namely $\{P(\mathbf{FI}(i, j), y_{OWA_{least}}), P(\mathbf{FI}(i, j), y_{OWA_{most}}), P(\mathbf{FI}(i, j), y_{OWA_{many}})\}$. Then, we take as solution of the *penalty-based function* the aggregated value y_{OWA} for the pixel (i, j) that has a minimum penalty P .

Theorem 2. (Calvo and Beliakov (2010)) *Any averaging aggregation function can be represented as a penalty-based function in the sense of Definition 11.*

In Calvo and Beliakov (2010) several well-known penalty-based aggregation functions are discussed in further detail.


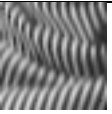
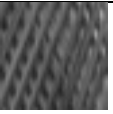
	Patch 1 	Patch 2 	Patch 3 	Global
γ_{mean}	261.65	278.19	294.95	285.45
γ_{median}	255.66	295.22	310.63	303.72

Table 1: MSE as penalty-based function calculated for the aggregation functions *arithmetic mean* (Eq. (1)) and *median* (Eq. (2)) on patches of 40x40 and on the entire image (512x512). The patches are extracted from the ‘Barbara image’. The starting left-corner pixels are: Patch 1: (155, 130); Patch 2: (200, 270); Patch 3: (20, 395).

4.1. Consensus

The penalty-based function allows us to select the aggregation function that minimizes the overall disagreement among a set of aggregation functions and the considered input candidates. But if the final image is obtained using one aggregation function for the entire image, we can just obtain the best global result. While if the best aggregation function is considered locally, it is reasonable to find another aggregation function that works better at pixel level. Accordingly, if the final image is built taking from the set of aggregation functions the best ones found locally, the final global result also improves. In Table 1 an example is shown using the well-known *barbara* image that represents this phenomenon at patches level. We use the mean square error (MSE) (Tang and Cahill (1992)) as a penalty-based function, defined as $MSE(\mathbf{x}, y) = \frac{1}{n} \sum_{i=1}^n (x_i - y)^2$; where $n = 4$, equal to the number of filtered images from a barbara noisy image. We take as possible output functions the $\gamma_{mean}(\mathbf{x})$ (Eq. (1)) and $\gamma_{median}(\mathbf{x})$ (Eq. (2)). Then, we calculate the MSE for the entire image and over small cropped neighborhoods of (40×40) pixels. The results show that the function γ_{mean} presents a better overall behaviour than γ_{median} , but for instance, γ_{median} is sometimes better in small patches, as in *Patch 1*.

Therefore, the proposed consensus methodology selects for each pixel the best aggregation function among the considered ones, instead of taking just one aggregation function as a solution. In our case, we combine the three different OWA operators in all their possible combinations at pixel level. Then, the penalty for all the combinations is measured and we take as solution the image with minimal penalty, and thus, the consensus final image.

In other words, the consensus methodology works as follows. Starting already from a stack of images (Figure 4(a)). For each pixel (i, j), a multi-

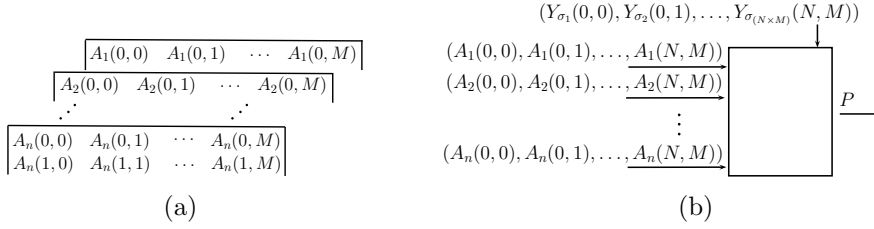


Figure 4: (a) Representation of a stack of n images of size $N \times M$. (b) Representation of a consensus problem by a penalty function, where σ is the set of arrangements with repetition of q idempotent functions taken in groups of $N \times M$.

fuzzy set, $A(i, j) = \{A_1(i, j), \dots, A_n(i, j)\}$ is built, where n is the number of images. Then, we apply q different aggregation functions to $A(i, j)$ ¹, i.e. we get $f_1(A), \dots, f_q(A)$ where f_k is any idempotent aggregation function, $f_k(A) = f_k(A_1, \dots, A_n)$.

Afterwards, using the obtained images $f_1(A), \dots, f_q(A)$, we build a new set of images conformed by all the possible permutations, where the best permutation is the one which minimizes the penalty (Figure 4(b)). So, $Y_{\sigma_k}(i, j)$ is the value for the pixel (i, j) in the image $\sigma_k \in \{f_1(A), \dots, f_q(A)\}$, selected for the position $k \in \{0, 1, \dots, N \times M\}$ in the candidate image. The consensus image is the $Y_{\sigma_1}(0, 0), Y_{\sigma_2}(0, 1), \dots, Y_{\sigma_{(N \times M)}}(N, M)$ that is minimum.

It can be deduced that the performance of consensus depends on the considered penalty function, as well as the selected aggregation functions. Moreover, from its definition, consensus is applied over all the pixels that conforms the image as it is shown in Figure 4(b). But this approach is demanding computationally and unmanageable calculations for a nowadays PC. E.g. a 256x256 image considering three aggregation functions, will need to compute 3^{65536} possible alternatives. We consider the possibility to apply the consensus over subgroups of pixels. In our case, square blocks of four neighbor elements. These new considerations lead us to use *penalty functions over cartesian product of lattices*.

4.2. Cartesian product of lattices

Because applying consensus over the entire image is computationally demanding and time consuming, we propose the alternative to apply consensus in small groups of pixels that makes faster the execution. As a result, we study penalty functions over a cartesian product of lattices.

¹We change the notation from $A(i, j)$ to A to simplify the notation.

Definition 12. A poset (P, \leq) is a set P with a relation \leq which is reflexive, antisymmetric and transitive. A chain is a totally ordered set. The length of a chain is given by the cardinality of the chain minus one.

Definition 13. A lattice $L = \{L, \leq, \vee, \wedge\}$ is a poset with the partial ordering \leq in L and operations \vee and \wedge which satisfy the properties of absorption, idempotency, commutativity, and associativity. That is, a poset such that any two elements have a unique minimal upper bound and a unique maximal lower bound in L .

Proposition 9. (Bustince et al. (2011)) Let $L_1 = \{L_1, \leq_1, \vee_1, \wedge_1\}$ and $L_2 = \{L_2, \leq_2, \vee_2, \wedge_2\}$ be two lattices. The Cartesian product $L_1 \times L_2 = \{L_1 \times L_2, \leq, \vee, \wedge\}$ with \leq defined componentwise by

$$(x_1, x_2) \leq (y_1, y_2) \text{ if and only if } x_1 \leq_1 y_1 \text{ and } x_2 \leq_2 y_2$$

and

$$\begin{aligned} \vee((x_1, x_2), (y_1, y_2)) &= (\vee_1(x_1, y_1), \vee_2(x_2, y_2)) \\ \wedge((x_1, x_2), (y_1, y_2)) &= (\wedge_1(x_1, y_1), \wedge_2(x_2, y_2)) \end{aligned}$$

is a lattice.

4.3. Penalty functions over a cartesian product of lattices: Building method

Using a cartesian product of lattices we can build penalty functions over it, and so we can apply the consensus technique over a small group of pixels. In Theorem 3 we present a building method.

Theorem 3. (Bustince et al. (2011)) Let $\mathcal{F}(U)$ be the set of all fuzzy sets defined on the finite referential and non-empty set $U = \{u_1, \dots, u_n\}$ ($\#(U) = n$). Let denote by B_{y_q} the fuzzy set over U such that all its membership values are equal to $y_q \in [0, 1]$; that is, $B_{y_q}(u_i) = y_q$ for all $u_i \in U$. Let $Y = (y_1, \dots, y_m) \in [0, 1]^m$ and $B_Y = (B_{y_1}, \dots, B_{y_m}) \in \mathcal{F}(U)^m$. Let C^* be a chain, that is a linear lattice, whose elements belong to $[0, 1]$ and let L_m^* be the cartesian product $L_m^* = C^* \times \dots \times C^*$. Let $K_i : \mathbb{R} \rightarrow \mathbb{R}^+$ be convex functions with a unique minimum at $K_i(0) = 0$, ($i = 1, \dots, m$), and take the taxi-distance between fuzzy sets in U defined as

$$D(A, B) = \sum_{i=1}^n |A(u_i) - B(u_i)| \quad (3)$$

Then the mapping $P_{\nabla} : \mathcal{F}(U)^m \times L_m^* \rightarrow \mathbb{R}^+$ given by

$$P_{\nabla}(\mathbf{A}, Y) = \sum_{q=1}^m K_q(D(A_q, B_{y_q})) = \sum_{q=1}^m K_q \left(\sum_{p=1}^n |A_q(u_p) - y_q| \right) \quad (4)$$

satisfies

1. $P_{\nabla}(\mathbf{A}, Y) = 0$ if and only if $A_q = y_q$ for every $q = 1, \dots, m$;
2. $P_{\nabla}(\mathbf{A}, Y)$ is convex in y_q for every $q = 1, \dots, m$.

Example 6.

- From the hypothesis in Theorem 3 we take as convex functions with unique minimum at zero the following: $K_q(x) = x^2$ for all $q \in \{1, \dots, m\}$, then

$$P_{\nabla}(\mathbf{A}, Y) = \sum_{q=1}^m \left(\sum_{p=1}^n |A_q(u_p) - y_q| \right)^2 \quad (5)$$

- If $K_q(x) = x$ for all $q \in \{1, \dots, m\}$, then

$$P_{\nabla}(\mathbf{A}, Y) = \sum_{q=1}^m \sum_{p=1}^n |A_q(u_p) - y_q| \quad (6)$$

Theorem 4. (Bustince et al. (2011)) In the setting of Theorem 3, the mapping

$$F(\mathbf{A}) = \mu(\arg \min_Y P_{\nabla}(\mathbf{A}, Y)) \quad (7)$$

where μ is the rounding to the smallest closest element, is an averaging aggregation function.

Proof. Just observe that

$$\begin{aligned} \arg \min_{(y_1, \dots, y_m)} P_{\nabla}(\mathbf{A}, (y_1, \dots, y_m)) &= \arg \min_{(y_1, \dots, y_m)} \sum_{q=1}^m K_q \left(\sum_{p=1}^n |A_q(u_p) - y_q| \right) \\ &= \sum_{q=1}^m \arg \min_y K_q \left(\sum_{p=1}^n |A_q(u_p) - y_q| \right) \end{aligned}$$

so it is enough to consider each of the quantities

$$\arg \min_y K_q \left(\sum_{p=1}^n |A_q(u_p) - y_q| \right)$$

but each of these functions is an aggregation function and since K_q is convex, the result follows. \square

To reach a consensus in our experiments, we use the penalty-based function over a cartesian product of lattices presented in Eq. (7), as well as the penalty function over a cartesian product of lattices from Eq. (5).

5. Results and discussion

Different experiments are carried out to illustrate the consensus behavior facing the blind denoising task. To be able to compare the results to a ground truth, we work with images with 256 gray levels artificially corrupted with noise. Two databases are used: Live Image Quality Assessment Database (live (2013)) and a magnitude MR T1 volumen originally noise-free from the BrainWeb database (brainWeb (2013)). The first database is corrupted with Gaussian and Poisson noise. While the second one is corrupted with Rician noise. In both cases, the noisy images are processed using different noise removal filters (see Section 2.1). The aggregation functions used to reach a consensus are the OWA operators: ‘at least half’, ‘as many as possible’ and ‘most of them’ (see Section 3.4). Finally, the based-penalty function (Eq.(7)) is applied over a cartesian product of lattices in groups of four neighbors pixels using the penalty function from Eq. (5).

To quantify the restoration performance of different methods, the PSNR and MSE (Tang and Cahill (1992)) are calculated. These are not bounded. A higher MSE represents worst quality, while a higher PSNR means better quality. They are simple to calculate, have clear physical meaning, and are mathematically convenient in the context of optimization. However it is not very well matched to perceived visual quality. This is our motivation to use also other quality indexes. In addition, the Mean Structural Similarity Index (MSSIM) (Wang et al. (2004)) and the Quality Index based on Local Variance (QILV) (Aja-Fernandez et al. (2006)) are used. Both give a measure of the structural similarity between the ground truth and the estimated images. Nonetheless, the former is more sensitive to the level of noise in the image and the latter to any possible blurring of the edges. This way we are able to assess the noise cleaning and border preserving capability of the

Filter	MSE		PSNR		MSSIM		QILV	
	mean	std	mean	std	mean	std	mean	std
Noisy	382.847	23.298	22.309	0.289	0.427	0.115	0.703	0.163
Impulse	379.338	23.043	22.218	0.263	0.432	0.114	0.711	0.165
Poisson	188.827	42.166	25.459	0.926	0.584	0.100	0.903	0.085
Gaussian	73.906	39.302	29.818	2.323	0.840	0.054	0.939	0.044
Rician	119.418	39.214	27.464	1.434	0.744	0.034	0.952	0.032
OWA_{least}	170.218	29.878	25.730	0.708	0.640	0.084	0.947	0.044
OWA_{many}	170.511	31.353	25.842	0.751	0.647	0.065	0.953	0.037
OWA_{most}	117.657	30.864	27.542	1.075	0.700	0.059	0.965	0.021
<i>Consensus</i>	116.530	31.335	27.548	1.081	0.700	0.061	0.965	0.022

Table 2: Results for the database live (2013), that contains 18 images 512x512 contaminated with Gaussian noise with $\sigma = 20$.

different schemes. Both indexes are bounded; the closer to one, the better the image.

A first experiment was accomplished with the 18 images from the live (2013) database corrupted with Gaussian noise. Table 2 contains the averages and the standard deviations achieved for this experiment. The Gaussian filter obtains the best results in average as expected, except for the QILV measure. However the difference between the best QILV result and the one obtained for the Gaussian filter is negligible. Furthermore, the reached consensus is just behind the best, as the second better approach. We can also observe that if we would have not applied consensus, we could have obtained worse results. For instance, in case we would have just decided to use a single aggregation function, because we do not know beforehand what aggregation function is better to use. Although in this experiment the results for the operator ‘most of them’ are similar to the proposed approach.

In a second experiment we want to analyze how the noise level affects the consensus performance. For it, we executed several times the same database (live (2013)) contaminated with various sigma values for Gaussian noise, from low to a high noise level. Figure 5 presents the graphs for the different quality measures. The image quality is affected as the noise level increases, although not all the measures and filters are affected in the same way. For example, the Rician filter preserves its performance for MSSIM as the noise amount increases due to its conservative behavior around the borders; or the Poisson filter worsens its achievement with the noise increasing. However, the consensus performance keeps as one of the three better approaches. The

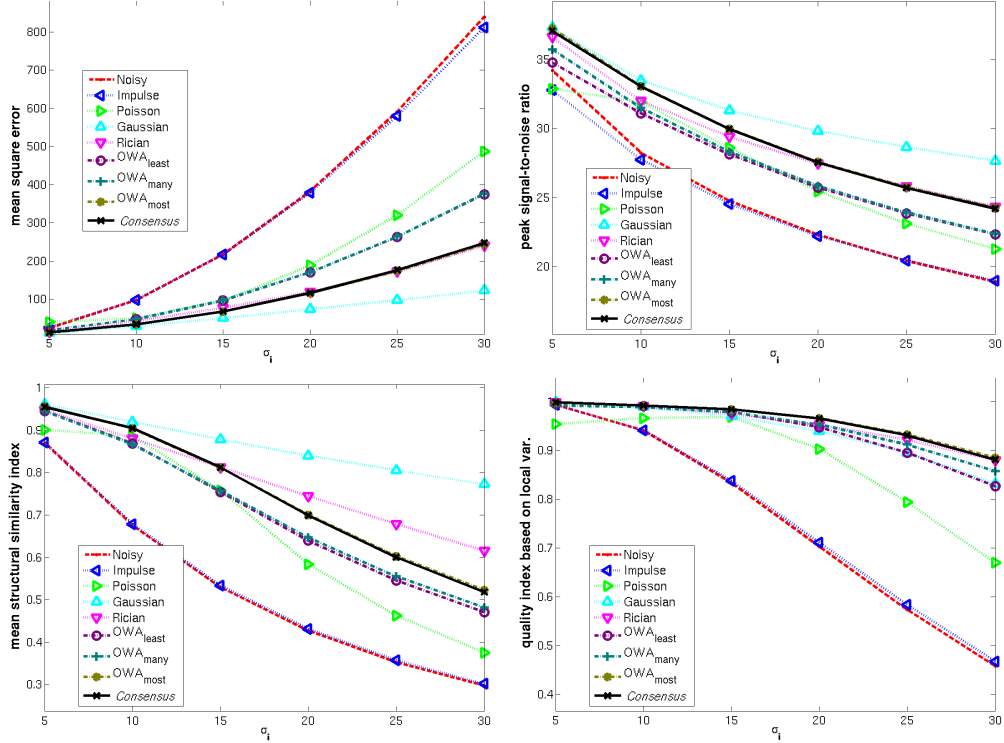


Figure 5: Quality results for the different used filters, the OWA operators and the reached consensus for different executions with various sigmas ($\sigma_i = \{5, 10, 15, 20, 25, 30\}$). It represents the mean values for the 18 images contained in the database live (2013). By order, from left to right, and from up to down: MSE, PSNR, MSSIM and QILV.

noise level affects its performance because is affected by the errors of the used filtered and aggregated images that are also altered by noise. However, the proposed approach still shows a good compromise.

Consensus is an approach that can be used under different noise models without previous information. Then, in a new experiment we contaminated the same database (live (2013)) with Poisson noise. We can recognize a coherent and consistent behaviour in Table 3, also in line with the previous experiments. The reached consensus obtains the best MSE, even better than for the Poisson filter, and a MSSIM close to the best achievement by the Poisson filter. Furthermore, we can see that consensus obtains better results than any of the single aggregation functions. As we already mention thanks to the reached consensus locally, the global result also improves.

Filter	MSE		PSNR		MSSIM		QILV	
	mean	std	mean	std	mean	std	mean	std
Noisy	111.974	11.221	27.665	0.424	0.666	0.099	0.926	0.068
Impulse	113.504	10.734	27.246	0.830	0.669	0.098	0.928	0.066
Poisson	49.455	33.528	32.007	2.722	0.896	0.031	0.965	0.022
Gaussian	72.211	41.597	30.005	2.540	0.820	0.076	0.871	0.100
Rician	115.462	82.705	28.101	2.275	0.792	0.069	0.879	0.091
OWA_{least}	78.704	16.978	28.983	0.946	0.790	0.052	0.982	0.010
OWA_{many}	86.148	20.770	28.884	0.992	0.769	0.052	0.974	0.015
OWA_{most}	53.378	12.056	30.919	0.953	0.804	0.046	0.986	0.008
<i>Consensus</i>	46.309	23.611	31.911	2.180	0.878	0.032	0.937	0.046

Table 3: Results for the database live (2013), which contains 18 images 512x512 contaminated with Poisson noise.

A last experiment was performed with a different noise model, the Rician distribution. For this experiment a MRI volumen from the brainWeb (2013) database has been used. The volumen contains 181 images free of noise that were contaminated with Rician noise. To avoid any bias in the results due to the background, the quality measures are only applied to those areas of the image that are relevant, in other words, inside of the skull. Table 4 and Table 5 show once again that consensus is one of the better approaches, just behind the best ones. We can also observe that the Rician filter performs worse than the Gaussian filter. This may be due to the contaminated image that closely approximates to the Gaussian distribution than a Rayleigh or Rician distribution; or even to the conservative nature of the Rician filter, that in case the data does not fit the model, the filter prefers to preserve the original data. Moreover, when the noise level increases (Table 5) the consensus works similarly to the previous case, where achievement gets affected by noise. It is natural since consensus looks for a cooperation between images, that also get affected by noise.

In summary, one can see that consensus does not get the best results, however it finds a cooperation between the considered filters. It obtains always a good performance and assures a result better than the worst of the individual solutions. Therefore, consensus is a good approach in situations where we do not know which filter to use, or when each filter performs better in different image regions. The main gains of this approach is the flexibility provided for an unknown noise model, where we can use several filters randomly, and the presented methodology still gets a good compromise.

Filter	MSE		PSNR		MSSIM		QILV	
	mean	std	mean	std	mean	std	mean	std
Noisy	60.882	20.126	30.803	1.951	0.871	0.043	0.970	0.056
Impulse	60.615	20.141	30.803	1.959	0.872	0.043	0.972	0.053
Poisson	21.358	7.505	35.395	2.089	0.960	0.015	0.991	0.008
Gaussian	15.819	6.317	36.966	2.900	0.970	0.013	0.994	0.004
Rician	34.489	11.636	33.446	2.370	0.942	0.019	0.994	0.004
OWA_{least}	41.044	13.625	32.503	1.952	0.920	0.026	0.990	0.022
OWA_{many}	36.161	12.665	33.139	2.179	0.924	0.027	0.993	0.010
OWA_{most}	31.064	10.373	33.742	2.008	0.927	0.024	0.992	0.016
Consensus	23.878	8.403	34.980	2.342	0.957	0.014	0.995	0.003

Table 4: Results for the MR volumen, which contains 181 MR images contaminated with Rician noise with $\sigma = 10$.

Filter	MSE		PSNR		MSSIM		QILV	
	mean	std	mean	std	mean	std	mean	std
Noisy	239.592	80.229	24.866	1.986	0.720	0.089	0.826	0.154
Impulse	237.456	80.157	24.905	2.005	0.720	0.100	0.835	0.150
Poisson	120.111	39.777	27.836	1.915	0.808	0.061	0.940	0.114
Gaussian	41.602	16.145	32.629	2.483	0.927	0.030	0.970	0.021
Rician	86.739	29.130	29.310	2.156	0.873	0.040	0.973	0.032
OWA_{least}	126.857	40.974	27.578	1.864	0.825	0.055	0.959	0.084
OWA_{many}	94.412	33.743	29.012	2.318	0.854	0.052	0.977	0.029
OWA_{most}	76.856	26.550	29.853	2.170	0.865	0.032	0.978	0.032
Consensus	78.344	26.951	29.764	2.151	0.863	0.045	0.977	0.035

Table 5: Results for the MR volumen, which contains 181 MR images contaminated with Rician noise with $\sigma = 20$.

Quantitative results already prove it, but it is also supported qualitatively with the images shown in Fig. 6 to 10. We can see how the reached consensus from the different images contaminated with different noise models (Gaussian, Poisson and Rician noise) exhibit better visual quality than any of the single filters. For instance, the Gaussian filtered images show over-filtering and loss of details in textured areas. In Fig. 6(c) the area in the nose is over-filtered and the areas closeby have lost details in the texture. A similar situation occurs in Fig. 7(c) where the image is in general over-filtered. For Fig. 8(c) the general quality looks pleasant, although the face area is blurred, as well as some areas have lost the textures. The Gaussian filtered image in the MRI approach, Fig. 9(c), presents a good quality.

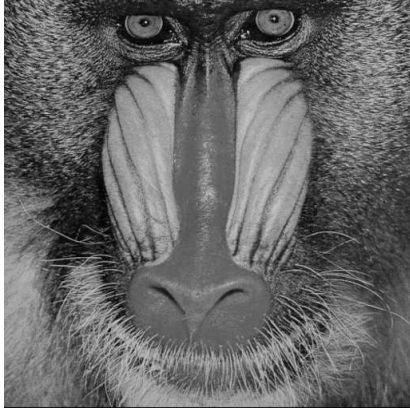
Nonetheless, the zoomed image, Fig. 10(c), shows ringing artifacts close to the edges. A similar outcome is gotten for the Poisson filter in the Rician case, Fig. 9(e), that also shows ringing artifacts. On the contrary, the Poisson filter achieves the best result for the Poisson approach, Fig. 8(e), and comparable to the consensus image, Fig. 8(f), as discussed for the results in Table 3. On the other hand, if we compare the remaining filtered images to the original ones, they are noisier than the reached consensus. The impulse filtered images, Fig. 6(b), 7(b), 8(b) and 9(b), are not an alternative. Neither the Poisson filter for the Gaussian problems, Fig. 6(e) and 7(e). Finally, in the case of the Rician filtered images, they usually keep some noise. Although the effect could be visually pleasant for some approaches, as the Gaussian and Rician approach, Fig.6(d), 7(d) and 9(d).

6. Conclusion

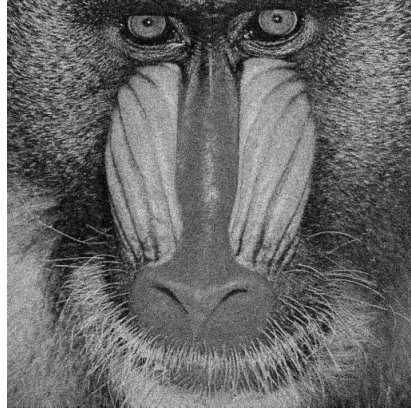
Results show that *consensus* presents a powerful tool for denoising with unknown noise distribution, because the noise is not known, the noise model does not follow the assumptions, or the image contains different noise sources that cannot be properly modeled. Therefore, consensus is a good alternative in situations where we do not know beforehand the best filter to apply, or when a combination of different filters performs better than any single filter. Consensus always brings a good final result, consistent and stable. Although, it must be contemplated that consensus computational time is dependent on the number of aggregation functions considered. For instance, if the number of used aggregation functions increases, the computational cost increases exponentially. This is a drawback that needs further research. Additionally, the results tend to the mean when the number of idempotent functions increases. The challenge is to find a compromise on the number of aggregation functions to obtain a good performance. Furthermore, the used penalty function also affects the final results. Nonetheless, instead of considering it as a drawback, it can be seen as an advantage that brings flexibility to the system. Therefore, the system can be considered as a framework. Further research in the aggregation functions and penalty function selection is carried out.

7. Acknowledgement

This work is supported by the European Commission under contract no. 238819 (MIBISOC Marie Curie ITN). H. Bustince was supported by Project TIN 2010-15055 of the Spanish Ministry of Science.



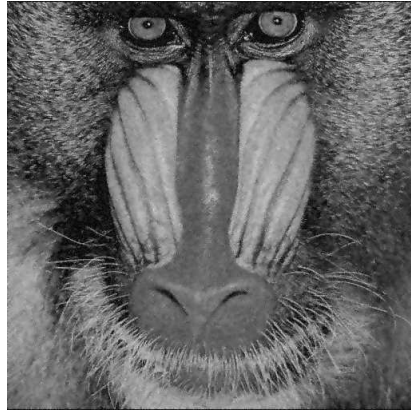
(a) Original



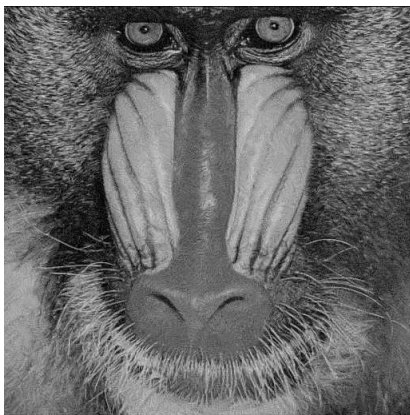
(b) Impulse



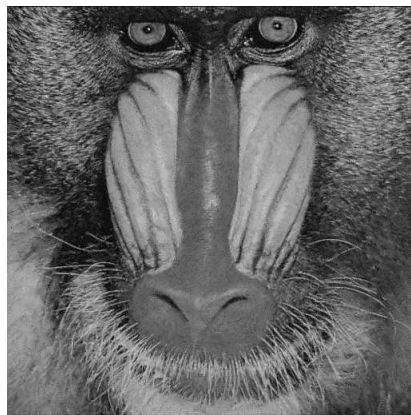
(c) Gaussian



(d) Rician

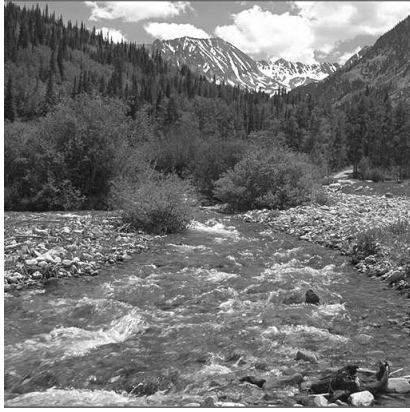


(e) Poisson

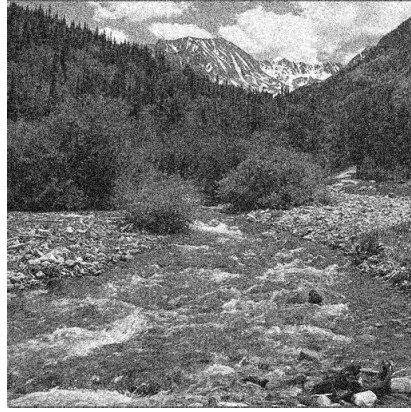


(f) Consensus

Figure 6: Noise removal results for a Gaussian noisy image contaminated with $\sigma = 20$ using different filters (Impulse, Gaussian, Rician and Poisson) and the reached consensus image.



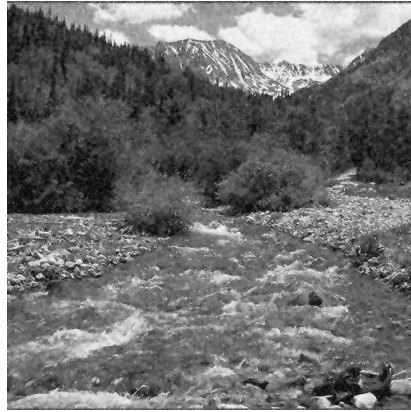
(a) Original



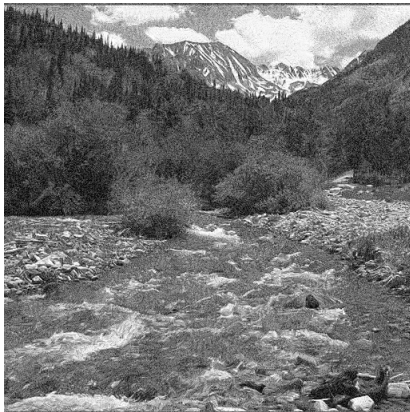
(b) Impulse



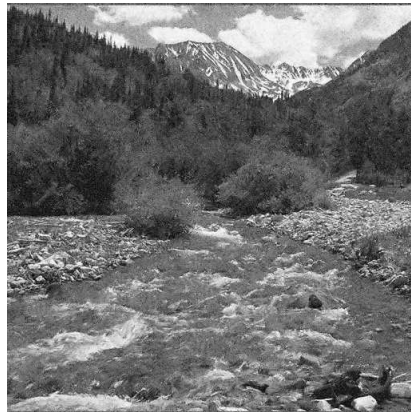
(c) Gaussian



(d) Rician



(e) Poisson



(f) Consensus

Figure 7: Noise removal results for a Gaussian noisy image contaminated with $\sigma = 30$ using different filters (Impulse, Gaussian, Rician and Poisson) and the reached consensus image.



(a) Original



(b) Impulse



(c) Gaussian



(d) Rician



(e) Poisson



(f) Consensus

Figure 8: Noise removal results for a Poisson noisy image using different filters (Impulse, Gaussian, Rician and Poisson) and the reached consensus image.

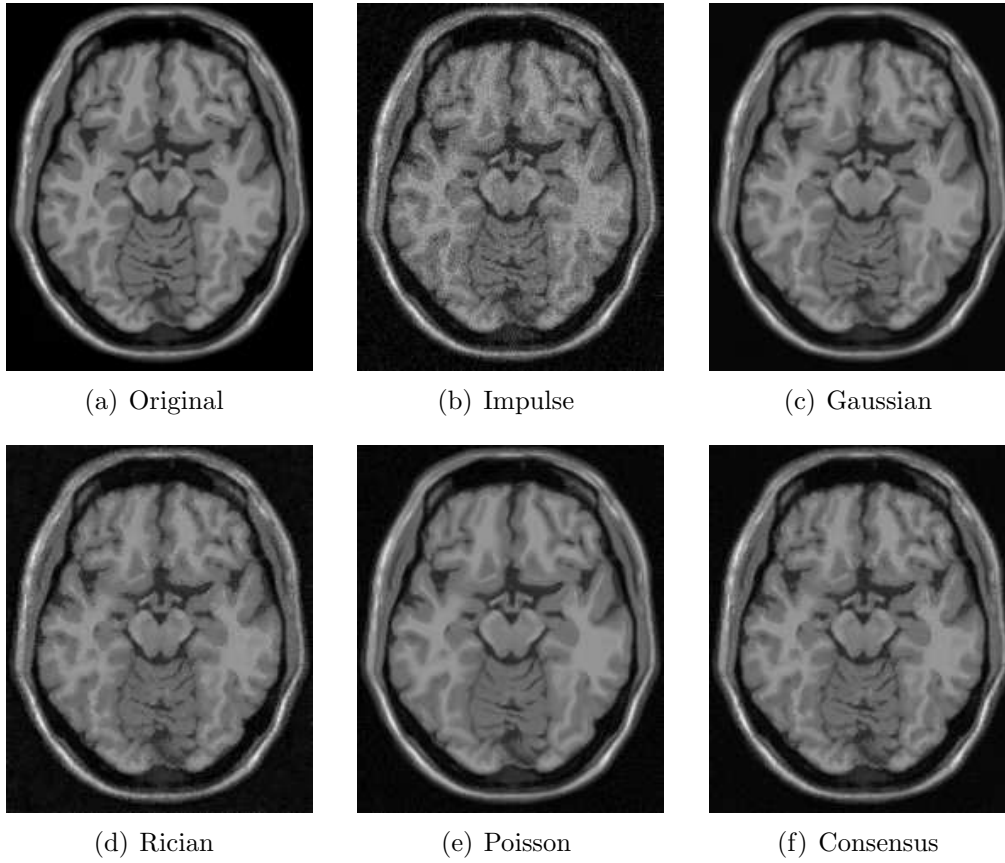


Figure 9: Noise removal results for a Rician noisy image contaminated with $\sigma = 10$ using different filters (Impulse, Gaussian, Rician and Poisson) and the reached consensus image.

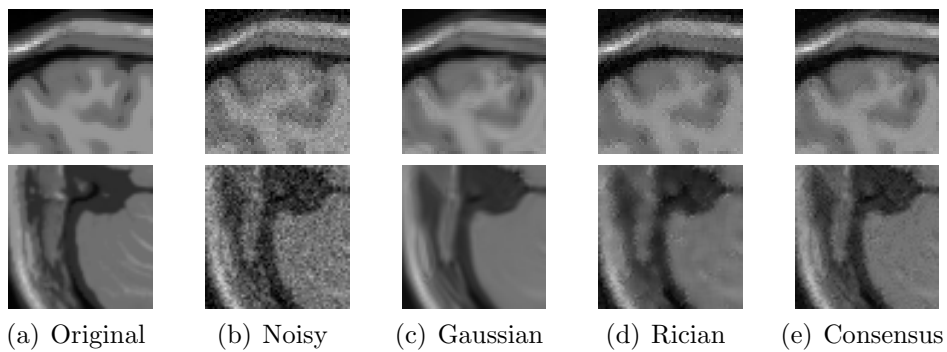


Figure 10: Region extracted from two MR brain images contaminated with Rician noise. The former row is the image contaminated with $\sigma = 10$; the second row with $\sigma = 20$.

References

- Aja-Fernandez, S., Alberola-Lopez, C., Westin, C.-F., 2008. Noise and signal estimation in magnitude MRI and Rician distributed images: A LMMSE approach. *IEEE Transactions on Image Processing* 17 (8), 1383–1398.
- Aja-Fernandez, S., Estepar, R. S. J., Alberola-Lopez, C., Westin, C. F., 2006. Image quality assessment based on local variance. In: *Conference Proceedings. Annual International Conference of the IEEE Engineering in Medicine and Biology Society (IEEE Cat. No. 06CH37748)*. IEEE EMB, IEEE, Piscataway, NJ, USA, pp. 4815–4818.
- Aja-Fernandez, S., Tristan-Vega, A., Alberola-Lopez, C., 2009. Noise estimation in single- and multiple-coil magnetic resonance data based on statistical models. *Magnetic Resonance Imaging* 27 (10), 1397–1409.
- Bernstein, M. A., Thomasson, D. M., Perman, W. H., 1989. Improved detectability in low signal-to-noise ratio magnetic-resonance images by means of a phase-corrected real reconstruction. *Medical Physics* 16 (5), 813–817.
- Borsdorf, A., Raupach, R., Flohr, T., Hornegger, J., Dec. 2008. Wavelet Based Noise Reduction in CT-Images Using Correlation Analysis. *Medical Imaging, IEEE Transactions on* 27 (12), 1685–1703.
- brainWeb, Feb. 2013. BrainWeb: Simulated Brain Database.
URL <http://mouldy.bic.mni.mcgill.ca/brainweb/>
- Buades, A., Coll, B., Morel, J. M., 2005. A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation* 4 (2), 490–530.
- Bustince, H., Barrenechea, E., Calvo, T., James, S., Beliakov, G., Nov. 2011. Consensus in multiexpert decision making problems using penalty functions defined over a Cartesian product of lattices. In “*Information Fusion*”.
- Calvo, T., Beliakov, G., 2010. Aggregation functions based on penalties. *Fuzzy Sets and Systems* 161 (10), 1420–1436.
- Chiclana, F., Herrera, F., Herrera-Viedma, E., 1998. Integrating three representation models in fuzzy multipurpose decision making based on fuzzy preference relations. *Fuzzy Sets and Systems* 97 (1), 33–48.
- Deledalle, C.-A., Tupin, F., Denis, L., 2010. Poisson NL means: Unsupervised non local means for poisson noise. In: *2010 IEEE International Conference on Image Processing (ICIP); IEEE Signal Process Soc, IEEE, 345 E 47TH ST, New York, NY 10017 USA*, pp. 801–804.
- Fodor, J. C., Roubens, M. R., 1994. *Fuzzy preference modelling and multicriteria decision support*. Vol. 14. Springer.
- González-Jaime, L., Nachtgael, M., Kerre, E., Bustince, H., Dec. 2012. Use of Idempotent Functions in the Aggregation of Different Filters for Noise Removal. In *2012 International Conference on “Intelligent Systems and Knowledge Engineering (ISKE)”*, Tsinghua University, China.
- Goossens, B., Luong, Q., Pizurica, A., Philips, W., 2008. An improved non-local denoising algorithm. In: *Local and Non-Local Approximation in Image Processing, International Workshop, Proceedings*. pp. 143–156.
- Hsieh, J., 1998. Adaptive streak artifact reduction in computed tomography resulting from excessive X-ray photon noise. *Medical Physics* 25 (11), 2139–2147.
- Klir, G. J., Yuan, B., 1995. *Fuzzy sets and fuzzy logic*. Prentice Hall New Jersey.
- La Riviere, P. J., Billmire, D. M., Jan. 2005. Reduction of noise-induced streak artifacts

- in X-ray computed tomography through spline-based penalized-likelihood sinogram smoothing. *Medical Imaging, IEEE Transactions on* 24 (1), 105–111.
- live, Feb. 2013. LIVE Image Quality Assessment Database.
URL <http://live.ece.utexas.edu/research/quality/>
- Molina, R., Nunez, J., Cortijo, F. J., Mateos, J., 2001. Image restoration in astronomy - A Bayesian perspective. *IEEE Signal Processing Magazine* 18 (2), 11–29.
- Morris, T., Sep. 2003. *Computer vision and image processing*. Palgrave Macmillan.
- Pahl, P. J., Damrath, R., 2001. *Mathematical foundations of computational engineering: a handbook*. Springer Verlag.
- Rosenthal, M. S., Cullom, J., Hawkins, W., Moore, S. C., Tsui, B. M. W., Yester, M., 1995. Quantitative SPECT imaging - A review and recommendations by the focus committee of the society-of-nuclear-medicine computer and instrumentation council. *Journal of Nuclear Medicine* 36 (8), 1489–1513.
- Rudin, L. I., Osher, S., Fatemi, E., Nov. 1992. Nonlinear total variation based noise removal algorithms. *Physica D* 60 (1-4), 259–268.
- Skiadopoulos, S., Karatrantou, A., Korfiatis, P., Costaridou, L., Vassilakos, P., Apostolopoulos, D., Panayiotakis, G., 2009. Evaluating image denoising methods in myocardial perfusion single photon emission computed tomography (SPECT) imaging. *Measurement Science & Technology* 20 (10), 104023+.
- Srinivasan, K. S., Ebenezer, D., 2007. A new fast and efficient decision-based algorithm for removal of high-density impulse noises. *IEEE Signal Processing Letters* 14 (3), 189–192.
- Suzuki, S., 1985. A comparative-study on pre-smoothing techniques for projection data with poisson noise in computed-tomography. *Optics Communications* 55 (4), 253–258.
- Tang, H., Cahill, L. W., Nov. 1992. A new criterion for the evaluation of image restoration quality. In: *TENCON'92. Technology Enabling Tomorrow: Computers, Communications and Automation towards the 21st Century. 1992 IEEE Region 10 International Conference*. IEEE, pp. 573–577.
- Wang, Z., Bovik, A. C., Sheikh, H. R., Simoncelli, E. P., Apr. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13 (4), 600–612.
- Yager, R. R., 1988. On ordered weighted averaging aggregation operators in multicriteria decision-making. *IEEE Transactions on Systems Man and Cybernetics* 18 (1), 183–190.
- Yager, R. R., Oct. 1993. Families of OWA operators. *Fuzzy Sets and Systems* 59 (2), 125–148.
- You, Y. L., Xu, W. Y., Tannenbaum, A., Kaveh, M., Nov. 1996. Behavioral analysis of anisotropic diffusion in image processing. *IEEE Transactions on Image Processing* 5 (11), 1539–1553.
- Zadeh, L. A., 1983. A computational approach to fuzzy quantifiers in natural languages. *Computers & Mathematics with applications* 9 (1), 149–184.