Color image processing based on Graph Theory

PhD DISSERTATION

Color image processing based on Graph Theory

PhD DISSERTATION

Advisors
Dr. José Alberto Conejero Casares
Dra. Cristina Jordán Lluch
Dr. Samuel Morillas Gómez

Author
Cristina Pérez Benito

Valencia, May 2019

A mis padres y a mis hermanos,
por creer y confiar en mi en todo momento.
A mi tito Javi,
por todo el cariño que no tuve tiempo de devolverte.

Ideal

Agradecimientos

Hace ya tiempo que comenzó la nueva etapa que me ha traído hasta aquí. Han sido unos años maravillosos, pero también especialmente complicados personal y profesionalmente. Es por ello que quería aprovechar este momento para agradecer a las muchísimas personas que me he han ayudado, apoyado y enseñado durante este tiempo, y gracias a las cuales estoy hoy aquí.

En primer lugar, a mis directores, Dr. Samuel Morillas, Dra. Cristina Jordán y Dr. Alberto Conejero. Las especiales circunstancias del desarrollo de esta Tesis Doctoral también han requerido un gran esfuerzo por vuestra parte que querría agradeceros. Gracias por confiar en mí, por vuestra paciencia y por guiarme a lo largo de esta etapa.

A mis compañeros del Instituto de Biomecánica de Valencia, que sin duda han sido una pieza clave durante este tiempo. Gracias a todo el grupo de Antropometría por todo lo que me habéis enseñado y las facilidades que me brindasteis para el desarrollo de esta tesis. He aprendido muchísimo de todos vosotros.

A toda la gente que me arropó en Valencia, tanto a mi llegada, como en los años siguientes. En especial, gracias a Leti y María, gracias por todo vuestro cariño y por hacer de mi llegada un año inolvidable. Y, por supuesto, a Víctor, gracias por todas las veces que me has ayudado en este caos que llevo siempre y gracias por tus palabras de fuerza cuando sentía que no podía más.

A toda la gente de Barcelona que me ha hecho este primer año un poquito más fácil y que ha vivido con intensidad la recta final de estas tesis, preguntándome a diario y prestándome su ayuda.

A toda la maravillosa gente que Salamanca me ha regalado, desde la infancia hasta los últimos y geniales años de Universidad. En especial, gracias a mis *chicas* por haberme hecho la vida mucho más divertida y fácil y por haber estado en los buenos y malos momentos. Gracias Marta por ser un ejemplo de lucha y perseverancia. Gracias Irene por esa infinita sonrisa, energía y felicidad que desprendes y que has conseguido trasladadame tantas y tantas veces. Gracias Patri, por ser la interminable voz al otro lado del teléfono que siempre necesito y necesitaré escuchar. Y gracias, muchísimas gracias Alba, por todas y cada una de las veces que me has cogido de la mano en los peores

momentos, por mucho que doliese. Gracias a todas, no os imagináis lo mucho que he aprendido y crecido con vosotras.

A Néstor, gracias por aparecer en mi vida, por ser mi compañero de viaje y por haber estado a mi lado en todo momento. Gracias por llorar y reír conmigo cada día y haberme acompañado en cada uno de los días de agobio, cansancio y llanto, que no han sido pocos, durante este tiempo.

Y por supuesto, a mi familia, por su apoyo incondicional. A mis padres Jose y Puri, y a mis hermanos Javi y Óscar, porque no hay páginas suficientes para agradeceros todo lo que habéis hecho durante estos 26 años. Gracias por animarme a estudiar lo que me gustaba, por enseñarme lo que es el esfuerzo y la constancia y, por supuesto, por apoyarme en todo momento y en todas las decisiones que he tomado. A mi abuela Ignacia, por hacerme sentir cada día que puedo hacer todo lo que me proponga. A mi abuelo Isidoro, porque puedo imaginar la cara de orgullo que tendrías ahora mismo. A mi abuela Chelo, por lo que me ha costado tener que cambiar estas palabras, porque sé lo mucho que te habría gustado leer esto y porque sigo sintiendo ese inmenso amor y orgullo que sentía cada vez que hablabamos. Y finalmente, a mi tito Javi, por todo lo que me has querido como tío, padrino y amigo y porque, pase el tiempo que pase, siempre me acompañarás donde quiera que vaya. Gracias, sin vosotros no podría haber llegado hasta aquí.

A todos, de corazón, muchísimas gracias.

Abstract

Most people say that it is the intellect which makes a great scientist. They are wrong: It is character.

Albert Einstein.

Computer vision is one of the fastest growing fields at present which, along with other technologies such as Biometrics or Big Data, has become the focus of interest of many research projects and it is considered one of the technologies of the future. This broad field includes a plethora of digital image processing and analysis tasks. To guarantee the success of image analysis and other high level processing tasks as 3D imaging or pattern recognition, it is critical to improve the quality of the raw images acquired.

Nowadays all images are affected by different factors that hinder the achievement of optimal image quality, making digital image processing a fundamental step prior to the application of any other practical application. The most common of these factors are noise and poor acquisition conditions: noise artifacts hamper proper image interpretation of the image; and acquisition in poor lighting or exposure conditions, such as dynamic scenes, causes loss of image information that can be key for certain processing tasks. Image (pre-)processing steps known as smoothing and sharpening are commonly applied to overcome these inconveniences: Smoothing is aimed at reducing noise and sharpening at improving or recovering imprecise or damaged information of image details and edges with insufficient sharpness or blurred content that prevents optimal image (post-)processing.

There are many methods for smoothing the noise in an image, however in many cases the filtering process causes blurring at the edges and details of the image. Besides, there are also many sharpening techniques, which try to combat the loss of information due to blurring of image texture and need to contemplate the existence of noise in the image they process. When dealing with a noisy image, any sharpening technique may amplify the noise.

Although the intuitive idea to solve this last case would be the previous filtering and later sharpening, this approach has proved not to be optimal: the filtering could remove information that, in turn, may not be recoverable in the later sharpening step.

In the present PhD dissertation we propose a model based on graph theory for color image processing from a vector approach. In this model, a graph is built for each pixel in such a way that its features allow to characterize and classify the pixel. As we will show, the model we proposed is robust and versatile: potentially able to adapt to a variety of applications. In particular, we apply the model to create new solutions for the two fundamentals problems in image processing: smoothing and sharpening.

To approach high performance image smoothing we use the proposed model to determine if a pixel belongs to a flat region or not, taking into account the need to achieve a high-precision classification even in the presence of noise. Thus, we build an adaptive soft-switching filter by employing the pixel classification to combine the outputs from a filter with high smoothing capability and a softer one to smooth edge/detail regions.

Further, another application of our model allows to use pixels characterization to successfully perform a simultaneous smoothing and sharpening of color images. In this way, we address one of the classical challenges within the image processing field.

We compare all the image processing techniques proposed with other state-of-the-art methods to show that they are competitive both from an objective (numerical) and visual evaluation point of view.

Resumen

La visión artificial es uno de los campos en mayor crecimiento en la actualidad que, junto con otras tecnlogías como la Biometría o el Big Data, se ha convertido en el foco de interés de numerosas investigaciones y es considerada como una de las tecnologías del futuro. Este amplio campo abarca diversos métodos entre los que se encuentra el procesamiento y análisis de imagenes digitales. El éxito del análisis de imágenes y otras tareas de procesamiento de alto nivel, como pueden ser el reconocimiento de patrones o la visión 3D, dependerá en gran medida de la buena calidad de las imágenes de partida.

Hoy en día existen multitud de factores que dañan las imagenes dificultando la obtención de imágenes de calidad óptima, esto ha convertido el (pre-) procesamiento digital de imágenes en un paso fundamental previo a la aplicación de cualquier otra tarea de procesado. Los factores más comunes son el ruido y las malas condiciones de adquisición: los artefactos provocados por el ruido dificultan la interpretación adecuada de la imagen y la adquisición en condiciones de iluminación o exposición deficientes, como escenas dinámicas, causan pérdida de información de la imagen que puede ser clave para ciertas tareas de procesamiento. Los pasos de (pre-)procesamiento de imágenes conocidos como suavizado y realce se aplican comúnmente para solventar estos problemas: El suavizado tiene por objeto reducir el ruido mientras que el realce se centra en mejorar o recuperar la información imprecisa o dañada. Con estos métodos conseguimos reparar información de los detalles y bordes de la imagen con una nitidez insuficiente o un contenido borroso que impide el (post-)procesamiento óptimo de la imagen.

Existen numerosos métodos que suavizan el ruido de una imagen, sin embargo, en muchos casos el proceso de filtrado provoca emborronamiento en los bordes y detalles de la imagen. De igual manera podemos encontrar una enorme cantidad de técnicas de realce que intentan combatir las perdidas de información, sin embargo, estas técnicas no contemplan la existencia de ruido en la imagen que procesan: ante una image ruidosa, cualquier técnica de realce provocará también un aumento del ruido. Aunque la idea intuitiva para solucionar este último caso sería el previo filtrado y posterior realce, este enfoque ha demostrado no ser óptimo: el filtrado podría eliminar información que, a su vez, podría no ser recuperable en el siguiente paso de realce.

En la presente tesis doctoral se propone un modelo basado en teoría de grafos para el procesamiento de imágenes en color. En este modelo, se construye un grafo para cada píxel de tal manera que sus propiedades permiten caracterizar y clasificar dicho píxel. Como veremos, el modelo propuesto es robusto y capaz de adaptarse a una gran variedad de aplicaciones. En partic-

ular, aplicamos el modelo para crear nuevas soluciones a los dos problemas fundamentales del procesamiento de imágenes: suavizado y realce.

Se ha estudiado el modelo en profundidad en función del umbral, parámetro clave que asegura la correcta clasificación de los píxeles de la imagen. Además, también se han estudiado las posibles características y posibilidades del modelo que nos han permitido sacarle el máximo partido en cada una de las posibles aplicaciones.

Basado en este modelo se ha diseñado un filtro adaptativo capaz de eliminar ruido gaussiano de una imagen sin difuminar los bordes ni perder información de los detalles. Además, también ha permitido desarrollar un método capaz de realzar los bordes y detalles de una imagen al mismo tiempo que se suaviza el ruido presente en la misma. Esta aplicación simultánea consigue combinar dos operaciones opuestas por definición y superar así los inconvenientes presentados por el enfoque en dos etapas.

Resum

La visió artificial és un dels camps en major creixement en l'actualitat que, junt amb altres tecnlogies com la Biometria o el Big Data, s'ha convertit en el focus d'interés de nombroses investigacions i és considerada com una de les tecnologies del futur. Aquest ampli camp comprén diversos mètodes entre els quals es troba el processament digital d'imatges i anàlisis d'imatges digitals. L'èxit de l'anàlisis d'imatges i altres tasques de processament d'alt nivell, com poden ser el reconeixement de patrons o la visió 3D, dependrà en gran manera de la bona qualitat de les imatges de partida,

Avui dia existeixen multitud de factors que danyen les imatges dificultant l'obtenció d'imatges de qualitat òptima, açò ha convertit el (pre-) processament digital d'imatges en un pas fonamental previ a la l'aplicació de qualsevol altra tasca de processament. Els factors més comuns són el soroll i les males condicions d'adquisició: els artefactes provocats pel soroll dificulten la interpretació adequada de la imatge i l'adquisició en condicions d'il·luminació o exposició deficients, com a escenes dinàmiques, causen pèrdua d'informació de la imatge que pot ser clau per a certes tasques de processament. Els passos de (pre-) processament d'imatges coneguts com suavitzat i realç s'apliquen comunament per a resoldre aquests problemes: El suavitzat té com a objecte reduir el soroll mentres que el realç se centra a millorar o recuperar la informació imprecisa o danyada. Amb aquests mètodes aconseguim reparar informació dels detalls i bords de la imatge amb una nitidesa insuficient o un contingut borrós que impedeix el (post-)processament òptim de la imatge.

Existeixen nombrosos mètodes que suavitzen el soroll d'una imatge, no obstant això, en molts casos el procés de filtrat provoca emborronamiento en els bords i detalls de la imatge. De la mateixa manera podem trobar una enorme quantitat de tècniques de realç que intenten combatre les pèrdues d'informació, no obstant això, aquestes tècniques no contemplen l'existència de soroll en la imatge que processen: davant d'una image sorollosa, qualsevol tècnica de realç provocarà també un augment del soroll. Encara que la idea intuïtiva per a solucionar aquest últim cas seria el previ filtrat i posterior realç, aquest enfocament ha demostrat no ser òptim: el filtrat podria eliminar informació que, al seu torn, podria no ser recuperable en el següent pas de realç.

En la present Tesi doctoral es proposa un model basat en teoria de grafs per al processament d'imatges en color. En aquest model, es construïx un graf per a cada píxel de tal manera que les seues propietats permeten caracteritzar i classificar el píxel en qüestió. Com veurem, el model proposat és robust i capaç d'adaptar-se a una gran varietat d'aplicacions. En particular,

apliquem el model per a crear noves solucions als dos problemes fonamentals del processament d'imatges: suavitzat i realç.

S'ha estudiat el model en profunditat en funció del llindar, paràmetre clau que assegura la correcta classificació dels píxels de la imatge. A més, també s'han estudiat les possibles característiques i possibilitats del model que ens han permés traure-li el màxim partit en cadascuna de les possibles aplicacions.

Basat en aquest model s'ha dissenyat un filtre adaptatiu capaç d'eliminar soroll gaussià d'una imatge sense difuminar els bords ni perdre informació dels detalls. A més, també ha permés desenvolupar un mètode capaç de realçar els bords i detalls d'una imatge al mateix temps que se suavitza el soroll present en la mateixa. Aquesta aplicació simultània aconseguix combinar dues operacions oposades per definició i superar així els inconvenients presentats per l'enfocament en dues etapes.

Presentation

Lo bueno de la ciencia es que es cierta
independientemente de si crees o no en ella.

Neil deGrasse Tyson.

Image filtering is probably the most common image processing task. Filtering an image means, in general, to transform that image into a more appropriate one for a certain purpose. Image filtering step known as smoothing is commonly applied to reduce the noise that may be present in an image and that may alter the information it contains. The noise that contaminates an image can be a serious inconvenience, both visually and for other imaging tasks such as image analysis or pattern recognition. As a consequence, image smoothing becomes an essential step in any computer vision system.

Along with smoothing, sharpening is another of the most common tasks within image processing. Sharpening an image consists of recovering imprecise or damaged information of details or edges with insufficient sharpness, blurred edges or dark images that hinder a good interpretation of the image. As denoising, this processing may preceed image processing tasks of many kinds: from the visual improvement of the image to the application of a subsequent treatment. In addition, each different objective can have different degrees of needed or optimal sharpness: If we seek a visual improvement we will try to recover the edges and details without exaggerating excessively; On the other hand if we need to detect objects or recognize patterns, we will need an approach much more aggressive that could even seem to worsen the visual quality of the image.

First image processing solutions were developed for gray-scale, one-channel, images. In the last years, the interest in using multichannel signals, and in particular color images, has impressively grown in a variety of applications.

The earliest solutions to process color images were component-wise approaches that used some gray-scale image filter in each color channel. In

this way, each channel was processed independently from the other channels. However, it is known that the existing correlation among the image channels should be taken into account. Otherwise, many color artifacts and other undesired effects may appear in the output images. This implied the need of specific color image filtering solutions.

One of the most studied approaches for color image processing is the vector approach. According to this approach, each image pixel is treated as a vector comprised of the color components and the image is treated as a vector field. Therefore, all image channels are jointly processed and the correlation among the image channels is necessarily taken into account.

This PhD aims at developing a novel tool that allows to process color images, from a vector point of view, for different objectives. We develop a graph-based tool able to understand and describe the image in a local and practical way. As we will see in the following chapters, it has applications to edge detection, image denoising, and image sharpening.

This dissertation is divided into three parts where each part consists of several chapters. Please note that each chapter is followed by the bibliographic references used in it.

Part I includes preliminaries concerning the area of research of this dissertation. This part is divided into 3 chapters, Chapter 1 presents the state-of-the-art of separate smoothing and sharpening as well as techniques approaching both smoothing and sharpening. The reader can find here the context of the problem in question and the work done in this field to date from different points of view. We would like to point out that sharpening is reviewed within the broader concept of image enhancement, where along with shaperning techniques we can find other image processing tasks pursuing image quality improvement. Notice that this review has been published by the author in "Pérez-Benito, Morillas, S, C., Jordán, C., Conejero, J. A.,. (2017). Smoothing vs. sharpening of colour images: Together or separated. *Applied Mathematics and Nonlinear Sciences*, 2(1), 299-316". Chapter 2 describes how the quality of both denoising and sharpening can be objectively measured. Finally, Chapter 3 introduces some basic concepts about graph theory intended to illustrate the reader who is unfamiliar with this field.

Part II and Part III presents the scientific contributions made in this PhD thesis that have been already published in specialized journals. Part II includes only one chapter and could be considered the heart of this dissertation as it presents the local graph model for color images used in the following. This model allows us to understand and characterize a color image for different purposes. Two applications of this model are those presented in the two chapters that compose Part III. In Chapter 5 we present the application of the graph-based model for smoothing color images. The model is used to define an adaptive filter able to remove Gaussian noise from color images. Finally, Chapter 6 contains the most novel application of this model, its use for simultaneous smoothing and sharpening of color images. The technique

introduced in this chapter removes noise and sharpens edges and details of the image in a simultaneous way which is both efficient and effective.

Finally, Part IV presents the overall conclusions and some potential future research lines.

With the exception of chapters 2, 3 and 7, the rest of the chapters of this thesis constitute a set of articles published in international journals or conferences. The content of the chapter 1 is almost entirely the published article. Chapter 4 is an extended version of the corresponding journal paper to improve the model description and comprehension. Finally, chapters 5 and 6, are the full published journal articles. Notice that due to the self-contained nature of the papers, probably some contents may be repeated along the document. However, in spite of this, we have preferred to include the original content of each published paper for the best understanding of the reader.

Table of Contents

Part I

Preliminaries

1 State-of-the-art of Smoothing and Sharpening

Pérez-Benito,C., Morillas, S, C., Jordán, C., Conejero, J. A., (2017). Smoothing vs. sharpening of colour images: Together or separated. *Applied Mathematics and Nonlinear Sciences*, 2(1), 299-316.

Abstract

It is still a challenge to improve the efficiency and effectiveness of image denoising and enhancement methods. There exists denoising and enhancement methods that are able to improve visual quality of images. This is usually obtained by removing noise while sharpening details and improving edges contrast. Smoothing refers to the case of denoising when noise follows a Gaussian distribution. Both operations, smoothing noise and sharpening, have an opposite nature. Therefore, there are few approaches that simultaneously respond to both goals. We will review these methods and we will also provide a detailed study of the state-of-the-art methods that attack both problems in colour images, separately.

1.1 Introduction

Several factors impact on color images and they do not only affect visual perception of the image. They also hinder the identification and distinction of image features that are relevant for different applications such as segmentation or pattern recognition. Noise is one of the most common of these factors and it can significantly affect visual quality of images, as well as the performance of most image processing tasks. It is the result of errors in the image acquisition process.

In several cases, images are taken under not suitable conditions: low light, too much clarity or poor weather conditions. A deficient quality equipment can hamper image acquisition because of transmissions errors, problems with networked cables, signal disturbances, troubles with sensors, etc. Therefore, pixel intensity values do not reflect true colors of the real scene we are shooting. For these reasons, lots of methods have been developed in order to recover

lost image information and to enhance image details. Color image *smoothing* is part of pre-processing techniques intended for removing possible image perturbations without losing image information.

Analogously, *sharpening* is a pre-processing technique that plays an important role for feature extraction in image processing. But even in this last case, smoothing will be needed in order to obtain a robust solution. This has motivated the study and development of methods that were able to cope with both operations.

The initial approach is usually to consider it as a two-steps process: first smoothing and later sharpening, or the other way around. However, this approach usually leads to many problems. On the one hand, if we first apply a smoothing technique, then we could be losing information that cannot be recovered in the succeeding sharpening step. On the other hand, if we first apply a sharpening method over a noisy image, we will amplify the noise present in it. The ideal way to address this problem is to consider a method that was able to sharp image details and edges while removing noise. Nevertheless, this is not a simple task given the opposite nature of these two operations.

Many methods for both sharpening and smoothing have been proposed in the literature, but if we restrict ourselves to methods that consider both of them simultaneously, the state-of-the-art is not so extensive. In this work we will also survey several methods of two-steps approaches in order to intensify the features of an image and to reduce the existing noise of the image. We will also review techniques that address both goals simultaneously.

In this way, the paper is organized as follows: Section 1.2 presents a brief review about image smoothing. In Section 1.3 we revisit some well-known techniques within enhancement and sharpening field. In Section 1.4.1 we introduce two-steps methods for smoothing and later sharpening and, alternatively, for sharpening and later smoothing. A comparison of both approaches will be shown. This will motivate the need of techniques that simultaneously address both processes, that will be exposed in Section 1.4.2. Finally, in Section 1.4.2 we compare the results given by the aforementioned methods.

1.2 Smoothing

Image smoothing techniques have the goal of preserving image quality. In other words, to remove noise without losing the principal features of the image. However, there are several types of noise. The main three types are: impulsive, additive, and multiplicative. *Impulsive noise* is usually characterized by some portion of image pixels that are corrupted, leaving the others unchanged. *Additive noise* appears when the values of the original image have been modified by adding random values which follow a certain probability distribution. Finally, *multiplicative noise* is more difficult to be removed from

images than additive noise, because in this case intensities vary along with signal intensity (e.g., speckle noise).

There are different sources of noise and plenty of denoising methods for each kind of noise. The most common one is probably the so-called *thermal noise*. This impulsive noise is due to CCD sensor malfunction in the image acquisition process.

Another interesting case is *Gaussian noise*, in which each pixel of the image will be changed from its original value by some small amount that follows a Gaussian distribution. This kind of noise is modelled as an *additive white Gaussian noise*. So that, its presence can be simulated by adding random values from a zero-mean Gaussian distribution to the original pixel intensities in each image channel independently, where the standard deviation σ of the Gaussian distribution characterizes the noise intensity [52].

The elimination of this kind of noise is known as *smoothing*, and this will be the type of noise elimination considered in this work. There are plenty of nonlinear methods for smoothing. In the rest of the section, we will review some of them.

1.2.1 Arithmetic Mean Filter

First approaches for Gaussian noise smoothing were based on linear strategies. These methods, such as the *Arithmetic Mean Filter* (AMF), see for instance [52], are able to suppress noise, because they take advantage of its zero-mean property. However, they blur edges and texture significantly. This motivated the development of non-linear methods that try to alleviate these problems by, firstly, detecting image edges and details, and secondly, by smoothing edges less than other parts of the image.

1.2.2 Bilateral Filter (BF)

Within nonlinear methods, a wide class of them uses averaging to take advantage of the zero-mean property of the Gaussian noise. This class includes the well-known *Bilateral Filter* (BF) [67] and its variants [13]. BF is a non-linear method able to smooth an image while respecting strong edges. This can be done by processing each pixel as a weighted average of its neighbours, where the weights depend on the spatial and intensity distance of each pixel with respect to the others. Several variants of the BF have been developed, for instance, the integration of a BF with an edge detection algorithm proposed in [24], or an adaptation of the BF with fuzzy metrics, as it is proposed in [42].

Another non-linear method respectful with image structure is the *Smallest Univalue Segment Assimilating Nucleus* (SUSAN)[64]. Here, a feature extraction algorithm is used to reduce noise using only sections from the local image structure that have been selected as similar pixels. The original value of each pixel is estimated using a weighted mean of the closest neighbours to it.

1.2.3 Fuzzy Noise Reduction Filters

Given the difficulty of distinguishing between noisy pixels and those belonging to details or edges of the image, fuzzy sets, which are capable of dealing with uncertainty, are very appropriate for image filtering tasks. In fact, the ability to manage uncertainty that is inherently adaptive implies that fuzzy filtering is useful for the suppression of different types of noise, including Gaussian noise.

Over the last years a huge amount of fuzzy filters were developed for images corrupted with impulse noise. They use fuzzy adaptive approaches that outperform rank-order filter schemes (such as the median filter). Although these filters are especially developed for grey-scale images, they can be used to filter color images by applying them on each color component separately. However, this approach generally introduces many color artifacts mainly on edge and texture elements.

To overcome these problems several fuzzy filtering approaches for colour images were successfully introduced. The vector median operations are extended to fuzzy numbers in [1]. In [2] a fuzzy rule based system determines the filter output. The vector median and some fuzzy measures are used in [6, 7] for calculating the fuzzy coefficients to determine the output as a weighted average of the inputs. In [3, 4] fuzzy coefficients determine the filter output by selecting the most representative input vector or as the combination of the vectors inside the filter window. The result of the detection method, which is applied on each color component separately, is used to calculate the noise-free color component differences of each pixel. These differences are then used by the noise reduction method so that the color component differences are preserved.

One of the best-known nonlinear filters is the *Fuzzy Noise Reduction Method* (FNRM) [61]. The core idea behind this method is to denoise each pixel using pixels within its neighborhood but using two sub-filters. FNRM provides very successful results. However, its drawback is that it respects image edges but at the expense of removing less noise.

To overcome the shortcomings of this kind of filters, linear and non-linear methods are combined in order to exploit the benefits of each of them for denoising colour images respecting details. In [23] graph theory is used to propose *Soft-Switching Graph Denoising* method (SSGD) that combines AMF and FNRM where AMF is more relevant in homogeneous regions and FNRM is more suitable for processing details. This method has been computationally improved in [48].

The filters introduced in [25] give detection rules based on differences between the *peer group* of a pixel and the peer groups of pixels in its neighbourhood. In [41], an averaging operation of the *fuzzy peer group* of each pixel is used for processing, which is called *Fuzzy Peer Group Averaging* (FPGA). Other methods have been developed using fuzzy logic or soft-switching strategies, such as those in [53, 43]. Methods based on different optimizations of

weighted averaging are proposed in [35, 62]. Another important family of filters are the *partition based filters* [62, 36], that classify each pixel to be processed into several signal activity categories which, in turn, are associated to appropriate processing methods.

1.2.4 Annisotropic Filtering (PM)

The *Annisotropic Filtering* was introduced by Perona and Malik (PM) [47]. There, a nonlinear adaptive diffusion process, called *anisotropic diffusion*, is considered. It consists of adapting the diffusion coefficient with a double goal: to reduce the smoothing effects near the edges for preserving image details, while smoothing flatter areas. There are several methods inspired by the PM model such as the one in [70], where a model based on a directional Laplacian is shown.

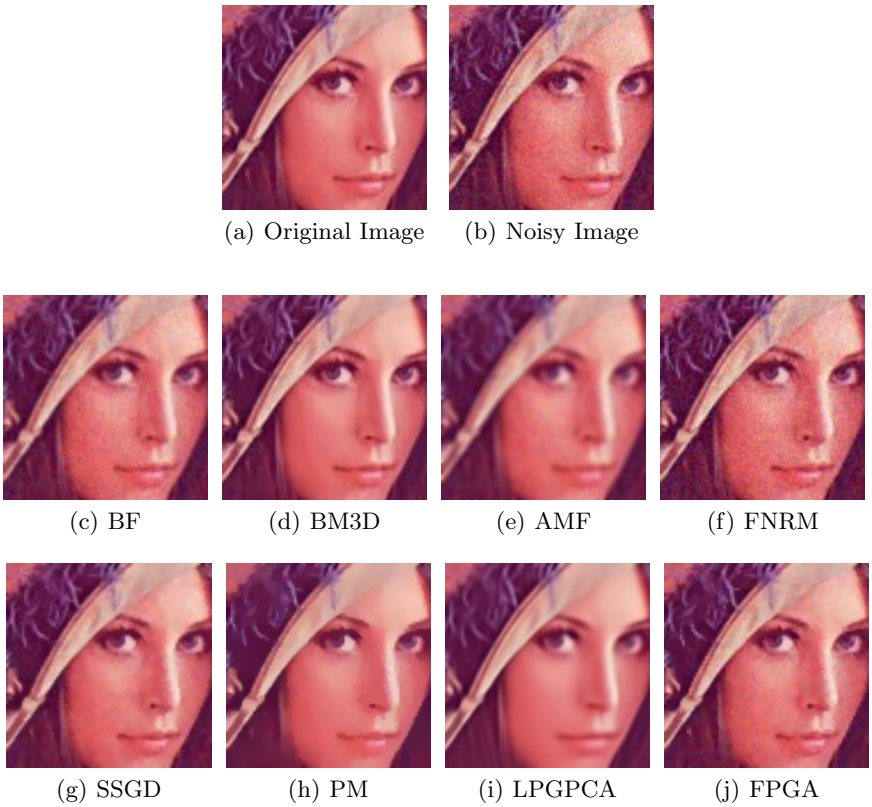
Guo et al [16] presented an adaptive PM filter able to segment the noisy image into two different regions, inner ones and borders. Then diffusion is applied by adapting it depending on the region we are considering.

1.2.5 Block-Matching and 3D Filtering (BM3D)

In [10], Dabov et al introduced collaborative filtering strategies which are probably the ones that provide the most impressive results within the block matching based denoising. The method presented there is called *Block-Matching and 3D Filtering* (BM3D). It is based on grouping, by *matching*, similar 2D fragments of the image into a 3D data matrix in order to use a different filter for each group. Details on matching algorithms can be shown in [20]. More precisely, filtering is achieved by the combination of collaborative non-local means and a transform-domain shrinkage. It can be summarized into 3 steps. First, a 3D transformation of each 3D group; secondly, a shrinkage of the spectrum of the transform; and lastly, an inverse 3D transformation. This technique is applied to each channel of the luminance-chrominance colour space, such as *YCbCr* or *YIQ*.

1.2.6 Principal Component Analysis (PCA)

Methods based on *Principal Component Analysis* (PCA) in the spatial domain have been applied in image denoising [45, 65]. The use of this technique allows us to reduce dimensionality, by transforming input data into the PCA domain in order to only preserve the most significant components. Muresan and Parks propose to divide every image into patches that are in turn divided into sub-windows, each of one has an associated vector built from pixels of the corresponding sub-window. Then, PCA is applied over these vectors for selecting a few principal components that are later used for smoothing [44]. This method has been refined in [75], where the training sample is selected by



(a) Original Image (b) Noisy Image

(c) BF (d) BM3D (e) AMF (f) FNRM

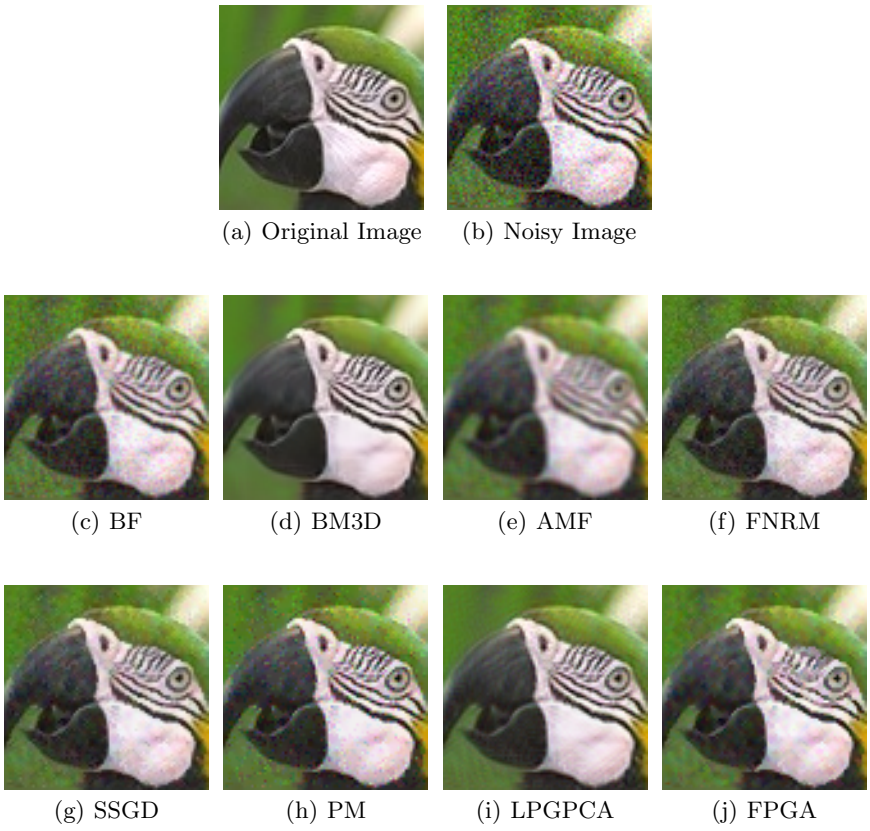
(g) SSGD (h) PM (i) LPGPCA (j) FPGA

Fig. 1.1: Results under different smoothing methods applied to Lenna image corrupted by a Gaussian noise with standard deviation $\sigma = 10$.

grouping pixels with similar local spatial structures using *Local Pixel Grouping* (LPG) before performing PCA. Additionally, the method in [44] has also inspired the development of a filter for images obtained from single-sensor digital cameras, named *Colour Filter Array* (CFA) [76].

1.2.7 Wavelet methods

Wavelet representation has become very popular within smoothing of images field [37]. It consists on decomposing an image signal into multiple scales, which represent its different frequency components. There are plenty of wavelets families, such as the ones of Haar, Daubechies, Coiflet, Symlet, Meyer, Morlet or the Mexican Hat, among others. In these methods, smoothing is applied in the image by using a threshold for removing detail coefficients. In this way, a hard scale-dependent threshold is proposed in



(a) Original Image (b) Noisy Image

(c) BF (d) BM3D (e) AMF (f) FNRM

(g) SSGD (h) PM (i) LPGPCA (j) FPGA

Fig. 1.2: Results under different smoothing methods applied to Parrots image corrupted by a Gaussian noise with standard deviation $\sigma = 20$.

[46]. Statistical modeling can be performed instead of thresholding to operate over wavelet coefficients to suppress noise [39, 51]. Wavelet transformation also works for data regularization as it is proposed in [17].

1.2.8 Results

In Figures 1.1 and 1.2 we can see the performance of some of the smoothing filters reported in this section. They have been applied to classical *Lenna* and *Parrots* images corrupted by some additive white Gaussian noise. BM3D method offers impressive results in comparison to the others, as we can see with Lenna images in Figure 1.1. PM smoothing method also presents good results, since it smooths well the noise without losing details and edge infor-

mation. However, if the level of noise is high, PM can produce some artifacts in the image, as we can see in the PM filtered image of Parrots in Figure 1.2.

1.3 Sharpening

Image enhancement process consists of a collection of techniques whose purpose is to improve image visual appearance and to highlight or recover certain details of the image for conducting an appropriate analysis by a human or a machine.

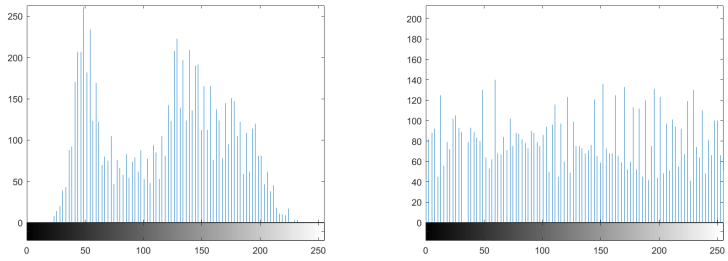
During the acquisition process, several factors can influence on the quality of the image such as illumination conditions, ambient pressure or temperature fluctuations. In order to enhance the image, we try to convert it for getting details that are obscured, or to sharpen certain features of interest. There is a large number of applications of these techniques that include medical image analysis, remote sensing, high definition television, microscopic imaging, etc. The existence of such a variety implies that there will also be very different goals within image enhancement, according to each particular application. In some cases, the purpose is to enhance the contrast, in others, to emphasize details and/or borders of the image. We will refer to this last process as *sharpening*, although the difference is not always clear. The choice of the most suitable techniques for each purpose will be a function of the specific task to be conducted, the image content, the observer characteristics, and the viewing conditions.

In this section we present a brief overview about the principal sharpening techniques. They can be classified into two different groups depending on the image domain: *spatial based* and *frequency based* techniques. In the first case, we directly operate over the pixel, while in the second we do it over the transform (Fourier or wavelet) coefficients of the image. Here, the effect of the transformation can only be noticed once we recover the image by the inverse transform.

1.3.1 Spatial domain techniques

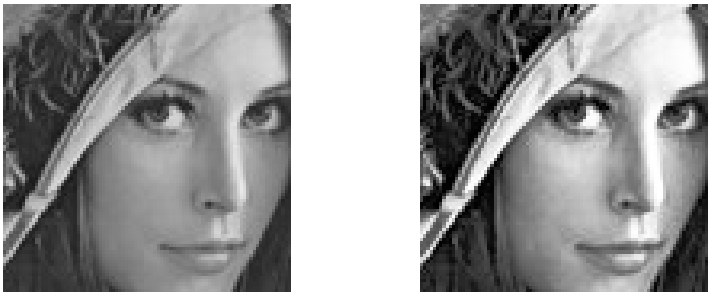
Spatial domain techniques for sharpening an image are based on manipulations of pixel values. One of the ways to improve it is by augmenting the contrast among different parts of the image.

There are several methods for image sharpening in the spatial domain. One of the most well-known is *Histogram Equalization* (HE). It is based on an adjustment of the contrast by using the histogram of the input image. It is manipulated in order to separate intensity levels of higher probability respect to their neighbour levels. In Figure 1.3 we can see the initial histogram for a gray-scale image of Lenna and the one obtained after having applied HE over the image. In Figure 1.4, we can see the input and output images



(a) Histogram (b) Equalized histogram

Fig. 1.3: Histogram of a gray-scale image of Lenna and the histogram of the resulting image after having applied HE.



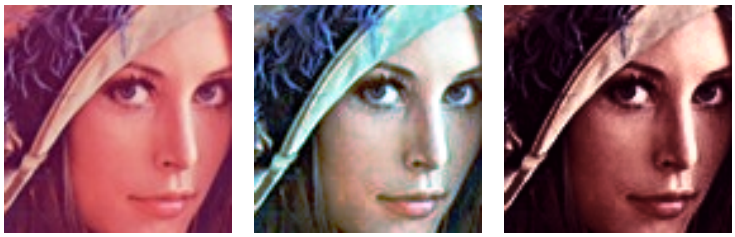
(a) Original Image (b) HE

Fig. 1.4: Comparation between a gray-scale image of Lenna and the resulting image after having applied HE.

corresponding to these histograms and how HE method works over a gray-scale image increasing the global contrast.

The application of this technique in colour images is not a simple task. Histogram equalization is a non-linear process and involves intensity values of the image and not the colour components. For these reasons, channel splitting and equalizing each channel separately is not the proper way for equalization of contrast. So, the first step is to convert the colour space of the image from RGB into other colour space which separates intensity values from colour components such as HSV, YCbCr or Lab, and apply to the equalization over the H, Y or L channel respectively. There are other approaches that generalize histogram equalization to colour spaces. Among the most well-known is 3D histogram [68].

There are lots of works seeking to improve HE techniques such as *Brightness Bi-Histogram Equalization* (BBHE) [27], where the image histogram is divided into two sub-histograms and they are independently equalized



(a) Original Image (b) HE in RGB space (c) HE in Lab space

Fig. 1.5: Comparation between HE applied over RGB channels separately and over L channel in Lab space.

(a) Original Image (b) BPFDHE (c) CLAHE

Fig. 1.6: Original image of Lenna and the output images obtained after applying BPDFHE and CLAHE methods.

later. *Dualistic Sub-Image Histogram Equalization* (DSIHE) [69] is similar to BBHE, but in this case the median value is used as a separation intensity level of reference in order to divide the histogram into two sub-histograms.

With the *Brightness Preserving Dynamic Histogram Equalization* (BPDHE) [19], we smooth the input histogram by using a Gaussian kernel and by avoiding a re-mapping of peaks unlike with the HE. This technique does not carry on the imprecision of gray-values while processing crisp histograms. In order to improve this technique, a fuzzy version of BPDHE is proposed to handle inaccuracy of gray levels, which is called the *Brightness Preserving Dynamic Fuzzy Histogram Equalization* (BPDFHE) [63]. A more rigorous study of methods that are based on histograms is presented in [72]. Besides, we can find there a method proposed with the goal of maximization of the expected contrast, called the *Optimal Contrast-Tone Mapping* (OCTM) method.

The aforementioned methods do not use spatial information neighbours of a given pixel. They are confined to use the intensity values of all pixels of the image. Local histogram equalization based methods were introduced in order to adapt these techniques by using local information. In this way, the *Contrast Limited Adaptative Histogram Equalization* method (CLAHE) is proposed in

order to enhance image contrast by applying CLHE on small data regions for adjusting the local contrast of an image [77]. The results locally obtained are joined together by bilinear interpolation to get the output image.

We can see in Figure 1.6 the result of applying BPDFHE and CLAHE methods to Lenna image. As it is indicated above, with this last method we improve the performance through a local approach that allows us to extract more information of the image structure.

Another well-known technique within spatial domain sharpening is the *Contrast Stretching* (CS), which is based on modifying the dynamic range, i.e., the range between the minimum and maximum intensity values of the image of the gray levels in the image being processed [73]. *Linear Contrast Stretch* (LCS) is the simplest contrast stretch algorithm that stretches pixel values of a low or high contrast image by extending the dynamic range across the whole image spectrum. One of the disadvantages of this method is that some details may be loss due to saturation and clipping.

In the *un-sharp masking* (UM) approach [55] an edge image is computed by using a fraction of the high-pass filtered version of the original image. This edge image is added to the original one to form the enhanced image. The main advantage of this method is their simplicity, however, this technique produce an large amplification of noise which often makes this method not useful in practice. Several approaches have been suggested for reducing the noise sensitivity of the linear UM technique. Many of these methods are based on the use of nonlinear operators in the correction path. A quadratic filter that can be approximately characterized as a local-mean-weighted adaptive high-pass filter is described in [56, 40]. An approach based on the order statistics Laplacian operator is described in [28]. An adaptative approach that prevents sharpening in flat regions is proposed in [54], that makes the method more robust in presence of noise.

1.3.2 Frequency domain techniques

Frequency domain techniques are based on the use of transformations like the Discrete Fourier (or Cosine) Transform or Wavelet Transforms. We remind that each one of these methods is not unique and, in fact, they compile a family of methods that are in essence the same, but each one with slight differences respect to the others. They work as follows: First, we apply one of these transformation methods, after we process the transform under one of these methods and, finally, the inverse transformation of the processed image gives us the result.

This approach has a wide advantage, the facility to distinguish between regions in an image. Higher frequencies are related to edges or details and lower ones correspond to smooth areas of the image. This easy separation allows to process the image appropriately depending on the goal. However, this also comprises that we are processing details of different regions at the same time in a indistinguishable way. This also happens with smooth regions.

Wavelet theory has become a potent image processing tool in the last years, this technique provide us image spatial and frequency information. An enhancement of the image can be obtained by adding high-pass or substracting low-pass filtered versions from the image [38, 37]. One of the early works on contrast sharpening in the wavelet domain is reported in [34], where a parametrised hyperbolic function is applied to the gradient of the wavelet coefficients. Since then, lots of works have been developed in the wavelet domain. For instance, Loza et al. proposed a non-linear enhancement method based on the local dispersion of the wavelet coefficients [33]. This algorithm enhances the contrast in images adaptively, based on local statistics of the wavelet coefficients of the image.

A contrast enhancement technique using a scaling of the internal noise of a dark image in the *Discrete Cosine Transform* (DCT) domain is developed in [22, 21]. It is based on a concept of physics called *Dynamic stochastic resonance* (DSR), that uses noise to improve the performance of a system [14]. The proposed algorithm enhances the contrast on colour images by applying the DSR method iteratively on the DCT coefficients of the image. DSR based methods in the wavelet domain have been also proposed in [9]. DSR based techniques are mainly centered in enhancement and not so much in sharpening edges or details of the image. They provide a better outcome when applied to low lighted images.

1.3.3 Results

In Figure 1.7 we can see the output of the UM and CLAHE methods for Parrot image. We also can see enlarged images of detail regions of them, where we can appreciate the sharpening effect over edges. This is an example of sharpening technique as opposite to the examples showed in Figure 1.6, which were methods more tied to contrast enhancement. They can be compared in Figure 1.7, where we can see an example of contrast enhancement, using CLAHE, versus sharpening using UM.

1.4 Simultaneous Smoothing and Sharpening

In this section we discuss about techniques that jointly considered smoothing and sharpening. The first idea we come up is to process the image in two different steps: first, by implementing one operation and then, over the processed image, carrying on the second process. Here, the order in which we carry the operations can greatly change the output. If we sharpen before smoothing, we can increase the relevance of image noise, which will complicate the smoothing task. If, by contrast, we smooth before sharpening, we may loss information in the smoothing process that the sharpen method could not recover. In general, the second approach usually provides better outcomes, however, it is still not an optimal solution. For that reason, techniques that

Fig. 1.7: First row, original Parrot image, filtered with UM and with CLAHE. Second row, a little detail region.

were able to combine simultaneously both smoothing and sharpness have been suggested in the last few years.

1.4.1 Two steps approach

Two-step methods for smoothing and sharpening consist on the sequential application of two methods, one of each type. In Figure 1.8 we can compare two-step methods based on BF for smoothing and CLAHE for sharpening. In the first case, we start with BF, and in the second one with CLAHE. This last method is applied to blurred Lenna and Parrot images in Figure 1.9.

We have seen the result of smoothing an image and subsequently apply a sharpening technique over the denoised image. In the first step, we lost a lot of information about the image, and then the second step was not good enough to recover the lost information. To overcome this drawback, we can first apply a sharpening, and in a second step we smooth the image. Results of both approaches can be seen in Figures 1.8 and 1.9.

Another example of a unified two-step method for both smoothing and sharpening over low light colour images is proposed in [29]. There two different steps are applied too. BM3D filter is combined with a structural filter for smoothing. Afterwards, a luminance adaptive contrast is applied in order to sharp the details of the smoothed image.

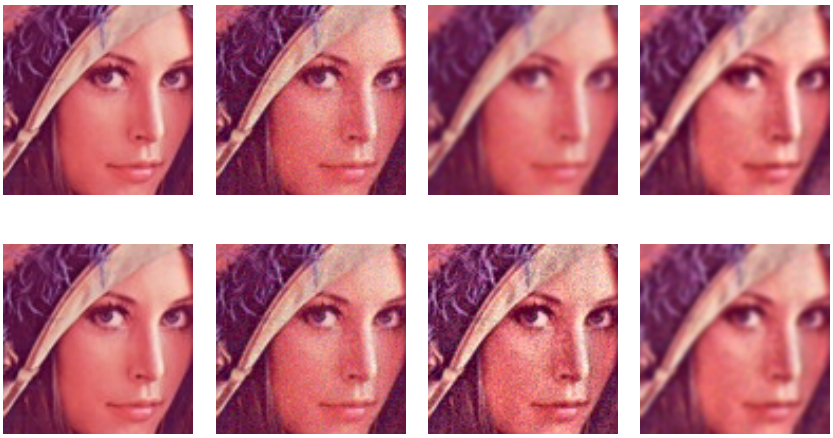


Fig. 1.8: First row, original image, original image blurred with Gaussian noise with $\sigma = 10$, filtered image with BF and finally ouput of BF and posterior CLAHE. Second row, original and noisy image, the enhanced image with CLAHE and finally output of CLAHE and posterior BF.

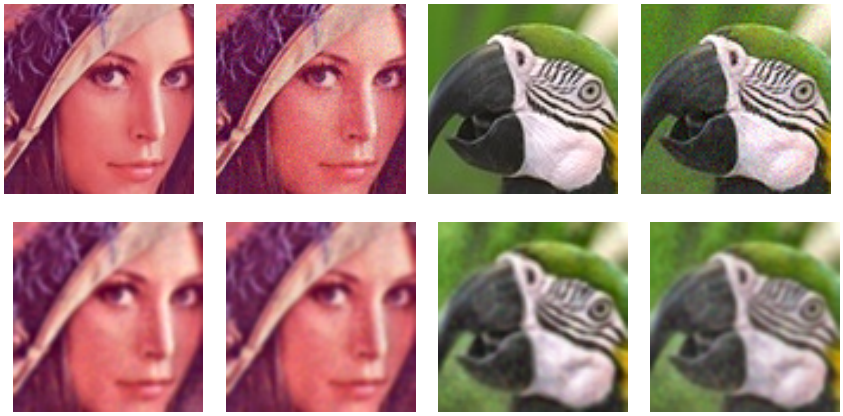


Fig. 1.9: First row, original images and original images blurred with Gaussian noise with $\sigma = 10$. Second row, the result of applying CLAHE and subsequently BF to both images and then the opposite approach, BF and subsequently CLAHE.

1.4.2 Simultaneous approach

Although smoothing and sharpening are apparently opposite operations, the necessity of using both techniques at the same time is ever increasing. Both of them have been extensively studied and the techniques developed for each process are very different. However, this does not happen if we talk about doing both operations at the same time. The state of the art in terms of methods that are able to sharp details while removing noise is still relatively reduced. In this section we present some of these techniques.

Two smooth and sharpening techniques, such as PM and CLAHE, have been combined simultaneously by means of a synchronization algorithm [12], where we can appreciate the improvement respect the corresponding two-step methods based on them. The method draw on the advantage of these original models and combine it for constructing a good tool for medical images, more concretely for magnetic resonance.

As we saw in Section 1.4.1, PM is based in a non-linear forward diffusion process geared by a diffusion variable that permits to control the smoothing effects over the image. In this way, it is tempting to use backward diffusion in order to obtain a sharpened image. However, backward diffusion is unstable and an ill-posed problem. Nevertheless, Gilboa et al. show that it is possible to combine forward and backward nonlinear diffusion processes for getting the *Forward-and-Backward* (FAB) diffusion process [15]. FAB is able to sharpen details while removing the noise. An adaptative control of the local degree of diffusion depending on the local gradient and inhomogeneity is considered to introduce the *Local Variance-Controlled Forward-and-Backward* (LVCFAB) [71].

Nevertheless, in the same way that te backward diffusion process, the FAB diffusion is unstable and ill-posed. In order to overcome this drawback Vadim and Yehoshua proposed the use of *Telegraph-Diffusion*(TeD) [57, 58], instead of the diffusion equation, giving rise to a stable smoothing and sharpening method, called (TeD-FAB).

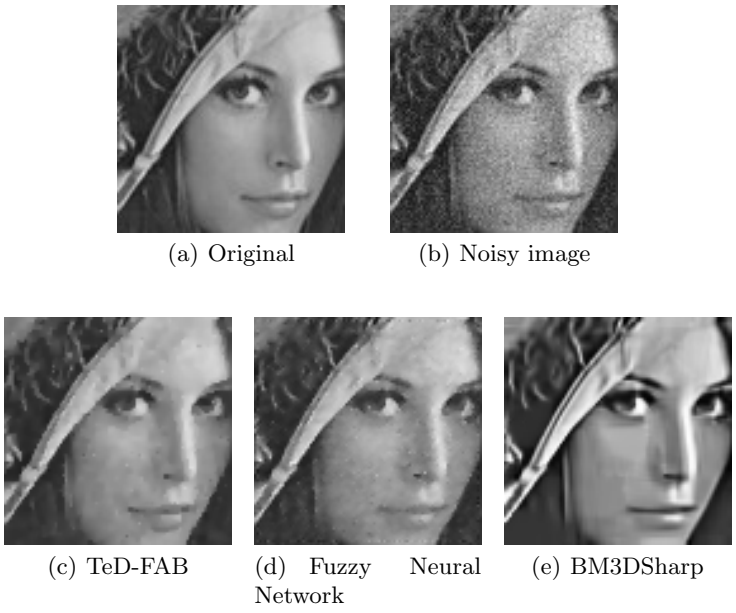
In [11], the authors proposed to combine BM3D with a transform-domain sharpening technique, applied to blocks, in order to sharpen while noise is being removed. We will refer to this method as (BM3DSharp).

We can also find fuzzy based methods with this double purpose. Russo proposed, in [59, 60], a fuzzy neural network technique that consists on a multiple-output processing system that adopts fuzzy networks in order to combine sharpening and smoothing. In particular, three fuzzy networks are combined; the first and third one smooth the image and the second one is responsible of the sharpening. The aforementioned methods can be compared in Figures 1.10, 1.11, and 1.12.

As we have mentioned UM has the disadvantages of increasing the noise in homogeneous regions and of not being able to sharpen all details due to its use of a fixed sharpening strength. With the purpose of overcome this drawback and to remove the noise at the same time that edges are sharpened, Kim et

al. have developed an adaptive unsharp mask, called *Optimal Unsharp Mask* (OUM) [26]. It is based on the classical approach of the UM but changing its parameter according to the local edge strength.

In [74], an *Adaptive Bilateral Filter* (ABF) based on the classical BF is presented. BF is reformulated by integrating a shift-variant technique to increase the slope of the edges and to smooth the noise. ABF presents a similar sharpening performance as OUM, but without producing the artefacts of OUM. Moreover, ABF achieves a better noise suppression than OUM. However, ABF significantly increases the computational complexity, that is proportional to the window size.



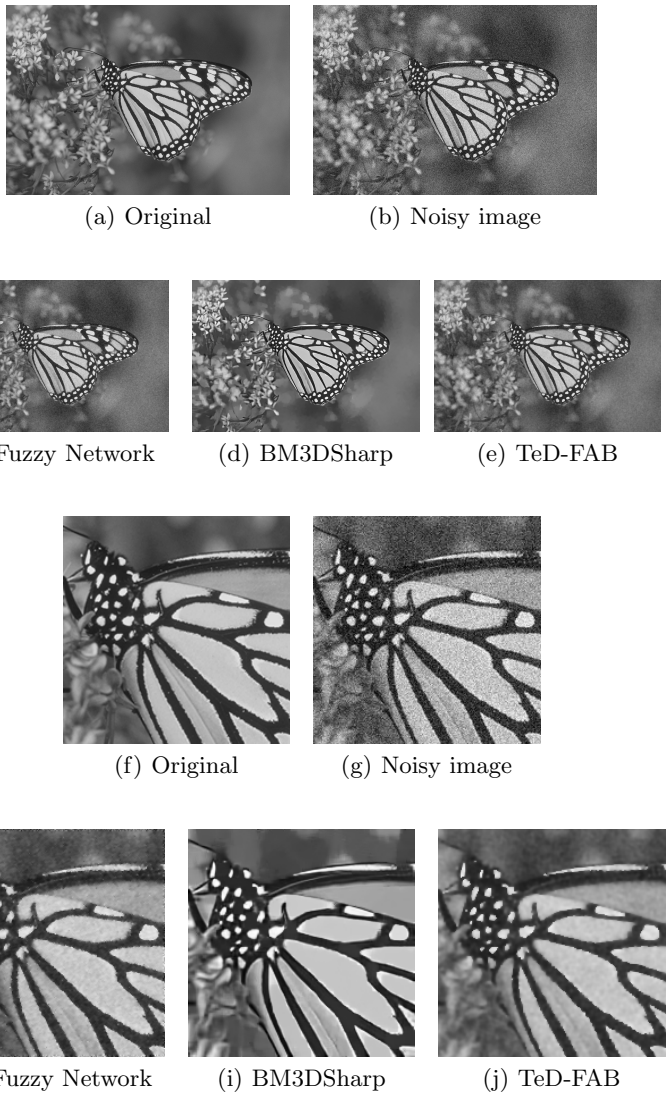
(a) Original (b) Noisy image

(c) TeD-FAB (d) Fuzzy Neural Network (e) BM3DSharp

Fig. 1.10: Denoising results for Lenna image corrupted by Gaussian noise with standard deviation $\sigma = 20$.

To overcome this problem, an *Adaptative Guided Image Filtering* (AGF), that combines a *guided* filter with the shift-variant technique, has been proposed in [49, 50].

In a few words, the guided filter is a linear translation-variant filter in which each pixel is replaced by a linear transform of a guidance image (input image or another one). Saini et al. proposed a modification of the ABF that firstly considers a segmentation of the image in clusters with similar structure [66]. This clustering is based on features that describe the local structure of



(a) Original (b) Noisy image

(c) Fuzzy Network (d) BM3DSharp (e) TeD-FAB

(f) Original (g) Noisy image

(h) Fuzzy Network (i) BM3DSharp (j) TeD-FAB

Fig. 1.11: Results of smoothing & sharpening with different methods an image corrupted by Gaussian noise with $\sigma = 30$.

Fig. 1.12: First row, Lenna image corrupted by Gaussian noise with standard deviations $\sigma = 10$, $\sigma = 20$ and $\sigma = 30$. Second row, the output of Fuzzy Network filter. Third row, output of BM3DShar and in the last one, the output of TeD-FAB.

the image. After a segmentation, each pixel is processed with a weighted mean that uses bilateral weights of the corresponding cluster.

Wavelet based methods have also been proposed for dealing with smoothing and sharpening simultaneously. In [32] the image on the HSV space is transformed into the wavelet domain by *Dual-Tree Complex Wavelet Transform* (DT-CWT), where the wavelets coefficients are adjusted in order to obtain a smooth and enhanced image. In this line, Li-na et al. applied wavelet methods colour images in the HSV space [30]. Their method uses properties of each canal to get the desired result. In this way, the saturation channel is smoothed according to a simple transformation by using the maximum and minimum values of the RGB space. The luminance channel is smooth by using a wavelet threshold and it is also enhanced by compression of the low frequencies of the image. Finally, the hue channel is kept invariable.

In [18], the authors apply smoothing and sharpening process on images in the YIQ space. This method depends on the surface texture of each pixel in order to smooth flat regions of the image while sharpening details. This is done by a combination of a Gaussian derivative filter that divides the Y image into flat and edge areas. The first ones are sharpened by using a Gaussian derivative operator and the second ones are smoothed using SUSAN method.

A combined method based on the graph Laplacian operator is performed in [31] where the output image is the solution of a minimization problem of a function with two different terms: one is a standard sparse coding formulation for image smoothing and the other one allows to sharpen the image thanks to the Laplacian operator.

Conclusions

In this paper, the main techniques for removing white Gaussian noise in colour images have been revisited. Also, we have reviewed the typical techniques for colour images smoothing and sharpening, both in spatial and in frequency domain.

Both operations have an opposite nature, the aim of smoothing an image is to remove the noise. However, the aim of sharpening is somehow the opposite, since it tries to emphasize details. These techniques are responsible for making more visible variations and details or edges of the images. We have seen that the application of both techniques in two steps, one after the other, produce wrong results because of loosing some relevant information or sharpening the noise.

The reduced number of approaches that simultaneously respond to both goals lies on the difficulty of combining these apparently contradictory process. We have reported the most remarkable of these methods.

References

[1] V. Chatzis, I. Pitas, Fuzzy scalar and vector median filters based on fuzzy distances, *IEEE Transactions on Image Processing* 8 5 (1999) 731-734.

[2] K. Arakawa, Median filter based on fuzzy rules and its application to image restoration, *Fuzzy Sets and Systems*, 77 1 (1996) 3-13.

[3] R. Lukac, K.N. Plataniotis, B. Smolka, A.N. Venetsanopoulos, cDNA Microarray Image Processing Using Fuzzy Vector Filtering Framework, *Fuzzy Sets and Systems: Special Issue on Fuzzy Sets and Systems in Bioinformatics*, 152 1 (2005) 17-35.

[4] R. Lukac, K.N. Plataniotis, B. Smolka, A.N. Venetsanopoulos, A Multichannel Order-Statistic technique for cDNA Microarray Image Processing, *IEEE Transactions on Nanobioscience* 3 4 (2004) 272-285.

[5] Y. Shen, K. Barner, *Fuzzy vector median-based surface smoothing*, IEEE Transactions on Visualization and Computer Graphics 10 3 (2004) 252-265.

[6] Y. Shen. K.E. Barner, Marginal fuzzy median and fuzzy vector median filtering of color images, *in Proc. 37th Annual Conf. Inf. Sciences & Systems* (2003).

[7] Y. Shen, K.E. Barner, Optimization of fuzzy vector median filters, *in Proc. 38th Annual Conf. Inf. Sciences & Systems* (2004).

[8] S. Hore, B. Qiu, and H.R. Wu, Improved vector filtering for color images using fuzzy noise detection, *Optical Engineering* , 42 6 (2003) 1656-1664.

[9] R. Chouhan, C. P. Kumar, R. Kumar, and R. K. Jha, (2012), Contrast enhancement of dark images using stochastic resonance in wavelet domain. *International Journal of Machine Learning and Computing*, 2(5):711-715. doi 10.7763/IJMLC.2012.V2.220

[10] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, (2007), Image denoising by sparse 3D transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16(8):2080–2095. doi 10.1109/TIP.2007.901238

[11] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, (2007), Joint image sharpening and denoising by 3D transform-domain collaborative filtering. In *Proceedings of the International Workshop on Spectral Methods for Multirate Signal Process, SMMSP 2007*, volume 2007.

[12] F. H. Di Jia, J. Yang, Y. Zhang, D. Zhao, and G. Yu, (2010), A synchronization algorithm of MRI denoising and contrast enhancement based on PM-CLAHE model. *JDCTA*, 4(6):144–149.

[13] M. Elad, (2002), On the origin of the bilateral filter and ways to improve it. *IEEE Transactions on Image Processing*, 11(10):1141–1151. doi 10.1109/TIP.2007.901238

[14] L. Gammaitoni, P. Hänggi, P. Jung, and F. Marchesoni, (1998), Stochastic resonance. *Reviews of Modern Physics*, 70:223–287. doi 10.1103/RevModPhys.70.223

[15] G. Gilboa, N. Sochen, and Y. Y. Zeevi, (2002), Forward-and-backward diffusion processes for adaptive image enhancement and denoising. *IEEE Transactions on Image Processing*, 11(7):689–703. doi 10.1109/TIP.2002.800883

[16] Z. Guo, J. Sun, D. Zhang, and B. Wu, (2012), Adaptive Perona–Malik model based on the variable exponent for image denoising. *IEEE Transactions on Image Processing*, 21(3):958–967. doi 10.1109/TIP.2011.2169272

[17] B.-b. Hao, M. Li, and X.-c. Feng, (2008), Wavelet iterative regularization for image restoration with varying scale parameter. *Signal Processing: Image Communication*, 23(6):433–441. doi 10.1016/j.image.2008.04.006

[18] T. Horiuchi, K. Watanabe, and S. Tominaga, (2007), Adaptive filtering for color image sharpening and denoising. In *14th International Conference on Image Analysis and Processing Workshops, ICIAPW 2007*, 196–201.

[19] H. Ibrahim and N. S. P. Kong, (2007), Brightness preserving dynamic histogram equalization for image contrast enhancement. *IEEE Transactions on Consumer Electronics*, 53(4):403-410. doi 10.1109/TCE.2007.4429280

[20] A. K. Jain, M. N. Murty, and P. J. Flynn, (1999), Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3):264–323. doi 10.1145/331499.331504

[21] R. K. Jha, R. Chouhan, and K. Aizawa, (2014), Dynamic stochastic resonance-based improved logo extraction in discrete cosine transform domain. *Computers & Electrical Engineering*, 40(6):1917–1929. doi j.compeleceng.2013.07.024

[22] R. K. Jha, R. Chouhan, P. K. Biswas, and K. Aizawa, (2012), Internal noise-induced contrast enhancement of dark images. In *19th IEEE International Conference on Image Processing (ICIP), 2012*, pages 973–976.

[23] C. Jordán, S. Morillas, and E. Sanabria-Codesal, (2012), Colour image smoothing through a soft-switching mechanism using a graph model. *IET Image Processing*, 6(9):1293–1298. doi 10.1049/IET-IPR.2011.0164

[24] W.-C. Kao and Y.-J. Chen, (2005), Multistage bilateral noise filtering and edge detection for color image enhancement. *IEEE Transactions on Consumer Electronics*, 51(4):1346–1351. doi 10.1109/TCE.2005.1561866

[25] C. Kenney, Y. Deng, B. Manjunath, and G. Hewer, (2001), Peer group image enhancement. *IEEE Transactions on Image Processing*, 10(2):326–334. doi 10.1109/83.902298

[26] S. H. Kim and J. P. Allebach, (2005), Optimal unsharp mask for image sharpening and noise removal. *Journal of Electronic Imaging*, 14(2):023005. doi 10.1117/1.1924510

[27] Y.-T. Kim, (1997), Contrast enhancement using brightness preserving bi-histogram equalization. *IEEE transactions on Consumer Electronics*, 43(1):1–8. doi 10.1109/30.580378

[28] Y. H. Lee and S. Y. Park, (1990), A study of convex/concave edges and edge-enhancing operators based on the Laplacian. *IEEE Transactions on Circuits and Systems*, 37(7):940–946. doi 10.1109/31.55069

[29] X. Li, (2007), On modeling interchannel dependency for color image denoising. *International Journal of Imaging Systems and Technology*, 17(3):163–173. doi 10.1002/ima.20112

[30] H. Li-na, G. Guo-hua, X. Jie, and X. Zheng-Long, (2009), Real-color image denoised and enhanced synchronously based on wavelet transform. In *Second International Conference onIntelligent Computation Technology and Automation ICICTA'09*, 1:658–661. doi 10.1109/AICI.2009.251

[31] X. Liu, G. Cheung, and X. Wu, (2015), Joint denoising and contrast enhancement of images using graph Laplacian operator. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2015*, 2274–2278. doi 10.1109/ICASSP.2015.7178376

[32] A. Łoza, M. Al-Mualla, P. Verkade, P. Hill, D. Bull, and A. Achim, (2014) Joint denoising and contrast enhancement for light microscopy image sequences. In *IEEE 11th International Symposium on Biomedical Imaging (ISBI), 2014*, 1083–1086. doi 10.1109/ISBI.2014.6868062

[33] A. Łoza, D. R. Bull, P. R. Hill, and A. M. Achim, (2013), Automatic contrast enhancement of low-light images based on local statistics of wavelet coefficients. *Digital Signal Processing*, 23(6):1856–1866. doi 10.1109/ICIP.2010.5651173

[34] J. Lu and D. Healy, (1994), Contrast enhancement via multiscale gradient transformation. In *IEEE International Conference Image Processing, ICIP-94*, 2:482–486. doi 10.1109/ICIP.1994.413617

[35] L. Lucchese and S. K. Mitra, (2004), A new class of chromatic filters for color image processing. theory and applications. *IEEE Transactions on Image Processing*, 13(4):534–548. doi 10.1109/TIP.2003.822609

[36] Z. Ma, H. R. Wu, and D. Feng, (2007), Fuzzy vector partition filtering technique for color image restoration. *Computer Vision and Image Understanding*, 107(1):26–37. doi 10.1016/j.cviu.2006.11.017

[37] S. Mallat, (1999), *A wavelet tour of signal processing*. Academic press, 1999. doi 10.1162/comj.2007.31.3.83

[38] S. G. Mallat, (1989), A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693. doi 10.1109/34.192463

[39] M. K. Mihcak, I. Kozintsev, K. Ramchandran, and P. Moulin, (1999), Low-complexity image denoising based on statistical modeling of wavelet coefficients. *IEEE Signal Processing Letters*, 6(12):300–303. doi 10.1109/97.803428

[40] S. K. Mitra, H. Li, I.-S. Lin, and T.-H. Yu, (1991), A new class of nonlinear filters for image enhancement. In *1991 International Conference on Acoustics, Speech, and Signal Processing, ICASSP-91*, 2525–2528. doi 10.1109/ICASSP.1991.150915

[41] S. Morillas, V. Gregori, and A. Hervás, (2009), Fuzzy peer groups for reducing mixed gaussian-impulse noise from color images. *IEEE Transactions on Image Processing*, 18(7):1452–1466. doi 10.1109/TIP.2009.2019305

[42] S. Morillas, V. Gregori, and A. Sapena, (2006), Fuzzy bilateral filtering for color images. In *International Conference Image Analysis and Recognition*, 138–145. doi 10.1007/11867586_13

[43] S. Morillas, S. Schulte, T. Mélange, E. E. Kerre, and V. Gregori, (2007), A soft-switching approach to improve visual quality of colour image smoothing filters. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, 254–261. doi 10.1007/978-3-540-74607-2_23

[44] D. D. Muresan and T. W. Parks, (2003), Adaptive principal components and image denoising. In *International Conference on Image Processing, 2003*, 1, 1–101. doi 10.1109/ICIP.2003.1246908

[45] E. Oja, (1992), Principal components, minor components, and linear neural networks. *Neural Networks*, 5(6):927–935. doi 10.1016/S0893-6080(05)80089-9

[46] Q. Pan, L. Zhang, G. Dai, and H. Zhang, (1999), Two denoising methods by wavelet transform. *IEEE Transactions on Signal Processing*, 47(12):3401–3406. doi 10.1109/78.806084

[47] P. Perona and J. Malik, (1990), Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629-639. doi 10.1109/34.56205

[48] C. Pérez-Benito, C. Jordán, S. Morillas, and J. Conejero, (2018), A model based on local graphs for colour images and its application for Gaussian noise smoothing. *Journal of Computational and Applied Mathematics*. doi 10.1016/j.cam.2017.05.013

[49] C. C. Pham, S. V. U. Ha, and J. W. Jeon, (2011), Adaptive guided image filtering for sharpness enhancement and noise reduction. In *Pacific-Rim Symposium on Image and Video Technology*, 323–334. doi 10.1109/78.806084

[50] C. C. Pham and J. W. Jeon, (2014), Efficient image sharpening and denoising using adaptive guided image filtering. *IET Image Processing*, 9(1):71–79. doi 10.1049/iet-ipr.2013.0563

[51] A. Pizurica, W. Philips, I. Lemahieu, and M. Acheroy, (2002), A joint inter-and intrascale statistical model for bayesian wavelet based image denoising. *IEEE Transactions on Image Processing*, 11(5):545–557. doi 10.1109/TIP.2002.1006401

[52] K. Plataniotis and A. N. Venetsanopoulos, (2013), *Color image processing and applications*. Springer Science & Business Media. doi 10.1088/0957-0233/12/2/703

[53] K. N. Plataniotis, D. Androutsos, and A. N. Venetsanopoulos, (1999), Adaptive fuzzy systems for multichannel signal processing. *Proceedings of the IEEE*, 87(9):1601–1622. doi 10.1109/5.784243

[54] A. Polesel, G. Ramponi, and V. J. Mathews, (2000), Image enhancement via adaptive unsharp masking. *IEEE Transactions on Image Processing*, 9(3):505–510. doi 10.1109/83.826787

[55] W. K. Pratt, (2001), Digital image processing: PIKS Inside, John Wiley & sons. *Inc.* doi 10.1002/0471221325

[56] G. Ramponi, N. Strobel, S. K. Mitra, and T.-H. Yu, (1996), Nonlinear unsharp masking methods for image contrast enhancement. *Journal of Electronic Imaging*, 5(3):353–366. doi 10.1117/12.242618

[57] V. Ratner and Y. Y. Zeevi, (2011), Denoising-enhancing images on elastic manifolds. *IEEE Transactions on Image Processing*, 20(8):2099–2109. doi 10.1117/12.242618

[58] V. Ratner and Y. Y. Zeevi, (2013), Stable denoising-enhancement of images by telegraph-diffusion operators. In *20th IEEE International Conference on Image Processing (ICIP), 2013*, 1252–1256. doi 10.1109/ICIP.2013.6738258

[59] F. Russo, (2000), Noise removal from image data using recursive neuro-fuzzy filters. *IEEE Transactions on Instrumentation and Measurement*, 49(2):307–314. doi 10.1109/IMTC.1999.776134

[60] F. Russo, (2002), An image enhancement technique combining sharpening and noise reduction. *IEEE Transactions on Instrumentation and Measurement*, 51(4):824–828. doi 10.1109/TIM.2002.803394

[61] S. Schulte, V. De Witte, and E. E. Kerre, (2007), A fuzzy noise reduction method for color images. *IEEE Transactions on Image Processing*, 16(5):1425–1436. doi 10.1109/TIP.2007.891807

[62] M. Shao and K. E. Barner, (2006), Optimization of partition-based weighted sum filters and their application to image denoising. *IEEE Transactions on Image Processing*, 15(7):1900–1915. doi 10.1109/TIP.2006.873436

[63] D. Sheet, H. Garud, A. Suveer, M. Mahadevappa, and J. Chatterjee, (2010), Brightness preserving dynamic fuzzy histogram equalization.

IEEE Transactions on Consumer Electronics, 56(4), 2475–2480. doi 10.1109/TCE.2010.5681130

[64] S. M. Smith and J. M. Brady, (1997), SUSAN – A new approach to low level image processing. *International Journal of Computer Vision*, 23(1):45–78. doi 10.1023/A:1007963824710

[65] A. R. Teixeira, A. M. Tomé, K. Stadlthanner, and E. W. Lang, (2008), KPCA denoising and the pre-image problem revisited. *Digital Signal Processing*, 18(4):568–580. doi 10.1016/j.dsp.2007.08.001

[66] K. K. V. Toh and N. A. M. Isa, (2011), Locally adaptive bilateral clustering for image deblurring and sharpness enhancement. *IEEE Transactions on Consumer Electronics*, 57(3). doi 10.1109/TCE.2011.6018878

[67] C. Tomasi and R. Manduchi, (1998), Bilateral filtering for gray and color images. In *Sixth International Conference on Computer Vision, 1998*, 839–846. doi 10.1109/ICCV.1998.710815

[68] P. Trahanias and A. Venetsanopoulos, (1992), Color image enhancement through 3D histogram equalization. In *Pattern Recognition, 1992. Vol. III. 11th IAPR International Conference on Image, Speech and Signal Analysis*, 545–548. doi 10.1109/ICPR.1992.202045

[69] Y. Wang, Q. Chen, and B. Zhang, (1999), Image enhancement based on equal area dualistic sub-image histogram equalization method. *IEEE Transactions on Consumer Electronics*, 45(1):68–75. doi 10.1109/30.754419

[70] Y. Wang, J. Guo, W. Chen, and W. Zhang, (2013), Image denoising using modified Perona–Malik model based on directional Laplacian. *Signal Processing*, 93(9):2548–2558. doi 10.1016/j.sigpro.2013.02.020

[71] Y. Wang, L. Zhang, and P. Li, (2007), Local variance-controlled forward-and-backward diffusion for image enhancement and noise reduction. *IEEE Transactions on Image Processing*, 16(7):1854–1864. doi 10.1109/TIP.2007.899002

[72] X. Wu, (2011), A linear programming approach for optimal contrast-tone mapping. *IEEE Transactions on Image Processing*, 20(5):1262–1272. doi 10.1109/TIP.2010.2092438

[73] M. F. Zakaria, H. Ibrahim, and S. A. Suandi, (2010), A review: Image compensation techniques. In *2nd International Conference on Computer Engineering and Technology (ICCET), 2010*, 7: V7-404–V7-408. doi 10.1109/ICCET.2010.5485499

[74] B. Zhang and J. P. Allebach, (2008), Adaptive bilateral filter for sharpness enhancement and noise removal. *IEEE Transactions on Image Processing*, 17(5):664–678. doi 10.1109/TIP.2008.919949

[75] L. Zhang, W. Dong, D. Zhang, and G. Shi, (2010), Two-stage image denoising by principal component analysis with local pixel grouping. *Pattern Recognition*, 43(4):1531–1549. doi 10.1109/TIP.2008.2011384

[76] L. Zhang, R. Lukac, X. Wu, and D. Zhang, (2009), PCA-based spatially adaptive denoising of CFA images for single-sensor digital cam-

eras. *IEEE Transactions on Image Processing*, 18(4):797–812. doi 10.1109/TIP.2008.2011384

[77] H. Zhu, F. H. Chan, and F. K. Lam, (1999), Image contrast enhancement by constrained local histogram equalization. *Computer Vision and Image Understanding*, 73(2):281–290. doi 10.1006/cviu.1998.0723

2 Assessment of color image denoising and sharpening methods

A key piece in the development of any image (pre-)processing technique is the ability to measure its performance. That is to say, to measure how much the image has been improved after removing noise, enhancing edges, increasing the contrast, and so on. In this chapter we are going to explain how to proceed when measuring the performance of a (pre-)processing method and we will review the main techniques to carry out this task and that have been used in this dissertation to evaluate denoising/smoothing and sharpening performance.

To start, we need to select a set of test images that we consider interesting. This selection will vary according to the objective we are looking for, if we need sharp borders, a more homogeneous image, certain fine details or some specific type of image such as medical image. In our case, we will use generic/domestic images of everyday things that present edge and detail areas and also homogeneous zones. Figure 2.1 shows several test images, some of them very well-known, that are used by the scientific community and that are also used in this dissertation.

A simple and intuitive way to evaluate the performance of a filter or the quality of an image is visual inspection by an expert observer. However, the visual perception can fluctuate between subjects which makes this kind of evaluation a weak tool. Although psychophysics evaluations could be used to solve this issue, this kind of evaluation takes a long time and it is difficult to replicate. For instance, a set of observers could be used so that results are statistically processed including inter-observer and intra-observer analysis. Besides, this requires a good sample selection and strict control of the experimental conditions, so that they are the same for all users: lighting, screen, distance, etc. Even controlling all the conditions, the results will depend on the knowledge of the users, their experience, seriousness, availability, repeatability of their answers, etc. As a consequence, numerical computational evaluations are preferred in general.

From now on, we are going to focus on the evaluation of denoising/smoothing performance given that the techniques used include those for assessing sharpening that we will highlight later.

Therefore, in order to evaluate smoothing, the test images are contaminated with some kind of simulated noise. There are several types of noise

(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 2.1: Some classical test images used for filter assessment: (a) Lenna (90×90), (b) Pills (50×50), (c) Peppers (50×50), (d) Statue (200×200), (e) Parrots (80×80), (f) Window (100×100, (e) Parrots1 (80×80), (e) Micro (51×51)

that can corrupt color images. Probably the most common is the noise associated to the camera sensor or thermal noise. This noise is modeled as additive *White Gaussian noise* having the following probability distribution in each color channel:

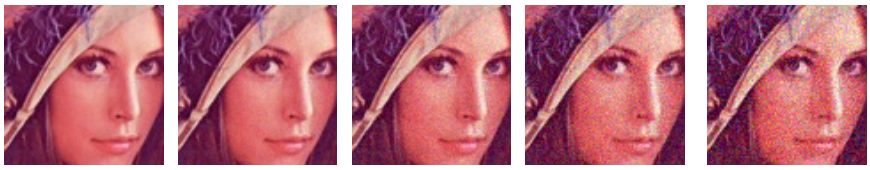
$$p(x) = \frac{1}{(2\pi\sigma)^{\frac{1}{2}}} e^{\frac{-x^2}{2\sigma^2}} \tag{2.1}$$

where σ denotes the standard deviation of the distribution. This noise is introduced independently in each color channel. However, it can be assumed that all three color channels have the same average noise magnitude with constant noise variance over the entire image plane. In Figure 2.2 we can see some examples of noisy images. The Lenna image of the Figure 2.1 (a) has been taken and Gaussian white noise with different standard deviations has been added, specifically $\sigma = 2.5$, $\sigma = 5$, $\sigma = 10$ and $\sigma = 20$.

Then, the corrupted image is filtered using the filtering/smoothing/denoising method to be assessed and the processed image is obtained.

Finally, the processed image is used to compute the image quality through an *Objective Image Quality Assessment* (IQA) method. Over the last few years, many methods of objective image quality assessment have been developed that we could classify within three types:

– Full-Reference Image Quality Assessment (FR-IQA)
– Reduced-Reference Image Quality Assessment (RR-IQA)
– Non-Reference Image Quality Assessment (NR-IQA)



(a) $\sigma = 2.5$ (b) $\sigma = 5$ (c) $\sigma = 10$ (d) $\sigma = 15$ (e) $\sigma = 20$

Fig. 2.2: Examples of images with Gaussian noise with different standard deviations from a reference image of the Figure 2.1(a)

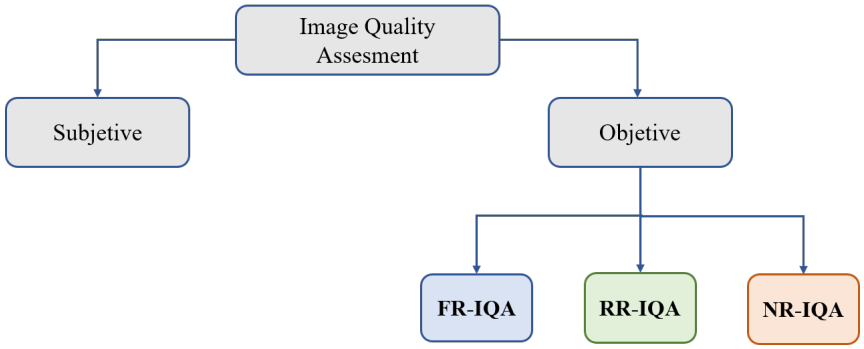


Fig. 2.3: Image Quality Assesment (IQA).

The full reference techniques (FR-IQA) refers to assessing the quality of the output image by comparing with the original, believed to be free-of-noise version of the same image or, in general, with an ideal of reference, which is a serious requirement. The filter performance is calculated by measuring the deviation of a filtered image from the reference image in terms of an objective score.

Reduced Reference Image Quality Assessment (RR-IQA) encompasses methods that use only partial information of the reference image rather than the full image to evaluate the quality of the filtered image. The reference limitation presented by the FR-IQA methods remains in the RR-IQA. Also, it is of paramount importance to carefully select the information used to assess to quality successfully for the particular application. However, despite all their limitations, RR-IQA techniques are widely used in satellite image quality assessment and remote sensing, among other problems.

Nevertheless, in most real cases, as it happens in the sharpening/enhancement field, the ideal of reference image is not available. In the case of sharpening, this image does not exist. This limitation led to the development of the third type of techniques: the Non-Reference Image Quality Assessment(NR-IQA). These methods determine the quality of an image through the extraction of a series of image features and statistics from the output image that

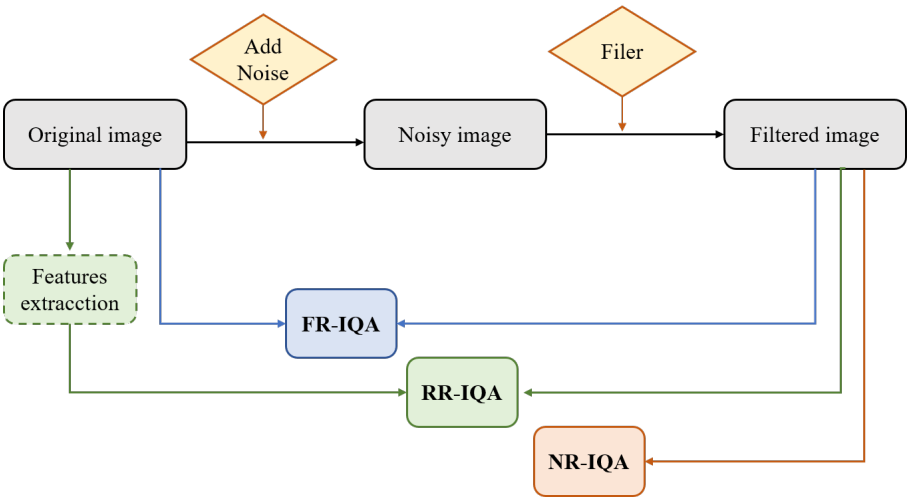


Fig. 2.4: Scheme of the filter evaluation.

are further processed in different ways to obtained the predicted quality, and they do no longer need a reference image. This methods are of special interest to this dissertation as in Chapter 6 we introduce a technique for simultaneous smoothing and sharpening of color images that needs to be evaluated through NR-IQA.

A review of the main methods to measure image quality is featured below. We will review the most popular among the three types listed providing some extra details on those used in the following chapters in this dissertation.

2.1 Full-Reference Image Quality Assessment

As commented above, for this type of methods, the quality is calculated by measuring the deviation of the output image from the reference image. Different functions can be used to measure this deviation. In order to properly assess the quality of the filtering both the noise suppression and the detail preserving abilities have to be evaluated. The *Mean Absolute Error* (MAE) is the most used function to approach the detail-preserving assessment and the *Peak Signal to Noise Ratio* (PSNR) is the function usually used to express the noise suppression ability. This last one objective quality measure has been used in this dissertation and it is defined as follows [23]:

$$MAE = \frac{\sum_{i=1}^{N \cdot M} \sum_{q=1}^{Q} \left| F_i^q - \hat{F}_i^q \right|}{N \cdot M \cdot Q} \tag{2.2}$$

$$PSNR = 20 \log \left(\frac{255}{\sqrt{\frac{1}{NMQ} \sum_{i=1}^{N \cdot M} \sum_{q=1}^{Q} \left(F_i^q - \hat{F}_i^q \right)^2}} \right) \tag{2.3}$$

where M, N are the image dimensions, Q is the number of channels of the image ($Q = 3$ for color images), and F_i^q and $\hat{F}_i^q$ denote the q^{th} component of the original image vector and the filtered image, at pixel position i, respectively.

In addition, the *Normalized Color Difference* (NCD) measure is also widely used since it approaches better the human perception of color differences [17] and is defined as:

$$NCD_{Lab} = \frac{\sum_{i=1}^{N \cdot M} \Delta E_{Lab}}{\sum_{i=1}^{N \cdot M} E^*_{Lab}} \tag{2.4}$$

where $\Delta E_{Lab} = [(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2]^{\frac{1}{2}}$ denotes the perceptual color error and $E^*_{Lab} = [(L^*)^2 + (a^*)^2 + (b^*)^2]^{\frac{1}{2}}$ is the *norm* or *magnitude* of the original image color vector in the $L^*a^*b^*$ color space.

Although these are probably the most used measurements in the literature, we can find many other measurements with which to assessment the quality of the images. Below are two methods that have also been used in this dissertation.

2.1.1 Structural similarity index

The widely-used PSNR technique does not always correlate with human visual perception and image quality [4]. To tackle this limitation, other figures of merit were proposed. One of the most popular ones is the *structural similarity index* (SSIM). This measure is based on the some properties of the human visual system to recognize structural similarity.

The main idea behind the method is to divide the images into blocks to compute similarity between pairs of blocks in the same image location. These similarities are later averaged to compute the overall image similarity. For the blocks similarity, the following three factor expression that evaluates similarity in luminance, contrast, and structure are used:

$$SSIM(x, y) = [l(x, y)]^{\alpha} [c(x, y)]^{\beta} [s(x, y)]^{\gamma} \tag{2.5}$$

where $\alpha > 0$, $\beta > 0$ and $\gamma > 0$ are parameters used to adjust the relative importance of the components and the luminance, contrast and structure similarity are respectively given by:

$$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \tag{2.6}$$

$$c(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \tag{2.7}$$

$$s(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \tag{2.8}$$

where μ_x, μ_y, σ_x, σ_y and σ_{xy} are the local means, standard deviations, and covariance of the image windows x and y. The constant C_1 is added in order to avoid the instability when $\mu_x^2 + \mu_y^2$ is closer to 0 and is defined as $C_1 = (K_1 L)^2$ being L the dynamic range of the pixel values (255 for 8-bit grayscale images), and $K_1 \ll 1$ a small constant. In the same way, $C_2 = (K_2 L)^2$ with $K_2 \ll 1$. Finally C_3 is defined as $C_3 = \frac{C_2}{2}$.

Setting the weights α, β and γ equal to 1 the reduced formula is obtained:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{2.9}$$

This measure was later extended to color using fuzzy logic in the *Fuzzy Color Structural Similarity* (FCSS) [2] measure, where more appropriate metrics were used to better process color correlation.

2.2 Reduced-Reference Image Quality Assessment

Reduced reference image quality metrics provide an intermediate solution between the FR and NR models. They are designed to predict the perceptual quality of distorted images with only partial information about the reference images. RR methods are useful in several applications. For example, in real-time visual communication systems, they can be used to track degradation of image quality and control streaming resources. In these techniques, a minimum set of image parameters are extracted and then used to determine image quality.

Jinjan et al. [15] developed a Reduced-Reference Image Quality Assessment based on visual information fidelity. They have proposed an index and used 30 bit data and achieve high consistency with human perception. Redi et al. [16] used descriptors based on color correlogram. They have analysed the alternations between the color distributions of an image for RR-IQA. Rehman and Zhou [18] proposed a RR-IQA method from the estimation of SSIM. Lin et al. [20] proposed an RRIQA by statically modeling the DCT distribution. Experimental analysis determines that only a small number of reduced reference parameters are sufficient to estimate the image quality. Xu et al. [21] introduced an approach for RRIQA which measures the differences of spatial arrangement between the reference image and distorted image in terms of spatial regularity measured by fractal dimension. Bhatijaet al. [22]

developed an application specific for smart cameras in which performance improvement and robustness can be achieved by intelligent moderation of the parameters both at algorithm and hardware level.

2.3 Non-Reference Image Quality Assessment

Currently, there is a growing number of non-reference image quality assessment methods due to the need to estimate a score of image quality without prior information or reference images, as it happens in sharpening and smoothing methods, among others. Also, the access to high performance machine learning methods is being very helpful.

The NR-IQA methods are based on the extraction of certain features of the image that will be employed to train some type of machine learning model. This training is what allows them to determine the quality of an image without the need to have information from the original image. However, as with any learning method, there is an intrinsic dependence on the database used for training. This dependence will be transferred to the application of the methods to real images, with which we will have to be careful to use them successfully.

In the following we review the most popular methods in this family.

2.3.1 BRISQUE

Mittal et al. [7] proposed Blind/Referenceless Image Spatial Quality Evaluator (BRISQUE), as an image quality metric in spatial domain. This algorithm is based on the idea that the normalized luminance coefficients of natural images obey generalized Gauss probability distribution and the image distortion will change the statistical characteristics of the normalized coefficient. By measuring the change in statistical characteristics, distortion types can be predicted and the visual quality of the image can be assessed.

This algorithm uses locally normalized luminance [9], i.e., Mean Subtracted Contrast Normalized (MSCN) image (I') which is calculated as shown below

$$I' = \frac{I(i,j) - \mu(i,j)}{\sigma(i,j) + 1} \tag{2.10}$$

$$\mu(i,j) = \sum_{k=-K}^{K} \sum_{l=-L}^{L} w_{k,l} I(i+k, j+l) \tag{2.11}$$

$$\sigma(i,j) = \sqrt{\sum_{k=-K}^{K} \sum_{l=-L}^{L} w_{k,l} [I(i+k, j+l) - \mu(i,j)]^2} \tag{2.12}$$

where I is the intensity of the image, $i \in 1, 2, \ldots, M$, $j \in 1, 2, \ldots, N$ being $M \times N$ the image size and $w = w_{k,l}|k = -K, \ldots, K$, $l = -L, \ldots, L$ a two-dimensional circularly symmetric Gaussian weighting function.

The idea on which BRISQUE is based is that MSCN have statistical properties which vary in the presence of distortions so that, if these changes are quantified, the quality of an image can be measured. In this way, with this MSCN image, a total of 16 different features in two different scales for each image are computed.

Finally, using Support Vector Machine (SVM) regressor (SVR) [24] for mapping from feature space to quality scores, BRISQUE is able to predict the final score. We have used BRISQUE to assess our proposal in Chapter 6 in this dissertation.

2.3.2 GSVD

Gradient Singular Value Decomposition (GSVD) is based on quality spatial features extracted from the product of energies of local dominant orientation, from the assumption that noise and blur affect the dominant direction of the local energy of the images.

For this, they start from a MSCN image [9], as BRISQUE. Once I' is computed, the image is partitioned in patches of different size. The horizontal and vertical components of the gradient are extracted for each patch, in this way, the main structure information of the image is obtained.

Using the Singular Value Decomposition of the gradient (SVD) computes the dominant orientation on each patch. Finally some features are extracted from the eigenvalue product of the SVD, such as the maximum of the histogram.

Then, as in BRISQUE, a training database and the under assessment image, the final score than quantifies the image quality is computed as the Euclidean distance between the features of the image to be assessed and the training database.

2.3.3 BLIINDS

Blind Image Integrity Notator using DCT Statistics(BLIINDS) [29] is based on a function of the representation of the features selected to represent the visual quality of the image being assessed. In particular, a feature representing image contrast is evaluated. In addition, the features we extract are also expected to take into account the sharpness of the image (without explicitly measuring blur distortion or any other specific distortion).

This method studies how the statistics of spatial frequency domain characteristics vary in natural and in distorted images. To do so, the discrete cosine transform (DCT) is employed to extract a number of features and model their statistics.

It seeks to observe how certain perceptually relevant statistical features of images change as an image becomes distorted, and then use these features to train a statistical model that is developed to make non-reference predictions about the quality of the assessed image.

A total of 4 features are computed in two different scales. Finally a probabilistic model is trained on a subset of the LIVE image database [26] to determine the parameters of the probabilistic model by distribution fitting. Two probabilistic models are chosen: the multivariate Gaussian distribution and the multivariate Laplacian distribution.

2.3.4 Other methods

In addition to the methods detailed above, among the most prominent NR-IQA techniques we can find: BIQI [27], a two-step method which involves distortion classification and a distortion quality assessment; DIVINE [28] as a extension of BIQI in which a series of features in the wavelet domain are used to predict the assessment; and BLINDS-II [29], that extracts features in the block-based DCT domain.

References

[1] Chen, Ming-Jun, and Alan C. Bovik. Fast structural similarity index algorithm. *Journal of Real-Time Image Processing* 6.4 (2011): 281-287.

[2] Grečova, Svetlana, and Samuel Morillas. Perceptual similarity between color images using fuzzy metrics.*Journal of Visual Communication and Image Representation* 34 (2016): 230-235.

[3] Hore, E. S., Qiu, B. and Wu, H. R. (2005). *Noise estimation in spherical coordinates for color image restoration*. Optical Engineering, 44(4), 047002.

[4] Huynh-Thu, Quan, and Mohammed Ghanbari. Scope of validity of PSNR in image/video quality assessment. *Electronics letters* 44.13 (2008): 800-801.

[5] Kamble, Vipin, and K. M. Bhurchandi. No-reference image quality assessment algorithms: A survey. *Optik-International Journal for Light and Electron Optics* 126.11-12 (2015): 1090-1097.

[6] Li, Chaofeng, and Alan C. Bovik. Content-partitioned structural similarity index for image quality assessment. *Signal Processing: Image Communication* 25.7 (2010): 517-526.

[7] Mittal, Anish, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing* 21.12 (2012): 4695-4708.

[8] Phadikar, Baisakhi Sur, Goutam Kumar Maity, and Amit Phadikar. Full Reference Image Quality Assessment: A Survey. *Industry Interactive Innovations in Science, Engineering and Technology*. Springer, Singapore, 2018. 197-208.

[9] Ruderman, Daniel L. The statistics of natural images. *Network: computation in neural systems* 5.4 (1994): 517-548.

[10] Sheikh, Hamid R., and Alan C. Bovik. Image information and visual quality. *IEEE Transactions on image processing* 15.2 (2006): 430-444.

[11] Sheikh, Hamid R., Alan C. Bovik, and Gustavo De Veciana. An information fidelity criterion for image quality assessment using natural scene statistics. *IEEE Transactions on image processing* 14.12 (2005): 2117-2128.

[12] Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli,E. P. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* 13.4 (2004): 600-612.

[13] Z. Wang, A.C. Bovik, A universal image quality index. *IEEE Signal Process.* Lett . 9 (3) (2002): 81-84

[14] Wu, J., Lin, W., Shi, G. and Liu, A. (2013). Reduced-reference image quality assessment with visual information fidelity. *IEEE Transactions on Multimedia*, 15(7), 1700-1705.

[15] Wu, J., Lin, W., Shi, G. and Liu, A. (2013). Reduced-reference image quality assessment with visual information fidelity. *IEEE Transactions on Multimedia*, 15(7), 1700-1705.

[16] Redi, J. A., Gastaldo, P., Heynderickx, I. and Zunino, R. (2010). Color distribution information for the reduced-reference assessment of perceived image quality. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(12), 1757-1769.

[17] B. Smolka, A. Chydzinski, Fast detection and impulsive noise removal in color images, *Real-Time Imaging* 11 5-6 (2005) 389-402.

[18] Rehman, A. and Wang, Z. (2012). Reduced-reference image quality assessment by structural similarity estimation. IEEE Transactions on Image Processing, 21(8), 3378-3389.

[19] Soundararajan, R. and Bovik, A. C. (2012). RRED indices: Reduced reference entropic differencing for image quality assessment. *IEEE Transactions on Image Processing*, 21(2), 517-526.

[20] Ma, L., Li, S., Zhang, F. and Ngan, K. N. (2011). Reduced-reference image quality assessment using reorganized DCT-based image representation. *IEEE Transactions on Multimedia*, 13(4), 824-829.

[21] Xu, Y., Liu, D., Quan, Y. and Le Callet, P. (2015). Fractal analysis for reduced reference image quality assessment. *IEEE Transactions on Image Processing*, 24(7), 2098-2109.

[22] Bhateja, V., Kalsi, A., Srivastava, A. and Lay-Ekuakille, A. (2015). A reduced reference distortion measure for performance improvement of smart cameras. *IEEE Sensors Journal*, 15(5), 2531-2540.

[23] K.N. Plataniotis, A.N. Venetsanopoulos, *Color Image processing and applications*, Springer-Verlag, Berlin, 2000.

[24] Schölkopf, B., Smola, A. J., Williamson, R. C. and Bartlett, P. L. (2000). New support vector algorithms. *Neural computation*, 12(5), 1207-1245.

[25] Charrier, C., Saadane, A. and Fernandez-Maloigne, C. (2015, February). Comparison of no-reference image quality assessment machine learning-based algorithms on compressed images. *Image Quality and System Performance XII* (Vol. 9396, p. 939610). International Society for Optics and Photonics.

[26] H.R. Sheikh, Z.Wang, L. Cormack and A.C. Bovik. LIVE Image Quality Assessment Database Release 2, http://live.ece.utexas.edu/research/quality.

[27] Sheikh, H. R., Bovik, A. C. and Cormack, L. (2005). No-reference quality assessment using natural scene statistics: JPEG2000. *IEEE Transactions on Image Processing*, 14(11), 1918-1927.

[28] Chen, J., Zhang, Y., Liang, L., Ma, S., Wang, R. and Gao, W. (2008, December). A no-reference blocking artifacts metric using selective gradient and plainness measures. *Pacific-Rim Conference on Multimedia* (pp. 894-897). Springer, Berlin, Heidelberg.

[29] Saad, M. A., Bovik, A. C. and Charrier, C. (2010). A DCT statistics-based blind image quality index. *IEEE Signal Processing Letters*, 17(6), 583-586.

[30] Moorthy, A. K. and Bovik, A. C. (2011). Blind image quality assessment: From natural scene statistics to perceptual quality. *IEEE transactions on Image Processing*, 20(12), 3350-3364.

3 Graph Theory

Graph theory is a tool that allows us to model the relationships between data elements and it has found wide application in science and engineering. Computer vision and image processing have a long history of using graph models, but these models have become increasingly dominant in recent literature. Graphs can be used to model spatial relationships between near and distant pixels, between image regions, between features, or as models of objects and parts.

In the next chapter we will see a brief introduction of the image processing techniques based on the most commonly used graph theory. From this introduction we will build the graph-based model that shall constitute the key piece of this dissertation.

In order to understand all the details of the model to be presented, as well as to show the reader all the possibilities offered by graph theory, in this chapter we will review the main concepts of graph theory most used in the image processing field, [5] [8] [4].

We will review terms such as *weighted graph*, base of a large part of the image processing models based on graphs, *neighborhood* of a node, which keeps an immediate analogy with the term neighborhood of a pixel, or *connected component*, one of the base concepts of the model to be presented. As we will see in the next chapter, the analysis of the related components of a graph will help us to distinguish the different areas of an image [7].

Although not all the concepts exposed next have been finally necessary in the final development of the model, they compose the background that has allowed to reach it. Knowing the possibilities of the graphs will help us to better understand the image and transfer the properties of the graphs to properties of the image.

3.1 Basic Graph Theory

Definition 3.1.1 *A graph G is a finite non-empty set $V(G)$ of objects called nodes (or, equivalently, vertices) and a set $L(G)$ of unordered pairs of distinct nodes/vertices of G called edge. Each edge $e \in L(G)$ has a unordered pair (v_i, v_j) of nodes/vertices associated called $tail(e) = v_i$ and $head(e) = v_j$.*

They are called the endpoints of e and denoted by endpts(e). If $v_i = v_j$, the link e is called a loop.

From now on throughout the dissertation, we will equivalently use either node/nodes or vertex/vertices for referring to the elements of $V(G)$.

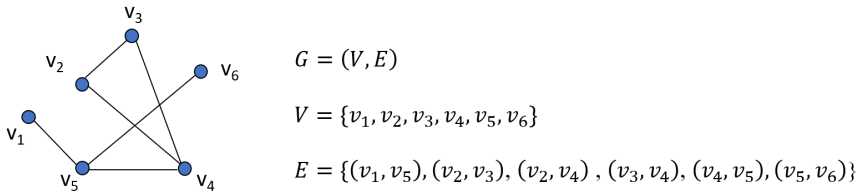


Fig. 3.1: Example of a graph with 6 nodes and 6 links.

In order to avoid confusion with the image processing terminology, we will call links to the elements of $L(G)$ instead of edges, as it is common practice.

We consider two functions $s : L(G) \longrightarrow V(G)$ and $t : L(G) \longrightarrow V(G)$. Function s is called the source function and t the target function. Given a link $e_{ij} = (v_i, v_j) \in L(G)$, we say that $s(e_{ij}) = v_i$ is the *origin* or *source* of e_{ij} and and $t(e_{ij}) = v_j$ is the *endpoint* or *target* of e_{ij}.

Definition 3.1.2 *Adjacent nodes are two nodes $v_i, v_j \in V(G)$ that are joined by a link $e_{ij} = (v_i, v_j)$.*

Definition 3.1.3 *Adjacent links are two links that have an endpoint in common.*

The links d and i of the graph of the Figure 3.2 are loops.

Definition 3.1.4 *An acyclic graph is a graph without cycles.*

Definition 3.1.5 *If node $v_i \in V(G)$ is an endpoint of a link $e_{ij} \in L(G)$, then v_i is said to be incident on e_{ij}, and e_{ij} is incident on v_i.*

Definition 3.1.6 *The degree of a node v_i in a graph G, deg(v), is the number of proper links incident on v_i plus twice the number of self-loops.*

Definition 3.1.7 *The degree sequence of a graph is a sequence formed by arranging the node degree in non-decreasing order.*

Definition 3.1.8 *In a graph, a walk form node v_0 to node v_n is an alternate sequence*

$$W =< v_0, e_1, v_1, e_2, \ldots, v_{n-1}, e_n, v_n >$$

of nodes and links, such that $tail(e_i) = v_{i-1}$ and $head(e_i) = v_i$ for $i = 1, \ldots, n$.

$$V = \{v_1, v_2, v_3, v_4, v_5\}$$
$$E = \{a, b, c, d, e, f, g, h, i\}$$

Edge	a	b	c	d	e	f	g	h	i
endpts	v_1	v_7	v_6	v_2	v_2	v_5	v_5	v_3	v_4
	v_7	v_5	v_2	v_2	v_5	v_3	v_4	v_4	v_4

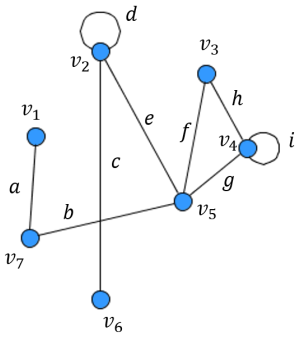


Fig. 3.2: A graph and the corresponding endpts of each link.

It is said that two nodes v_i and v_j are connected if there is a walk W that joins them.

Definition 3.1.9 *The length of a walk is the number of link-step in the walk sequence.*

Definition 3.1.10 *A walk is said to be closed if the starting node is the same as the ending node, that is $v_0 = v_n$. A walk is said to be open otherwise.*

Definition 3.1.11 *A subwalk of a walk $W =< v_0, e_1, v_1, e_2, \ldots, v_{n-1}, e_n, v_n >$ is a sequence of consecutive entries $S =< v_j, e_{j+1}, v_{j+1}, \ldots, e_k, v_k >$ such that $0 \leq j \leq k \leq n$, that begins and ends at a node. Thus, the subwalk is itself a walk.*

Definition 3.1.12 *In a graph, the distance from node s to node t is the length of a shortest walk form s to t, or ∞ if there is no walk from s to t.*

Definition 3.1.13 *A trail is a walk with no repeated links.*

Definition 3.1.14 *A path is a sequence of nodes $v_0, \ldots, v_n$ all different such that every pair (v_i, v_{i+1}) is an link.*

In other words, a path is a trail where all nodes are distinct.

Definition 3.1.15 *If a walk $W = v_0, v_1, e_2, \ldots, v_l$ is such that $l \geq 3$, $v_0 = v_l$ and the nodes v_i, $0 < i < l$, are distinct from each other and v_0, then W is said to be a cycle.*

Definition 3.1.16 *If every pair of distinct nodes of a graph without loops are joined by a link we say that the graph is complete.*

Definition 3.1.17 *When each link (u, v) has an associated value $w(u, v)$, we say that the graph is weighted.*

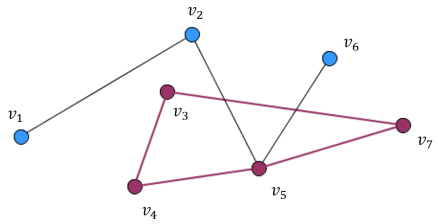
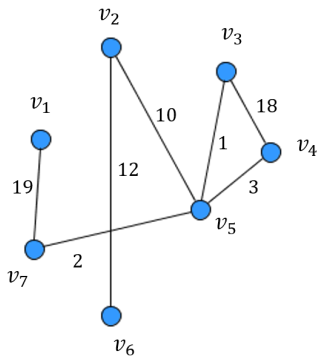


Fig. 3.3: An example of a cycle in a graph.



$w(v_1, v_7) = 19$ $\quad$ $w(v_2, v_5) = 10$

$w(v_5, v_7) = 2$ $\quad$ $w(v_4, v_5) = 3$

$w(v_2, v_6) = 12$ $\quad$ $w(v_3, v_4) = 18$

$w(v_3, v_5) = 1$

Fig. 3.4: Example of a weighted graph.

The $w(u, v)$ value is known as the weight of the link (u, v).

Definition 3.1.18 *A subgraph of a graph G is a graph H whose nodes and links are all in G.*

Definition 3.1.19 *A proper subgraph H of G is a subgraph such that $V(H)$ is a proper subset of $V(G)$ of $E(G)$ is a proper subset of $E(G)$.*

Definition 3.1.20 *A subgraph H is said to be a spanning subgraph G if $V(H) = V(G)$.*

Definition 3.1.21 *For a given graph G the subgraph induced on a node subset U of $V(G)$, denoted by $G[U]$, is the sugbraph of G whose node-set is U and whose link set consist of all links in G that have both endpoints in U. That is,*

$$V(G[U]) = U$$
$$E(G[U]) = \{e \in E(G) : endpts(e) \subseteq U\}$$

Definition 3.1.22 *For a given graph G the subgraph induced on a link subset d of $E(G)$, denoted by $G[D]$, is the sugbraph of G whose link set is D and*

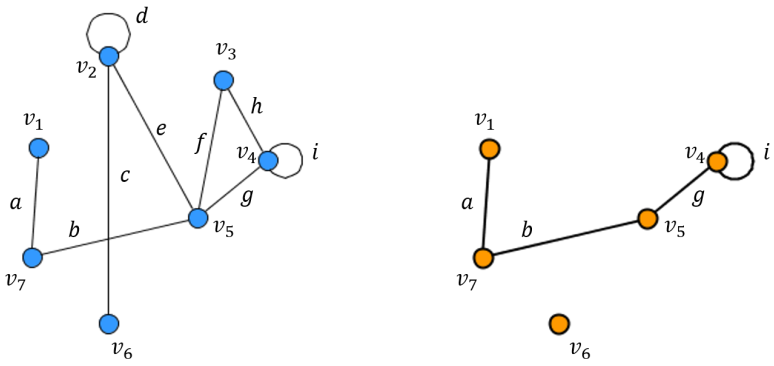


Fig. 3.5: The subgraph induced on $\{v_1, v_7, v_5, v_4\}$.

whose node set consist of all nodes that are incident with at least one link in D. That is,

$$E(G[D]) = D$$
$$V(G[D]) = \{v \in V(G) : v \in endpts(e), \text{ for some } e \in D\}$$

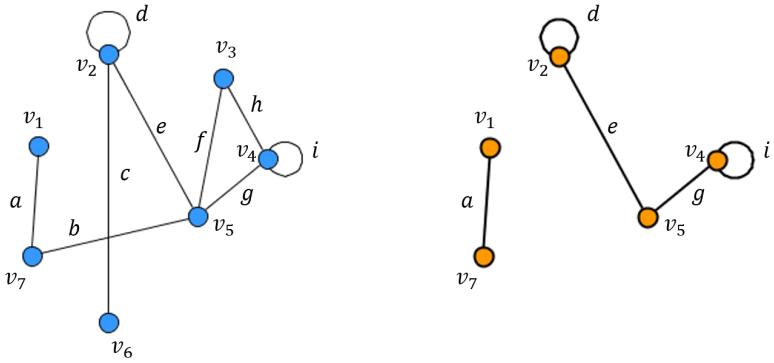


Fig. 3.6: The subgraph induced on $\{d, e, g, i\}$.

Definition 3.1.23 *A subgraph S of G is called a clique if its complete and maximal respect the links.*

In other words, a clique is a subgraph in which each node is connected to all the other links of the subgraph and there is not another complete subgraph that contains it strictly. This is tantamount to saying that the subgraph induced by S is a complete graph.

Definition 3.1.24 *A size of a clique is the number of nodes it contains.*

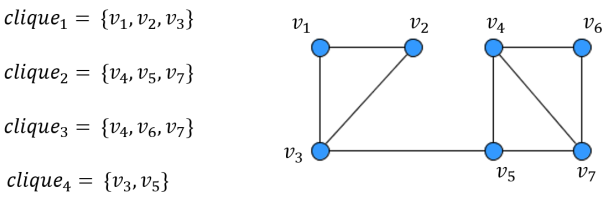


Fig. 3.7: Example of a graph with cliques.

Definition 3.1.25 *Any node adjacent to a node is said to be its neighbor.*

Definition 3.1.26 *The neighborhood of a node v, $N_G(v)$, is the subgraph induced by the node set consisting of v and all its neighbors.*

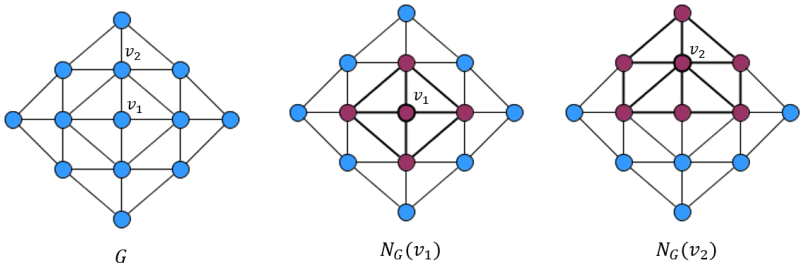


Fig. 3.8: From left to right, an example of graph, the neighborhood of node v_1 and the neighborhood of the node v_2.

3.2 Paths, Trees and Connectivity

Connectivity is one of the basic concepts of graph theory and one of the most relevant concepts in the field of image processing. The study of the connections between nodes and the characterization of a graph regarding its connectivity will allow us to study and model the pixels of an image and work with the tools provided by this theory. The concepts presented below will be the basis of the processing model presented in this thesis as we will see in the following chapters.

Definition 3.2.1 *A graph is connected if for every pair v_i, v_j of distinct nodes there is a walk from v_i to v_j.*

Definition 3.2.2 *A connected component of a graph G is a connected subgraph H of G such that there is not a connected subgraph of G that contains H strictly.*

Intuitively, the components of a non-connected graph are the "whole pieces" it comprises.

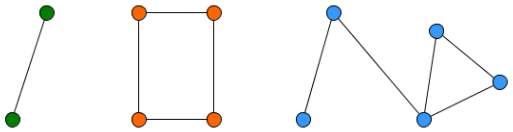


Fig. 3.9: A graph with three components.

Different components of the same graph do not have any common nodes because of the following theorem.

Theorem 1. *If the graph G has a node v that is connected to a node of the component H of G, then v is also a node of H.*

Identify the components of a small graph is trivial. But larger graphs that are specified by some computer representation require a computer algorithm.

Definition 3.2.3 *A tree is a connected acyclic graph.*

Definition 3.2.4 *Let G be a graph and T a tree which is a subgraph of G. A frontier link of G is an link which has exactly one endpoint in T.*

Definition 3.2.5 *A forest is a disjoint union of trees.*

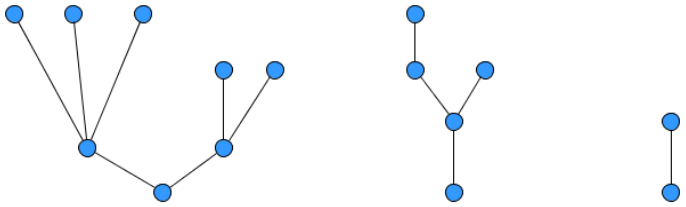


Fig. 3.10: A forest with three trees.

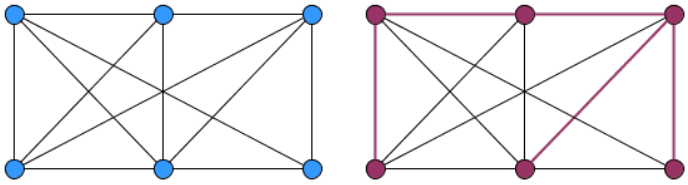


Fig. 3.11: Example of a graph and a spanning tree.

Definition 3.2.6 *A spannig tree of a graph is a spanning subgraph that is a tree.*

Proposition 3.2.7 *A subgraph H of a connected graph G is a subgraph of some spanning tree if and only if H is acyclic.*

A spanning tree of the graph that minimizes the cost (compared to all spanning trees of the graph) is called a minimal spanning tree. The different weights ensure that the minimum spanning tree of a graph is unique. Without this condition, there may be several different minimum spanning trees. The problem of finding a minimal spanning tree of a graph appears in many applications. There are various algorithms for finding minimum spanning trees, and we present the one proposed by Prim. The principle of Prims algorithm is to progressively build a tree, finding a minimum spanning tree without explicitly examining all the spanning trees.

Data: A weighted connected graph G
Result: A minimum spanning tree T
Initialize the Prim tree T as node s;
Initialize the set of frontier links for tree T as empty;
while *Prim tree T does not yet span G* **do**
  Update the set of frontier links for T;
  Let e be a frontier link for T with the smallest link-weight;
  Let v be the non-tree end point for link e;
  Add link e and node v to tree T;
end

Algorithm 1: Prims algorithm minimum spanning tree.

The shortest path between two nodes of a graph consists on the path with minimum weight, that is, the sum of the weights of the links that make up the path is minimal. There are many algorithms that solve this problem, among the best known are the Bellman-Ford algorithm, the Floyd-Warshall algorithm or the Dijkstra algorithm.

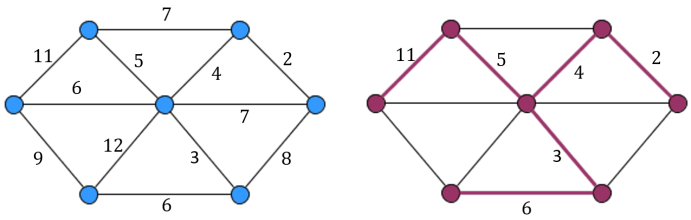


Fig. 3.12: A weighted graph and its minimum spanning tree.

Dijkstra algorithm allows us to efficiently find the shortest paths from a node s of a general graph with positive weights to all other nodes. Note that the only restriction is that the weights of the graph are required to be nonnegative, condition that in several real-life applications is satisfied. The strategy of this algorithm is similar to the one used for Prims algorithm, growing a tree, starting at a node s, by adding, at each iteration, a frontier link whose non-tree endpoint is as close as possible to s.

Data: A weighted connected graph G whose link-weights are non-negative; and the initial node s of G
Result: A spanning tree T of G, rooted at node s, whose path from s to each node v is a shortest path from s to v in G; and the distance from s to each node.
Initialize the Dijkstra tree T as node s;
Initialize the set of frontier links for tree T as empty;
$dist(s) = 0$ Write label 0 on node s **while** *Dijkstra tree T not yet span G* **do**
  Update the set of frontier links for T;
  for *each frontier link e for T* **do**
    Let x be the labeled endpoint of link e.;
    Let y be the unlabeled endpoint of link e. ;
    Set $P(e) = dist(x) + w(e)$.
  end
  Let e be a frontier link for T that has the smallest P-value ;
  Let x be the labeled endpoint of link e;
  Let y be the unlabeled endpoint of link e;
  Add link e and node y to tree T;
  $dis(y) = P(e)$;
  Write label $dist(y)$ on node t. ;
end
Algorithm 2: Dijkstra algorithm for finding the shortest path.

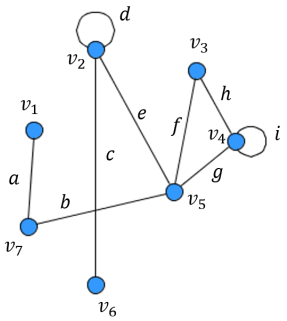
3.3 Graph representations

The graphical representation of graphs that we have seen in the different examples offers us a visual and aesthetic tool to understand and work with graphs. However it is not a very operative tool when carrying out more complicated processes or working with large graphs.

The matrix representation of a graph is often convenient if one intends to use a computer to obtain some information or solve a problem concerning the graph. This kind of representation of a graph is conducive to study properties of the graph by means of algebraic methods. We can consider several computer representations of graphs. The data structures used to represent the graphs can have a significant influence on the size of the problems that can be performed on a computer and the speed with which they can be solved. That is why it is so important to know the different graph representations.

In addition to the computational advantage, it is possible to derive incidence ratios, circuits and cutting sets using theorems and matrix manipulations. In other words, graph theory once again provides us a powerful tool to model and understand data, in our case, images.

Below we introduce the most common matrix representations of the graphs.



$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Fig. 3.13: A graph and its adjacency matrix.

Definition 3.3.1 *The adjacency matrix of a graph G, denoted by A_G, is the matrix whose rows and columns are both indexed by identical orderings of $V(G)$, such that*

$$A_G[v_i, v_j] = \begin{cases} 1 \text{ if } e_{ij} = (v_i, v_j) \in E(G) \\ 0 \text{ in other case} \end{cases}$$

A weighted graph may be represented using the weight as the entry.

Definition 3.3.2 *The adjacency matrix of a weighted graph G is given by*

$$A_G[v_i, v_j] = \begin{cases} w(v_i, v_j) \text{ if } (v_i, v_j) \in E(G) \\ \infty \text{ otherwise} \end{cases}$$

$$\begin{pmatrix} \infty & \infty & \infty & \infty & \infty & \infty & 5 \\ \infty & \infty & \infty & \infty & 9 & 4 & \infty \\ \infty & \infty & \infty & 1 & 1 & \infty & \infty \\ \infty & \infty & 1 & \infty & 7 & \infty & \infty \\ \infty & 9 & 1 & 7 & \infty & \infty & 2 \\ \infty & 4 & \infty & \infty & \infty & \infty & \infty \\ 5 & \infty & \infty & \infty & 2 & \infty & \infty \end{pmatrix}$$

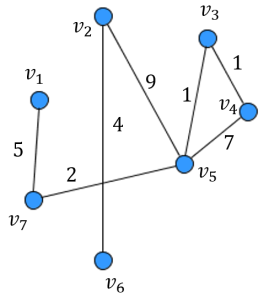


Fig. 3.14: A weighted graph and its adjacency matrix.

Definition 3.3.3 *The incidence matrix of a graph G is the matrix I_G whose rows and columns are indexed by some ordering of $V(G)$ and $E(G)$ respectively, such that*

$$I_G[v, e] = \begin{cases} 0 \text{ if } v \text{ is not an endpoint of } e \\ 1 \text{ if } v \text{ is an endpoint of } e \\ 2 \text{ if } e \text{ is a loop at } v \end{cases}$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 2 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

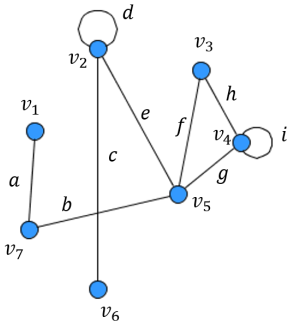


Fig. 3.15: A graph and its incidence matrix.

Proposition 3.3.4 *The sum of the entries in any row of an incidence matrix is the degree of the corresponding node.*

Proposition 3.3.5 *The sum of the entries in any column of an incidence matrix is equal to 2.*

References

[1] Biggs, Norman, E. Keith Lloyd, and Robin J. Wilson. Graph Theory, 1736-1936. *Oxford University Press*, 1976.

[2] Gibbons, Alan. Algorithmic graph theory. *Cambridge university press*, 1985.

[3] Biggs, Norman, et al. Algebraic graph theory. Vol. 2. *Cambridge: Cambridge university press*, 1974.

[4] Gross, J.L., Yellen, Graph Theory and its applications. *Chapman& Hall/CRC* , Boca Raton (Fla.), London, New York (2006.

[5] Jordán, C.; Torregrosa, J.R. Introducción a la Teoría de Grafos y sus Algoritmos. *Reverté/Universitat Politècnica de València*, València, Spain, (1996).

[6] Diestel, Reinhard. Graph theory. Springer Publishing Company, Incorporated, 2017.

[7] Pérez-Benito, C., Morillas, S., Jordán, C., Conejero, J. A. . Determinación de componentes conexas en el análisis de zonas homogéneas y de detalle en imágenes a color. *Modelling in Science Education and Learning*. Universitat Politècnica de València, 2018. p. 5-14.

[8] Jordán C., Morillas S., Sanabria-Codesal E.: Colour image smoothing through a soft-switching mechanism using a graph model, *IET Image Processing*, 6 (9),1293-1298 (2012).

Part II

Model based on Graph Theory

4 Model based on Graph Theory for Color Image processing

After discussing some basic concepts about image processing and graph theory, as well as having reviewed the state of the art of smoothing and sharpening, we provide a brief overview of the use of Graph Theory in image processing, that will serve as an introduction of a graph-based model of images.

More precisely, we describe a model for the description of the structure of an image that is based on graph theory[1].

This model permits us to distinguish the different areas of the image with high precision and to characterize the nature of each pixel in the image. This characterization turns the model into a powerful tool for different applications of image processing, not only to segment and distinguish areas of the image, but also to smooth and enhance the image. This model will be also used in the following chapters.

In this chapter, we show how this model can be applied in order to to detect the edges of a color image, by eliminating noise and even enhancing its borders.

4.1 Modeling color images with Graph Theory: Introduction

Graph theory offers us a way to represent and understand an image in a very simple and visual way, as well as it also provides us with a powerful tool for different image processing tasks. One of these tasks is *segmentation*, which consists on partitioning an image into several disjoint subsets such that each subset corresponds to a meaningful part of the image. This problem is commonly modeled in terms of partitioning a graph into several sub-graphs, so that each of these sub-graphs represents one of these meaningful parts. In this line, the most commonly used techniques are based on the use of minimal spanning trees and graph cuts, with cost functions or based on Markov

[1] Translation and further details of Pérez-Benito, C., Morillas, S., Jordán, C., Conejero, J. A. (2018, February). Determinación de componentes conexas en el análisis de zonas homogéneas y de detalle en imágenes a color. *In Modelling in Science Education and Learning* (Vol. 11, No. 1, pp. 5-14). Universitat Politècnica de València.

random fields. It is worth to mention that minimum spanning trees have been also used for smoothing, see [10, 12, 13, 11].

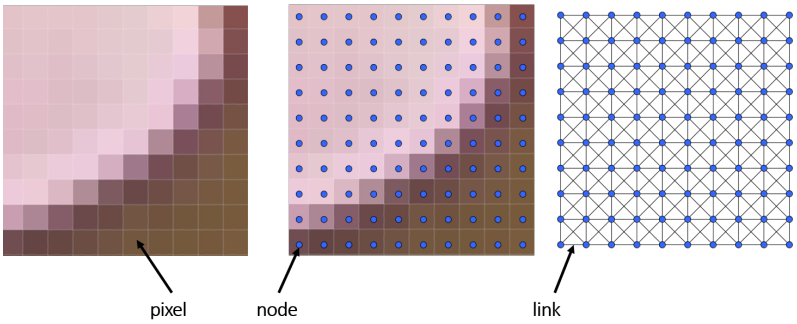


Fig. 4.1: Example of mapping an image onto an 3-connected graph.

There are several ways to model an image using a graph, the most intuitive and the most used one is to map each pixel of the image into a node of a graph. In this way, nodes can be connected in different ways, for example with spatial criteria. Then, we can assign weights to the links that join pairs of nodes. An intuitive approach is to assign to this weight the difference between the corresponding pixel intensities of the nodes, which is the one that will be considered in our work. Nevertheless, weights can also be given by more complex weight function. Both, the construction of the graph associated with the image and its subsequent treatment will be what defines the processing method.

Methods based on minimum spanning trees (mST) let us build a tree associated with each image, trying to ensure that the edges of the graph define as best as possible the similarity between the pixels they are joining. The underlying idea is that in this tree the most similar pixels will be kept together, so that, when cutting the mST into subtrees, we obtain a segmented decomposition of the image.

Graph cut methods [2, 3, 4, 5, 6, 7] can also be considered as an alternative way of converting the image into aff graph. Apart from considering each pixel as a node, we add two (or more) additional nodes, one for each zone that we want to distinguish within the image. We connect all the nodes among them and assign weights to these links in order to keep the similarity of the pixels. If we consider the additional nodes as sources and sinks, using Ford-Fulkerson max-flow/min-cut algorithm it will return us a decomposition of the graph into subgraphs corresponding to the image pixels, providing us different clusters of the image.

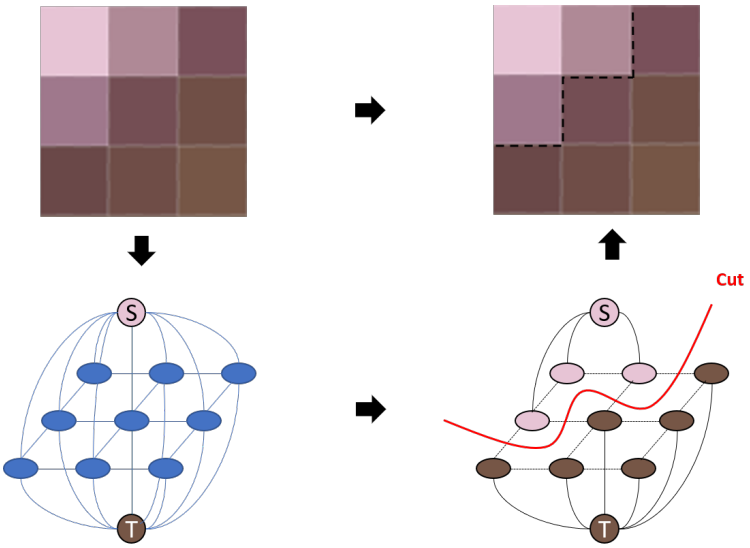


Fig. 4.2: Segmentation using Graph Cuts.

The performance of the previous approaches depend heavily on the weight function associated with the links. The better that this function represents local features, the better that the image processing will be.

4.2 Definition of the model

We describe our proposed model and how it will allows us to characterizes a color image in a local way. For each pixel of the image, we consider also all the pixels in its neighborhood. Despite we will also consider the general case of a neighborhood given by a $n \times n$ square centered at each pixel, the examples will be illustrated for 3×3 neighborhoods.

For assigning weights to the links, we consider the decomposition of the image into color channels, giving a description of the color image as a vector field. In our work we will considered the decomposition into red, green, and blue channels (RGB). Then, we consider each pixel of the image as a 3 component vector, with one component for each channel. We also consider that the intensity of a pixel in each channel takes a value between 0 and 255.

Given a color image $\mathbf{F}$, represented in the RGB color space, we construct a graph for each image pixel. For any arbitrary pixel of the image $\mathbf{F}_0$ we considered a window centered on it of size $N \times N$ where $N = 2n + 1$ and $n = 1, 2, \ldots$. The remaining pixels in the window are denoted by $\mathbf{F}_i, i = 1, \ldots, N^2 - 1$. Each pixel $\mathbf{F}_i$ will be represented by the 3-tuple (F_i^R, F_i^G, F_i^B), $i = 0, \ldots, N^2 - 1$ of its three color components in the RGB space.

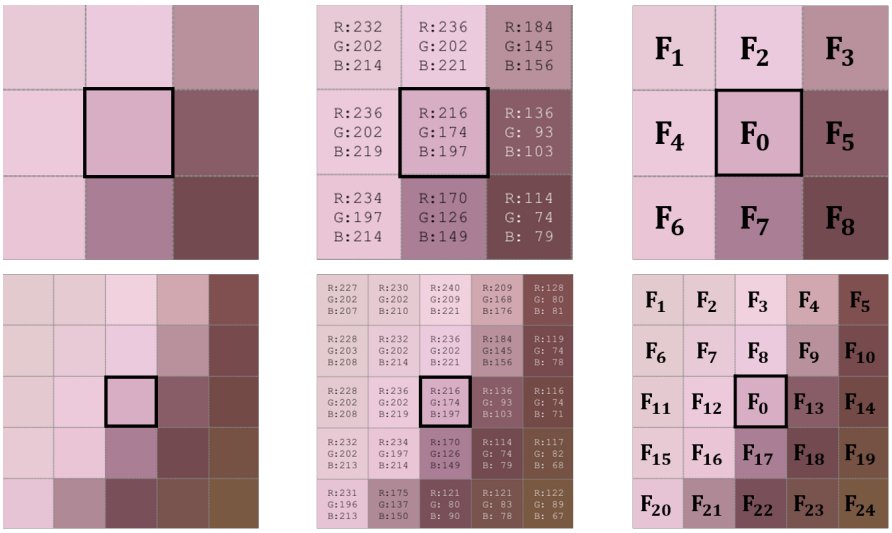


Fig. 4.3: Example of 3×3 and 9×9 window, its RGB color components and the notation of each pixel of the window.

We define the *local weighted graph* $G_{\mathbf{F}_0}$ associated to the pixel $\mathbf{F}_0$, with the following set of nodes and links.

$$V(G_{\mathbf{F}_0}) = \{\mathbf{F}_i,\, i = 0, \ldots, N^2 - 1\} \tag{4.1}$$

$$L(G_{\mathbf{F}_0}) = \{(\mathbf{F}_i, \mathbf{F}_j),\, i \neq j,\, ||\mathbf{F}_i - \mathbf{F}_j||_2 < \mathcal{U}\} \tag{4.2}$$

That is, there is an edge between the pixels $\mathbf{F}_i$ and $\mathbf{F}_j$, $i \neq j$, if the euclidean distance between their color vectors in the RGB space is lower than a certain threshold $\mathcal{U}$. If such a link exists, its weight is

$$w(\mathbf{F}_i, \mathbf{F}_j) = ||\mathbf{F}_i - \mathbf{F}_j||_2, \tag{4.3}$$

where $||\cdot||_2$ stands for the Euclidean norm.

The Euclidean distance gives us a measure of similarity between pixels. So that, the shorter the Euclidean distance between them, the more similar they are. This model aims to associate each pixel of the image with a graph whose connections represent the similarities between pixels.Then, if two pixels are connected in the graph we can assume that they are similar enough.

We must also bear in mind that we will work with noisy images and Gaussian Noise affects all pixels. This causes many difficulties to differentiate whether a pixel belongs to a detail zone or, on the contrary, is a noisy pixel. In general, the presence of noise will cause all Euclidean distances between pixels to increase and therefore the value of the threshold $\mathcal{U}$ should be increased in order to correctly differentiate homogeneous and detailed areas.

Experimentally, it is possible to verify that the value of $\mathcal{U}$ is key in the structure of the local graphs since it will allow us to determine the connected component of the graph $G_{\mathbf{F}_0}$ that contains the vertex F_0. This connected component is fundamental when it comes to classify image pixels as belonging to homogeneous regions or regions of details.

The fact that the connected component that contains the central pixel has many nodes and/or many edges will be a sign that the pixels are very similar between them and therefore we will be in a homogeneous area. On the other hand, if this component has few edges or few nodes, we will know that the similarity between pixels is not so clear and we could be in an area of detail. The structure that will allow us to know in which zone we are is determined by the aforementioned threshold $\mathcal{U}$.

The setting of the parameter $\mathcal{U}$ will depend on the specific application of the model. As we have mentioned before, there are different features of graphs that we can use to characterize the pixels of the image. The choice of one or the other feature, and therefore the optimization of the threshold, will depend on the pursued objective. In the next chapter we will see two possible applications of the model presented, in each of them we will use two different features:

- The cardinal of the link set of the connected component that contains the central pixel, $\mathbf{card}(L(H_{\mathbf{F}_0}))$.
- The cardinal of the node set of the connected component that contains the central pixel, $\mathbf{card}(V(H_{\mathbf{F}_0}))$.

The first one is related to the structure of the graph and gives us a wider range of values to classify the pixel. Given an $N \times N$ window, $\mathbf{card}(L(H_{\mathbf{F}_0}))$ will take values from 0 to $\binom{N}{2}$, which represents that the pixel belongs to an area of very much detail, for the case of 0, or to a very homogeneous one, for the case of $\binom{N}{2}$. This feature will be very useful when the main objective is to segment an image as accurately as possible.

The second feature offers us a smaller range of values, from 0 to N, and therefore a less detailed classification. However, this limited range makes the selection of the threshold very restrictive and a good choice of threshold will divide the graph into different connected components, each of of them identifying an area of the image. This approach will let us classify pixels between those that belong to the same area as the central pixel, and those that belong to other areas.

We are going to illustrate it with an example: in the Figure 4.4 there is a 3×3 window corresponding to a detail area of the Pills image, specifically the neighborhood of the pixel $(26, 42)$. Visually it is clear that we are in a edge or detail zone of the image, and we can easily appreciate two areas. In order to simplify the subsequent examples we will enumerate the pixels/nodes of the window/network as described in the figure.

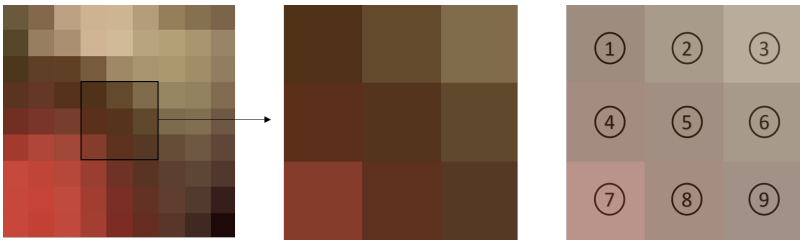


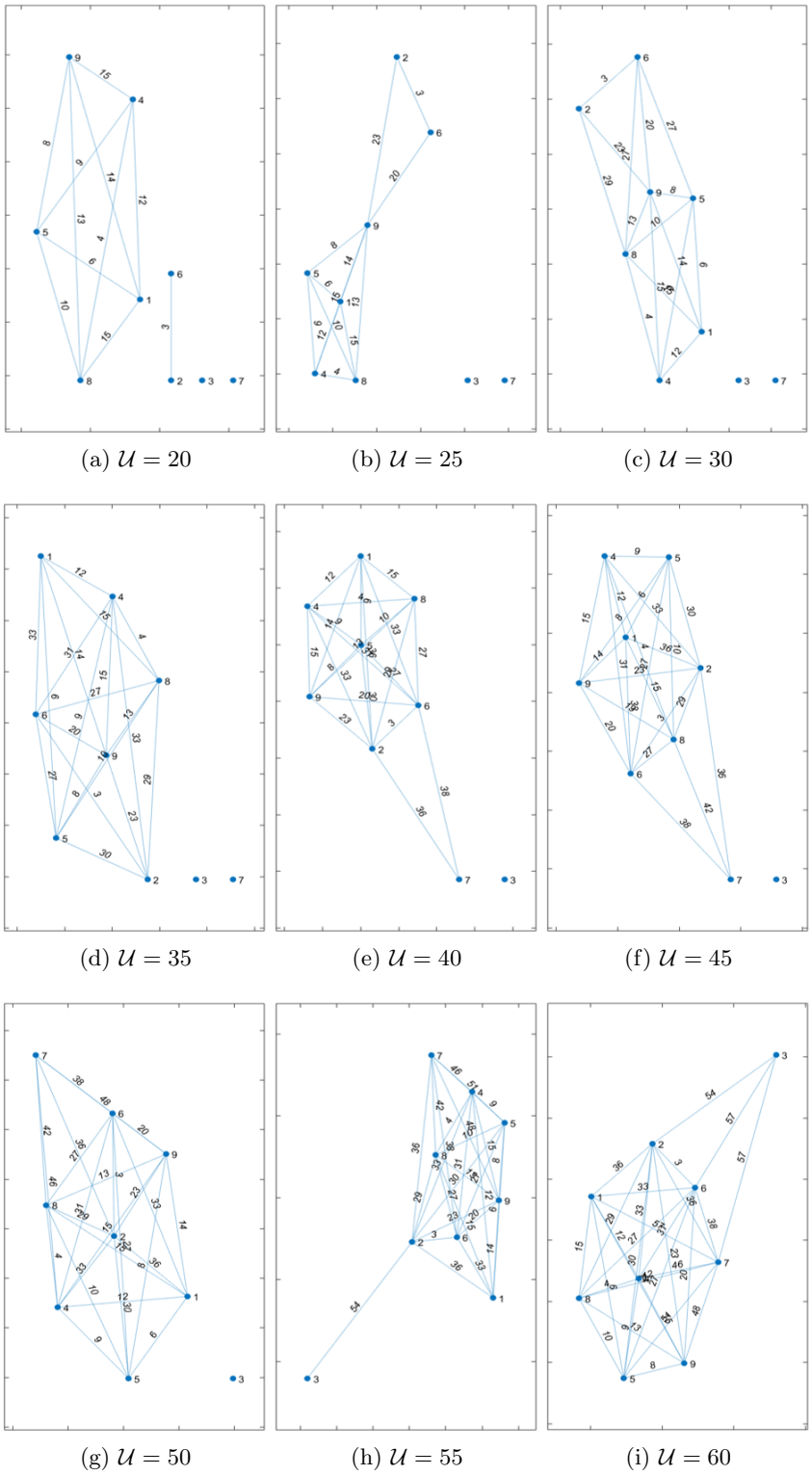
Fig. 4.4: From left to right: Example of a detail region of the image, zoom in 3×3 window and finally the numeration of the pixels of the window.

In figure 4.5 we can see how the graph associated with this window is transformed, reducing their number of edges and increasing the number of connected components, as we increase the threshold.

In Figure 4.6 we can see how the number of edges in the connected component of the central pixel increases as the threshold increases. When the threshold is greater than the maximum of the weight links, the graph will not change anymore, and all nodes will remain connected among them.

On the other hand, in the graphic of Figure 4.7 we see that the cardinal of the nodes of the central pixel connected component takes the values 5, 7, 8 and 9 as the threshold increases.

It is clear that the use of link set cardinal provides a wider tange of values and facilitates subsequent pixel classifications. However, as we will see below, for certain applications the precision will not be as important as the existence of a characterization of the different connected components in the graph.



(a) $\mathcal{U} = 20$ (b) $\mathcal{U} = 25$ (c) $\mathcal{U} = 30$

(d) $\mathcal{U} = 35$ (e) $\mathcal{U} = 40$ (f) $\mathcal{U} = 45$

(g) $\mathcal{U} = 50$ (h) $\mathcal{U} = 55$ (i) $\mathcal{U} = 60$

Fig. 4.5: Evolution of the graph as a function of the threshold.

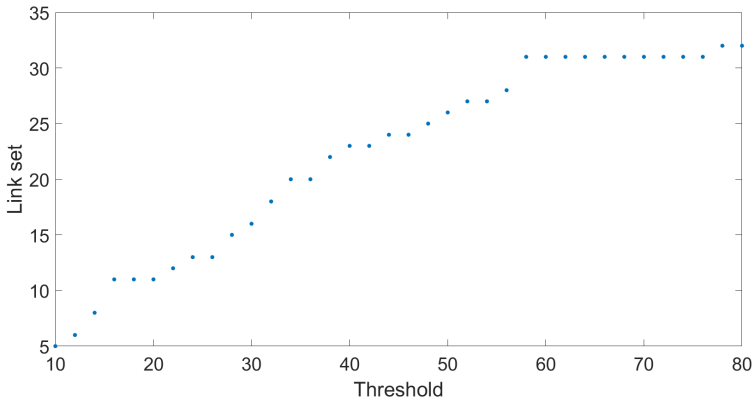


Fig. 4.6: Value $\mathbf{card}(L(H_{\mathbf{F}_0}))$ in function of the threshold.

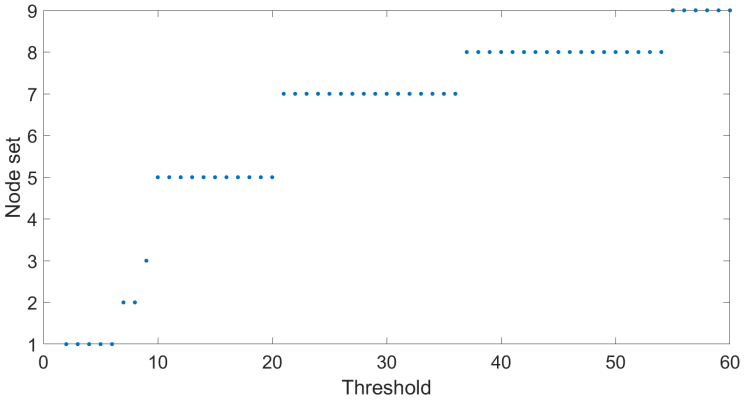


Fig. 4.7: Value $\mathbf{card}(V(H_{\mathbf{F}_0}))$ in function of the threshold.

4.3 A edge detector based on the graph-model

Our model has a number of applications within image processing field. Probably, the more direct application is border detection. We have seen that the feature that characterizes whether a pixel $\mathbf{F}_0$ belongs to a flat or edge/detail region with the greatest precision is the cardinal of the links set of its connected component, $\mathbf{card}(L(H_{\mathbf{F}_0}))$. A higher cardinality indicates great similarity between all the pixels in the window and, therefore, pixel $\mathbf{F}_0$ will be

associated to a flat region of the image. On the other hand, lower values indicates indicate less similarity between pixels and it will associate $\mathbf{F}_0$ to textures, edges, or details.

In Figure 4.8 we show an example of a homogeneous region of an image (left). We choose a 3×3 window around the central pixel (middle) and we compute the associated subgraph $G_{\mathbf{F}_0}$ for a threshold $\mathcal{U} = 38$ (right), as we have already indicated. Here, we can see that all nodes belong to a unique connected component. This structure of the graph indicates us that we are in a flat zone of the image, without details or edges. On the other hand, in Figure 4.9 we show an example of an edge region, where we can see two different connected components of the subgraph $G_{\mathbf{F}_0}$, one for each perfectly differentiated zone in the 3×3 image window.

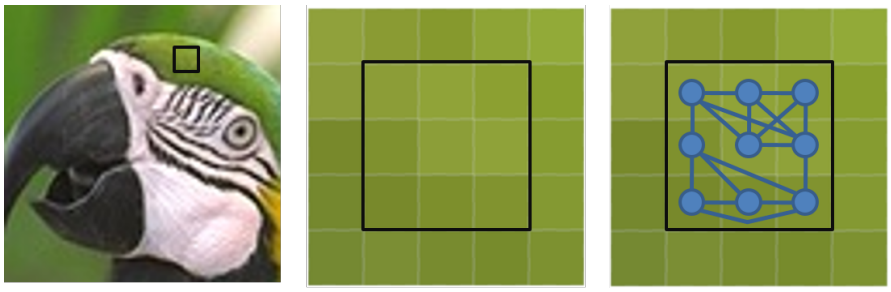


Fig. 4.8: Example of homogeneous region, from left to right: flat region of the image, a zoom of a 3×3 window to be processed, and finally the graph associated to this window for $\mathcal{U} = 38$.

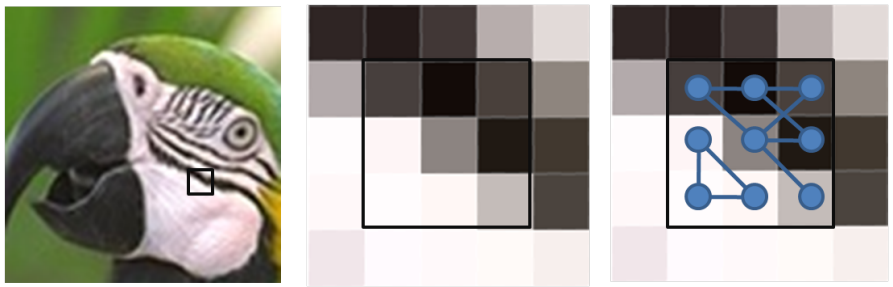


Fig. 4.9: Example of detail region, from left to right: detail region of the image, a zoom in the 3×3 window to be processed and finally, the graph associated to this window for $\mathcal{U} = 38$.

Now, if we create a gray-scale image of the given one, where the intensity of each pixel is proportional to the cardinal of the set of links of the corre-

Fig. 4.10: Evolution of the border detection in function of the threshold of the Lenna image, from $\mathcal{U} = 10$ to $\mathcal{U} = 60$

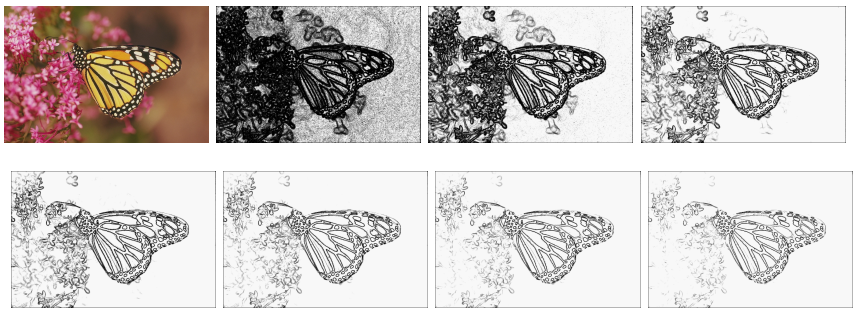


Fig. 4.11: Evolution of the border detection in function of the threshold, from $\mathcal{U} = 5$ to $\mathcal{U} = 70$

sponding local graph, we will be able to see in a visual way the details of the image. Figures 4.10 and 4.11 shows this gray-scale images and the effect of the threshold in the classification. n both figures, we can see that for low thresholds more pixels are classified as borders and for high thresholds, on the contrary, we have more homogeneous zones.

Although the threshold value may vary according to specific objectives in each case, we will try to select an optimal parameter, understanding by optimal the one that best distinguishes the different areas of the image. For this, we take as golden standard the output of the fuzzy edge detector [16] and select the threshold that provides the grayscale image more similar to our golden standard. To measure this similarity, we will use the well-known mutual information that will allow us to compare our grayscale image with the output of the fuzzy edge detector, see Figure 4.12.

To illustrate the robustness of our model, we show its operation on several images with different levels of noise intensities. The results are shown in Figure 4.13. We see that, as the noise increases, the model gives us a less marked grayscale image. This is due to the need of increasing the threshold when the noise intensity is very high in order to avoid confusing noise with detail areas. Even so, we can see that edges and homogeneous areas of the image are distinguished with enough clarity, even when high noise appears.

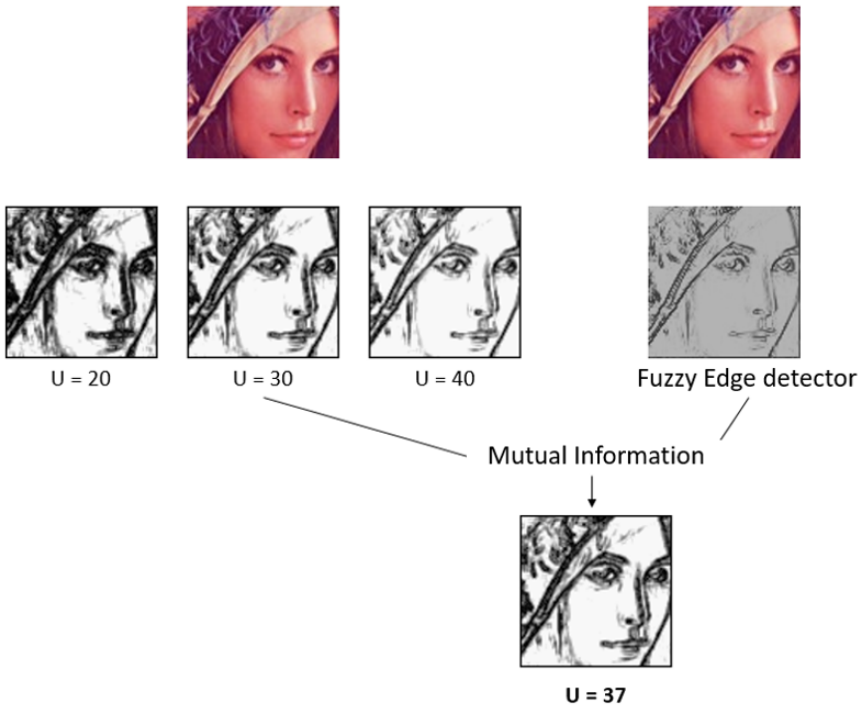
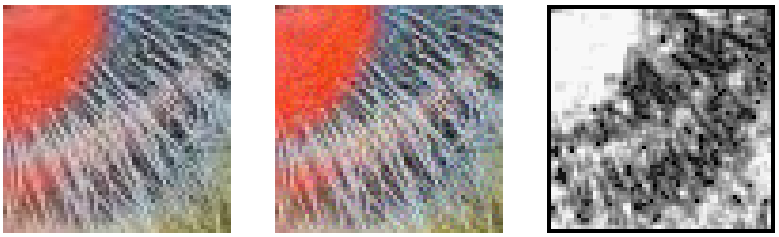
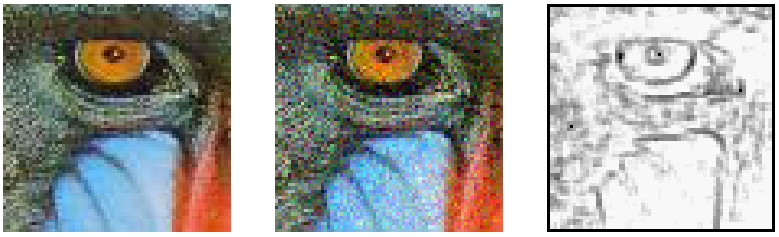


Fig. 4.12: Schedule of the setting threshold parameter.

Note also that the optimal threshold value for classification not only depends on the level of noise that it presents, but also on the image characteristics. A very homogeneous image with few edges will not require the same threshold as an image with a lot of detail. In the same way, we will not need a very high threshold when the different zones of the image present lot of contrast. However, but we will need to raise this parameter when we find more blurred edges or if they present less contrast. An example of this can be seen in the image of Parrots1 in Figure 4.13, where we can clearly see in the grayscale image the most contrasted edges, as for example those that delimit black and white areas. However, in areas with less contrasted edges, such as the area of the green head of the parrot and the green background, the differences are not so clear.



(a) Baboon (b) Baboon $\sigma = 10$ (c) Grayscale image



(d) Baboon2 (e) Baboon2 $\sigma = 20$ (f) Grayscale image

(g) Parrots1 (h) Parrots1 $\sigma = 30$ (i) Grayscale image

Fig. 4.13: Gray scale images, where the intensity of each pixel is proportional to $\mathbf{card}(L(H_{\mathbf{F}_0}))$, for images with different levels of noise ($\sigma = 10, 20$ y 30).

References

[1] R. Lukac, B. Smolka, K. Martin, K.N. Plataniotis, A.N. Venetsanopoulos, Vector Filtering for Color Imaging, *IEEE Signal Processing Magazine, Special Issue on Color Image Processing* 22 1 (2005) 74-86.

[2] Rother, C., Kolmogorov, V., and Blake, A. (2004, August). Grabcut: Interactive foreground extraction using iterated graph cuts. *In ACM transactions on graphics (TOG)* (Vol. 23, No. 3, pp. 309-314). ACM.

[3] Boykov, Y., and Kolmogorov, V. (2004). An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE transactions on pattern analysis and machine intelligence*, 26(9), 1124-1137.

[4] Li, Y., Sun, J., Tang, C. K., and Shum, H. Y. (2004, August). Lazy snapping.*ACM Transactions on Graphics (ToG)* (Vol. 23, No. 3, pp. 303-308). ACM.

[5] Boykov, Y., Veksler, O., and Zabih, R. (2001). Fast approximate energy minimization via graph cuts. *IEEE Transactions on pattern analysis and machine intelligence*, 23(11), 1222-1239.

[6] Veksler, O. (2008, October). Star shape prior for graph-cut image segmentation.*European Conference on Computer Vision* (pp. 454-467). Springer, Berlin, Heidelberg.

[7] Felzenszwalb, P. F., and Huttenlocher, D. P. (2004). *Efficient graph-based image segmentation. International journal of computer vision*, 59(2), 167-181.

[8] Morris, O. J., Lee, M. D. J., and Constantinides, A. G. (1986, April). Graph theory for image analysis: an approach based on the shortest spanning tree. *IEE Proceedings F (Communications, Radar and Signal Processing)* (Vol. 133, No. 2, pp. 146-152). IET Digital Library

[9] Zuo, Y., Wu, Q., Zhang, J., and An, P. (2017, July). Minimum spanning forest with embedded edge inconsistency measurement for color-guided depth map upsampling.*Multimedia and Expo (ICME), 2017 IEEE International Conference on* (pp. 211-216). IEEE.

[10] Bao, L., Song, Y., Yang, Q., Yuan, H., and Wang, G. (2014). Tree filtering: Efficient structure-preserving smoothing with a minimum spanning tree. *IEEE Transactions on Image Processing*, 23(2), 555-569.

[11] Zhang, F., Dai, L., Xiang, S., and Zhang, X. (2015). Segment graph based image filtering: fast structure-preserving smoothing.*Proceedings of the IEEE International Conference on Computer Vision* (pp. 361-369).

[12] Stawiaski, J., and Meyer, F. (2009, November). Minimum spanning tree adaptive image filtering.*Image Processing (ICIP), 2009 16th IEEE International Conference on* (pp. 2245-2248). IEEE.

[13] Koga, T., and Suetake, N. (2011, September). Structural-context-preserving image abstraction by using space-filling curve based on minimum spanning tree.*Image Processing (ICIP), 2011 18th IEEE International Conference on* (pp. 1465-1468). IEEE.

[14] Maes, F., Collignon, A., Vandermeulen, D., Marchal, G. and Suetens, P. (1997). *Multimodality image registration by maximization of mutual information.* IEEE transactions on Medical Imaging, 16(2), 187-198.

[15] Viola, P. and Wells III, W. M. (1997).*Alignment by maximization of mutual information.* International journal of computer vision, 24(2), 137-154.

[16] MathWorks. Fuzzy Logic Image Processing. https://es.mathworks.com/help/fuzzy/examples/fuzzylogic- image-processing.html

Part III

Applications of the model

Introduction

This part, consisting of two chapters, presents some applications to image processing tasks of the graph-based model explained in Part II. These chapters correspond to the following journal papers that, despite having some overlapping with other parts of this dissertation, are included entirely for the better understanding of the reader:

- Pérez-Benito, C., Morillas, S., Jordán, C., Conejero, J. A. (2018). A model based on local graphs for colour images and its application for Gaussian noise smoothing. *Journal of Computational and Applied Mathematics*, 330, 955-964.
- Pérez-Benito, C., Jordán, C., Conejero, J. A., Morillas, S. (2018). Graph-based methods for simultaneous smoothing and sharpening of color images. *Journal of Computational and Applied Mathematics*, 350, 380-395.

The first chapter presents a characterization of the pixels of the image based on the features of the local graphs that are constructed for each pixel. This characterization allows to distinguish homogeneous and detailed areas even in the presence of noise which has led to the design of a hybrid filter of color images. The presented filter removes the Gaussian noise from color images preserving edges and details of the image and thus avoiding the introduction of blur.

The second chapter shows more advanced application of the model, made possible through taking advantage of the features of the local graph defined for each pixel. The image processing task addressed concerns the simultaneous smoothing and sharpening of color images. These two operations have an opposite nature which makes its simultaneous addressing challenging. The method proposed in this chapter achieves to remove the Gaussian noise of an image while improving the sharpness of details and edges. Not only the proposed method achieves the goal but also does it in a single step, avoiding the drawbacks of two-step solutions.

5 Contribution (i)

Pérez-Benito, C., Morillas, S., Jordán, C., Conejero, J. A. (2018). A model based on local graphs for colour images and its application for Gaussian noise smoothing. *Journal of Computational and Applied Mathematics*, 330, 955-964.

Abstract

In this paper, a new model for processing colour images is presented. A graph is built for each image pixel taking into account some constraints on links. Each pixel is characterized depending on the features of its related graph, which allows to process it appropriately. As an example, we provide a characterization of each pixel based on the link cardinality of its connected component. This feature enables us to properly distinguish flat image regions in relation to edge and detail regions. According to this, we have designed a hybrid filter for colour image smoothing. It combines a filter able to properly process flat image regions with another one that is more appropriate for details and texture. Experimental results show that our model performs appropriately. We also see that our proposed filter is competitive with respect to state-of-the-art methods. It is close closer to the corresponding optimal switching filter respect to other analogous hybrid method.

5.1 Introduction

Image denoising is a topic that has been extensively studied in computer vision and digital image processing fields. The denoising (or filtering) step is essential for almost every computer vision system because noise can significantly affect the visual quality of images, as well as the performance of most image processing tasks. Also, in the last years the use of colour images has gained much attention within the computer vision field and therefore colour image denoising has become an important research topic [1].

Among the different sources of noise in digital imaging, probably the most common one is the so-called *thermal noise*, which is due to CCD sensor malfunction. This kind of noise is modeled as an additive white Gaussian noise.

So that, the presence of thermal (or Gaussian) noise can be simulated by adding random values from a zero-mean Gaussian distribution to the original values of each image channel independently. The standard deviation σ of the Gaussian distribution characterizes the noise intensity [2]. Many methods for reducing image Gaussian noise in colour images have been proposed in the literature. We will review some of them.

The earliest approaches for Gaussian noise smoothing were based on linear approaches. These methods, such as the *Arithmetic Mean Filter* (AMF), see for instance [2], are able to suppress noise because they take advantage of its zero-mean property. However, they tend to blur edges and texture significantly. This fact motivated the development of many nonlinear methods that try to overcome these drawbacks by detecting image edges and details. This is intended for smoothing there less than in the rest of the image.

Within nonlinear methods, many of them use averaging to take advantage of the zero-mean property of the noise. This class includes the well-known *Bilateral Filter* (BF) [6] and its variants [7]-[11]. Besides, in [12, 13] the authors use an averaging operation which is restricted to the *(fuzzy) peer group* members for each image pixel. Other methods are developed using fuzzy logic or soft switching methods, such as those in [14]-[23]. Several methods based on different optimizations of weighted averaging are proposed in [24]-[27]. Another important family of filters are partition based filters [27]-[29], that classify each pixel to be processed into several signal activity categories which, in turn, are associated to appropriate processing methods. Other filters follow a regularization approach [30]-[40] based on the minimization of appropriate energy functions by means of Partial Differential Equations (PDEs). Wavelet theory has also been used to design image filtering methods [41]-[50]. The combination of collaborative non-local means and wavelet filtering is proposed in [51, 52], and a method using the wavelet transformation and data regularization is proposed in [53]. Other recent methods make use of a combination of image analysis techniques for image segmentation followed by an appropriate smoothing of each image region [54]-[56]. More recently, methods using graph modeling colour images have provided competitive filtering solutions as [57, 58].

Despite that many works have consider this question up to date, the problem remains open. Recently, very few works have been published (just 2 articles [58, 40] in prestigious journals in four years, 2013-2016). This is due, in part, to new image models being needed to develop new filtering solutions. Therefore, in this paper we propose a new model for colour images which is based on graph theory and vector processing. In the model, a local graph is built for each image pixel taking into account some constraints on links. Each pixel is characterized depending on the features of its related graph so that it can be properly processed. As an application of the model, we provide a characterization of each pixel based on the link cardinality of its connected component. This feature is able to properly distinguish flat image

regions respect to edge and detail regions. According to this characterization, we have designed a hybrid filter for colour image smoothing that combines a filter able to properly process flat image regions with another one more appropriate for details and texture. This approach follows the methodology in [23, 57]. The experimental results show that the proposed filter is competitive with analogous filters and closer to the corresponding optimal soft-switching filter.

The paper is organized as follows: Section 6.2 described the local graph model for colour images, Section 5.3 details the pixel characterization, and Section 6.3 introduces the hybrid filter. Finally, experimental results and conclusions are given in Sections 6.4 and 5.5.1, respectively.

5.2 Image model based on local graphs

A graph G is defined as a finite nonempty set $V(G)$ of objects called *vertices* and a set $L(G)$ of unordered pairs of distinct vertices of G which, in order to avoid confusion with the image processing terminology, we will call *links* instead of edges, as it is common practice. Two vertices u and v joined by a link (u, v) are said to be *adjacent*. When each link (u, v) has an associated value $w(u, v)$, we say that the graph is *weighted*.

A graph H is called a subgraph of G if $V(H) \subseteq V(G)$ and $L(H) \subseteq L(G)$. A walk W from a node v_0 to a node v_l in a graph is a sequence of vertices say $v_0, v_1, \ldots, v_l$ where $(v_{i-1}, v_i) \in L(G)$, $0 < i \leq l$.

A graph is *connected* if for every pair v_i, v_j of distinct vertices there is a walk from v_i to v_j.

A connected component of a nondirected graph G is a connected subgraph H of G such that there is not a connected subgraph of G that contains H strictly.

For a colour image $\mathbf{F}$, which is represented in the RGB colour space, we build a graph-based model for each pixel in $\mathbf{F}$. In doing so, we take the neighbours around each image pixel $\mathbf{F}_0$ in a window centered on it of size $N \times N$ where $N = 2n + 1$ and $n = 1, 2, \ldots$. The rest of the neighbour pixels in the window are denoted as $\mathbf{F}_i, i = 1, \ldots, N^2 - 1$. The central pixel $\mathbf{F}_0$ is in turn defined by the tern (F_0^R, F_0^G, F_0^B) of its three RGB colour components. In the following we will use $n = 1$ as it is common practice in colour image filtering.

Given a pixel $\mathbf{F}_0$, we define a local weighted graph $G_{\mathbf{F}_0}$ where $V(G_{\mathbf{F}_0}) = \{\mathbf{F}_i,\, i = 0, \ldots, N^2 - 1\}$ and $L(G_{\mathbf{F}_0}) = \{(\mathbf{F}_i, \mathbf{F}_j),\, i \neq j,\, ||\mathbf{F}_i - \mathbf{F}_j||_2 < \mathcal{U}\}$. That is, a link exists between pixel $\mathbf{F}_i$ and $\mathbf{F}_j$, $i \neq j$, if the euclidean distance between their colour vectors is lower than a certain threshold $\mathcal{U}$. If such a link exists, its weight is $w(\mathbf{F}_i, \mathbf{F}_j) = ||\mathbf{F}_i - \mathbf{F}_j||_2$, where $||\cdot||$ stands for the Euclidean norm, see the example in Figure 1.

The value of $\mathcal{U}$ critically influences the structure of each local graph since it determines the connected component of $G_{\mathbf{F}_0}$ that contains the node $\mathbf{F}_0$,

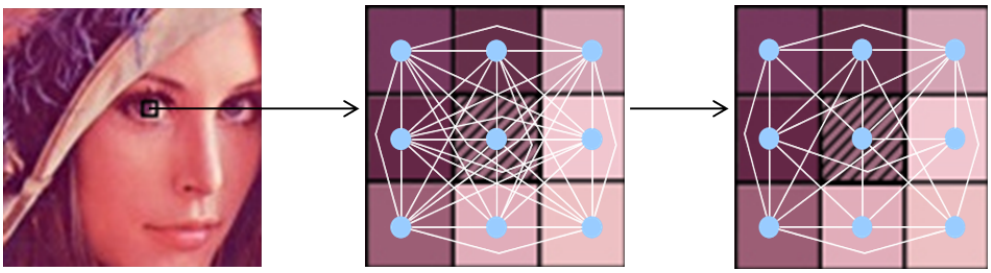


Fig. 5.1: Example of 3 window in an image (left) and its associated graph with all the links (center), and with the links lower than the threshold $\mathcal{U}$ (right) that is $G_{\mathbf{F}_0}$.

noted as $H_{\mathbf{F}_0}$. This connected component will play an important role in order to classify the different regions of the image into flat or detail/texture regions. We will discuss extensively the adjustment of the threshold $\mathcal{U}$ in the following section. Our global image model is the composition of all local graphs that characterize each image pixel.

5.3 A characterization of colour image pixels for smoothing

As an example of application of our model we aim to develop a procedure for smoothing colour images. To this end, it is critical to distinguish flat image regions in front of edges and details. This is because optimal smoothing needs to process differently flat regions, where smoothing can be more aggressive, from texture and detail regions, where smoothing should be done with special care. So, we need to devise a characterization based on our model to make such a classification.

We have seen that the feature that better characterizes whether a pixel $\mathbf{F}_0$ belongs to a flat or edge/detail region is the cardinal of the links set of its connected component, $\mathbf{card}(L(H_{\mathbf{F}_0}))$. Lower cardinality is associated to texture, edges and details whereas higher values correspond to flat image regions, as we can see if we compare the images in Figure 5.3, that were created assigning grayscale image levels proportional to $\mathbf{card}(L(H_{\mathbf{F}_0}))$, with the corresponding original images in Figure 5.3, that were created assigning grayscale image levels proportional to $\mathbf{card}(L(H_{\mathbf{F}_0}))$, with the corresponding original images shown in Figure 5.2.

However, for this characterization to be as accurate as possible it is critical to properly set the value of $\mathcal{U}$ for each input image. We have applied a method for this as follows.

(a) Lenna (b) Parrots (c) Statue (d) Peppers

Fig. 5.2: Set of training images

Fig. 5.3: Grayscale image where intensity of each pixel is proportional to $\mathbf{card}(L(H_{\mathbf{F}_0}))$.

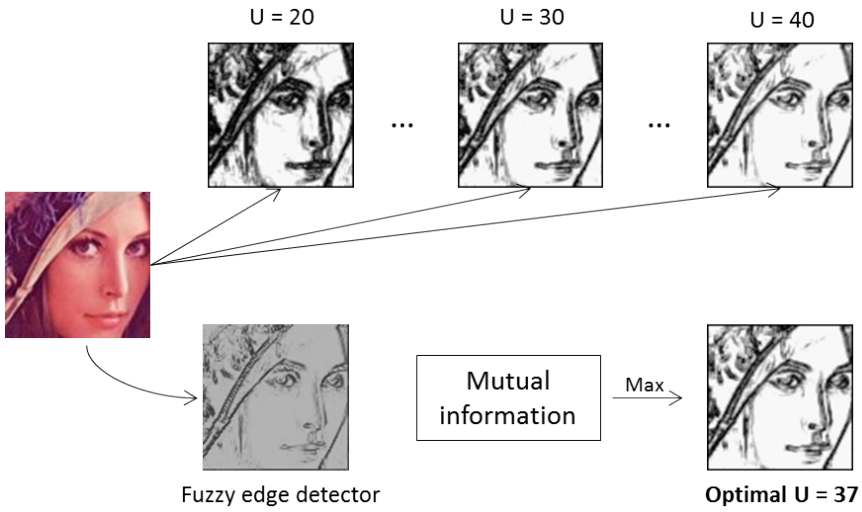


Fig. 5.4: Scheme of the method for set the optimal threshold $\mathcal{U}$

5.3.1 Adjustment of $\mathcal{U}$ parameter

The role of $\mathcal{U}$ is to avoid that very different pixels in the image were connected. In the context of image smoothing, we have to find a setting that is robust to the presence of noise or at least we need to adapt it to the density of

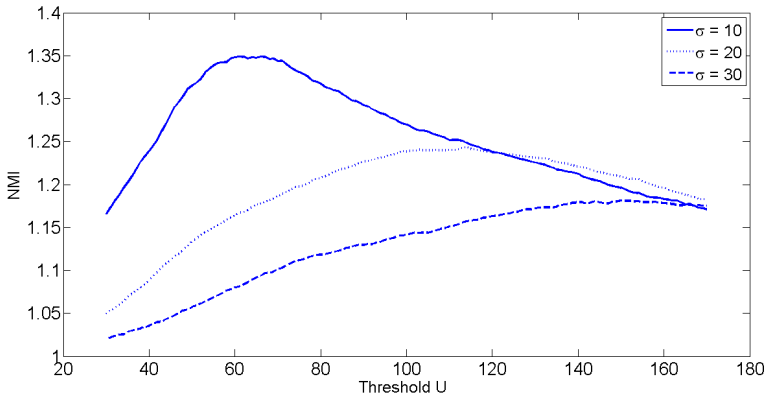


Fig. 5.5: NMI as a function of $\mathcal{U}$ for Parrots with different levels of noise

contaminating noise. Given that the feature that better characterizes whether a pixel $\mathbf{F}_0$ belongs to a flat or detail region is $\mathbf{card}(L(H_{\mathbf{F}_0}))$, we focus the adjustment of $\mathcal{U}$ to maximize the correlation between $\mathbf{card}(L(H_{\mathbf{F}_0}))$ and the presence of edges/texture.

Therefore, we first have taken the four training colour images in Figure 5.2 and for each of them we have obtained a groundtruth image of edges by means of the fuzzy edge detection method [64] as it is implemented in MATLAB© R2016b.

Secondly, for each noise free training image we have computed the value of $\mathcal{U}$ that maximizes the images mutual information (NMI) [62, 63] between the grayscale image obtained with the $\mathbf{card}(L(H_{\mathbf{F}_0}))$ of each pixel and the corresponding groundtruth image of the first step. Then, using each optimal value of $\mathcal{U}$ and the values $\mathbf{card}(L(H_{\mathbf{F}_0}))$ we obtain four edge/texture reference images that we use in the next step.

Thirdly, since our method will process noisy images with unknown noise variance, it would be desirable to have robustness against noise or at least adaptiveness to noise. So that, we have contaminated the training colour images with different densities of additive white Gaussian noise ($\sigma \in \{10, 20, 30\}$) according to the model in [2]. For each of the 12 noisy images we obtain the value of $\mathcal{U}$ that maximizes the NMI between the grayscale image obtained with the $\mathbf{card}(L(H_{\mathbf{F}_0}))$ of each pixel and the corresponding edge/texture reference image of the second step. It can be seen in Figure 5.5 that the higher the image noise is, the greater the optimal threshold is, too. Finally, we have conducted a linear regression analysis over all optimal $\mathcal{U}$ to be able to appropriately set $\mathcal{U}$ for any input image. We have also used an estimation of the standard deviation of the noise, $\hat{\sigma}$, in the input image, which is obtained using the method in [61] (we average the estimate in each

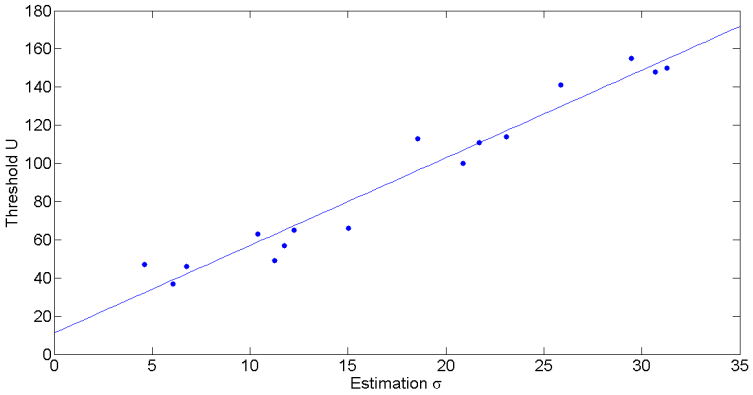


Fig. 5.6: Regression bewteen the estimation of the noise and the optimal threshold

of the RGB channels). The regression, that can be seen graphically in Figure 5.6, concludes that we can safely set $\mathcal{U}$ as

$$\mathcal{U} = 4.59\hat{\sigma} + 11.16, \tag{5.1}$$

given that correlation coefficient r equals 0.9187. The scheme in Figure 5.4 summarizes the procedure applied.

5.4 Proposed hybrid smoothing method

Recent smoothing methods commonly present the drawback that, as the higher the noise in the image is, the more confused is the noise in homogeneous regions with the image structure that should be preserved. So that, it cannot be properly reduced.

There are some filtering structures more suitable for smoothing, and others more powerful for preserving borders. We take, for instance, one filter of each type: AMF to process image flat regions and the nonlinear method called *Fuzzy Noise Reduction Method* (FNRM) [15] FNRM for the rest of the image. We propose to combine them following the reasoning in [57].

The switching between AMF and FNRM is performed in a soft fashion so that when the class of the image pixel is not clearly determined the results of both methods are combined. The proposed filter follows the idea behind the *Soft-Switching Graph Denoising* (SSGD) method in [57], but using our new model and characterization based on $\mathbf{card}(L(H_{\mathbf{F}_0}))$. In addition, notice that although we have used the AMF and FNRM, any other methods can be used within the same structure and analogous improvements are expected.

The combination of the aforementioned methods is performed as follows: Let us consider a pixel $\mathbf{F}_0$. Since $L(H_{\mathbf{F}_0})$ is a connected component of $G_{\mathbf{F}_0}$,

the parameter $\mathbf{card}(L(H_{\mathbf{F}_0}))$ takes discrete values, between 0 and $\binom{N^2}{2}$. In our case $N = 3$ and

$$\mathbf{card}(L(H_{\mathbf{F}_0})) \in \{0, \ldots, 36\}, \tag{5.2}$$

We classify the image pixels of any image into one of these 37 different categories, one for each admissible value of $\mathbf{card}(L(H_{\mathbf{F}_0}))$. In this way, we build $\beta = \{\beta_1, \ldots, \beta_{37}\}$, with $0 \leq \beta_i \leq 1$. If $\mathbf{card}(L(H_{\mathbf{F}_0})) = i - 1$, we make $\mathbf{F}_0$ in correspondence with β_i. These values $\beta_i's$ shall determine the soft-switching between AMF and FNRM to process each image pixel.

This new method will be called a *Soft-Switching Local Graph Denoising* method (SSLGD). For each image pixel $\mathbf{F}_0$, if $\mathbf{card}(L(H_{\mathbf{F}_0})) = i$, the output of SSLGD is

$$SSLGD_{out}(\mathbf{F}_0) = (1 - \beta_i)AMF_{out}(\mathbf{F}_0) + \beta_i FNRM_{out}(\mathbf{F}_0) \tag{5.3}$$

where $\beta_i \in [0, 1]$.

Notice that when $\beta_i = 1$ the $SSLGD$ method behaves as the $FNRM$, and when $\beta_i = 0$ it coincides with the AMF. Thus, the value of β_i should depend on the nature of the pixel under process. Therefore, if the pixel $\mathbf{F}_0$ belongs to an homogeneous region of the image β_i should be large (close to 1), otherwise, β_i should be lower (close to 0).

In this method it is critical to find a setting for the values β_i in equation (5.3) in order to obtain the better combination between AMF and FNRM. To do it we use the ascending gradient method for maximizing the *Peak Signal to Noise Ratio* (PSNR) [2] between the filter output and the original noise free image. In this optimization we have used as initial vector $B^0 = (\beta_i^0)$, where $\beta_i^0 = 1$, and a step $\delta = 0.05$. We find the optimization for the 4 training images each of them contaminated with 3 different densities of noise ($\sigma \in \{10, 20, 30\}$) which provides 12 optimized sets of β_i's.

Now, by using these sets we compute three default sets of $\beta = (\beta_i)_{1 \leq i \leq 37}$'s, one for low noise ($\beta^{10}, \sigma = 10$), another for medium noise ($\beta^{20}, \sigma = 20$) and a third one for high noise ($\beta^{30}, \sigma = 30$). To process an input image where noise is unknown we use the noise estimation $\hat{\sigma}$ and we choose among $\beta^{10}, \beta^{20}, \beta^{30}$ the set with superscript closest to $\hat{\sigma}$, that we call it now $\bar{\beta}$. The choice of $\bar{\beta}$ for each image is determined by the noise estimation. In this sense, if the estimation of noise level is the same for two images, they will be processed analogously, but two images having the same real noise level may have different estimation of noise. As we have said before, the noise is unknown in general, thus using a noise estimator provide us a more realistic and robust approach when processing any image.

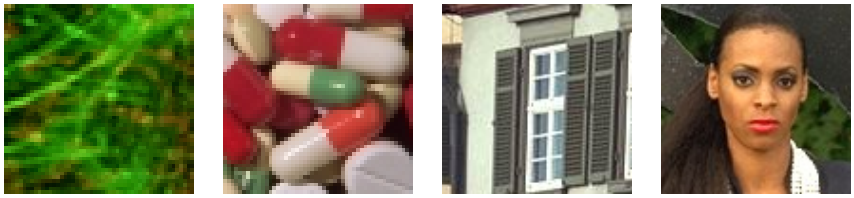
Although it is true that using more images makes better the learning, the inclusion of more images will adjust the values in a unnecessary precision since the step of β in the optimization is $\delta = 0.05$ and then the differences in the ranges in which we move are imperceptible. For this reason and taking

into account that our goal is to find an appropriate general robust setting that could be used to process any unknown image, we consider four images, with different structures that provide us enough information for set the beta values in a general way.

Notice that we have chosen a window size 3×3. According to previous works [65, 66], using $N > 3$ results in higher noise smoothing capability but much more blurred images that make increasing the window not a good choice in general. If the interest is to increase noise reduction capability, it has been reported to be a much better choice to apply several iterations of the same method, that is, filtering the output image again and again until convergence is reached.

5.5 Experimental results

In this section, we compare the performance of the SSLGD filter respect to other filters with the aim of validating the parameter settings. In Figure 6.10, we show the validation set of images. We have added Gaussian noise with standard deviations $\sigma \in \{10, 20, 30\}$ to them, obtaining an experimental set of 12 images.



(a) Micro (b) Pills (c) Window (d) img58

Fig. 5.7: Images used for the validation.

We process all 12 images with SSLGD using two different parameter setting for $\mathcal{U}$ and β: one with the optimal settings for the particular image and noise $\mathcal{U}_{op}, \beta_{op}$ which we denote by $SSLGD_{\mathcal{U}_{op},\beta_{op}}$, and another with the estimated $\mathcal{U}_e$ and $\bar{\beta}$ which we denote by $SSLGD_{\mathcal{U}_e,\bar{\beta}}$. We compare the performance for the experimental set of images with respect to the methods AMF, FNRM, and SSGD, which is a method following the same structure of SSLGD. In addition, we compare with the optimal hybrid method associated to SSLGD and SSGD that we call *Optimal Soft Switching* (OSS), and which is defined as the best combination between AMF and FNRM, defined for each pixel $\mathbf{F}_0$ by:

$$OSS_{out}(\mathbf{F}_0) = \alpha_i AMF_{out}(\mathbf{F}_0) + (1-\alpha_i) FNRM_{out}(\mathbf{F}_0) \qquad \alpha_i \in [0,1] \quad (5.4)$$

where $\alpha_i = argmin_{[0,1]}||\mathbf{F}_0 - OSS_{out}(\mathbf{F}_0)||, i \in \{0, 1, \ldots, 36\}$ can be easily derived analytically if the original image $\mathbf{F}$ is known.

As figures of merit for objective evaluation we have used the PSNR, SSIM [60], and the *Fuzzy Colour Structural Similarity* [59], denoted by FCSS. These latter two methods have proved to correlate with human perception better than PSNR.

	σ	**10**			**20**			**30**		
		PSNR	**FCSS**	**SSIM**	**PSNR**	**FCSS**	**SSIM**	**PSNR**	**FCSS**	**SSIM**
Micro	AMF	28.430	0.890	0.775	27.147	0.889	0.694	25.580	0.884	0.606
	FNRM	31.334	**0.943**	0.866	27.964	**0.925**	0.741	25.216	0.902	0.617
	SSGD	**31.34**	**0.943**	**0.867**	28.06	**0.925**	0.746	25.63	0.907	0.634
	$SSLGD_{\beta_{op},\mathcal{U}_{op}}$	**31.34**	**0.943**	**0.867**	**28.36**	0.920	0.752	26.05	**0.908**	0.648
	$SSLGD_{\bar{\beta},\mathcal{U}_e}$	31.154	0.936	0.865	28.333	0.918	**0.754**	**26.097**	0.907	**0.65**
	OSS	**32.218**	**0.95**	**0.891**	**29.788**	**0.942**	**0.810**	**27.638**	**0.933**	**0.728**
Pills	AMF	25.913	0.888	0.885	25.211	0.882	0.856	24.139	0.874	0.809
	FNRM	32.417	0.944	0.962	28.143	0.919	0.901	25.138	0.891	0.825
	SSGD	32.62	0.945	0.966	28.89	0.925	0.921	**26.59**	0.907	**0.862**
	$SSLGD_{\beta_{op},\mathcal{U}_{op}}$	**32.69**	**0.946**	**0.967**	**28.97**	**0.926**	**0.923**	26.28	**0.909**	0.864
	$SSLGD_{\bar{\beta},\mathcal{U}_e}$	32.26	0.943	0.965	28.70	0.923	0.919	26.14	0.907	0.860
	OSS	**34.433**	**0.961**	**0.978**	**30.81**	**0.947**	**0.948**	**28.07**	**0.933**	**0.903**
Window	AMF	22.36	0.832	0.715	22.00	0.834	0.689	21.42	0.833	0.650
	FNRM	31.13	0.930	0.946	27.46	0.912	0.878	24.59	0.889	0.802
	SSGD	31.14	0.929	0.94	27.4	0.91	0.873	24.65	0.892	0.8
	$SSLGD_{\beta_{op},\mathcal{U}_{op}}$	**31.21**	**0.931**	**0.951**	**27.67**	**0.916**	**0.889**	**24.96**	**0.897**	**0.815**
	$SSLGD_{\bar{\beta},\mathcal{U}_e}$	30.34	0.92	0.95	26.93	0.906	0.879	24.47	0.888	0.792
	OSS	**32.87**	**0.946**	**0.965**	**29.59**	**0.936**	**0.925**	**27.05**	**0.925**	**0.7877**
img58	AMF	27.78	0.912	0.831	26.58	0.908	0.746	25.29	0.898	0.659
	FNRM	33.06	0.946	0.904	28.32	0.911	0.750	25.28	0.869	0.622
	SSGD	33.24	**0.958**	0.905	29.38	**0.931**	0.77	26.69	**0.911**	0.67
	$SSLGD_{\beta_{op},\mathcal{U}_{op}}$	**33.45**	0.947	0.9156	**29.42**	0.93	**0.804**	**26.77**	**0.911**	**0.70**
	$SSLGD_{\bar{\beta},\mathcal{U}_e}$	33.04	0.946	0.899	28.68	0.922	0.74	25.90	0.889	0.599
	OSS	**35.27**	**0.962**	**0.946**	**31.16**	**0.949**	**0.861**	**28.56**	**0.936**	**0.780**

Table 5.1: Results in terms of PSNR, SSIM and FCSS of the validation set.

	σ	10		20		30	
		PSNR	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**
Window	SSGD	36.78	0.989	33.55	0.972	31.67	0.949
	$SSLGD_{\bar{\beta},\mathcal{U}_e}$	37.66	0.989	33.64	0.971	31.11	0.951
img58	SSGD	39.51	0.977	36.88	0.965	33.96	0.945
	$SSLGD_{\bar{\beta},\mathcal{U}_e}$	40.33	0.982	35.04	0.941	31.74	0.900
Micro	SSGD	40.60	0.982	35.18	0.945	33.62	0.915
	$SSLGD_{\bar{\beta},\mathcal{U}_e}$	41.86	0.983	37.83	0.967	35.42	0.948
Pills	SSGD	38.44	0.991	35.27	0.983	33.02	0.974
	$SSLGD_{\bar{\beta},\mathcal{U}_e}$	39.83	0.992	36.44	0.985	34.123	0.978

Table 5.2: Results in terms of PSNR and SSIM comparing with OSS

From the results in Table 5.1 we can see that both in terms of PSNR, FCSS and SSIM the performance of $SSLGD_{\bar{\beta},\mathcal{U}_e}$ is very close to $SSLGD_{\beta_{op},\mathcal{U}_{op}}$. This means that the methods for parameter setting are performing appropriately. Also, we see that the performance of our method is competitive with SSGD and both SSGD and SSLGD are a little bit below OSS, which implies that the proposed method is competitive with respect to state-of-the-art methods.

To see which of SSGD or SSLGD is closer to the optimal OSS we have also computed the PSNR and SSIM for SSGD and SSLGD with respect to OSS. These results are shown in Table 5.5, where we can see that SSLGD is close to the optimal OSS, which is a strong point for our method and the model behind it. We consider that this comparation is more important than the one in Table 5.1, since the original image is irretrievable and OSS image is the maximum to what can be reached with this kind of hybrid filter.

In Figure 5.8 we can see the qualitative results of AMF, FNRM, SSGD and the new proposed filter SSLGD. They have been applied separately to the set of validation images with standard deviation $\sigma = 20$.

As it can be seen, on the one hand AMF smooths the noise fine but it blurs the image. On the other hand, we can see how in all the cases FNRM does not blur the image. However, it does not remove so much noise. SSGD improves this drawback combining both methods and achieving a good denoising without blurring the image. Finally, our propose method follows the line of SSGD, being very close to it. The differences between both methods can be appreciated in detail regions such as the blinds of Window, or the edges of the capsules of Pills.

Although the result of SSLGD is close to SSGD, it can be seen more globally as the result of SSLGD is closer to OSS image than the SSGD, what is the most important for us, being especially appreciated in the homogeneous regions.

5.5.1 Computational Complexity

We will analyze the computational complexity of the proposed method and SSGD for each pixel of the image. The number of operations for each pixel depends on the window NxN considered, a total of N^2 pixels in the window. Since both methods depend on two basic generic filters, one for homogeneous regions and other for borders regions, we will focus on the complexity of SSLGD and SSGD. We denote the computational complexity of the homogeneous-regions and detail-regions generic filters as $H(N^2)$ and $E(N^2)$, respectively.

In the proposed method, for each pixel we have to compute the distance between the central pixel and its neighbours, that is to say, a total of $\frac{N^2(N^2-1)}{2}$ distances, which means $\frac{9N^2(N^2-1)}{2}$ distances in total. Once the distances are calculated, we compare all of them with the fixed threshold in

order to compute $\mathbf{card}(L(H_{\mathbf{F}_0}))$. This would amount to a total of $\frac{N^2(N^2-1)}{2}$ comparisons. In this way, we have a total of $\frac{10N^2(N^2-1)}{2}$ comparisons. The value of $\mathbf{card}(L(H_{\mathbf{F}_0}))$ allows us to choose the corresponding β for combine appropriately the based filters. Thus SSLGD needs a number of operations of order $\mathcal{O}(N^2) + H(N^2) + E(N^2)$.

SSGD filter, as SSLGD do, needs to compute the distances between the central pixel and its neighbours ($\frac{9N^2(N^2-1)}{2}$ operations). Kruskal algorithm, whose computational cost is $\mathcal{O}\left(N^2 \log(N^2)\right)$, is applied twice in order to compute the maximum and minimum spanning tree by considering the mentioned distance. Minimum spanning tree's weight allows to compute the coefficient that will be used in the linear combination of the generic filters. The computational complexity of this method is therefore of order $\mathcal{O}(N^2 \log(N^2)) + H(N^2) + E(N^2)$.

The proposed SSLGD method is computationally more efficient than SSGD. SSLGD has the advantage of having fixed parameters *beta* for processing the pixels. In contrast, SSGD performs all the soft switching mechanism completely for every pixel, which increases the number of operations and the computational time.

Conclusions

In this paper, we have presented a new model based on local graphs for low level image processing. In the model, each pixel is associated to a graph whose features allow to characterize it. We show an application of the model for Gaussian noise smoothing which is based on using each pixel graph to decide whether a pixel belongs to a flat region or not. The model allows to distinguish appropriately flat regions and border regions in an image even in the presence of noise. Related to this classification a soft-switching filter is built by using a filter with good smooth capability in flat regions and another to smooth border regions. Also, parameters of the method have been analyzed and it has been proposed how to set them automatically for any input image, so that the filter is very easy to use.

Performance of the new proposed method, SSLGD, in terms of PSNR, SSIM and FCSS shows that it is competitive with respect to state-of-the-art methods, decreasing the computational complexity thanks to the global characterization of the parameters, which allows us to reduce the computational cost. Also, objective comparison with respect to the corresponding optimal hybrid filter claims that our method is closer to the optimal than another soft-switching filter with the same structure.

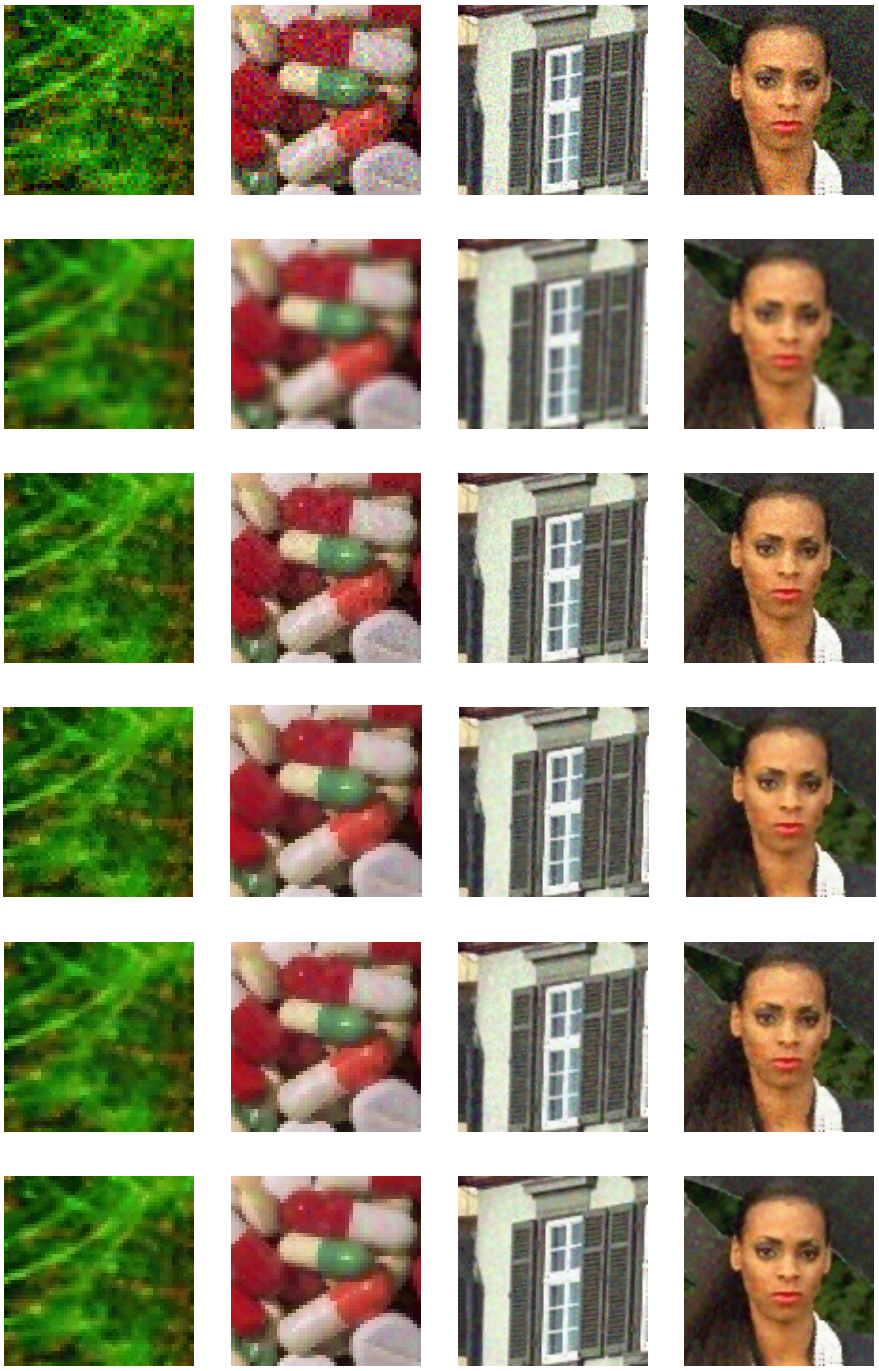


Fig. 5.8: Filtered image: first row the original images blurred with Gaussian noise with $\sigma = 20$, second row the filtered images using AMF, third row using FNRM, the fourth row with SSGD,the fifth row with SSGLD and finally OSS images.

References

[1] Celebi, M.E., Lecca, M., Smolka, B.: Color Image and Video Enhancement: *Springer-Verlag*, Berlin (2015).

[2] Plataniotis, K.N., Venetsanopoulos, A.N.: Color Image processing and applications. *Springer-Verlag*, Berlin (2000).

[3] Lukac, R., Smolka, B., Martin, K., Plataniotis, K.N., Venetsanopoulos, A.N.: Vector Filtering for Color Imaging, *IEEE Signal Processing Magazine, Special Issue on Color Image Processing*, 22, 74-86 (2005).

[4] Lukac, R., Plataniotis, K.N.: A taxonomy of color image filtering and enhancement solutions.*Hawkes, P.W. (ed) Advances in Imaging and Electron Physics*, 140, pp. 187-264. Elsevier Acedemic Press (2006).

[5] Buades, A., Coll, B., Morel, J.M.: Nonlocal image and movie denoising. *International Journal of Computer Vision*, 76, 123-139 (2008).

[6] Tomasi, C., Manduchi, R.: Bilateral filter for gray and color images. *Proceedings of IEEE International Conference Computer Vision*, 839-846 (1998).

[7] Elad, M.: On the origin of bilateral filter and ways to improve it. *IEEE Transactions on Image Processing*, 11, 1141-1151 (2002).

[8] Kao, W.C., Chen, Y.J.: Multistage bilateral noise filtering and edge detection for color image enhancement. *IEEE Transactionc on Consumer Electronics*, 51, 1346-1351 (2005).

[9] Garnett, R., Huegerich, T., Chui, C., He, W.: A universal noise removal algorithm with an impulse detector. *IEEE Transactions on Image Processing*, 14, 1747-1754 (2005).

[10] Morillas, S., Gregori, V., Sapena, A.: Fuzzy Bilateral Filtering for color images, *Lecture Notes in Computer Science*, 4141, 138-145 (2006).

[11] Zhang, B., Allenbach, J.P.: Adaptive bilateral filter for sharpness enhancement and noise removal. *IEEE Transactions on Image Processing*, 17, 664-678 (2008).

[12] Kenney, C., Deng, Y., Manjunath, B.S., Hewer, G.: Peer group image enhancement. *IEEE Transactions on Image Processing*, 10, 326-334 (2001).

[13] Morillas, S., Gregori, V., Hervás, A.: Fuzzy peer groups for reducing mixed Gaussian-impulse noise from color images. *IEEE Transactions on Image Processing*, 18, 1452-1466 (2009).

[14] Plataniotis, K.N., Androutsos, D., Venetsanopoulos, A.N.: Adaptive fuzzy systems for multichannel signal processing. *The Proceedings of the IEEE*, 87, 1601-1622 (1999).

[15] Schulte, S., De Witte, V., Kerre, E.E.: A fuzzy noise reduction method for colour images. *IEEE Transactions on Image Processing*, 16, 1425-1436 (2007).

[16] Shen, Y., Barner, K.: Fuzzy vector median-based surface smoothing. *IEEE Transactions on Visualization and Computer Graphics*, 10, 252-265 (2004).

[17] Lukac, R., Plataniotis, K.N., Smolka, B., Venetsanopoulos, A.N.: cDNA Microarray Image Processing Using Fuzzy Vector Filtering Framework, *Fuzzy Sets and Systems*, 152, 17-35 (2005).

[18] Smolka, B.: On the new robust algorithm of noise reduction in color images, *Computers & Graphics*, 27, 503-513 (2003).

[19] Van de Ville, D., Nachtegael, M., Van der Weken, D., Philips, W., Lemahieu, I., Kerre, E.E.: Noise reduction by fuzzy image filtering,*IEEE Transaction on Fuzzy Systems*, 11, 429-436 (2003).

[20] Schulte, S., De Witte, V., Nachtegael, M., Van der Weken, D., Kerre, E.E.: Histogram-based fuzzy colour filter for image restoration, *Image and Vision Computing*, 25, 1377-1390 (2007).

[21] Nachtegael, M., Schulte, S., Van der Weken, D., De Witte, V., Kerre, E.E.: Gaussian noise reduction in grayscale images, *International Journal of Intelligent Systems Technologies and Applications*, 1, 211-233 (2006).

[22] Schulte, S., De Witte, V., Nachtegael, M., Mélange, T., Kerre, E.E.: A New Fuzzy Additive Noise Reduction Method. *Lecture Notes in Computer Science*, 4633, 12-23 (2007).

[23] Morillas, S., Schulte, S., Mélange, T., Kerre, E.E., Gregori, V.: A soft-switching approach to improve visual quality of colour image smoothing filters. *Proceedings of Advanced Concepts for Intelligent Vision Systems ACIVS07, Lecture Notes in Computer Science*, 4678, 254-261 (2007).

[24] Lucchese, L., Mitra, S.K.: A new class of chromatic filters for color image processing: theory and applications,*IEEE Transactions on Image Processing*, 13, 534-548 (2004).

[25] Lee, J.A., Geets, X., Grégoire, V., Bol, A.: Edge-preserving filtering of images with low photon counts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30, 1014-1027 (2008).

[26] Russo, F.: Technique for image denoising based on adaptive piecewise linear filters and automatic parameter tuning, *IEEE Transactions on Instrumentation and Measurement*, 55, 1362-1367 (2006).

[27] Shao, M., Barner, K.E.: Optimization of partition-based weighted sum filters and their application to image denoising. *IEEE Transactions on Image Processing*, 15, 1900-1915 (2006).

[28] Ma, Z., Wu, H.R., Feng, D.: Partition Based Vector Filtering Technique for Suppression of Noise in Digital Color Images. *IEEE Transactions on Image Processing*, 15, 2324-2342 (2006).

[29] Ma, Z., Wu, H.R., Feng, D.: Fuzzy Vector Partition Filtering Technique for Color Image Restoration. *Computer Vision and Image Understanding*, 107, 26-37 (2007).

[30] Perona, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12, 629-639 (1990).

[31] Sroubek, F., Flusser, J.: Multichannel blind iterative image restoration, *IEEE Transactions on Image Processing* 12, 1094-1106 (2003).

[32] Hu, J., Wang, Y., Shen, Y.: Noise reduction and edge detection via kernel anisotropic diffusion. *Pattern Recognition Letters*, 29, 1496-1503 (2008).

[33] Li, X.: On modeling interchannel dependency for color image denoising. *International Journal of Imaging Systems and Technology, Special issue on applied color image processing*, 17, 163-173 (2007).

[34] Keren, D., Gotlib, A.: Denoising Color Images using regularization and correlation terms. *Journal of Visual Communication and Image Representation*, 9, 352-365 (1998).

[35] Lezoray, O., Elmoataz, A., Bougleux, S.: Graph regularization for color image processing. *Computer Vision and Image Understanding*, 107, 38-55 (2007).

[36] Elmoataz, A., Lezoray, O., Bougleux, S.: Nonlocal discrete regularization on weighted graphs: A framework for image and manifold processing.*IEEE Transactions on Image Processing*, 17, 1047-1060 (2008).

[37] Blomgren, P., Chan, T.: Color TV: total variation methods for restoration of vector-valued images. *IEEE Transactions on Image Processing*, 7, 304-309 (1998).

[38] Tschumperlé, D., Deriche, R.: Vector-valued image regularization with PDEs: A Common framework from different applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27, 506-517 (2005).

[39] Plonka, G., Ma, J.: Nonlinear regularized reaction-diffusion filters for denoising of images with textures, IEEE Transactions on Image Processing, 17, 1283-1294 (2007).

[40] Moreno, J.C., Prasath, V.B.S., Neves, J.C., Color image processing by vectorial total variation with gradient channels coupling. *Inverse Problems and Imaging* 10 (2), pp. 461-497 (2016)

[41] Melange, T., Zlokolica, V., Schulte, S., De Witte, V., Nachtegael, M., Pizurca, A., Kerre, E.E., Philips, W.: A new fuzzy motion and detail adaptive video filter- *Lecture Notes in Computer Science*, 4678, 640-651 (2007).

[42] De Backer, S., Pizurica, A., Huysmans, B., Philips, W., Scheunders, P.: Denoising of multicomponent images using wavelet least-squares estimators. *Image and Vision Computing*, 26, 1038-1051 (2008).

[43] Dengwen, Z., Wengang, C.: Image denoising with an optimal threshold and neighboring window- *Pattern Recognition Letters*, 29, 1694-1697 (2008).

[44] Schulte, S., Huysmans, B., Pizurica, A., Kerre, E.E., Philips, W.: A New Fuzzy-Based Wavelet Shrinkage Image Denoising Technique. *Proceedings of Advanced Conceptps for Intelligent Vision Systems ACIVS06, Lecture Notes in Computer Science*, 4179, 12-23 (2006).

[45] Pizurica, A., Philips, W.: Estimating the probability of the presence of a signal of interest in multiresolution single and multiband image denoising, *IEEE Transactions on Image Processing*, 15, 654-665 (2006).

[46] Scheunders, P.: Wavelet thresholding of multivalued images, *IEEE Transactions on Image Processing*, 13, 475-483 (2004).

[47] Sendur, L., Selesnick, I.W.: Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency. *IEEE Transactions on Signal Processing*, 50, 2744-2756 (2002).

[48] Balster, E.J., Zheng, Y.F., Ewing, R.L.: Feature-based wavelet shrinkage algorithm for image denoising, *IEEE Transactions on Image Processing*, 14, 2024-2039 (2005).

[49] Miller, M., Kingsbury, N.: Image denoising using derotated complex wavelet coefficients. *IEEE Transactions on Image Processing*, 17, 1500-1511 (2008).

[50] Zhang, B., Fadili, J.M., Starck, J.L.: Wavelets, Ridgelets, and Curvelets for poisson noise removal.*IEEE Transactions on Image Processing*, 17, 1093-1108 (2008).

[51] Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising by sparse 3D transform-domain collaborative filtering, *IEEE Transactions on Image Processing*, 16, 2080-2095 (2007).

[52] Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space. *Proceedings of the IEEE International Conference on Image Processing* ICIP2007 , 313-316 (2007).

[53] Hao, B.B., Li, M., Feng, X.C.: Wavelet iterative regularization for image restoration with varying scale parameter. *Signal Processing: Image Communication*, 23, 433-441 (2008).

[54] Zhao, W., Pope, A.: Image restoration under significat additive noise. *IEEE Signal Processing Letters*, 14, 401-404 (2007).

[55] Gijbels, I., Lambert, A., Qiu, P.: Edge-preserving image denoising and estimation of discontinuous surfaces- *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28, 1075-1087 (2006).

[56] Liu, C., Szeliski, R., Kang, S.B., Zitnik, C.L.,Freeman, W.T.: Automatic estimation and removal of noise from a single image, *IEEE Transactions on Pattern Analysis and Machine Intelligente*, 30, 299-314 (2008).

[57] Jordan C., Morillas S., Sanabria-Codesal E.: Colour image smoothing through a soft-switching mechanism using a graph model, *IET Image Processing*, 6 (9),1293-1298 (2012).

[58] Bao, L., Song, Y., Yang, Q., Yuan, H., Wang, G., Tree filtering: Efficient structure-preserving smoothing with a minimum spanning tree. *IEEE Transactions on Image Processing* 23 (2), 6665130, pp. 555-569 (2014).

[59] Grecova, S.,Morillas, S.: Perceptual similarity between color images using fuzzy metrics. *J. Vis. Commun. Image R.*, 34, 230-235 (2016).

[60] Wang, Z., Bovik, A. C., Sheikh, H. R., Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4), 600-612 (2004).

[61] Immerkaer, J.: Fast Noise Variance Estimation. *Computer Vision and Image Understanding*, 64 (2), 300-302 (1996)

[62] Viola, P., Wells, W.M.: Alignment by maximization of mutual information. *International Journal of Computer Vision* 24 (2), 137-154, (1997).

[63] Maes F., Collingnon A., Vandermeulen, G. Marchal G., P. Suetens P.: Multimodality image registration by maximization of mutual information. *IEEE Transactions on Medial Imaging* 16 (2), 187-198 (1997).

[64] MathWorks. Fuzzy Logic Image Processing. https://es.mathworks.com/help /fuzzy/examples/fuzzy-logic-image-processing.html. Retrieved on November, 29th 2016.

[65] Lukac, R., Plataniotis, K. N. (2006). A taxonomy of color image filtering and enhancement solutions. *Advances in imaging and electron physics* , 140, 188.

[66] Celebi, E., Lecca, M., Smolka, B. (Eds.). (2015). Color Image and Video Enhancement. Springer International Publishing.

6 Contribution (ii)

Pérez-Benito, C., Jordán, C., Conejero, J. A., Morillas, S. (2018). Graph-based methods for simultaneous smoothing and sharpening of color images. *Journal of Computational and Applied Mathematics*, 350, 380-395.

Abstract

In this work we introduce an image characterization of pixels based on local graphs that allows to distinguish different local regions around a pixel. This separation also permits us to develop a method for determining the role of each pixel in a neighborhood of any other, either for smoothing or for sharpening. Two methods for simultaneously conducting both processes are provided. Our solution overcome the drawbacks of the classic two steps sequential smoothing and sharpening process: enhancing details while reducing noise and not losing critical information. The parameters of the methods are adjusted in two different ways: through observers visual quality optimization and with an objective optimization criterion. The results show that our methods outperform other recent state-of-the-art ones.

6.1 Introduction

The use of digital images has grown over the last few years and is now present in almost every field, from domestic digital cameras to medical applications or artificial intelligence. This has led to a great growth of techniques devoted to improve the quality of images.

There are many factors that can affect the image quality, causing loss of information, poor visual quality, and difficulties in image processing. The presence of noise and the consequences of poor acquisition conditions, that make the image blurry or not well defined in the edges and/or textures, are the two most common causes of image degradation.

Noise may be introduced in digital images through different sources, but the most common one is a CCD sensor malfunction which introduces the so-called thermal noise. This kind of noise is modeled as an additive white

Gaussian noise, that can be simulated by adding random values from a zero-mean Gaussian distribution to the original values of each image channel independently. The noise intensity is characterized by the standard deviation of the Gaussian distribution.

Image smoothing, used to remove this kind of noise, has been an intensively studied problem in the image processing field for more than 25 years. The smoothing step is essential for almost every computer vision system since the noise can affect the performance of most image processing tasks. First approaches to solve this problem followed a linear approach such as the classical *arithmetic mean filter* or *Gaussian filter* [24]. However, these methods produced excessive smoothing near edges and details. This led to the development of several nonlinear approaches that later constituted not only methods for image smoothing but complex paradigms for image modelling with applications to many image processing tasks. The most popular frameworks are *anisotropic diffusion* [19], *bilateral filtering* [29], *mean shifting* [4], *scale-space* techniques [10], and *total variation* [27]. For more than 20 years, different filtering solutions have been proposed within these frameworks including a vast number of publications, which have been shown to share some commonalities [1]. More recently, the extension to color image smoothing has been studied [11] and new methods based on *Fourier transform*, *wavelet theory* [12, 34] *nonlocal means* [2], *collaborative filtering* [5], *fuzzy logic* [15, 23], *spatial-tonal* averages [30] and *graph models* [16] have been used in different smoothing solutions.

On the other hand, image sharpening is used to improve the definition of edges, texture, and details, which are of paramount importance for many image analysis applications such as segmentation or object detection. The *unsharping mask* method (UM) [25] and contrast enhancement techniques, such as the *histogram equalization* [9, 22, 31], or *linear contrast stretching* [32], are some recent and popular methods used for sharpening.

In practical applications, given that there are no pristine images, it is common to assume that they need some degree of both smoothing and sharpening to be appropriate for further processing. However, there is an interdependency of both operations since they both deal with high spatial frequencies in the images: smoothing can reduce or even remove small details or textures that a subsequent sharpening cannot recover properly; but also a sharpening can detect image noise as textures to be highlighted.

An intuitive approach to this problem is the use of methods that join two independent processing steps: a first step of smoothing to remove the noise, and a second step of sharpening to enhance the edges, or, in reverse order, a first step of sharpening and a second one of smoothing. Both approaches are easy ways of trying to achieve the goal, given the broad state-of-the-art in both smoothing and sharpening.

However, these approaches can also lead to some of the aforementioned problems. On the one hand, if we first apply a smoothing technique there

is a risk of losing detail or edge information that will not be recovered in the sharpening step. On the other hand, if we apply a sharpening method on a noisy image, we amplify the noise making more difficult the smoothing task, which may lead to over smooth the image or failing in reducing part of the noise. As a result, the application of these two complex processing steps could be ineffective for further applications and even inefficient.

A more efficient solution to address this problem is to consider a simultaneous perspective being able of sharpening image details while removing noise. This kind of solution can only be approached through nonlinear processes which are needed to locally adapt the operations to be done and apply appropriate intensities of smoothing and sharpening. However, this is not a simple task given the opposite nature of these two operations. Some authors have tried to address this problem, in general for gray-scale images, by using different approaches as those reviewed in [18]: *forward-and-backward diffusion* [26], *block-matching* and *3D filtering* [5], different reformulations of the classical bilateral filter such as the *adaptive bilateral filter* [33], the *guided image filtering* [20, 21], the *adaptive unsharping mask* [8] or *difference of Gaussians* [7]. Even though the state-of-the-art concerning smoothing or sharpening is very extensive, currently there are not many methods able to achieve both goals simultaneously. For a recent review of these methods we refer the reader to [18].

In this paper, we study how two nonlinear methods based on the computation of local graphs at each pixel provide enough information to simultaneously carry out the nonlinear process of smoothing the noise, while sharpening the edges and details. For every pixel, we consider a 3x3 window and we determine a weighted graph with the similarities between each pair of pixels in that window. Then, for the central pixel, the subgraph with the closer pixels to it is determined. In both methods, this subgraph will be used for smoothing and the subgraph given by the rest of the nodes will be used for sharpening. The nonlinear splitting of the local graph in these two subgraphs was already considered for defining a *smoothing soft switching filter* in [16]. Using an analogous model, we extend here this operation to simultaneous smoothing and sharpening of color images.

The paper is organized as follows: in Section 6.2 we describe our local graph based model for processing color images. In Section 6.3 two proposed techniques based on these local graphs for simultaneous sharpening and smoothing are explained. The quantitative and qualitative experimental results and conclusions are given in Section 6.4. Finally, in Section 5.5.1 we outline the conclusions.

6.2 Local graphs for color image modeling

In this section we briefly introduce the local graph-based model that allows us to characterize a pixel in a color image and the notation to be used throughout the paper.

A *graph* G is defined as a finite nonempty set $V(G)$ of objects, called *vertices*, and a set $L(G)$ of unordered pairs of distinct vertices of G which, in order to avoid confusion with the image processing terminology, we will call them *links* instead of edges, as it is common practice. Two vertices u and v joined by a link (u, v) are said to be *adjacent*. When each link (u, v) has an associated value $w(u, v)$, we say that the graph is *weighted*. A graph H is called a *subgraph* of G if $V(H) \subseteq V(G)$ and $L(H) \subseteq L(G)$. A *walk* W from a node v_0 to a node v_l in a graph is a sequence of vertices say $v_0, v_1, \ldots, v_l$ where $(v_{i-1}, v_i) \in L(G)$, $0 < i \leq l$. A graph is *connected* if for every pair v_i, v_j of distinct vertices there is a walk from v_i to v_j. A *connected component* of a nondirected graph G is a connected subgraph H of G such that there is no other connected subgraph of G that contains H, with its nodes and links, strictly.

Given a color image $\mathbf{F}$, we consider the neighbors around each image pixel $\mathbf{F}_0$ in a 3×3 supporting window centered on it. The rest of the neighbor pixels in the window are denoted as $\mathbf{F}_i, i = 1, \ldots, 8$, following a clockwise order. Each image pixel is represented by its three color components in the RGB space $\mathbf{F}_i = (F_i^R, F_i^G, F_i^B)$.

We define a local weighted graph $G_{\mathbf{F}_0} = (V(G_{\mathbf{F}_0}), L(G_{\mathbf{F}_0}))$ associated to any arbitrary pixel $\mathbf{F}_0$ and its associated 3×3 window, which makes our model to be inherently nonlinear in the spatial domain of the image. Each one of these graphs is defined as

$$V(G_{\mathbf{F}_0}) := \{\mathbf{F}_i,\, i = 0, \ldots, 8\}$$

$$L(G_{\mathbf{F}_0}) := \{(\mathbf{F}_i, \mathbf{F}_j),\, i \neq j,\, ||\mathbf{F}_i - \mathbf{F}_j||_2 < \mathcal{U}\}$$

with $||\cdot||$ standing for the Euclidean norm and $\mathcal{U}$ being a threshold, that is a key parameter of the model which, in turn, makes the model being nonlinear in the image range domain, as well. Last, if $(\mathbf{F}_i, \mathbf{F}_j) \in L(G_{\mathbf{F}_0})$, its weight will be denoted by $w(\mathbf{F}_i, \mathbf{F}_j) := ||\mathbf{F}_i - \mathbf{F}_j||_2$.

The parameter $\mathcal{U}$ is crucial in the definition of the graph $G_{\mathbf{F}_0}$. Depending on its value, $G_{\mathbf{F}_0}$ can have one or more connected components that, even in the presence of noise, will allow us to properly classify each pixel $\mathbf{F}_0$ as belonging to a flat or to a detail region [16]. When the values of $\mathcal{U}$ are low enough, the pixels in the connected component of the central pixel will permit us to smooth this central pixel acting as a low-pass filter. Links that are related to high values of the threshold $\mathcal{U}$ correspond to detail regions. In addition, the structure of the links and nodes remaining in each connected component of $G_{\mathbf{F}_0}$ permits to locally characterize the image, and then to use this information as an efficient edge detector [16, 17].

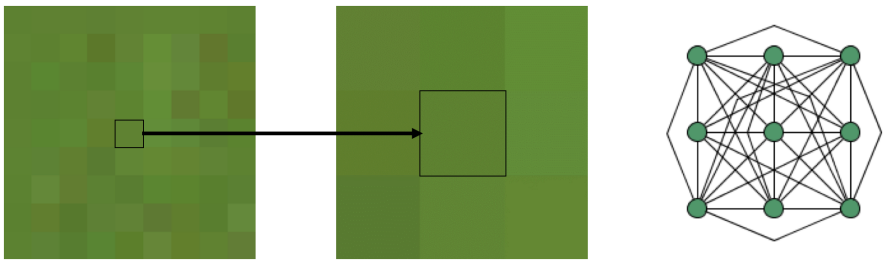


Fig. 6.1: Example of a homogeneous region, from left to right: flat region of the image, a zoom of a 3×3 window to be processed, and finally the graph associated to this window for $\mathcal{U} = 38$.

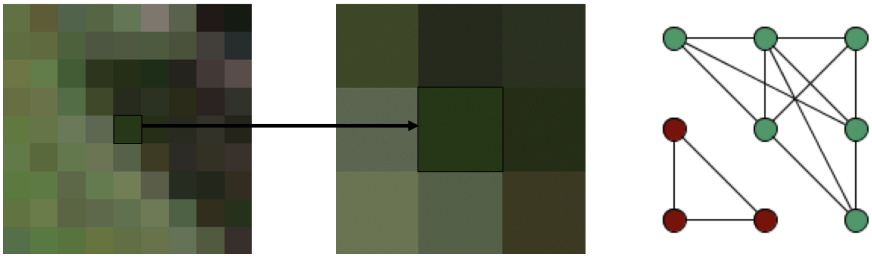


Fig. 6.2: Example of detail region, from left to right: detail region of the image, a zoom in the 3×3 window to be processed and finally, the graph associated to this window for $\mathcal{U} = 38$.

In Figure 6.1 we show an example of a homogeneous region of an image (left). We choose a 3×3 window around the central pixel (middle) and we compute the associated subgraph $G_{\mathbf{F}_0}$ for a threshold $\mathcal{U} = 38$, as we have already indicated (right). Here, we can see that all nodes belong to a unique connected component. This structure of the local graph indicates that the central pixel belongs to a flat zone of the image, without details or edges. On the other hand, in Figure 6.2, we show an example of a pixel in an edge region, where we can see two different connected components of the subgraph $G_{\mathbf{F}_0}$, one for each zone perfectly differentiated in the 3×3 image window.

The parameter $\mathcal{U}$ has been estimated through a linear regression analysis over all optimal $\mathcal{U}$ obtained for the images in a training dataset, see [16]. For those images, we also used an estimation of the standard deviation of the noise, $\hat{\sigma}$, that was obtained by using the method in [6]. The regression concluded that we can safely set $\mathcal{U} = 4.59\hat{\sigma} + 11.16$. For more details about this model, we refer the reader to [16], where it can be found further information regarding the choice of this parameter $\mathcal{U}$, and how the cardinal of the sets of vertices and edges of the connected component of the central pixel help us to characterize to which type of region the central pixel belongs. We will

see in the following section how these characterizations allow us to design a simultaneous smoothing and sharpening operation throughout the image.

6.3 Proposed methods for simultaneous smoothing and sharpening

The most basic spatial filters able to perform either smoothing or sharpening are *linear kernel-based filters*. There, each pixel is modified according to a linear combination of the pixels in its neighbourhood. If the coefficients of the convolution kernel are positive and their sum is 1, then the kernel represents a smoothing or low-pass filtering. It would smooth the homogeneous-like regions, as well as reduce sharp transitions in intensities. Also, it would reduce any high frequency white noise. However, if the coefficients of the convolution kernel are all negative, except the one corresponding to the pixel under process, and their sum is 0, then the kernel represents a sharpening filter that would highlight the local intensity contrast and, thus, it would sharpen edges and details.

Therefore, from the linear kernel point of view, the nature of these two operations is opposed. In addition, both methods apply the same linear combination all along the image, which determines the intensity of the smoothing or sharpening performed. Here, it relays the main inspiration of our proposals.

In the rest of the section, we explain how we use $G_{\mathbf{F}_0}$ to create nonlinear kernels able to simultaneously smooth and sharpen the image. Our methods are based on kernel type operations but the image processing is considered in a nonlinear way. That is, we do not keep constant the coefficients of the linear combination to be used for all image pixels. Indeed, we can not only switch from smoothing to sharpening kernels in different image regions, but we could also use positive and negative coefficients in the same kernel so that some pixels are used for smoothing and others for sharpening. So that, we are able to conduct both operations simultaneously. The key point behind this approach is to be able to determine at each image location which pixels, if any, should be used for smoothing and, consequently, for reducing noise, and which pixels, if any, should be used for sharpening. This information will be extracted from the local graph model built for each image pixel, which is, in turn, the key component of the method.

In Figure 6.3 we depict the points in the RGB space corresponding to the 3×3 windows appearing in Figures 6.1 and 6.2. On the one hand, in Figure 6.3 (left) all the points are grouped in a narrow region of the space, which correlates with the structure of the graph of Figure 6.1, where all nodes belong to a unique connected component. On the other hand, in Figure 6.3 (right) we can see two different groups of points in the color space corresponding to the connected components of the graph of Figure 6.2: one group with the points representing pixels similar to the central one, and another group with the three pixels that do not share that similarity with the central pixel.

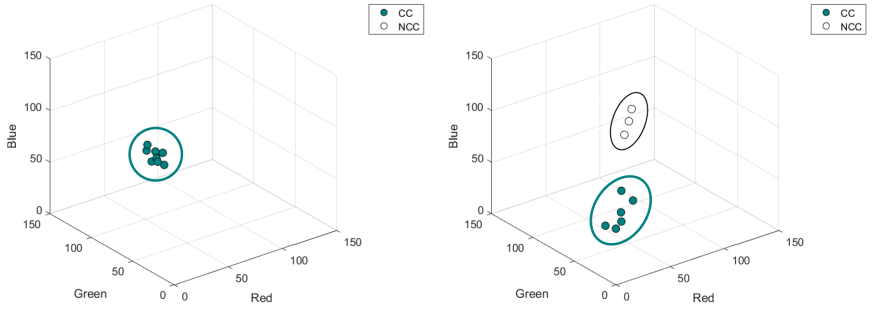


Fig. 6.3: On the left, the nine points in the RGB space corresponding to the 3×3 windows of Figure 6.1. On the right, the corresponding points obtained from Figure 6.2. The points in the connected component of the central pixel of each window are colored in green and denoted by (CC) and the others are colored in white and denoted by (NCC).

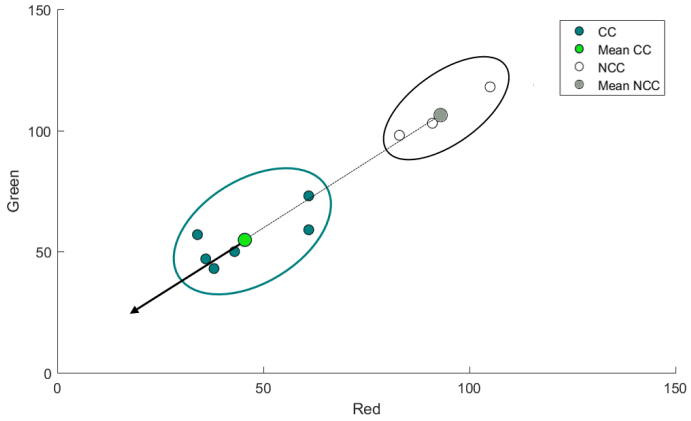


Fig. 6.4: Example in the Red-Green plane of the intuitive idea of the proposed technique

In our first method, for every pixel $\mathbf{F}_0$ we consider the pixels in the connected component of that pixel $\mathbf{F}_0$ in the graph $G_{\mathbf{F}_0}$, denoted by $CC_{\mathbf{F}_0}$, for smoothing, and the pixels in the subgraph of the other connected components, for sharpening. In Figure 6.4 we show this by using only two color channels: green and red. In this example we can achieve smoothing by replacing the value of the central pixel by a linear combination of the pixels of its connected component. Later, the result of the smoothing can be shifted into the color space in the direction of the arrow in order to increase its difference in relation to the pixels of the other connected component of this

local graph. The distance shifted can be used for determining the amount of sharpening achieved in the process.

We present a detailed description of our first proposed method, that we call as *Graph Method for Simultaneous Smoothing and Sharpening* (GMS^3). For each pixel $\mathbf{F}_0$ in the image, we build the local graph $G_{\mathbf{F_0}}$, and determine the pixels in the connected component of the pixel $\mathbf{F}_0$, $V(CC_{\mathbf{F}_0})$. Then, we can first make the smoothing operation by computing a smoothed version of the central pixel, $\mathbf{F}_0^S$, as

$$\mathbf{F}_0^S = \frac{\sum\limits_{i \in V(CC_{\mathbf{F}_0})} e^{-\frac{\|\mathbf{F}_i - \mathbf{F}_0\|_2}{2\alpha^2}} \mathbf{F}_i}{\sum\limits_{i \in V(CC_{\mathbf{F}_0})} e^{-\frac{\|\mathbf{F}_i - \mathbf{F}_0\|_2}{2\alpha^2}}} \tag{6.1}$$

where $\alpha > 0$ is a parameter that controls the smoothing effect. Here, we have assigned a weight to each pixel according to its distance to $\mathbf{F}_0$, giving greater weights to the nearest pixels to it, and thus achieving a nonlinear kernel-based smoothing adapted to the local information around the pixel.

Second, the sharpening operation, also nonlinear kernel-based, is made with the pixels outside of the connected component of $\mathbf{F}_0$, if there exists. In this case, to perform the sharpening we compute the value $\mathbf{F}_0^{GMS^3}$ as

$$\mathbf{F}_0^{GMS^3} := \mathbf{F}_0^S - \lambda \mathbf{v} \qquad \text{being} \qquad \mathbf{v} = \frac{\sum\limits_{i \notin V(CC_{\mathbf{F}_0})} \left(\mathbf{F}_i - \mathbf{F}_0^S\right)}{9 - \text{card}(V(CC_{\mathbf{F}_0}))}, \tag{6.2}$$

where $\lambda \in [0, 1]$ is a parameter for controlling the sharpening effect. In this last operation, we compute the mean vector of the differences between the central pixel and their dissimilar pixels. This is intended for increasing locally the contrast, which is expected to sharpen the border/detail features and to improve image quality. In fact, the use of the mean could also palliate the effect of noise in the pixels different to the central pixel. If for some channel of $\mathbf{F}_0^{GMS^3}$ the value lays outside the range of $[0, 255]$, we set it to the corresponding extreme value, 0 or 255.

Finally, we also consider of particular interest to propose as an alternative, a slight modification of the aforementioned method that consists on considering the normalized vector in equation (6.2). In this way, we would only consider the vector direction and, thus, the sharpening will be independent of the initial edge contrast. In other words, we would enhance all the edges equally, instead of enhancing edges proportionally to their initial contrast. We name this variation as the *Normalized Graph-Method for Simultaneous Smoothing and Sharpening* ($NGMS^3$) and its output will be obtained as

$$\mathbf{F}_0^{NGMS^3} = \mathbf{F}_0^S - \lambda \frac{\mathbf{v}}{\| \mathbf{v} \|_2}. \tag{6.3}$$

with $\mathbf{v}$ defined as in (6.2). Here, the values of λ can be taken greater than 1.

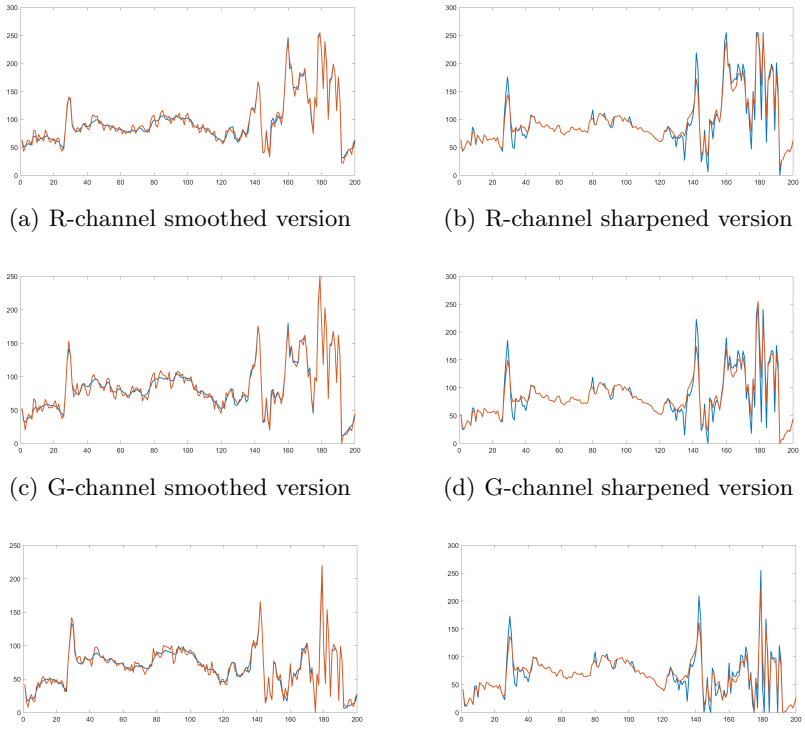
In Figure 6.5 we show an example of the separate smoothing and sharpening performance of the GMS^3 and $NGMS^3$ methods. To illustrate them, in the first column we have chosen a noisy image and we have filtered it with both methods with $\lambda = 0$, that is to say, without sharpening. We can appreciate the smoothing and noise reduction while maintaining edges and details of the image. In the next two columns, we have again the original image, without noise, and the output of both methods under different parameters. Here we can appreciate the opposite effect, the edges of the image are sharpened, keeping the homogeneous zones.

Fig. 6.5: Performance of Smoothing and Sharpening of GMS^3 and $NGMS^3$ methods. First column: original noised image with $\sigma = 10$ and outputs by GMS^3 and $NGMS^3$ using $\alpha = 5$ and $\alpha = 10$, resp. Second column: original image free of noise and outputs by GMS^3 using $\lambda = 0.5$ and $\lambda = 0.8$. Finally, third column: original image free of noise and outputs from $NGMS^3$ using $\lambda = 15$ and $\lambda = 25$.

For a better understanding of how the proposed filters work and their smoothing and sharpening capabilities, the intensity values before and after the filtering of one of the image rows are shown in Figure 6.6.

In the first column, the three RGB channels of the noisy image of Figure 6.5 are shown (orange lines) along with the smoothed version (blue lines). Here we can see how noisy areas are smoothed (left part of the graph) and the border and detail areas are kept intact (right part of the graph).

In the second column, the sharpening results for all three channels are shown. Here, we can compare the intensity values of the noise-free image (orange line) with the sharpening version of the same image (blue line). In these graphs we can see how homogeneous zones are preserved, while differences increase in the border areas. These correspond to more intensified peaks in the graphs.



(a) R-channel smoothed version

(b) R-channel sharpened version

(c) G-channel smoothed version

(d) G-channel sharpened version

(e) B-channel smoothed version

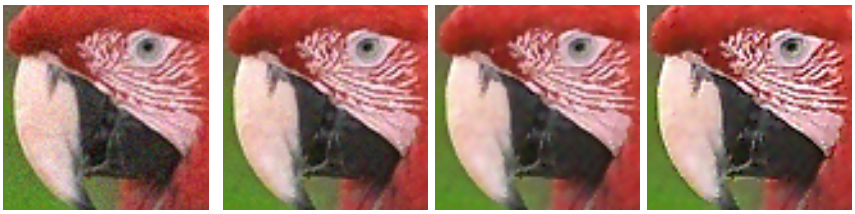
(f) B-channel sharpened version

Fig. 6.6: 1D-representation of each of the RGB channels of one of the rows of the noisy image in Figure 6.5 (orange line) versus its filtered version (blue line).

6.4 Experimental results

In this section we aim to assessing the performance of the GMS^3 and $NGMS^3$ methods using 8-bit per channel RGB images under different parameter settings, and also from different points of view.

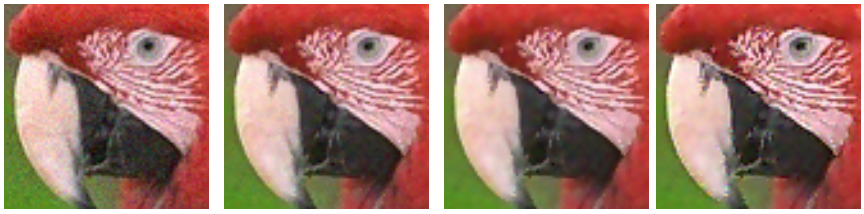
As a first approach to analyze the proposed methods performance, we show in Figures 6.7 and 6.8 some examples of how do they work with different choices of the parameters α and λ, and the effect when increasing their values.



(a) (b) (c) (d)

Fig. 6.7: Examples of application of GMS^3 method. From left to right: (a) original noisy image with $\sigma = 5$, (b) processed image with $\alpha = 3$, $\lambda = 0.2$, (c) processed image with $\alpha = 5$, $\lambda = 0.2$, and (d) processed image with $\alpha = 5$, $\lambda = 0.8$.

Figure 6.7(a) is an image with Gaussian noise ($\sigma = 5$) and Figure 6.7(b) is the processed image by GMS^3 with low parameters for both smoothing and sharpening ($\alpha = 3$ and $\lambda = 0.2$). Figure 6.7(c) displays the result if we raise the value of α, keeping the value of λ fixed ($\alpha = 5$ and $\lambda = 0.2$), and thus achieving greater smoothness. Finally, in Figure 6.7(d) we can see the result of also increasing the parameter λ, thereby enhancing details and edges of the image without intensifying the noise ($\alpha = 5$ and $\lambda = 0.8$).



(a) (b) (c) (d)

Fig. 6.8: Examples of application of $NGMS^3$: From left to right: (a) original noisy image with $\sigma = 5$, (b) processed image with $\alpha = 3$, $\lambda = 5$, (c) processed image with $\alpha = 5$, $\lambda = 5$, and (d) processed image with $\alpha = 5$, $\lambda = 30$.

A similar scheme as in Figure 6.7 is presented for the the $NGMS^3$ method in Figure 6.8. As we pointed out in Section 6.3, we can appreciate the differences in sharpening between the two methods by focusing our attention on Figures 6.7(d) and 6.8(d). We can see how GMS^3 intensifies edges with higher contrast, while $NGMS^3$ gives us a more homogeneous sharpening.

We will see how the GMS^3 and $NGMS^3$ methods can be used to:

1. improve the performance of an edge detector over the output image,
2. process the image for optimizing the visual quality determined by a set of observers, and

Finally, we will compare the proposed technique with the principal state-of-the-art methods. We will see a visual comparison of all of them and additionally, in order to obtain a more objective comparison we will optimize the parameters of each of the methods in terms of an objective quality measure.

6.4.1 Improvement of the performance of an edge detector

The objective of sharpening is not always just an improvement of the visual appearance of the image but also an improvement of the performance of subsequent image processing techniques. We illustrate in Figure 6.9 the improvement that the GMS^3 and $NGMS^3$ methods can provide for borders and details detection. To this end, we have applied the *Canny edge detector* to the noisy original image of Parrots with $\sigma = 5$, see Figure 6.7(a), and to the corresponding processed images by GMS^3 and $NGMS^3$, shown in Figures 6.7(d) and 6.8(d).

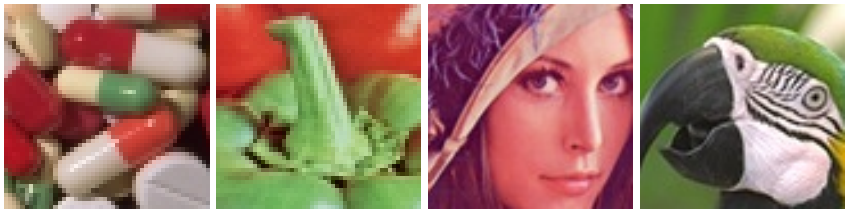
Figures 6.9(b) and 6.9(c) show the borders obtained by Canny edge detector after applying GSM^3 and $NGMS^3$, respectively. In both cases we can appreciate how the edges near the pick and the texture closer to the eye are better recovered, specially for GSM^3. Note that the improved best edge definition provided by this last method is due to the less homogeneous sharpening, that we pointed out in the previous section. This example also illustrates that although higher levels of sharpening may reduce the visual quality of the image, they may allow a better subsequent process of the image.

6.4.2 Optimizing the visual quality by a set of observers

We are going to show the preferred adjustment of the parameters of GSM^3 and $NGMS^3$ in terms of observers who evaluated the quality of the processed image. To this end, we have considered the set of images shown in Figure 6.10 to which we have added white Gaussian noise with standard deviations $\sigma \in \{2.5, 5, 10\}$. These levels of noise have been chosen taking into account that $\sigma = 2.5$ (1%) is a noise level near the perceptual threshold, $\sigma = 10$ (4%) is a noise clearly annoying, and $\sigma = 5$ (2%) is an intermediate value between both of them.

(a) (b) (c)

Fig. 6.9: Example of Canny edge detector applied to (a) the original noisy Parrot image with $\sigma = 5$, and applied to the sharpened-smoothed results obtained by the (b) GSM^3 and (c) $NGSM^3$ methods.



(a) Pills (b) Peppers (c) Lenna (d) Parrots

Fig. 6.10: Set of images considered for the observers optimization process.

A set of 6 observers was selected to adjust the model parameters. All observers visualized the set of images, randomly ordered, under the same conditions: in a dark room, with the same screen, at a distance of about 50cm, and after five minutes of visual adaptation. For each image, each observer has chosen the pair of parameters (α, λ) that yields the best quality image according to their opinion. In Table 6.1, we show the characteristic percentiles of the ordered observations of the parameters α and λ chosen by the observers. Then, we considered the median value of each parameter as an optimum for using it for further analysis.

		$\alpha_{observers}$	$\lambda_{observers}$
GMS^3	P5	0.4	0.05
	P25	2.25	0.1
	P50	**4.43**	**0.16**
	P75	6.1	0.175
	P95	6.8	0.5

		$\alpha_{observers}$	$\lambda_{observers}$
$NGMS^3$	P5	2	0.1
	P25	2	0.1
	P50	**8.67**	**4.54**
	P75	11.25	8.5
	P95	18	18.1

Table 6.1: Adjustment of parameters α and λ by observers for percentiles 5, 25, 50, 75, and 95.

In Figure 6.16, we can see an example of the performance of the GSM^3 and $NGMS^3$ methods using these parameters over Pills image with a Gaussian noise of standard deviation $\sigma = 10$. The results obtained by both methods are similar, with a slight decrease in smoothness being observed in the case of GMS^3. Once again, we appreciate that a more uniform enhancement is offered by the $NGMS^3$ method, as we formerly discussed in Section 6.3.

6.4.3 Comparison with the state-of-the-art techniques

We are going to compare the results of the proposed method with the results obtained from some state-of-the-art methods of simultaneous smoothing and sharpening in terms of an objective quality measure. The methods considered for the comparison will be:

1. the *forward-and-backward diffusion* method *(FAB)* [26],
2. the *fuzzy networks* based technique *(Fuzzy)* [28],
3. the *collaborative filtering* based method *(BM3D)* [5] and
4. the Laplacian matrices based method *(Laplacian)* [7].

Except for the last one, these methods are designed for gray-scale images. Therefore we have applied each one of these methods to each individual color channel of the images.

Fuzzy and *BM3D* methods are controlled by a unique parameter. The results depending on the evolution of their parameters can be appreciated in Figures 6.11 and 6.12.

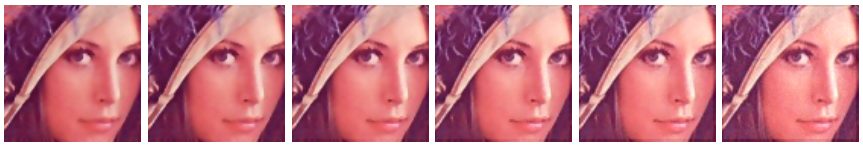


Fig. 6.11: Results of filtering the Lenna image with noise level equal to 5 with the Fuzzy technique progressively increasing the smoothing-sharpening parameter from 30 to 5 in steps of 5.

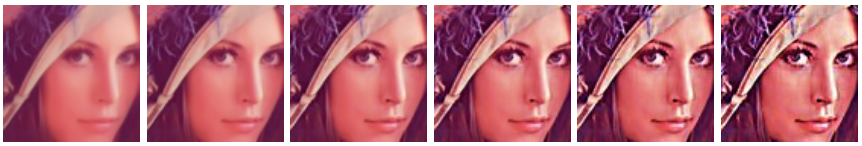


Fig. 6.12: Results of filtering the Lenna image with noise level equal to 5 with BM3D progressively increasing the sharpening parameter from 0.8 to 1.3 in steps of 0.1.

On the other hand, we represent the performance of the multi-parametric (GMS^3, *Laplacian* and FAB) methods respectively for both smoothing and sharpening. We show in Figures 6.13, 6.14 and 6.15 the smoothing feed rate from lowest to highest (horizontally) and how sharpening evolves, from lowest to highest (vertically), according to the parameters variation in each one of these methods.

From a qualitative point of view, we can appreciate the great capacity of smoothing and sharpening presented by *BM3D*. It can also be noticed how an increase of the enhancement is tied to an increase of the image contrast. In contrast, *FAB*, *Laplacian* and GMS^3 methods remove the image noise in an adaptive way: by trying to maintain edges while presenting good sharpening results, and by increasing the image sharpness without changing the image contrast. On the one hand, *Laplacian* method offers good results, but with a lower sharpening potential than GMS^3. We also can see the high sharpening potential of the *FAB* method, however, a lot of edge information is lost and not recovered with the sharpening part, in contrast with the performance of GMS^3. Less noise smoothing will be required in order that *FAB* would provide a final image of good quality.

Now, let us analyze the parameters adjustment from a quantitative point of view. Additionally, we will also compare the results, visually and quantitatively, with other techniques within the state-of-the-art. As figure of merit, we must use a non-reference quality measure given that no ideal output exists when sharpening is performed. So as to, we have chosen the well-known non-reference image quality assessment *BRISQUE* technique (*Blind/Referenceless Image Spatial Quality Evaluator*) [13, 14].

First, the optimal parameters α and λ of GMS^3 and $NGMS^3$ have been obtained by minimizing the sum of the squares of the *BRISQUE* score for the same set of 12 images used in the previous section for observers optimization, see Figure 6.10, using the *Interior Point Algorithm* [3]. These parameters are optimal ones for the whole image set, that is, independently of the image characteristics and noise, see Table 6.4.3.

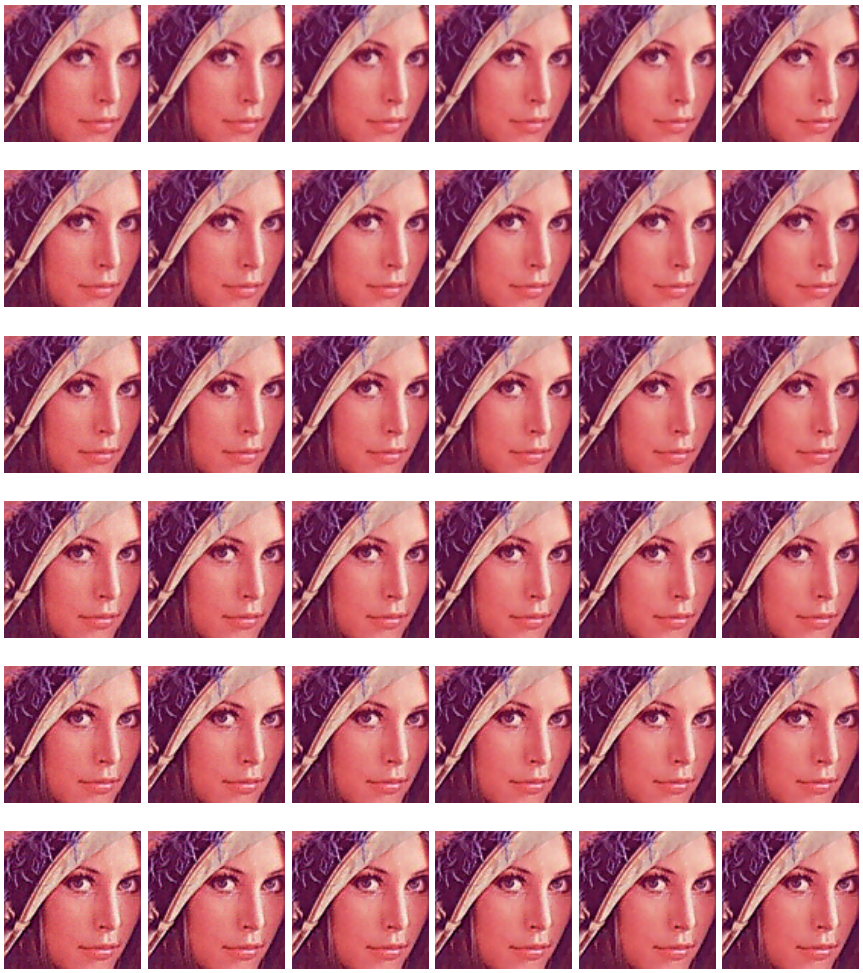


Fig. 6.13: Results of filtering the Lenna image with noise level equal to 5 with GMS^3 progressively increasing the smoothing and sharpening parameteres.

	$\alpha_{BRSIQUE}$	$\lambda_{BRISQUE}$
GMS^3	7	0.275
$NGMS^3$	5.5	3.5

Table 6.2: Optimal parameters for GMS^3 and $NGMS^3$ in terms of BRISQUE.

Results from observers criterion and the ones obtained from the *BRISQUE* score, are slightly similar. However, in the case of the GMS^3, observers smoothed and sharpened less than what *BRISQUE* score suggests. We also found the opposite effect with $NGMS^3$. In Figure 6.16 we can visually com-

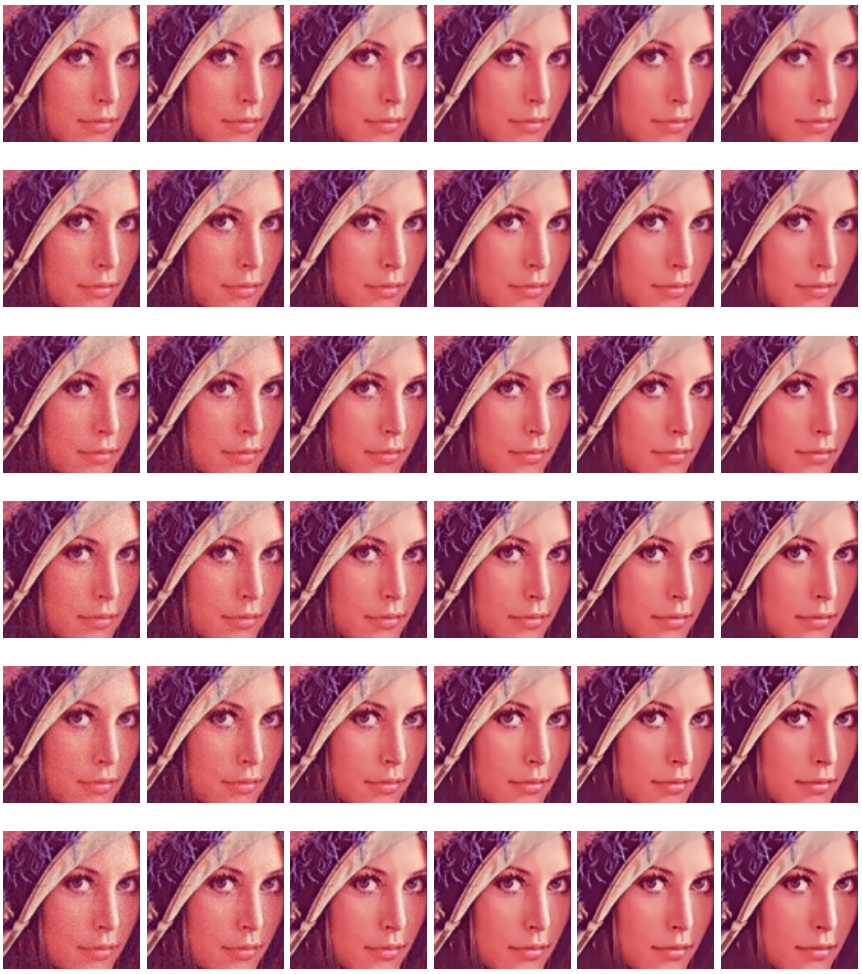


Fig. 6.14: Results of filtering the Lenna image with noise level equal to 5 with *Laplacian* progressively increasing the smoothing and sharpening parameteres.

pare the effects produced by both methods with the different optimal parameters obtained by each procedure.

Focusing our attention on Figure 6.16(c), corresponding to observers, and on Figure 6.16(e), corresponding to *BRISQUE*, we can see the greater sharpening given by the parameter fixed through *BRISQUE*. This effect cannot be appreciated so clearly with $NGMS^3$, as it can be seen in Figures 6.16(d) and 6.16(f), where only slight perceptual differences can be found.

In Figure 6.17, we can compare the performance of the methods looking at a sample row of one image. There, each one of the RGB channels is repre-

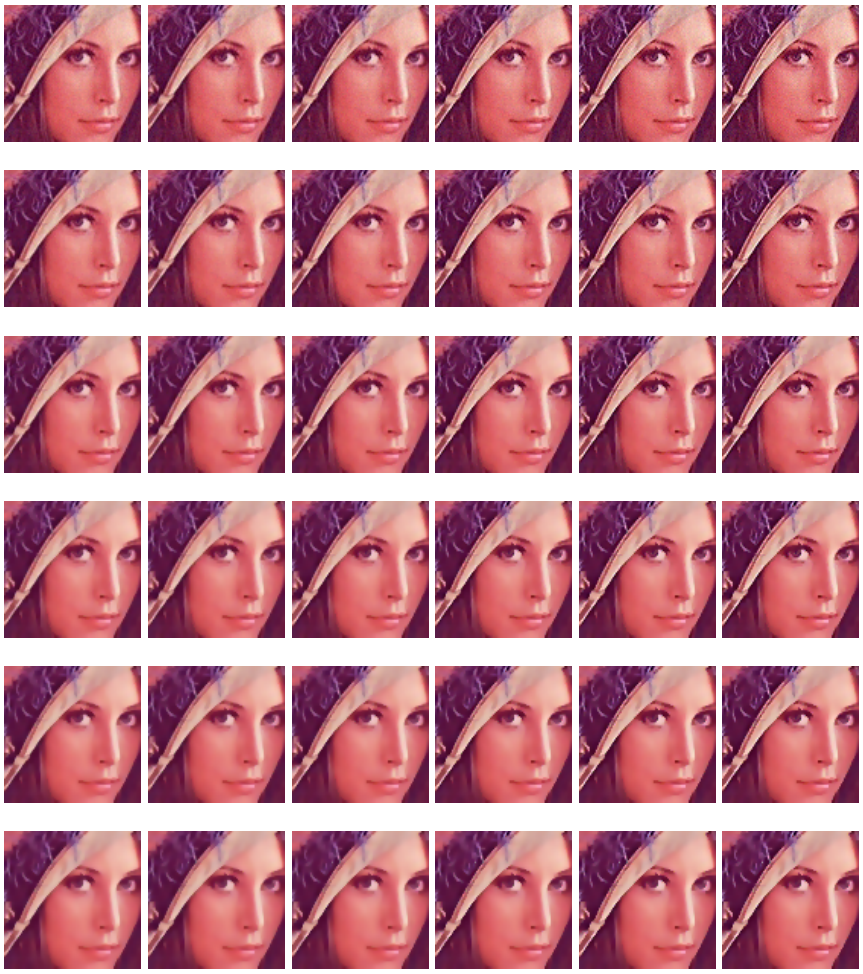
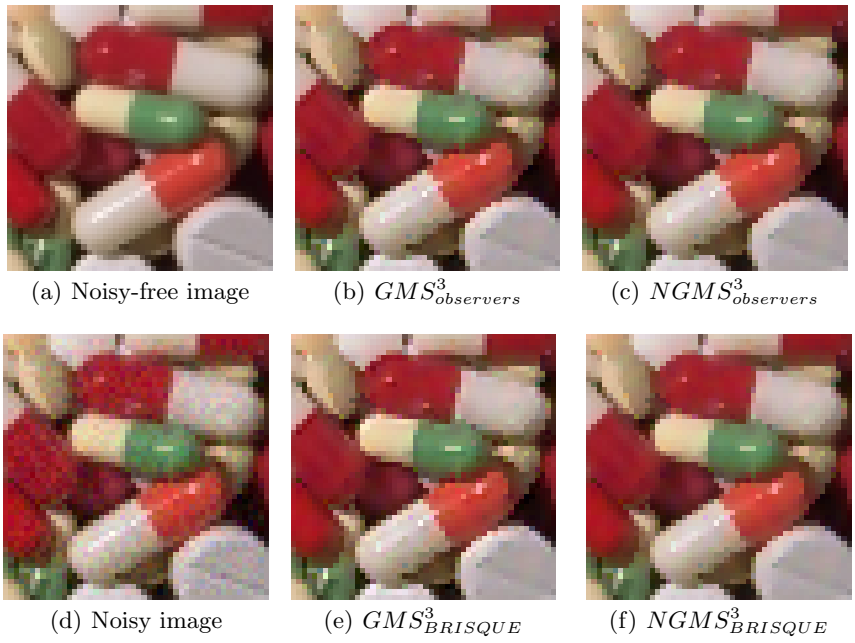


Fig. 6.15: Results of filtering the Lenna image with noise level equal to 5 with FAB progressively increasing the smoothing and sharpening parameteres.

sented for two images, the original noisy one, Figure 6.16(d) and its filtered version with $NGMS^3$, Figure 6.16(e).

In these graphics we can see that high fluctuation areas due to the noise (orange line) are softened in the filtered version of the image (blue line). This can be easily appreciated in the last part of the graphs. Additionally, we also note that the areas of peaks, associated with edges of the image, are intensified. For example, between pixels 30 and 35 the presence of a border or of a detail can be clearly observed, and how it has been intensified in the filtered version.



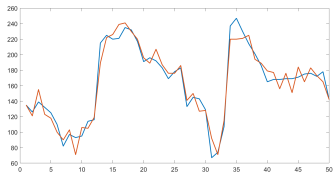
(a) Noisy-free image (b) $GMS^3_{observers}$ (c) $NGMS^3_{observers}$

(d) Noisy image (e) $GMS^3_{BRISQUE}$ (f) $NGMS^3_{BRISQUE}$

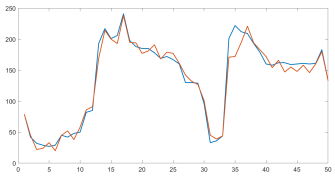
Fig. 6.16: Examples of filtering with GSM^3 and $NGSM^3$ using the parameters set by observers (b) & (c) and with the optimal parameters fixed by using *BRISQUE* score (e) & (f).

Finally, we are going to compare the results of GMS^3 and $NGMS^3$ with the parameters obtained through the *BRISQUE* fitting, respect to the results given by the other methods. The optimal parameters for running these methods have been obtained by minimizing the *BRISQUE* score, too, except for the *FAB* method, due to convergence problems. In this last case, the parameters have been set through observers optimal adjustment.

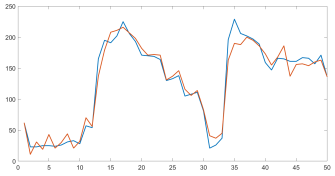
Table 6.3 summarizes the results of all these approaches in terms of the *BRISQUE* score. We remind that the lower the *BRISQUE* score is, the higher the image quality is. The GMS^3 presents similar results to $NGMS^3$ in terms of *BRISQUE* score. In addition, both GMS^3 and $NGMS^3$ outperforms in general the *FAB* and *Fuzzy* methods, and they are competitive respect to the *BM3D* method. We show in Figure 6.19 the output of all these methods for visual comparison. We can see that the sharpening level achieved by GMS^3 is the highest one. *FAB* output still shows a little of noise and the *Fuzzy* output looks quite blurred. Despite of *BM3D* offers very satisfactory results, the quality of the image details and borders of $NGMS^3$ and GMS^3 outputs look better. We would like to point out that *BM3D* is a non-local method and therefore it deals with a greater amount of image information for the image processing.



(a) Red Channel



(b) Green Channel



(c) Blue Channel

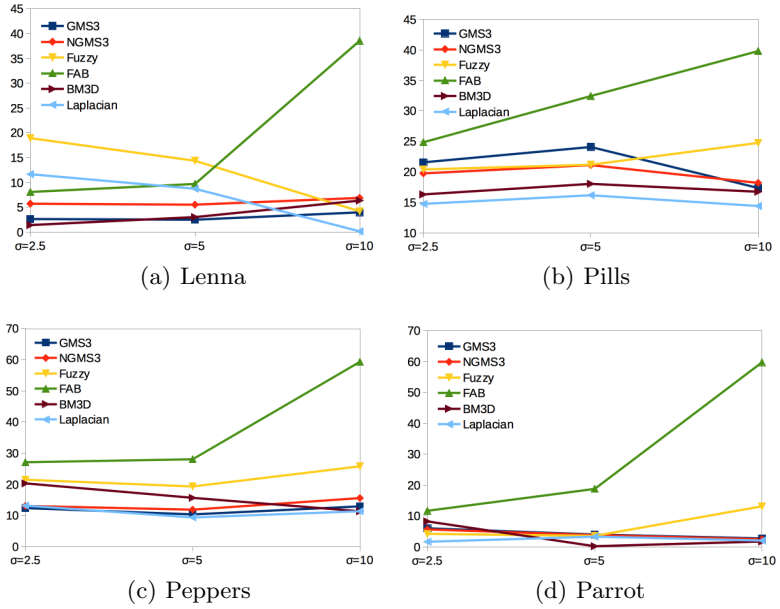
Fig. 6.17: 1D-representation of each of the RGB channels of the original noisy image 6.16(d) (orange line) versus its filtered version 6.16(e) (blue line)

	$\sigma = 2.5$				$\sigma = 5$				$\sigma = 10$			
	Lenna	Pills	Peppers	Parrot	Lenna	Pills	Peppers	Parrot	Lenna	Pills	Peppers	Parrot
GMS^3	2.64	21.57	**12.40**	6.03	**2.50**	24.09	10.37	4.02	3.99	17.37	12.96	2.75
$NGMS^3$	5.73	19.76	13.10	5.68	5.53	21.12	11.89	3.85	6.90	18.20	15.61	2.51
Fuzzy	18.94	20.41	21.50	4.25	14.38	21.18	19.33	3.58	4.20	24.76	25.80	13.15
FAB	8.09	24.86	27.11	11.69	9.71	32.44	28.06	18.78	38.57	39.81	59.34	59.71
BM3D	**1.42**	16.32	20.33	8.27	3.03	18.05	15.69	**0.23**	6.37	16.75	**11.38**	**1.77**
Laplacian	11.68	**14.76**	13.18	**1.68**	8.74	**16.18**	**9.39**	3.34	**0.16**	**14.41**	11.38	2.07

Table 6.3: Results in terms of *BRISQUE* score of our set of images.

Conclusions

In this work we have studied how pixels characterization based on local graphs can be used to perform a simultaneous smoothing and sharpening of color images, a topic little studied in the literature by now. The methods proposed in this work, GMS^3 and $NGMS^3$ are inspired on linear kernel methods, but carrying out the processing in a nonlinear way. Such processing not only changes at different image regions, but also, within each local region. This holds because, at every pixel, we use some pixels in the neighborhood for smoothing and some others for sharpening. Their choice depends on the information obtained from the local graph analysis of the pixels of the image.

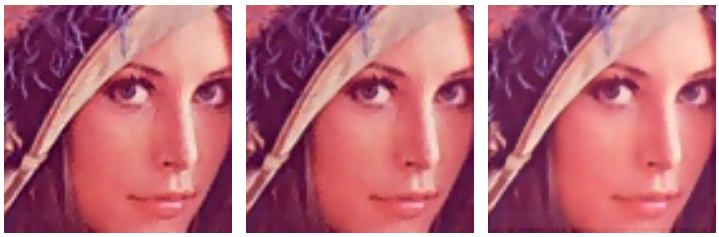


(a) Lenna (b) Pills

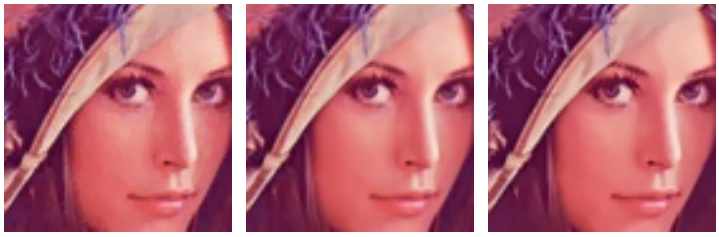
(c) Peppers (d) Parrot

Fig. 6.18: Charts illustrating the behaviour of GMS^3, $NGMS^3$, *Fuzzy*, *FAB*, *BM3D*, and *Laplacian* methods, shown in Table 6.3.

We have studied in detail the performance of our proposed methods, GMS^3 and $NGMS^3$, for different parameter adjustments, either by observers evaluation as well as by optimizing a given non-reference image quality measure, the *BRISQUE* score. We have also compared the outputs with the ones obtained from other state-of-the-art methods. The results show that our methods are competitive with them, both in terms of objective assessment, as well as of visual evaluation. Through all these analysis we have seen that the GMS^3 and $NGMS^3$ methods are highly versatile, so that we can tailor them to specific processing objectives.



(a) GMS3 (b) NGMS3 (c) Fuzzy



(d) FAB (e) BM3D (f) Laplacian

Fig. 6.19: Results of filtering the Lenna image with noise level equal to 5 with the different methods

References

[1] D. Barash. A Fundamental Relationship between Bilateral Filtering, Adaptive Smoothing, and the Nonlinear Diffusion Equation. *IEEE T. Pattern Anal.* 24(6):844–847 (2002).

[2] A. Buades, B. Coll, and J.M. Morel. Nonlocal image and movie denoising, *Int. J. Comput. Vision*, 76(2):123–139 (2008).

[3] R.H. Byrd, J.C. Gilbert, and J. Nocedal. A trust region method based on interior point techniques for nonlinear programming. *Math. Program.*, 89(1):149–185 (2000).

[4] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE T. Pattern Anal.*, 24(5):603–619 (2002).

[5] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE T. Image Process.*, 16(8): 2080–2095 (2007).

[6] J. Immerkaer. Fast noise variance estimation. *Comput. Vis. Image Underst.*, 64(2):300–302 (1996).

[7] A. Kheradmand and P. Milanfar. Non-linear structure-aware image sharpening with difference of smoothing operators. *Frontiers in ICT*, 2(22):1-12 (2015).

[8] S.H. Kim and J.P. Allebach. Optimal unsharp mask for image sharpening and noise removal. *J. Electron. Imaging*, 14(2):023005 (2005).

[9] Y.T. Kim. Contrast enhancement using brightness preserving bi-histogram equalization. *IEEE T. Consum. Electr.*, 43(1):1–8 (1997).

[10] T. Lindberg. Scale-space: A framework for handling image structures at multiple scales, *Technical Report* KTH, S-100 44 Stockholm, Sweden, 1996.

[11] R. Lukac and K.N. Plataniotis. A taxonomy of color image filtering and enhancement solutions. *Adv. Imag. Electron. Phys.*, 140:188-262 (2006).

[12] T. Mélange, V. Zlokolica, S. Schulte, V. De Witte, M. Nachtegael, A. Pižurica, E.E. Kerre, and W. Philips. A new fuzzy motion and detail adaptive video filter. In Blanc-Talon J., Philips W., Popescu D., Scheunders P. (eds) *Advanced Concepts for Intelligent Vision Systems. ACIVS 2007*. Lecture Notes in Computer Science, vo.l 4678. Springer, Berlin, Heidelberg, 640–651 (2007).

[13] A. Mittal, A.K. Moorthy, and A.C. Bovik. No-reference image quality assessment in the spatial domain. *IEEE T. Image Process.*, 21(12):4695–4708 (2012).

[14] A. Mittal, A.K. Moorthy, and A.C. Bovik. Referenceless image spatial quality evaluation engine. *45th Asilomar Conference on Signals, Systems and Computers*, 2011.

[15] S. Morillas, S. Schulte, T. Mélange, E. E. Kerre, and V. Gregori. A soft-switching approach to improve visual quality of colour image smoothing filters. In Blanc-Talon J., Philips W., Popescu D., Scheunders P. (eds) *Advanced Concepts for Intelligent Vision Systems. ACIVS 2007.* Lecture Notes in Computer Science, vol. 4678. Springer, Berlin, Heidelberg, 254–261 (2007).

[16] C. Pérez-Benito, C. Jordán, S. Morillas, and J.A. Conejero. A model based on local graphs for colour images and its application for Gaussian noise smoothing. *J. Comput. Appl. Math.*, 330:955–964 (2018).

[17] C. Pérez-Benito, S. Morillas, C. Jordán, and J.A. Conejero. Determination of connected components in the analysis of homogeneous and detail zones in color images. *Modelling in Science and Education Learning*, 11(1):5-14 (2018).

[18] C. Pérez-Benito, S. Morillas, C. Jordán, and J.A. Conejero. Smoothing vs. sharpening of color images-together or separated. *Applied Mathematics and Nonlinear Sciences*, 2(1):299–316 (2017).

[19] P. Perona, J. Malik, Scale-space and edge detection using anisotropic diffusion. *IEEE T. Pattern Anal.* 12(7): 629–639, (1990).

[20] C.C. Pham, S.V.U. Ha, and J.W. Jeon. Adaptive guided image filtering for sharpness enhancement and noise reduction. In: Ho YS. (eds) *Advances in Image and Video Technology. PSIVT 2011.* Lecture Notes in Computer Science, vol. 7087, 323–334 (2011).

[21] C.C. Pham and J.W. Jeon. Efficient image sharpening and denoising using adaptive guided image filtering. *IET Image Process.*, 9(1):71–79 (2014).

[22] S.M. Pizer, E.P. Amburn, J.D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter HaarRomeny, J. B. Zimmerman, and K. Zuiderveld. Adaptive histogram equalization and its variations. *Computer Vision, Graphics, and Image Processing*, 39(3):355–368 (1987).

[23] K.N. Plataniotis, D. Androutsos, and A.N. Venetsanopoulos. Adaptive fuzzy systems for multichannel signal processing. *Proceedings of the IEEE*, 87(9):1601–1622 (1999).

[24] K.N. Plataniotis and A.N. Venetsanopoulos.*Color image processing and applications*. Springer Science & Business Media (2013).

[25] W.K. Pratt. Digital Image Processing: Piks Inside. John Wiley & sons *Inc.* (2001).

[26] V. Ratner and Y.Y. Zeevi. Stable denoising-enhancement of images by telegraph-diffusion operators. In *20th IEEE International Conference on Image Processing (ICIP) 2013*, 1252–1256 (2013).

[27] L. Rudin, S. Osher, and E. Fatemi. *Non linear total variation based noise removal algorithms*, *Physica D* 60:259–268 (1992).

[28] F. Russo.An image enhancement technique combining sharpening and noise reduction. *IEEE T. Instrum. Meas.*, 51(4):824–828 (2002).

[29] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *IEEE Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271), Bombay, India, 1998*, 839-846 (1998).

[30] T. Wilkin and G. Beliakov. Robust image denoising and smoothing with generalised spatial-tonal averages. In *2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Naples, 2017*, 1–7 (2017).

[31] X. Xie and K.-M. Lam. Face recognition under varying illumination based on a 2D face shape model. *Pattern Recogn.*, 38(2):221–230 (2005).

[32] M.F. Zakaria, H. Ibrahim, and S. A. Suandi. A review: Image compensation techniques. In *2010 2nd International Conference on Computer Engineering and Technology, Chengdu, 2010*, 7:404–408 (2010).

[33] B. Zhang and J.P. Allebach. Adaptive bilateral filter for sharpness enhancement and noise removal.*IEEE Transactions on Image Processing*, 17(5):664–678 (2008).

[34] L. Zhang, R. Lukac, X. Wu, and D. Zhang. PCA-based spatially adaptive denoising of CFA images for single-sensor digital cameras. *IEEE T. Image Process.*, 18(4):797–812 (2009).

Part IV

Conclusions and Future Work

7 Conclusions and Future Work

7.1 Overall conclusions

In this dissertation a novel graph based model for color image processing has been developed. In the model, each pixel is associated to a graph whose features allow to characterize it for different purposes.

This model has been studied in depth to extract all the useful information and turn it into a versatile tool. We have studied the behaviour of the model in the presence of Gaussian noise of different intensities, the potential characteristics of graphs that help us to describe the image and the key parameter of the model, the threshold. We have seen that, depending on the objective or application of the model that we want to carry out, the characteristics of the graph that we will use, and therefore, how the model is used, changes. This is evident in the applications that are presented.

With this model we could classify each of the pixels in an image by defining local graphs, whose features determine the nature of the pixel. This information will be used to different tasks, the most immediate, edge detection. However, as demonstrated in contributions (i) and (ii), this opens the door to multiple types of image processing issues.

We have introduced an application of the model for Gaussian noise smoothing which is based on using each pixel graph to decide whether a pixel belongs to a flat region or not. For this application, the design of the model has taken into account the need for high precision in the classification of each pixel. Therefore, we have used the cardinal of the links set of the local graphs that gives us a wider range of values and so a finer classification than other features. In this way, the model allows to distinguish appropriately flat regions and edge/detail regions in an image even in the presence of noise. Related to this classification a soft-switching filter has been built by using a filter with high smoothing capability in flat regions and a softer one to smooth edge/detail regions. The parameters of the method have been analyzed and it has been proposed how to set them automatically for any input image thus facilitating its use.

Finally, the model has also allowed us to deal with one of the great challenges within image processing: simultaneous image smoothing and sharpening. These two operations are opposed by definition, which makes their simultaneous application a topic little studied in the literature so far.

We have studied how pixels characterization based on local graphs can be used to perform a simultaneous smoothing and sharpening of color images. In this case, local graph features used changed with respect to the previous application, and it was the connected components of the local graphs which allowed us to use a set of pixels to sharpening and another set to smoothing.

We have studied in detail the performance of the proposed methods, $SSLGD$, GMS^3 and $NGMS^3$, for different parameter settings. We have also compared results with those obtained with other state-of-the-art methods. The results show that our methods are competitive with them, both in terms of objective (numerical) assessment, as well as of visual evaluation.

7.2 Future work

Some lines that we find interest for future works include the following:

- To study the application of the model in other color spaces more faithful to human visual perception such as CIELab.
- To adapt the model and the metrics used to represents pixel colour differences in the images in a way closer to the human visual perception.
- After knowing the limitations of the methods of measurement of quality of images we believe that the local information provided by the graphs of the model could serve to design a method of measurement of quality of images and image similarity. The local graphs associated to each image pixels could be compared and thus compare two images, providing a new look at image similarity and quality assessment. Given two images, we could built the graph-model of each image and study global and local disparities between the two models that allow us to measure its resemblance.
- To study how to use the model to stand for those characteristics that are interesting from a visual point of view that in turn we could associate to a score of visual quality. In this way, we could use the model to design a non-reference image quality measure.
- To apply the model to other higher level image processing and analysis tasks such as pattern recognition, 3D imaging, etc.