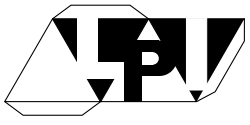
UNIVERSIDAD DE VALLADOLID

Master Universitario de Investigación
en Tecnologías de la Información y las Comunicaciones

MEDICAL IMAGE FILTERING: ENHANCEMENT AND RESTORATION



Laboratorio de
Procesado de Imagen
Santiago Aja Fernández
E.T.S.I. Telecomunicación
Valladolid, Febrero 2020

Medical Image Filtering: Enhancement and Restoration

Santiago Aja-Fernández[1] and Ariel H. Curiale[2]

[1] *Universidad de Valladolid, ETSI Telecomunicación, Valladolid, Spain*
[2] *CONICET, Medical Physics department, Centro Atómico Bariloche, Argentina.*

1 Medical Imaging Filtering

The first thing to consider when dealing with filtering in medical imaging, regardless of the modality, is that the data under consideration can incorporate very sensitive information. The knowledge contained into the intensity pattern that conforms the image has not been acquired with aesthetic purposes but with a clinical or research aim. Therefore, special care must be taken not to eliminate or modified that information: no filtering procedure can be done with simple artistic purposes. Although this premise is clearly shared by most medical imaging researchers, it is sometimes left aside when validating new filtering schemes using *visual comparison*.

From a practical viewpoint, filtering in medical imaging must be **conservative** under the following terms [2]:

1. No significant information present in the image must be erased of modified. For instance, an aggressive filtering can eliminate small calcifications in a mammogram, which could be a risk for diagnosis. In the same way, some filtering methodologies can alter the edges on the image causing a distortion of objects' sizes, which in the end may originate incorrect volume or distance measures.

2. Keep all the information relevant to the physicians. In many occasions noisy patterns have information useful for the expert. Before *cleaning* a specific area of the image, we must ensure the visual role of noise in diagnosis. For instance, in ultrasound imaging, very aggressive filtering that removes the speckle of the image can also remove valuable information about the mobility of certain structures.

3. Do not add information. Filtering artifacts can appear as a side effect of certain techniques. Sometimes these artifacts can be interpreted as anatomical features, and a false diagnosis can be derived.

Thus, the rule of thumb would be *"if you cannot keep all the important information, do not filter"*. Most of the approaches in literature are usually validated via spectacular visual results. However, the success of a filtering procedure is not to produce good-looking pictures, but to ensure that no relevant information is removed.

With this strong requirement in mind, the next step is to consider the final purpose of the filtering of the data. Every proposal in literature will present some advantages and disadvantages, and there is no suitable method for all purposes. Thus, the image processing has to be done attending the utterly use of the filtered image. Let us consider some possible scenarios:

1. **Visual quality:** the purpose of the filtering is to improve its visual quality. The processing must not only seek *good–looking* pictures, but to ease the visual understanding of the data by an expert.

2. **Further processing:** the purpose is to improve the response of different algorithms that will be used to extract information from the data. Note that, this time, the quality of a filtering method is no longer related to the nice appearance of the images, but to the improvement in the accuracy of the algorithms. Some significant applications are segmentation, measure of geometrical distances and numerical processing.

We want to recall the importance of selecting a filtering method totally adapted to the specific needs of the problem. There is no all-purpose filter that, with the same configuration parameters could perform excellent in all situations. Sometimes, very simple filtering techniques are enough for the requirements of the application.

In this chapter, the basic procedures to improve the quality of an image are reviewed. We will make a distinction between image enhancement and restoration:

1. *Image enhancement:* the term gathers all those techniques that seek to (1) improve the visual appearance of an image; or (2) transform the image to different shape more suited to be analyzed. Image enhancement techniques do not assume an underlying model for the image, and therefore their task is not to improve the fidelity to the original image. Some examples of these techniques are contrast manipulation, histogram processing and border detection,

2. *Image restoration:* is a process that seeks to estimate the original image that has been degraded in the acquisition step. A degradation model is needed and it usually involves blurring of the image and noise. Many different degradation models can be used, and they differ among different modalities.

In this chapter, the first sections are devoted to image enhancement: Point to point operations; spatial operations and operations in Transform domain. The second part is focus on image restoration and finally, some examples of medical imaging filtering techniques are given.

2 Point to point operations

Point to point operations are commonly used to enhance low contrast images as well as high luminance images, but they are not only restricted to these tasks. In these operations, each point (pixel or voxel) is modified according to a particular transformation, T, that only depends on the image intensity at each point (Fig. 1). For example, in a 2D gray level image, I, the enhanced image, $\hat{I}$, is defined by the transformation, T, as follows:

$$\hat{I}(\mathbf{x}) = T(I(\mathbf{x})) \tag{1}$$

where $\mathbf{x}$ corresponds to the spatial pixel location, $\mathbf{x} = (x, y)$, the image intensity belongs to $[0, \cdots, L-1]$ and the enhancement image intensity is in the range of $[0, \cdots, L'-1]$. It is straight forward to make an extension of Eq. (1) to color or higher dimensions.

Negative transformation, addition, subtraction, multiplication or division between images are some of the most simple point to point operations and they are defined as follows:

- Negative image:

$$T(I(\mathbf{x})) = (L-1) - I(\mathbf{x}), \qquad I \in [0, \cdots, L-1]. \tag{2}$$

- Addition, subtraction, multiplication and division:

$$T(I_1(\mathbf{x}), I_2(\mathbf{x})) = I_1(\mathbf{x})\langle\cdot\rangle I_2(\mathbf{x}), \qquad \langle\cdot\rangle \in \{+, -, *, /\}. \tag{3}$$

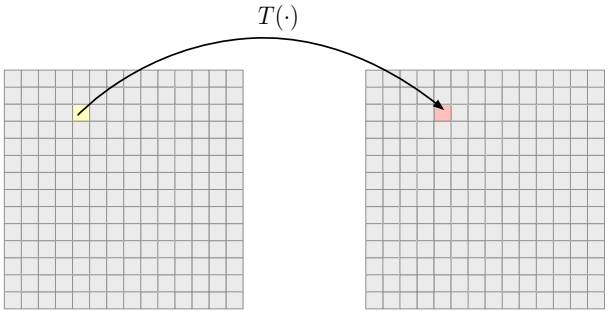
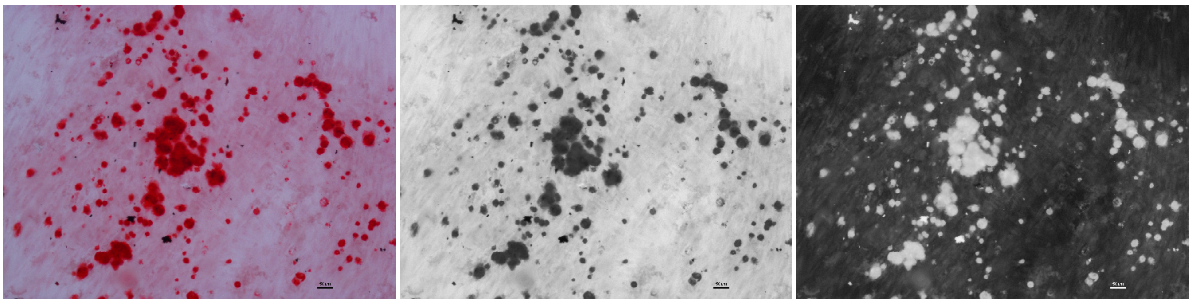


Figure 1: Schematic point to point transformation.

A color image is usually defined by three intensity levels. Each of these intensity levels represents a color level defined in a specific color mode. The most used color models are RGB (red, green and blue), YCM (yellow, cyan and magenta), HSL (hue, saturation and lightness) or HSV (hue, saturation and value), however, they are not the only one. Color point to point transformations can combine the color intensities, or they can be applied for each color. For example, the following point to point operation, T, transforms an RGB image into a grayscale:

$$T = 0,299\,\mathrm{R} + 0,587\,\mathrm{G} + 0,114\,\mathrm{B} \tag{4}$$

where R, G and B represents the input image intensity for each of the RGB channels, and T corresponds to the output image intensity. An example of this transformation can be seen in Fig. 2 where it is shown also the negative of the gray scale image according to Eq. (2).



(a) (b) (c)

Figure 2: (a) Original mesenchymal stem cell (image courtesy of Dr. Diego Bustos, Cell Signal Integration Lab. - IHEM CONICET UNCUYO, Mendoza, Argentina); (b) Gray scale image of (a); (c) Negative of the grayscale image (b).

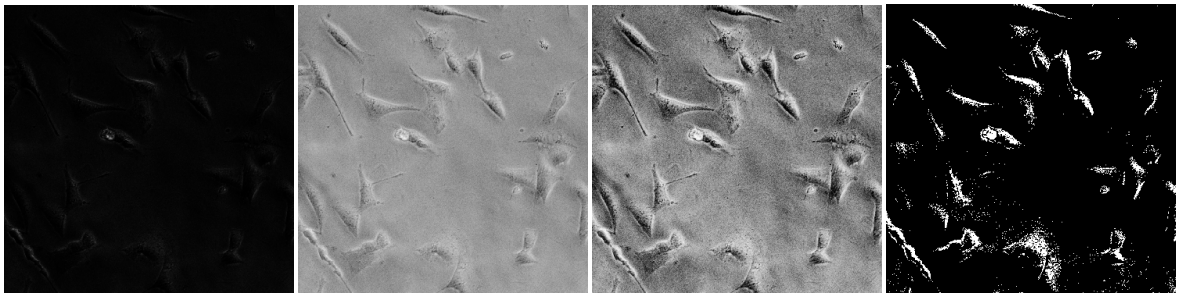
Contrast enhancement

In digital images, dynamic range refers to the ratio between the largest and smallest intensity values, and it is commonly associated to the contrast in an image. When images with high dynamic range are displayed on a monitor, the highest values dominate on the screen. This effect produces that details around lower intensity values seem to be lost. This is because the

image intensity values are linearly scaled to a fixed number of bits when they are displayed, for example 8 bits. To avoid this issue and allow us to better distinguish details on brightness or darkness regions, a logarithm transformations can be applied to compress the dynamic range as follows:

$$T(I(\mathbf{x})) = c * \log(1 + I(\mathbf{x})) \tag{5}$$

where c is a constant. As Figure 3 illustrates, cells cannot be easily distinguished on the dark image (Fig. 3a). This image present a very high dynamic range, indeed, the intensity values around the air bubble are extremely high compared with other pixels. So, it is quite difficult to identify each cells. However, when a dynamic range compression is carried out, the stem cells show up (Fig. 3b). Nevertheless, it can be seen that the image is still having low contrast. i.e. image intensity values are too similar.



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

Figure 3: (a) Dark confocal microscopy of stem cells with high dynamic range; (b) Results of using a log transformation; (c) Results of contrast stretching applied to image (b); (c) Results of a threshold applied to image (b). (Original image courtesy of Dr. Diego Bustos, Cell Signal Integration Lab. - IHEM CONICET UNCUYO, Mendoza, Argentina.)

Low contrast images could be acquired because of a poor illumination or lack of dynamic range in the sensors, among others. Image normalization, also called contrast stretching, aims to improve the contrast in an image by expanding a narrow range of input intensity values into a wide (stretched) range of output intensity values (usually the full range of gray values). The contrast stretching function in Fig. 4a is defined as follows:

$$T(I(\mathbf{x})) = \frac{1}{1 + (\frac{m}{I(\mathbf{x})})^k} \tag{6}$$

where k controls the slope of the function and m corresponds to the intensity value where the stretching will be performed. The result of applying this transformation is a high contrast image as it can be seen in Fig. 3c. In the limit case Eq. (6) becomes just a threshold function (Fig. 4b), and the output results in a binary image (see Fig. 3d).

Different contrast stretching functions can be used for contrast enhancement. However, one of the most simplest is the piecewise linear function depicted in Fig. 4c. Piecewise transformations use different linear functions to modify the output intensity levels. In fact, a piece wise transformation can be seen as a general contrast stretching transformation. The shape of the transformation is controlled by the number of linear transformations used and its connections points. Figure 4c shows a piecewise transformation where three lineal transformations are combined in the points (r_1, s_1) and (r_2, s_2). If $r_1 = s_1$ and $r_2 = s_2$ the piecewise transformation reduces to a linear transformation and no change occurs in the image contrast. Also, if $L = L'$ there is no change at all in the image intensity. If $r_1 = r_2$, $s_1 = 0$ and $s_2 = L' - 1$, it is a

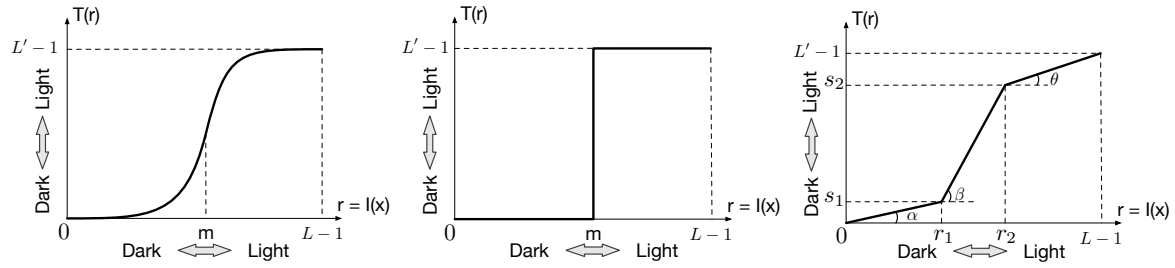


Figure 4: (a) Contrast stretching transformation; (b) Threshold transformations; (c) Piece wise transformation.

threshold function. The following equation is used for performing the piecewise transformation depicted in Fig. 4c:

$$T(r) = \begin{cases} \tan(\alpha)\, r & : 0 \leq r < r_1 \\ \tan(\beta)(r - r_1) + s_1 & : r_1 \leq r < r_2 \\ \tan(\theta)(r - r_2) + s_2 & : r_2 \leq r < L - 1 \end{cases} \tag{7}$$

where $\tan(\alpha) = s_1/r_1$, $\tan(\beta) = (s_2 - s_1)/(r_2 - r_1)$ and $\tan(\theta) = (L' - 1 - s_2)/(L - 1 - r_2)$. An example of this transformation with $r_1 = 5, r_2 = 100, s_1 = 0, s_2 = 200$ can be seen in Fig. 5c.

Another useful family of gray level transformations is commonly named power-law transformations. Power-low transformations have the basic form

$$T(I(\mathbf{x})) = c\,(\epsilon + I(\mathbf{x}))^{\gamma} \tag{8}$$

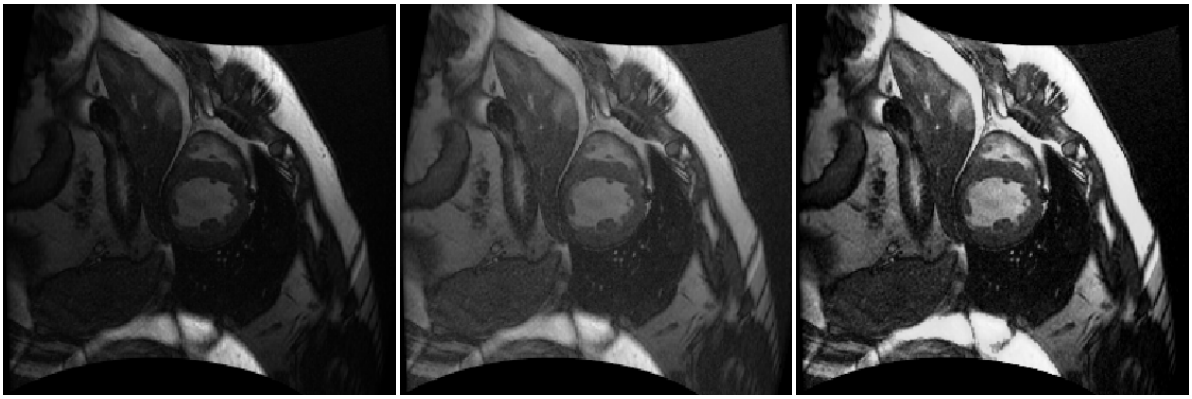
where c, γ and ϵ are constants; and ϵ accounts for an offset. Offsets typically are an issue for display calibration and they are normally avoided. So, equation (8) is simplified to $T(I(\mathbf{x})) = c\, I(\mathbf{x})^{\gamma}$.

Many medical imaging devices used for image capturing and displaying, respond according to non-linearity process. The luminance non-linearity introduced by these devices can often describes according to a power law. The process used for correcting this phenomena is called gamma correction. Images that are not properly corrected can look too dark. In addition to gamma correction, power-law transformations are useful for general-purpose contrast manipulation [6]. Figure 5b shows an example of a power-law transformation on a cardiac magnetic resonance image with $c = 1$ and $\gamma = 0.6$.

Histogram processing

In what follows we will describe some useful point to point operations based on modifying the image histogram. A histogram is an accurate and simple way to estimate the *probability density function* (PDF) for a random variable. It was first introduced by Karl Pearson in [11]. The histogram of an image, I, with L gray levels (i.e. $I(\mathbf{x}) \in [0, \cdots, L-1]$) is a discrete function $h(r_k) = n_k$ that describes the relative frequency, n_k, for a particular gray level intensity r_k. In particular, if the histogram is normalized by dividing each of its values by the total number of pixels/voxels in the image (i.e. $\sum_{r_k=0}^{L-1} h(r_k) = 1$), it gives an estimation that a pixel has a particular gray level.

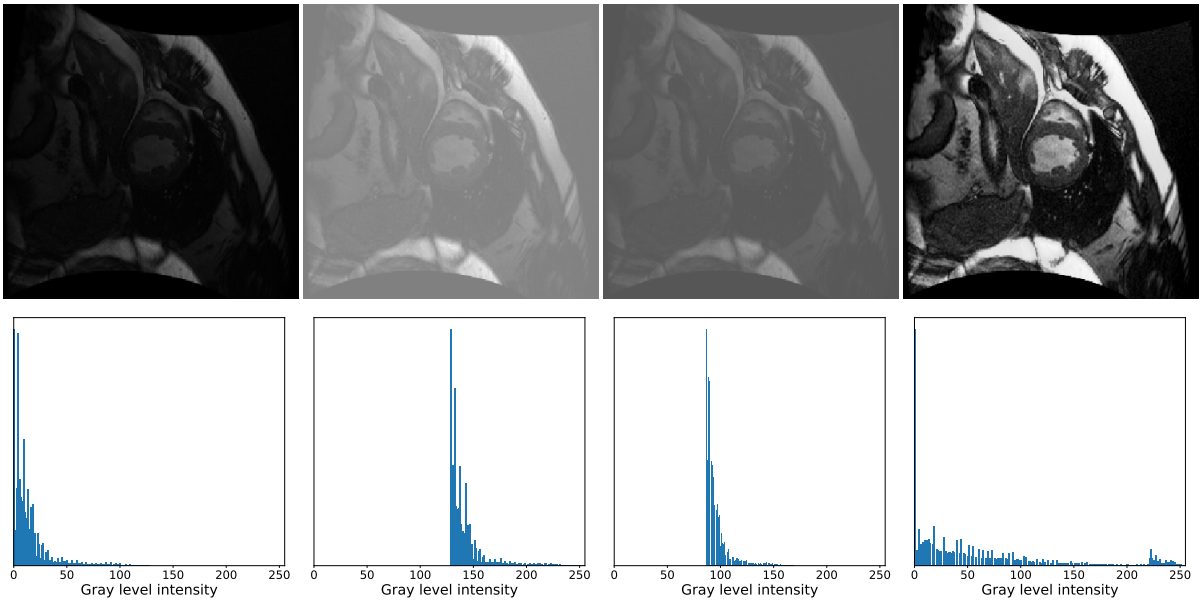
The histogram of an image is a powerful tool for describing basic gray-level characteristics such as dark, light, low contrast, and high contrast, among others. Figure 6 shows the normalized histogram for these four characteristic images. The vertical axis of each histograms



(a) (b) (c)

Figure 5: (a) Original cardiac magnetic resonance imaging (image courtesy of Cardiac Atlas Project); (b) The result of using a power-low transformation; (c) The result of using a pice wise transformation.

represents the relative number of pixel (it is normalized) with a particular gray level intensity. Dark and bright image tends to have more pixels concentrated in the low and high gray scale as the histogram shows in Fig. 6a and 6b. Thus, the histogram gives us information about the spatial distribution of the gray levels in an image. Besides, it can also bring information about the shape of this distribution. For example, the histogram of low-contrast images shows that most of the pixels are in a narrow range of gray levels(Fig.6c). On the contrary, they gray levels in a high-contrast image tends to be spread into a wide range of gray levels (Fig.6d).



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

Figure 6: (a) Dark cardiac magnetic resonance image; (b) Bright cardiac magnetic resonance image; (c) Low-contrast cardiac magnetic resonance image; (c) High-contrast cardiac magnetic resonance image. (Original image courtesy of Cardiac Atlas Project.)

Histogram equalization

Histogram equalization aims to find a transformation, T, that properly distributes the input image intensity where most frequent intensity levels will be assigned to new intensity levels with higher dynamic range. In fact, the main goal of histogram equalization is to transform the input image intensity in a such way that the intensity levels transformed will be uniformly distributed. Hence, the image contrast is enhanced.

As it was introduced before, the image intensity level can be seen as a random variable in the range of $[0, L-1]$. Let p_x be the PDF of a random variable x that represents the input intensity level of an image. Now, let p_y be the PDF of a random variable $y \in [0, L-1]$ that represents the output intensity level for a transformation $y = T(x)$ defined as the cumulative distribution function (CDF):

$$y = T(x) = (L-1) \int_0^x p_x(w) dw \,. \tag{9}$$

It easy to prove that $p_y(y) \sim U[0, L-1]$. A basic result from probability theory is that, if $p_x(x)$ and $T(x)$ are known and $T^{-1}(y)$ satisfy that it is single-valued and monotonically increasing in the interval $0 \leq y \leq L-1$. Then, the PDF $p_y(y)$ is defined as:

$$\begin{aligned} p_y(y) &= p_x(x) \left| \frac{dx}{dy} \right| && (10) \\ &= p_x(x) \left| \frac{1}{\frac{dy}{dx}} \right| && (11) \\ &= p_x(x) \left| \frac{1}{\frac{dT(x)}{dx}} \right| = \frac{1}{L-1}, \quad y \in [0, L-1] && (12) \\ p_y(y) &\sim U[0, L-1] && (13) \end{aligned}$$

where

$$\frac{dT(x)}{dx} = (L-1)\, p_x(x). \tag{14}$$

It is important to note, that unlike its continuous counterpart, the discrete CDF

$$y = T(x) = (L-1) \sum_{j=0}^{x} p_x(j), \quad j \in [0, L-1] \tag{15}$$

where

$$p_x(x) = \frac{n_x}{n}, \quad x \in [0, L-1], \tag{16}$$

cannot produce a discrete equivalent of a uniform probability density function, which would be a uniform histogram. However, as we see in Fig. 7, the use of Eq. (15) does have the general tendency of spreading the histogram of the input image which enhances the image contrast as it is expected. Note that the CDF of the image equalized (i.e. histogram equalization), plotted in red over the histogram in Fig. 7, is similar to a uniform distribution.

Histogram specification

Histogram equalization automatically determines a transformation that distributes the input image intensity that has a uniform histogram. However, there are some applications in which

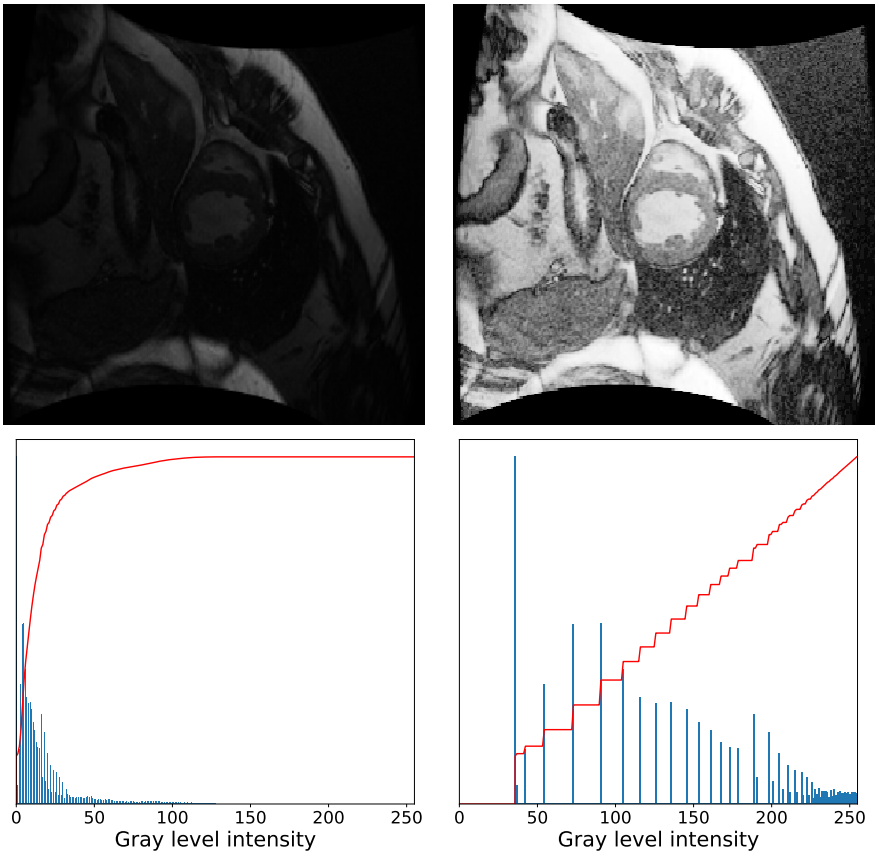


Figure 7: Dark cardiac magnetic resonance image (left) and the result of a histogram equalization (right) where the cumulative distribution function is plot in red over the histogram (bottom). (Original image courtesy of Cardiac Atlas Project.)

attempting to perform such transformation is not the best approach for image enhancement. In fact, we sometimes find useful to *specify* the shape of the histogram that we want to get. The method used for this task is known as *histogram specification* or *matching*.

Using the same notation as it was introduced before, let p_x be the PDF of a continuous random variable $x \in [0, L-1]$ that represents the image intensity level. Now, let p_z the specified PDF of another continuous random variable, $z \in [0, L-1]$, that represents the desired image intensity level. Suppose next that we define the transformation, $y = T(x)$, as the CDF in a similar ways as it was done for histogram equalization:

$$y = T(x) = \int_0^x p_x(w)dw. \tag{17}$$

It is important to note that now $y \in [0, 1]$ instead of $[0, L-1]$. Then, we define the transformation $G(z)$ as the CDF of the p_z as follows:

$$G(z) = \int_0^z p_z(w)dw, \tag{18}$$

and the property of $y = G(z) = T(x)$. Therefore, z must satisfy the following condition:

$$z = G^{-1}(y) = G^{-1}(T(x)). \tag{19}$$

Assuming that G^{-1} exists and it is singled-valued and monotonically increasing, it is possible to transform the image gray levels from the original image to get an image according to the specified probability density function p_z by using G^{-1} and T. Note that the problem to find G^{-1} is considerably simplified for the discrete case. However, in this case, only an approximation to the desired histogram is achieved. Since the gray levels in the image are integer, a simple approach can be used to estimate $\hat{z} = G^{-1}(y_k)$. Indeed, a good approximation is obtained by finding the smallest integer $\hat{z} \in [0, L-1]$ such that

$$(G(\hat{z}) - y_k) \geq 0 \quad k = 0, \ldots, L-1. \tag{20}$$

Figure 8 shows an example of the histogram specification approach (Fig. 8d) for a desired p_z (Fig. 8e plotted in red). In particular, this example shows that the histogram equalization is not the best approach for improving the contrast (Fig. 8b). Furthermore, it shows that the histogram specification avoid the saturation issue introduced when the histogram equalization is performed.

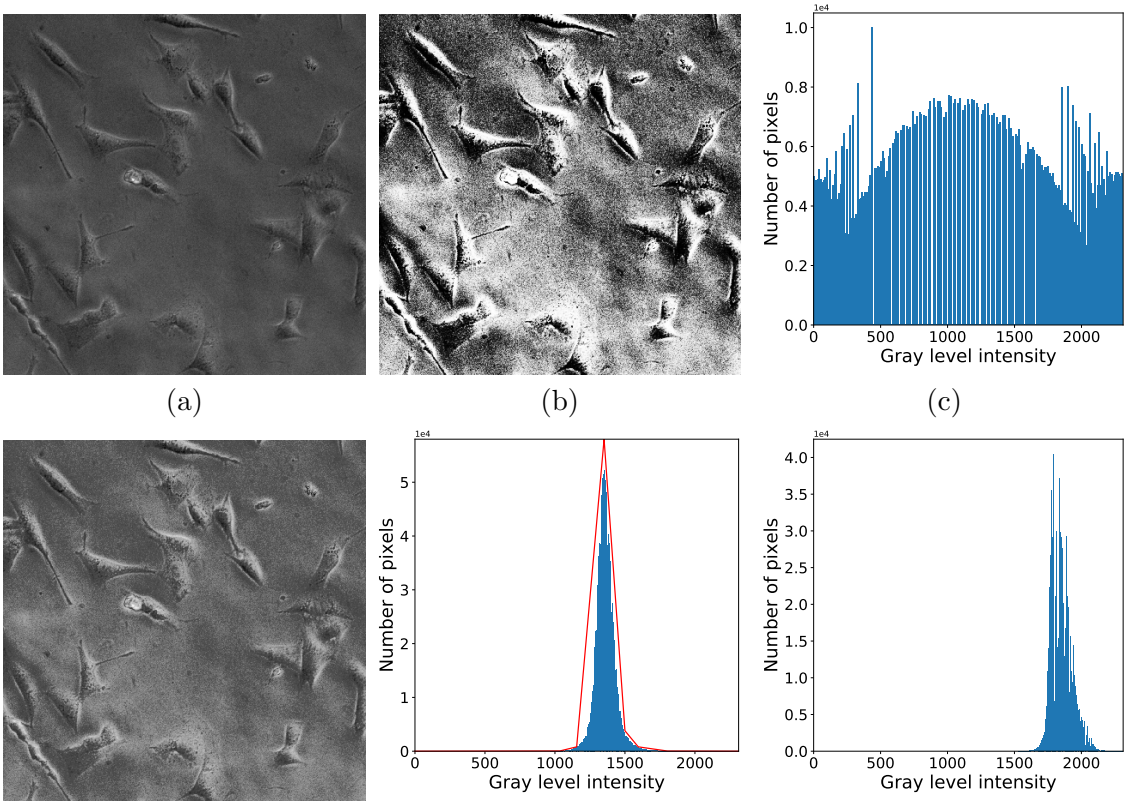


Figure 8: (a) Original confocal microscopy of stem cells; (b) Results of a histogram equalization; (c) Histogram of (b); (d) Results of histogram specification according to the desired distribution plotted in red on the image histogram (e); (f) Histogram of the specified histogram image. (Original image courtesy of Dr. Diego Bustos, Cell Signal Integration Lab. - IHEM CONICET UNCUYO, Mendoza, Argentina.)

3 Spatial operations

The term *spatial operations*, *spatial transformations* or *spatial filtering* refers to operations in the image plane itself, i.e. operations based on direct manipulation of pixels in an image. Approaches in this category commonly use information around the pixel to be transformed as it is depicted in yellow in Fig. 9. This neighborhood is commonly called *mask*, *filter* or *kernel*, and their values are referred to *coefficients*. Following the same notation that we have introduced in point to point operations, each point (pixel or voxel) in the input image is transformed according to a spatial transformation, T, that takes into account the information of the point and their neighborhood according to the values defined in the filter or kernel as follows:

$$\hat{I}(\mathbf{x}) = T(I(\mathbf{x}), I(\mathbf{k})), \tag{21}$$

where $I(\mathbf{x})$ is the intensity level of the point $\mathbf{x}$, and $I(\mathbf{k})$ corresponds to the intensity levels of the points in the neighborhood of $\mathbf{x}$ spatially defined by the kernel $\mathbf{k}$.

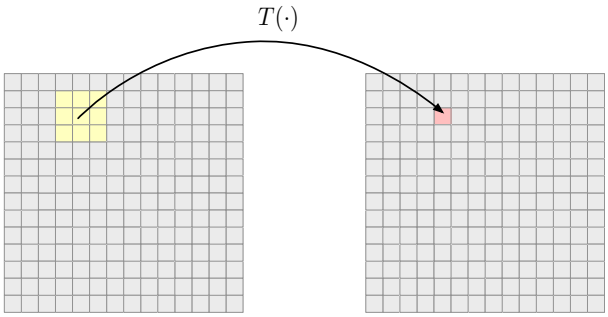


Figure 9: Schematic spatial transformation.

Most of the spatial operations discussed in this section can be formulate as linear operations between the input image and the kernel $\mathbf{k}$ as follows:

$$\hat{I}(\mathbf{x}) = I(\mathbf{x})\, w_0 + \sum_{i=1}^{N} I(\mathbf{x} + \mathbf{k}_i))\, w_i \tag{22}$$

where w_0 corresponds to the central coefficient of the kernel, and $\mathbf{k}_i$ represents the relative spatial kernel position for the coefficient w_i when the pixel $\mathbf{x}$ is located at the center(see Fig. 9 for a 2D example). Eq. (22) can be rewritten for an image I of size $M \times N$ with a filter mask of size $m \times n$ as follows:

$$\hat{I}(x, y) = \sum_{i=-a}^{a} \sum_{j=-b}^{b} w(i, j) I(x + i, y + j) \tag{23}$$

with $x \in [0, M-1]$, $y \in [0, N-1]$, $a = (m-1)/2$ and $b = (n-1)/2$. An example of a 3×3 filter mask is depicted in Fig. 10.

As it will be discussed in Section 4, linear spatial filtering often is referred to as convolving a mask with an image due to its similarities to the frequency domain concept called *convolution*. In this way, filter mask are sometimes called *convolution mask* or *convolution kernel*.

Regarding to implementation details, it is important to think what happens when the center of the filter approaches to the borders of the images. When the convolution mask reaches the image borders one possible consideration is to restrict the center of the mask to be a distance no less than (n-1)/2 where n is the mask size in the dimension of the border that is reached.

$w(-1,-1)$	$w(-1,0)$	$w(-1,1)$
$w(0,-1)$	$w(0,0)$	$w(0,1)$
$w(1,-1)$	$w(1,0)$	$w(1,1)$

Figure 10: 3×3 filter mask.

The resulting image will be smaller than the original, but the processed pixels will have been computed with the full mask. Other possible solutions are to complete the pixels outside the image with 0, or repeating the value in the border among others. These operation are commonly named padding. Most of the times the best option relies completely on the problem to solve and the filter to use. For example, on most of linear spatial operations, the zero-padding seems to be a proper way to deal with this issue. However, for the median nonlinear filter, using a zero-padding approach can introduce undesired effected on the borders.

In what follows we will introduce some of the most common spatial operations used for smoothing, border detection and sharpening. Furthermore, we will describe some nonlinear spatial transformation for similar purposes.

Smoothing filters

Smoothing spatial operations or filters are specially designed to smooth the data with the aim of removing small details on it. These operations are commonly used in image processing for object detection or noise reduction. The mean filter is the most intuitive and simplest smoothing filter for smoothing images. This filter consists of assigning for each point in the image the average value of the intensity levels in the neighborhood. Figure 11 shows the effects of the mean filter smoothing for a brain and its corrupted magnetic resonance image with a white noise[1] by using a squared mask of 5×5 as follows

$$\mathbf{k} = \frac{1}{25}\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

In addition, Fig. 11e show that the mean filter is specially efficient for removing white noise (Fig. 11d). However, it tends to lose small details as it is shown in Fig. 11f. According to Eq. (22), this filter can be rewritten as follows

$$I(\mathbf{x}) = \frac{1}{25}\sum_{i}^{25} I(\mathbf{x} + \mathbf{k}_i). \tag{24}$$

This formulation leads to a computationally more efficient algorithm avoiding to perform extra multiplications/divisions. Indeed, it requires just one multiplication/division for each pixel.

It is important to note, that all the coefficients in the mean filter contribute in the same way to the result. This filter tends to remove or reduce noise, however, it also tends to remove small object or details in the image which could be an undesired effect. This leads to a variation of the

[1] details on noise models can be found in Section 5

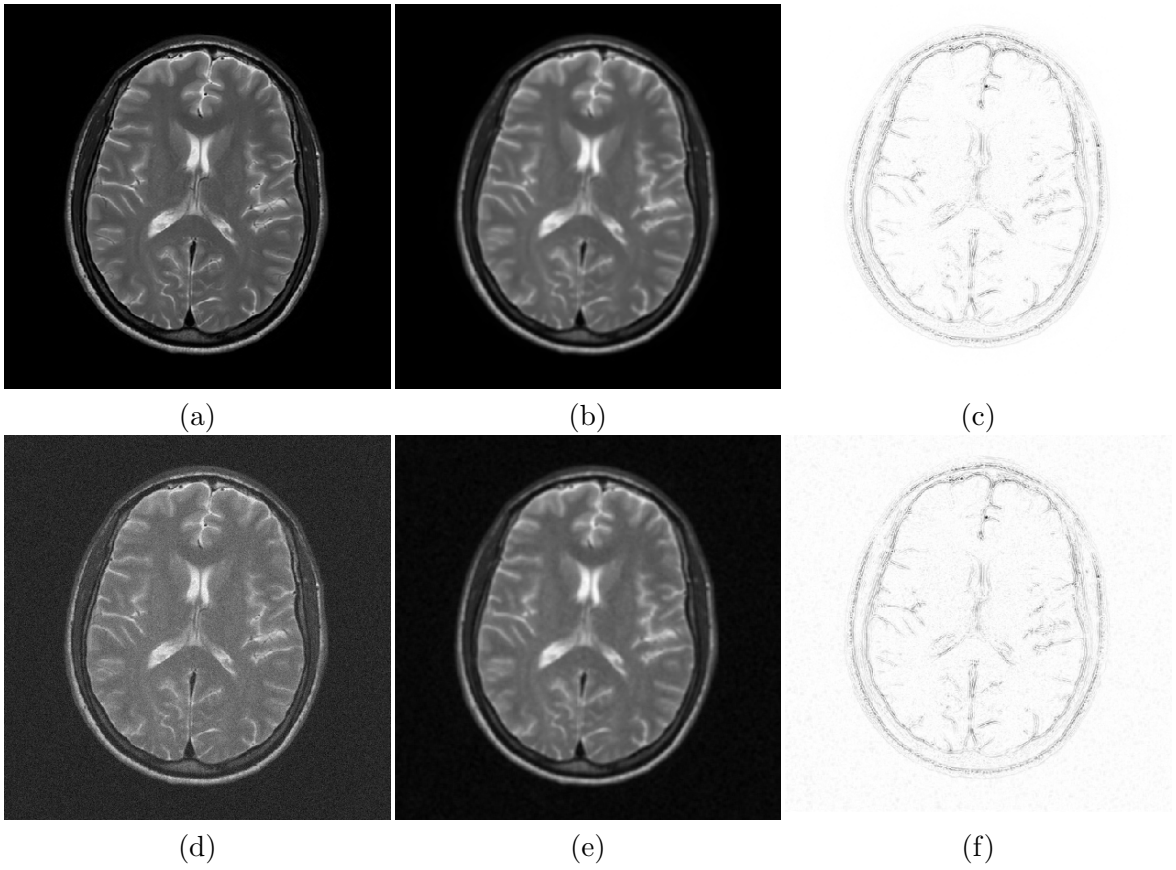


Figure 11: (a) Original brain magnetic resonance image; (b) Results of smoothing the image (a) with a squared mask of 5×5; (c) Differences between (a) and (b); (d) Original image (a) corrupted by a withe noise; (e) Results of smoothing the noisy image (d) with a squared mask of 5×5. (f) Differences between the original image (a) and the smoothed noise image (e). (Original image courtesy of Hospital Clínico Universitario de Valladolid, Spain.)

previous filter which is named weighted mean filter, where in this case, not all the pixels in the neighborhood contribute to the result in the same way. For example, a possible implementation can be done by using a convolution mask as follows

$$\mathbf{k} = \frac{1}{16}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 8 & 1 \\ 1 & 1 & 1 \end{bmatrix},$$

where the contribution of the central pixel is eight times higher than the others. A special case of weighted mean operator is the Gaussian smoothing where the kernel is defined by the Gaussian function

$$f(\mathbf{x}) = \frac{\exp(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}))}{\sqrt{(2\pi)^2|\Sigma|}} \tag{25}$$

where $\boldsymbol{\mu}$ and Σ correspond to the multidimensional mean and the covariance matrix respectively. In fact, if dimensions are uncorrelated (i.e. Σ is diagonal), the Gaussian filter can be generated from only 1D Gaussian filter. Furthermore, the kernel should be carefully designed to include at least 2σ (or 3σ) of the distribution values. In this sense, its size must increase with increasing σ to maintain the Gaussian nature of the filter. Figure. 12 shows an example of a 2D Gaussian convolution kernel with $\mu_x = \mu_y = 0$ and $\sigma_x = \sigma_y = 1$ where 3σ of the distribution values are

included. Additionally, results of applying a 2D Gaussian smoothing filter for different isotropic $\sigma \in [1, 2, 4]$ are depicted in Fig. 13.

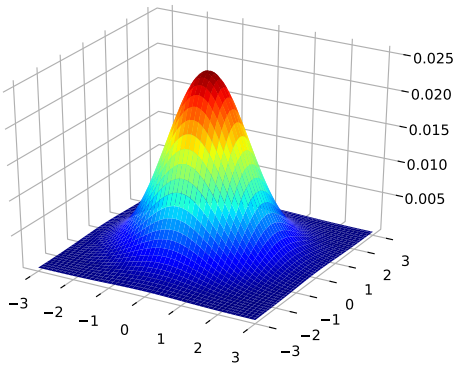


Figure 12: Example of 2D Gaussian kernel $\mu_x = \mu_y = 0$ and $\sigma_x = \sigma_y = 1$.

Highlighting borders and small details

The principal objective of the spatial filters described in this section is to highlight the boundaries of objects and small details on images. Thus, they can be seen as opposite spatial operations to those described for smoothing. In particular, these filter are based on first- and second-order derivatives, specially the gradient and Laplacian

$$\nabla I(\mathbf{x}) = \left(\frac{\partial I(\mathbf{x})}{\partial x_1}, \cdots, \frac{\partial I(\mathbf{x})}{\partial x_n} \right) \tag{26}$$

$$\Delta I(\mathbf{x}) = \nabla^2 I(\mathbf{x}) = \nabla \cdot \nabla I(\mathbf{x}) = \sum_i^n \frac{\partial^2 I(\mathbf{x})}{\partial^2 x_i} \tag{27}$$

where $\mathbf{x} \in \mathbb{R}^N$.

The derivatives of an image are defined in terms of differences and they provide an easy way to identify where the image intensity presents a high variation. A possible definition of the first-order derivative in 2D image, not the only one, is

$$\frac{\partial I(x,y)}{\partial x} = I(x+1, y) - I(x, y) \tag{28}$$

$$\frac{\partial I(x,y)}{\partial y} = I(x, y+1) - I(x, y). \tag{29}$$

There are different convolutions mask that implement Eq. (28) and (29), but the most well known are the Prewitt, Roberts and Sobel (Fig. 14).

Similarly, we define the second-order derivative as

$$\frac{\partial^2 I(x,y)}{\partial x^2} = I(x+1, y) - 2I(x, y) + I(x-1, y) \tag{30}$$

$$\frac{\partial^2 I(x,y)}{\partial y^2} = I(x, y+1) - 2I(x, y) + I(x, y-1). \tag{31}$$

In this way, the Laplacian (Eq. (27)) of an image can be computed by using one of the convolutions mask described in Fig. 15.

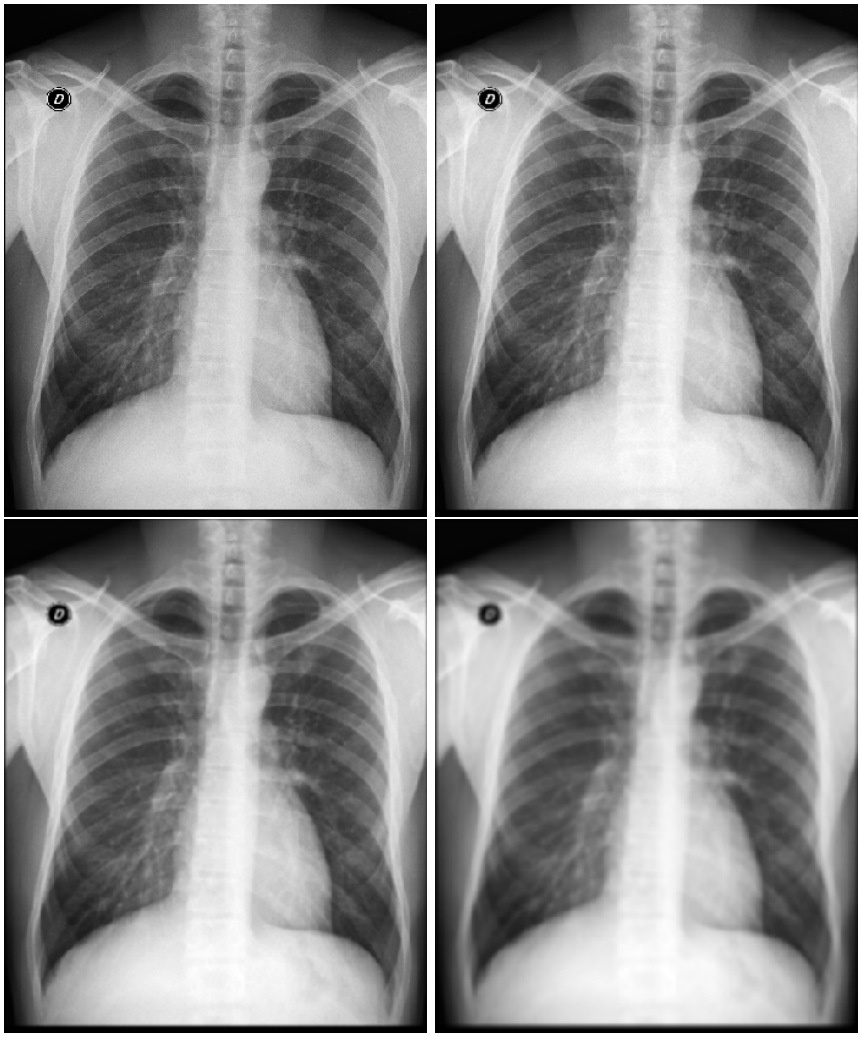


Figure 13: Results of applying a Gaussian smoothing filter to a chest X-ray image (upper left) for different isotropic $\sigma \in [1, 2, 4]$. (Original image courtesy of Hospital Universitario UNCUYO, Mendoza, Argentina.)

$$\underset{\text{(a)}}{\begin{bmatrix} -1 & 1 \end{bmatrix}} \qquad \underset{\text{(b)}}{\begin{bmatrix} 1 \\ -1 \end{bmatrix}} \qquad \underset{\text{(c)}}{\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}} \qquad \underset{\text{(d)}}{\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}}$$

$$\underset{\text{(e)}}{\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}} \quad \underset{\text{(f)}}{\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}} \quad \underset{\text{(g)}}{\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}} \quad \underset{\text{(h)}}{\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}}$$

Figure 14: Different mask used for the computation of the gradient vector in 2D. (a-b) Partial derivatives; (c-d) Roberts operators; (e-f) Prewitt kernel filters; (g-h) Sobel convolution masks.

The principal objective of border detection is to highlight transitions between objects boundaries in the image. A simple way to highlight this transitions is by using the magnitude of a first-order derivative as it is shown in Fig. 16b. In the same way, to highlight small details in

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

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

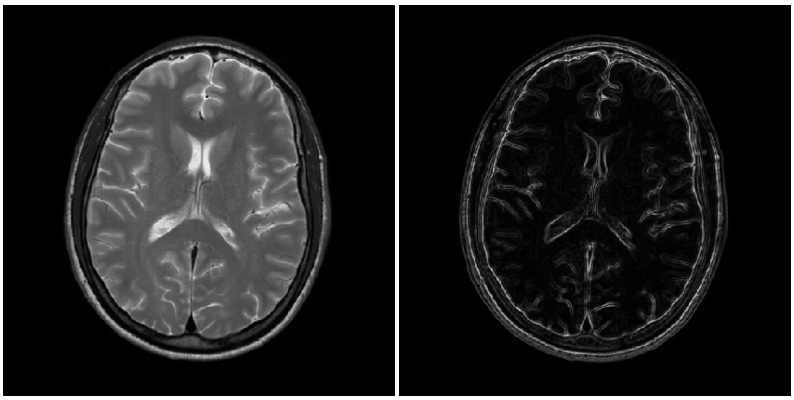
Figure 15: Laplacian masks with (c-d) and without (a-b) diagonal elements.

the image (Fig. 17c) it is common to add a second-order derivative, such as the Laplacian, to the original image as follows:

$$\hat{I}(\mathbf{x}) = \begin{cases} I(\mathbf{x}) - c\,\nabla^2 I(\mathbf{x}) & \text{if the center coefficient of the Laplacian mask is negative} \\ I(\mathbf{x}) + c\,\nabla^2 I(\mathbf{x}) & \text{if the center coefficient of the Laplacian mask is positive} \end{cases} \tag{32}$$

where the sharpening effect is controlled by the constant c. Another option to highlight small details in images is known as hihgh-boost filtering, and it is defined by using a blurred version of the original image, $\overline{I}$, as follows:

$$\hat{I}(\mathbf{x}) = A\,I(\mathbf{x}) - \overline{I}(\mathbf{x}) \qquad A \geq 1. \tag{33}$$



(a) (b)

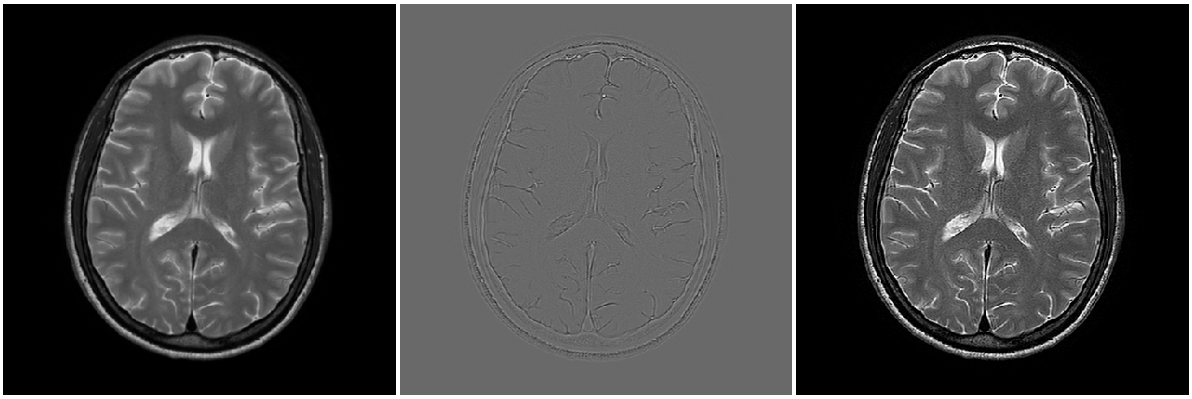
Figure 16: (a) Original brain magnetic resonance image; (b) Sobel gradient of (a). (Original image courtesy of Hospital Clínico Universitario de Valladolid, Spain.)

Nonlinear filters

Nonlinear spatial filters such as max, min, median or variance filters also operates on a neighborhood, and the mechanism of sliding a mask over the image is the same. However, the transformation, T, applied to a pixel using the information of its neighborhood cannot be described just as a weighted sum over the filter coefficients, or convolution, as it was done in Eq. (22).

Median Filter

The transformation, T, that defines the median filter, sorts the intensity values of the pixels in the neighborhood and replace its intensity with the one that corresponds to higher half of the intensity in the neighborhood. In particular, median filter is a powerful tool for noise and speckle reduction. In contrast to smoothing filters, median filter preserve the border in the image. as it can be seen in Fig. 19.



(a) (b) (c)

Figure 17: (a) Original brain magnetic resonance image (courtesy of Hospital Clínico Universitario de Valladolid, Spain.); (c) Laplacian of (a); (c) Sharpening of (a).

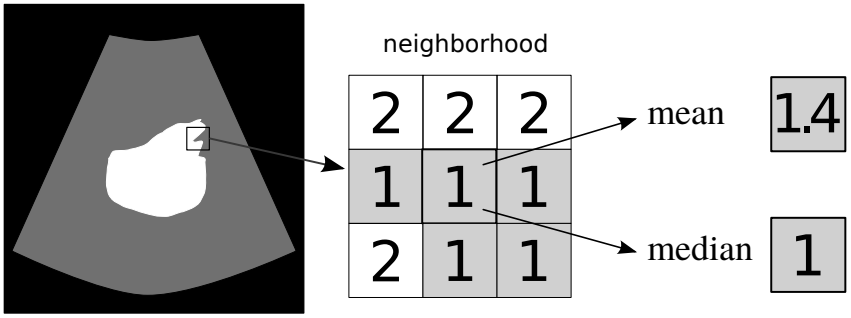


Figure 18: Example of median filtering in a 3×3 neighborhood compared to a mean filter of the same area.

We define the median filtering as

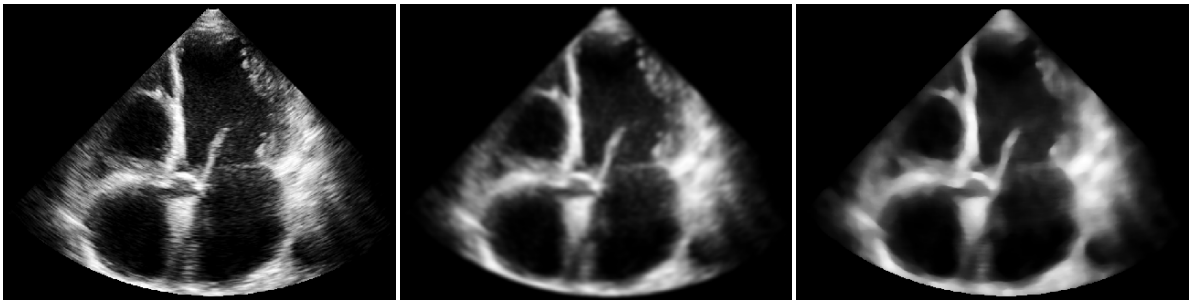
$$I_{\mathrm{MED}}(\boldsymbol{x}) = \mathrm{median}_{\eta(\boldsymbol{x})} I(\boldsymbol{x}) \tag{34}$$

where $\eta(\boldsymbol{x})$ is a centered neighborhood. Assuming a $P \times P$ window, the median filter works as follows:

1. The P^2 values of the pixels in the neighborhood are extracted.
2. Those values are ordered.
3. The output of the filter correspond to the value placed in the $\frac{P^2+1}{2}$ position.

An example the filtering of one particular neighborhood is depicted in Fig. 18 and in Fig. 19, and compared to the result of a linear *mean* filter. This particular example illustrates one of the distinct features of the median filter: the output value is one of the values already present in the image. This effect has two advantages: first, no new values are introduced into the image; second, there is no smooth or blur of edges.

One of the main drawbacks of the median filter is its computational cost, with the number of operations growing exponentially with the size of the window. An alternative would be the so-called *pseudomedian* filter [13]. If $\{M_N\}$ is a sequence of elements $m_1, m_2, \cdots, m_N$, the



(a) (b) (c)

Figure 19: (a) Four chamber cardiac ultrasound image; (c) Gaussian smoothing filter; (c) Median filter. (Original image courtesy of Dr. T. Perez Sanz, Rio Hortega, Valladolid, Spain.)

pseudomedian of the sequence is defined as

$$\text{pmed}\{M_N\} = \frac{\text{maximim}\{M_N\} + \text{minimax}\{M_N\}}{2} \tag{35}$$

where

$$\begin{aligned} \text{maximin}\{M_N\} &= \max\{\min(m_1, \cdots, m_L), \min(m_2, \cdots, m_{L+1}), \\ &\quad \cdots, \min(m_{N-L+1}, \cdots, m_N)\} \end{aligned}$$

and

$$\begin{aligned} \text{minimax}\{M_N\} &= \min\{\max(m_1, \cdots, m_L), \max(m_2, \cdots, m_{L+1}), \\ &\quad \cdots, \max(m_{N-L+1}, \cdots, m_N)\}, \end{aligned}$$

with $L = (N+1)/2$.

4 Operations in Transform Domain

Most of the operations previously introduced can be carried out in transform domains, alternatively to the presented spatial domain. As seen in the previous chapter, there is an equivalence between convolution in the spatial domain and multiplication of the FT of the signals.

4.1 Linear filters in frequency domains

The linear processing of an image using a kernel, as described in eq. (22) can be written as a convolution, using the notation of the previous chapter:

$$\hat{I}(x, y) = I(x, y) * h(x, y).$$

where $h(x, y)$ is the convolution kernel. If continuous signals are assumed, the Fourier transform of the convolution is carried out by the product of every term:

$$\widehat{I}(u, v) = I(u, v) \cdot H(u, v) \tag{36}$$

However, in practical situations in which images are limited and discrete, and the DFT is used, the equivalent of this operation in the frequency domain implicitly assumes a periodic expansion of the signal, an LCI system and a circular convolution:

$$\hat{I}(x, y) = I(x, y) \circledast h(x, y).$$

According to the previous chapter, the DFT of this convolution

$$\hat{I}[k_1, k_2] = I[k_1, k_2] \cdot H[k_1, k_2]. \tag{37}$$

where $H[k_1, k_2]$ is the DFT of $h(x, y)$.

4.2 Homomorphic processing

A very useful enhancement scheme can result from considering the image as the product of two components:

$$f(\boldsymbol{x}) = i(\boldsymbol{x}) \cdot r(\boldsymbol{x}) \tag{38}$$

where $i(\boldsymbol{x})$ is the *illumination* component and $r(\boldsymbol{x})$ the *reflectance*. The illumination is related to the shades and lights within the image while the reflectance is related to the objects in the image, i.e. the transitions and border. As a consequence, $i(\boldsymbol{x})$ is a low-pass signal and $r(\boldsymbol{x})$ is a high pass signal.

The separation and subsequent equalization of both sources may produce very different results, being the main purpose the correction of uneven illumination in the scene. In medical imaging this is particularly important in some modalities, like radiography.

In order to decouple both components, the logarithm is taken:

$$\log f(\boldsymbol{x}) = \log i(\boldsymbol{x}) + \log r(\boldsymbol{x}). \tag{39}$$

This way, a simple linear filering may separate one of the components in order to process differently each channel. As an example, in Fig. 20 a scheme for local contrast enhancement is presented. Homomorphic processing is also used to separate multiplicative noise from the original signal.

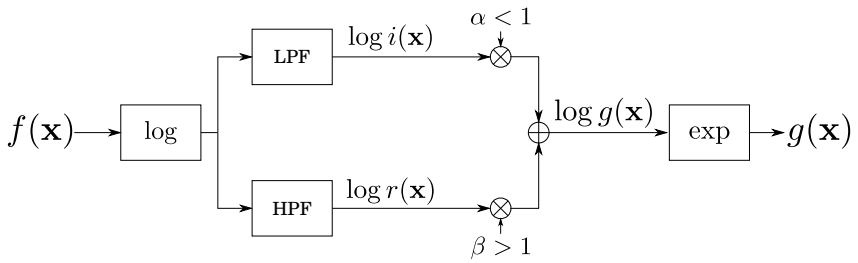


Figure 20: Example of homomorphic processing: local contrast enhancement scheme.

5 Model-based filtering: Image restoration

Image restoration aims to improve the image when for some reason it has been degraded. This degradation can occurs for multiple reasons, but noise and blurring are the most common causes. The emission and detection of light and all other types of electromagnetic or ultrasonic waves are stochastic processes by nature. In this sense, noise is inherent to the images acquisition process, and it refers to the image intensity level variation for a particular spatial position. Besides the noise introduced in the acquisition process, images are also corrupted during transmission mainly due to the interference in the channel used in the transmission process, such as wired and wireless networks. To measure the power of the noise in the image, an important measure from signal theory, the *signal-to-noise ratio* (SNR) is used. In particular, if this ratio is to small it means that the noise level is high compared with the image intensity, and the meaningful information will be lost in the noise.

On the other hand, blurring can be produced in medical image by different causes such as patient motion or those related to the image acquisition/reconstruction. For example, multiples echos in magnetic resonance or scattered radiation in computed tomography. A simplify general model (Fig. 21) that takes into account the noise and image degradation present in the observed image, I, can be modeled as:

$$I(\mathbf{x}) = g(\mathbf{x}) + \eta(\mathbf{x}) \tag{40}$$

where η corresponds to the noise model. The image formation process, g, is modeled by a linear and space invariant system where its impulse response, h, is convolved with the original image f as follows:

$$g(\mathbf{x}) = h(\mathbf{x}) * f(\mathbf{x}). \tag{41}$$

Additionally, the noise model, η could be signal *dependen* or *independent* ($\eta_1 \neq 0$ vs. $\eta_1 = 0$)

$$\eta(\mathbf{x}) = g(\mathbf{x}) \times \eta_1(\mathbf{x}) + \eta_2(\mathbf{x}). \tag{42}$$

where η_1 and η_2 are two particular noise probability density function. The reader is referred to [7] for more details on a possible generalization of an image observation model.

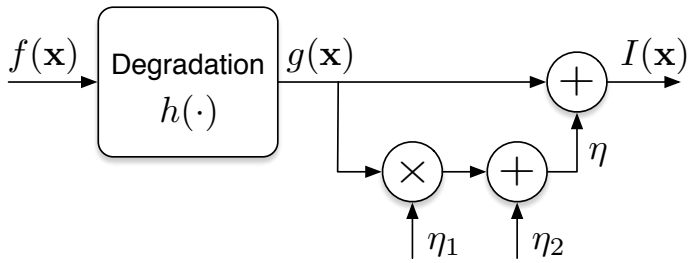


Figure 21: Simplify image degradation model.

Depending on the image acquisition process and transmission system used for imaging the body, Eq. (40)-(42) can be simplify by only considering η_1 or η_2 as follows

$$I(\mathbf{x}) = g(\mathbf{x}) + \eta \quad \text{(Signal independent noise model)} \tag{43}$$

$$I(\mathbf{x}) = g(\mathbf{x}) + g(\mathbf{x}) \times \eta \quad \text{(Signal dependent noise model)}, \tag{44}$$

where I refers to the observed image, g corresponds to the detected image, and η is a particular noise probability density function.

Noise models

The most common noise models in computer vision and medical images are *photon noise*, *white noise*, *salt & pepper*, *Rayleigh* and *Gamma* among others. The photon noise degradation refers to the inherent variation of photons collected by a digital sensor over a given time interval. This intensity variation is usually modeled as signal dependent (Eq. (44)) with a Possion PDF. However, in computer vision a widespread approximation is to model with a white noise, i.e. by modeling the image noise as signal independent (Eq. (43)) with a zero-mean Gaussian PDF

(Table 1). Additionally, withe noise also arises in an image due to factors such as electronic circuit noise. The impulse noise, also called *salt & pepper*, is often used for modeling malfunctioning of sensor cells, memory cell failure, or synchronization errors in the image digitalization or transmission, for example. Impulsive noise has only two possible intensity level, a and b (Table 1). If $b > a$, gray level b will appear as light dot in the image. Conversely, a gray level will appears as a dark dot. For an 8 bit/pixel image, the typical intensity values for a (pepper noise) is close to 0 and for b (salt noise) is close to 255 [6].

Ultrasound, synthetic aperture radar and optical coherence tomography are images with a particular granular pattern named speckle. This patter is generated by the reflection of transmitted coherent waves at fixed frequencies, as it happens in ultrasound waves. In ultrasound systems, as many others, the result of the interaction between those waves and different types of tissues give rise to the interference phenomenon known as speckle. This interference pattern, though it is textured with noisy visual aspect, remains unaltered under the same acquisition conditions [4], i.e. the same transducer aperture, pulse length and transducer angle. This behavior exhibits an inherent relationship with the tissue structure and it is commonly modeled as signal dependent (Eq. (44)) with a Rayleigh or Gamma PDF (Table 1). Figure 22 shows images and histogram resulting from adding some of the degradation model described before.

PDF		Mean	Variance
Gaussian	$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(z-\mu)^2/2\sigma^2}$	μ	σ^2
Uniform	$p(z) = \begin{cases} \frac{1}{b-a} & \text{if } a \le z \le b \\ 0 & \text{otherwise} \end{cases}$	$(a+b)/2$	$(a+b)^2/12$
Exponential	$p(z) = ae^{-a\,z}, \quad z \ge a$	$1/a$	$1/a^2$
Rayleigh	$p(z) = \frac{2}{b}(z-a)e^{-(z-a)^2/b}, \quad z \ge a$	$a + \sqrt{\pi\, b/4}$	$b(4-\pi)/4$
Gamma	$p(z) = \frac{a^b z^{b-1}}{(b-1)!} e^{-az}, \quad z \ge a$	b/a	b/a^2
Impulsive (salt & pepper)	$p(z) = \begin{cases} P_a & z = a \\ P_b & z = b \\ 0 & \text{otherwise} \end{cases}$		

Table 1: Different common noise probability density functions (PDF). The random variable, z represents the noise intensity (gray) level.

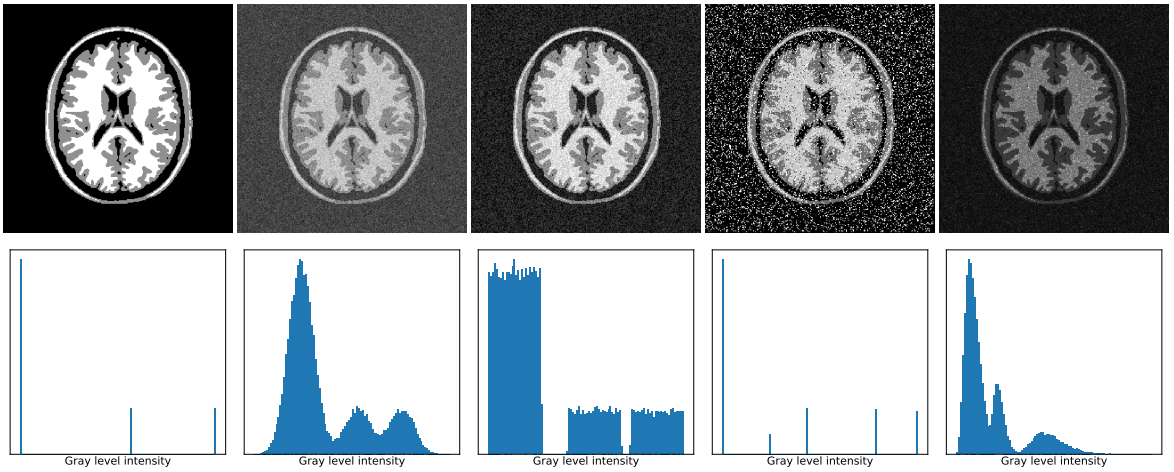
Restoration process

Most of the approaches used throughout this chapter for image restoration assume that the degradation on the image, also known as *point spread function*, can be modeled as linear and invariant space filter as it was described in Eq. (40)

$$I(\mathbf{x}) = h(\mathbf{x}) * f(\mathbf{x}) + \eta(\mathbf{x}). \tag{45}$$

Convolutions in the space domain correspond to multiplications in the frequency domain as it was introduced in section 4. So, Eq. (45) can be rewritten in the frequency domain as follows:

$$\mathcal{F}_{\mathbf{x}}[I(\mathbf{x})](\mathbf{w}) = H(\mathbf{w})\, F(\mathbf{w}) + N(\mathbf{w}) \tag{46}$$



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

Figure 22: Images and histogram resulting from adding a signal independent Gaussian (b), uniform (c), impulsive (d), and a signal dependent Gamma (e) noise to a synthetic brain imaging (a).

where each term in capital letter are the Fourier transform of the corresponding terms described before. These two equations settle down the bases of image restoration for linear space-invariant systems.

Image restoration only in presence of noise

In what follows, we assume that we only deal with degradation due to noise, i.e. the impulse response of the point spread function in the frequency domain, H, is the identity operator (Eq. 46). Figure 23 shows the noise reduction results for the arithmetic mean filter

$$\hat{I}(x,y) = \frac{1}{m\,n} \sum_{(s,t)\in S_{xy}} I(s,t), \tag{47}$$

the geometric mean filter

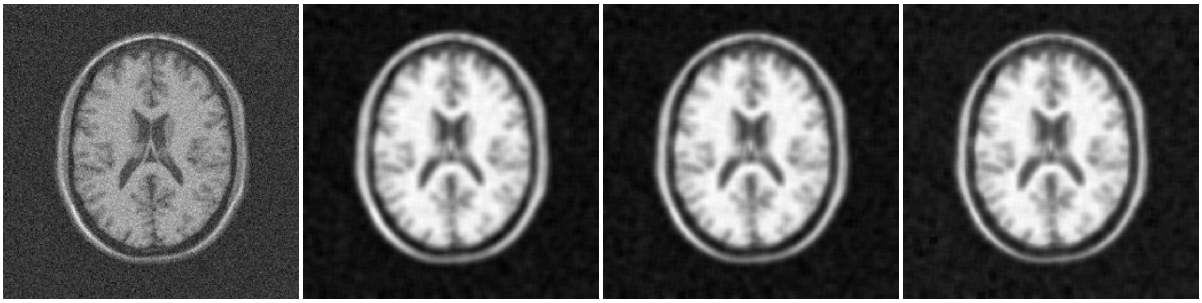
$$\hat{I}(x,y) = \left[\prod_{(s,t)\in S_{xy}} I(s,t) \right]^{\frac{1}{m\,n}}, \tag{48}$$

and the harmonic mean filter

$$\hat{I}(x,y) = \frac{m\,n}{\sum_{(s,t)\in S_{xy}} \frac{1}{I(s,t)}} \tag{49}$$

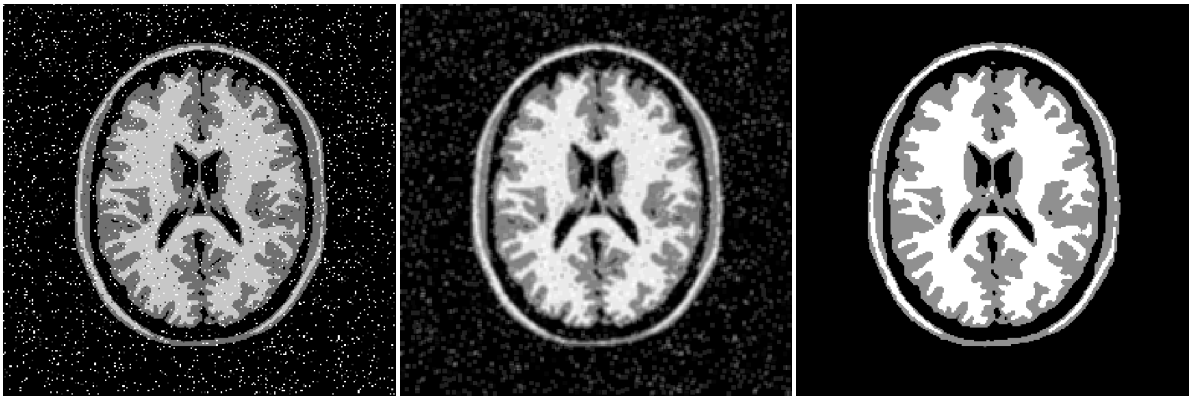
where S_{xy} represents the set of spatial coordinates for the noisy image, $I(x,y)$, in a window of $m \times n$.

In particular, geometric and harmonic mean filters are better than arithmetic mean filter for removing white noise because they preserve more fine details on the image. However, they tend to fail for removing impulsive salt & pepper noise (Fig. 24). In this case, median, min and max nonlinear filters are better options for noise reduction. Indeed, median filter presents a remarkable performance for impulsive noise reduction (Fig. 24c). Furthermore, median, max and min filters tend to preserve image borders as it was described before.



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

Figure 23: Results of noise reduction by using an arithmetic (b), geometric (c) and harmonic (d) mean filter for a synthetic MRI brain image corrupted with a withe noise (a).



(a) (b) (c)

Figure 24: Results of noise reduction by using an arithmetic mean (b) and median (c) filter for a synthetic MRI brain image corrupted with an impulsive salt & pepper noise (a).

6 Some examples of filtering techniques

In this final section we will show some examples with popular filtering schemes and their application to specific noise models.

6.1 Linear minimum mean square estimator

One simple and powerful way to estimate the original signal when it is corrupted with noise is using a statistical estimator, like the Linear Minimum Mean Square Error (LMMSE) estimator [8]. The general formulation of the LMMSE of a parameter θ in the presence of noise is defined as

$$\hat{\theta} = E\{\theta\} + \mathbf{C}_{\theta\mathbf{s}}\mathbf{C}_{\mathbf{ss}}^{-1}(\mathbf{s} - E\{\mathbf{s}\}) \tag{50}$$

being $\mathbf{s}$ the vector of available samples, $\mathbf{C}_{\mathbf{ss}}$ the covariance matrix of $\mathbf{s}$ and $\mathbf{C}_{\theta\mathbf{s}}$ the cross-covariance vector. When assuming a simple degradation model of signal $f(\mathbf{x})$ corrupted by additive Gaussian noise,

$$I(\mathbf{x}) = f(\mathbf{x}) + \eta\left(\mathbf{x}; 0, \sigma^2\right),$$

the LMMSE estimator becomes a simple version of the adaptive Wiener filter [9]

$$\widehat{f}(\mathbf{x}) = E\{f(\mathbf{x})\} + \frac{\mathrm{Var}(f(\mathbf{x}))}{\mathrm{Var}(I(\mathbf{x}))}(I(\mathbf{x}) - E\{I(\mathbf{x})\}). \tag{51}$$

A practical implementation of this estimator is achieved using local sample moments:

$$\widehat{f}(\mathbf{x}) = \langle I(\mathbf{x})\rangle_{\mathbf{x}} + \frac{\langle I^2(\mathbf{x})\rangle_{\mathbf{x}} - \langle I(\mathbf{x})\rangle_{\mathbf{x}}^2 - \sigma^2}{\langle I^2(\mathbf{x})\rangle_{\mathbf{x}} - \langle I(\mathbf{x})\rangle_{\mathbf{x}}^2} \left(I(\mathbf{x}) - \langle I(\mathbf{x})\rangle_{\mathbf{x}}\right) \tag{52}$$

where $\langle . \rangle_{\mathbf{x}}$ denotes the local mean estimator defined as:

$$\langle I(\mathbf{x})\rangle_{\mathbf{x}} = I(\mathbf{x}) * h(\mathbf{x}). \tag{53}$$

The term $h(\mathbf{x})$ is a low pass filter, as previously defined.

When doing the expansion to cope with MRI data, we can adopt different noise models, being the Rician the most common:

$$I(\mathbf{x}) = |f(\mathbf{x}) + \eta_1\left(\mathbf{x}; 0, \sigma^2\right) + j. * \eta_2\left(\mathbf{x}; 0, \sigma^2\right)|,$$

where $\eta_1(\mathbf{x})$ and $\eta_2(\mathbf{x})$ are the real and imaginary parts of a complex additive Gaussian noise process. Since the moments of the Rician distribution have a non-trivial integral expression but for even-order moments, in order to achieve a closed-form expression, the estimator was defined for the square signal $f^2(\mathbf{x})$ instead of $f(\mathbf{x})$. The LMMSE for this case becomes [1]

$$\widehat{f^2}(\mathbf{x}) = E\{f^2(\mathbf{x})\} + \mathbf{C}_{f^2 I^2}(\mathbf{x})\mathbf{C}_{I^2 I^2}^{-1}(\mathbf{x})\left(\mathbf{I}^2(\mathbf{x}) - E\{\mathbf{I}^2(\mathbf{x})\}\right) \tag{54}$$

The practical implementation of this estimator with local moments local sample moments becomes

$$\widehat{f^2}(\mathbf{x}) = \langle I^2(\mathbf{x})\rangle_{\mathbf{x}} - 2\sigma^2 + K(\mathbf{x})\left(I^2(\mathbf{x}) - \langle I^2(\mathbf{x})\rangle_{\mathbf{x}}\right) \tag{55}$$

with $K(\mathbf{x})$

$$K(\mathbf{x}) = 1 - \frac{4\sigma^2\left(\langle I^2(\mathbf{x})\rangle_{\mathbf{x}} - \sigma^2\right)}{\langle I^4(\mathbf{x})\rangle_{\mathbf{x}} - \langle I^2(\mathbf{x})\rangle_{\mathbf{x}}^2}. \tag{56}$$

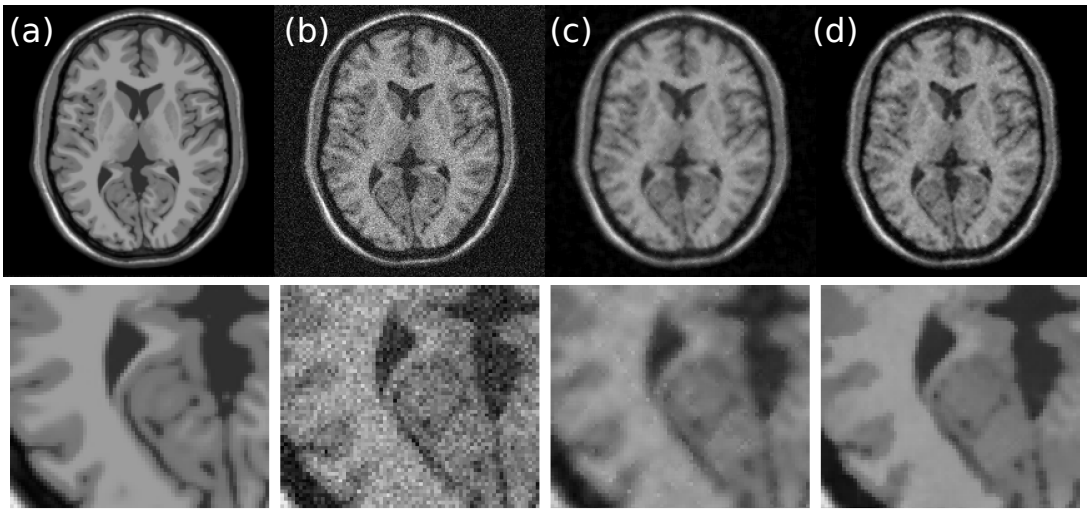


Figure 25: Example filtering techniques over a T_1 MRI synthetic image. (a) Original image; (b) Image corrupted with Rician noise ($\sigma = 20$); (c) LMMSE estimator for Rician noise; (d) Unbiased NLM.

One example of filtering can be found in Fig. 25 for a synthetic image. The filter may show a low visual filtering capability but note that it is a conservative scheme: it does not eliminate relevant information of the signal together with the noise. As a result, visual results are always less appealing than other methods, since it keeps some of the noise. As a counterpart, relevant information is not removed.

6.2 Non–local Means schemes

The non–local means (NLM) scheme was firstly described in [3] to denoise 2D natural images corrupted by an additive white Gaussian noise. This methodology is currently very popular due to its excellent visual performance. NLM is a non–linear filter based on a weighted average of pixels inside a search window that is relatively large compared to traditional neighborhood techniques. The structure of the image is preserved by applying an adaptive weight according to a similarity measure (usually the mean squared difference for natural images).

In its original formulation, the output of a NLM filter is computed as follows [3]:

$$I_{\mathrm{NLM}}(\mathbf{x}) = \sum_{\mathbf{y}\in\Omega} w(\mathbf{x},\mathbf{y}) I(\mathbf{y}) \tag{57}$$

where w is a set of weights computed as:

$$w(\mathbf{x},\mathbf{y}) = \frac{1}{Z(\mathbf{x})} \exp\left(-\frac{d(\mathbf{x},\mathbf{y})}{h^2}\right), \; Z(\mathbf{x}) = \sum_{\mathbf{y}\in\Omega} \exp\left(-\frac{d(\mathbf{x},\mathbf{y})}{h^2}\right) \tag{58}$$

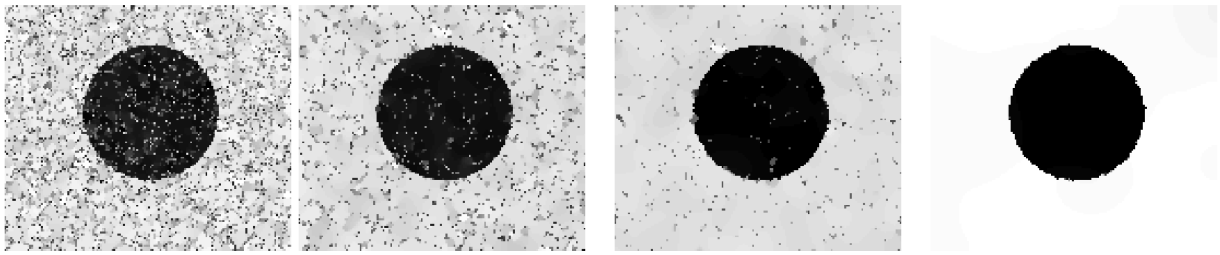
h is a parameter related to the noise power in the image and $d(\mathbf{x},\mathbf{y})$ is a distance between the voxels at positions p and q. Instead of using a geometrical distance, NLM uses a distance in the domain of the gray levels of the image, defined as:

$$d(\mathbf{x},\mathbf{y}) = \left(\mathbf{I}(\mathcal{N}_{\mathbf{x}}) - \mathbf{I}(\mathcal{N}_{\mathbf{y}})\right)^T G_\rho \left(\mathbf{I}(\mathcal{N}_{\mathbf{x}}) - \mathbf{I}_i(\mathcal{N}_{\mathbf{y}})\right) \tag{59}$$

where $\mathbf{I}(\mathcal{N}_{\mathbf{x}})$ and $\mathbf{I}(\mathcal{N}_{\mathbf{y}})$ are column vectors containing the gray values of the voxels in the neighborhoods $\mathcal{N}_{\mathbf{x}}$ and $\mathcal{N}_{\mathbf{y}}$ of voxels $\mathbf{x}$ and $\mathbf{y}$ respectively. G_ρ is a matrix that accounts for a Gaussian weighting that gives a higher weight to the voxels of the neighborhood closer to the central voxel.

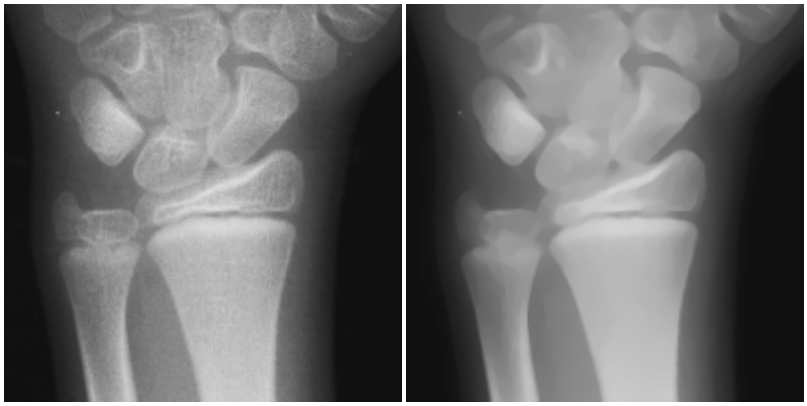
The computational cost associated to eq. (57) is prohibitive, so the domain Ω is usually substituted by a neighborhood $\mathcal{N}'_{\mathbf{x}}$ of voxel $\mathbf{x}$. Besides, it is proposed in [10] to change the weight $w(\mathbf{x},\mathbf{x})$ in eq. (58) by the maximum of $w(\mathbf{x},\mathbf{y})$ with $\mathbf{x} \neq \mathbf{y}$, to avoid over–weighting the central voxel of $\mathcal{N}'_{\mathbf{x}}$. A similar procedure is applied to the central coefficient of G_ρ. One example of filtering of the this scheme for MRI can be found in Fig. 25.

6.3 Anisotropic diffusion filters



(a) 150 iter. (b) 300 iter. (c) 500 iter. (d) 1000 iter.

Figure 26: Example of anisotropic diffusion filter for a noisy synthetic image. A small step is considered, so the diffusion takes place slowly. The synthetic nature of the image favor the goodness of the result.



(a) Original (b) 100 iter.

Figure 27: Example of anisotropic diffusion filter for a real image: a radiography of hand and wrist.

There is a huge family of filtering techniques based of Partial Differential Equations (PDEs). Among them, the most well-known is the so-called anisotropic diffusion (AD) filtering. The cornerstone in AD is the heat diffusion equation [14]:

$$\frac{\partial I(\mathbf{x},t)}{\partial t} = \text{div}(\mathbf{D} \cdot \nabla I(\mathbf{x},t)) \tag{60}$$

where a noisy image $I(\mathbf{x})$ feeds the PDE with initial condition $I(\mathbf{x}, t = 0) = I(\mathbf{x})$ and the temporal evolution is represented by variable t . Such a temporal evolution is governed by the divergence and gradient operator, denoted by div and ∇ respectively, and importantly, by $\mathbf{D}$, a symmetric positive definite tensor that depends on the local structure of the filtered image at time t, that is, $I(\mathbf{x}, t)$. When $\mathbf{D}$ becomes a tensor of order zero, i.e., a scalar function, the filter is usually termed as *isotropic non-homogeneous* diffusion filter. On the other hand, regarding the two-order tensor case, that is, when $\mathbf{D}$ is represented by a matrix, the term anisotropic diffusion filter is usually adopted [14].

Most of diffusion filters are modifications of the work of Perona and Malik [12] and its practical implementation by Gerig [5]. Both works are focused on a scalar diffusion coefficient $\mathbf{D} = c(\mathbf{x}, t)$. The method attempts to avoid diffusion closing to the boundaries (edges should be preserved) while filtering (diffusion) should be encouraged in homogeneous areas. To that end, since a natural approach to detect edges is looking at the gradient, the diffusion coefficient $c(\mathbf{x}, t)$ is defined as a decreasing function, $g(\cdot)$, of $\nabla I(\mathbf{x}, t)$

$$c(\mathbf{x},t) = g(||\nabla I(\mathbf{x},t)||). \tag{61}$$

where $||\cdot||$ is a prescribed norm, commonly the l_2 norm. When $\nabla I(\mathbf{x}, t) \to \infty$, $c(\mathbf{x}, t) \to 0$, since $c(\mathbf{x}, t)$ is, by construction, non-negative. Thus, $\frac{\partial I(\mathbf{x},t)}{\partial t} \to 0$, and smoothing is not applied. For practical implementation, $g(.)$ is usually defined as a decreasing function of $||\nabla I(\mathbf{x}, t)||$, as for instance:

$$\begin{aligned} g_1(||\nabla I(\mathbf{x},t)||) &= \exp\left(-\left(\frac{||\nabla I(\mathbf{x},t)||}{K}\right)^2\right) \\ g_2(||\nabla I(\mathbf{x},t)||) &= \left(1+\left(\frac{||\nabla I(\mathbf{x},t)||}{K}\right)^2\right)^{-1} \end{aligned}$$

where K is the *diffusivity parameter*, which plays the role of thresholding mechanism in order to control the sensitivity of edge detection. Under this formulation, the parameter K must be manually selected. Two practical examples of AD filtering can be seen in Fig. 26 and Fig. 27.

References

[1] Aja-Fernández, S., Alberola-López, C., Westin, C.F.: Noise and signal estimation in magnitude mri and rician distributed images: a lmmse approach. IEEE Tr. on Image Processing **17**(8), 1383–1398 (2008)

[2] Aja-Fernández, S., Vegas-Sánchez-Ferrero, G.: Statistical analysis of noise in MRI. Springer International Publishing AG (2016)

[3] Buades, A., Coll, B., Morel, J.M.: A review of image denoising algorithms, with a new one. Multiscale Modeling & Simulation **4**(2), 490–530 (2005)

[4] Burckhardt, C.B.: Speckle in ultrasound b-mode scans. IEEE Transactions on Sonics and ultrasonics **25**(1), 1–6 (1978)

[5] Gerig, G., Kübler, O., Kikinis, R., Jolesz, F.: Nonlinear anisotropic filtering of MRI data. IEEE Tr. on Medical Imaging **11**(2), 221–232 (Jun 1992)

[6] Gonzlez, R., Woods, R.: Digital Image Processing. Addison & Wesley, 3rd. edn. (2011)

[7] Jain, A.K.: Fundamentals of digital image processing. Englewood Cliffs, NJ: Prentice Hall, (1989)

[8] Kay, S.M.: Fundamentals of statistical signal processing. Volume I: Estimation theory. Prentice-Hall (1993)

[9] Lim, J.S.: Two-dimensional signal and image processing. Prentice Hall, Englewood Cliffs, NJ (1990)

[10] Manjón, J., Carbonell-Caballero, J., Lull, J., García-Martí, G., Martı-Bonmatı, L., Robles, M.: MRI denoising using Non-Local Means. Medical Image Analysis **12**, 514–523 (2008)

[11] Pearson, K.: Contributions to the mathematical theory of evolution. ii. skew variation in homogeneous material. Philosophical transactions of the Royal Society of London **186**(Part I), 343–424 (1895)

[12] Perona, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion. IEEE Tr. on Pattern Analysis and Machine Intelligence **12**(7), 629–639 (Jul 1990)

[13] Pratt, W.K.: Digital Image Processing. Wiley, New York, NY, USA, 3rd edn. (1991)

[14] Weickert, J.: Anisotropic Diffusion in image processing. Teubner-Verlag, Stuttgart, Germamy (1998)