Image Analysis Human vision allows us to perceive and interpret the world which surrounds us. Artifical vision aims at reproducing some of the human vision functionalities through *Image Analysis*. It is a difficult problem in part due to the fact that the available input data, images acquired with digital sensors, correspond to 3D-2D projections of the environment. These projections cause a loss of information. Moreover, images are not perfect as a result of various perturbations, *e.g.* sensors digitazing and optical deformations, and processing such noisy information appears to be a sensitive operation.

Image analysis is composed of several fields that can be roughly classified into three main categories:
1

- **Low level** image processing that requires few knowledge on the image content. It includes filtering, image improvement, image restoration, among others. We will abusively group under the term *image processing* all these low level operations.
- Intermediate level processing where some prior knowledge on image contents is assumed and such as feature extraction.
- High level processing that usually relies on lower level processing and such as reconstruction, recognition and cognitive processes in general.

Image (pre-)processing concerns all the methods aiming at improving the characteristics of an image (i.e. the input and the output are images).

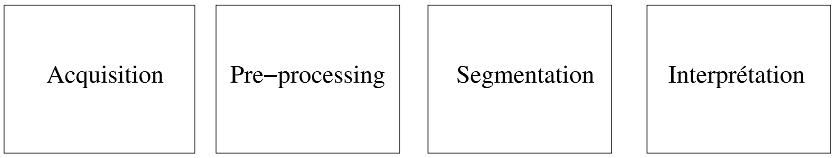


Figure 1: Image analysis is a data processing chain.

- Local filtering: The objective is to reduce noise, or small intensity variations, in an image. A Pixel value is modified by considering the pixel values over a neighborhood.
- Image improvement consists in modifying the visual characteristics of an image, e.g. its contrast, to ease its interpretation by humans. Histogram modifications are often used for that purpose.
- Image restoration aims at suppressing degradations in the image using prior knowledge on these degradations or on *good* images. For instance, numerous image denoising applications assume that the intensity gradient varies sparsely in a noise free image (Total Variation or TV denoising).

1 Local Filtering

Local filtering or smoothing consists in replacing a pixel value by a weighted combination of the values in a local neighborhood of the pixel. Weights depends then on the type of filtering that is desired.

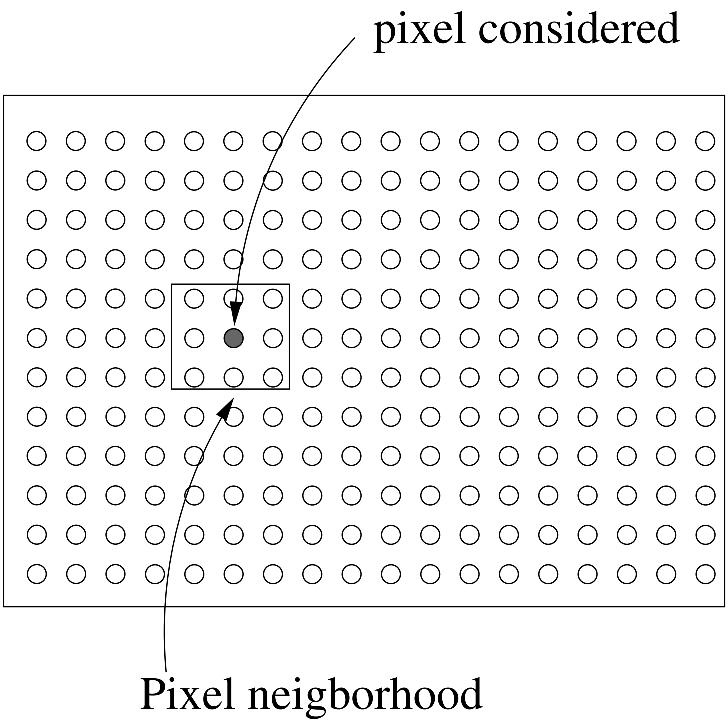
1.1 Averaging

A straightforward strategy assumes that the image information is redundant over local neighborhoods. The new value of a pixel is then simply the average of values. This linear operation can be seen as a discrete convolution of the image with a mask.

$$I'(i,j) = \sum_{(m,n)\in\mathcal{N}} h(m,n)\, I(i-m, j-n),$$

$$\sum_{(m,n)\in\mathcal{N}} h(m,n) = 1,$$

where I is the intensity function in the original image, I' is the intensiyt function in the filtered image, $\mathcal{N}$ is the local neighborhood considered and h the convolution mask.



the average mask over a 3x3 neighborhood is:

$$h = 1/9 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

☞ Averaging is a low-pass filter.

☞ It blurs the image, especially contours.

☞ It suppresses local small degradations which is valid only when objects in the images are of dimensions greater than these degradations.

In order to improve the average filter behavior, values in the associated mask can be modified, for instance increasing the influence of the central and current pixel:

$$h_1 = 1/10 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

This also true for the well known binomial filters whose coefficient values are obtained from Pascal's triangle (convolve $[11]$ with itself and iterate):

$$h_2 = 1/16 \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

☞ The binomial filter can be seen as a discrete approximation of a Gaussian filter that is thus well adapted to Gaussian noises.

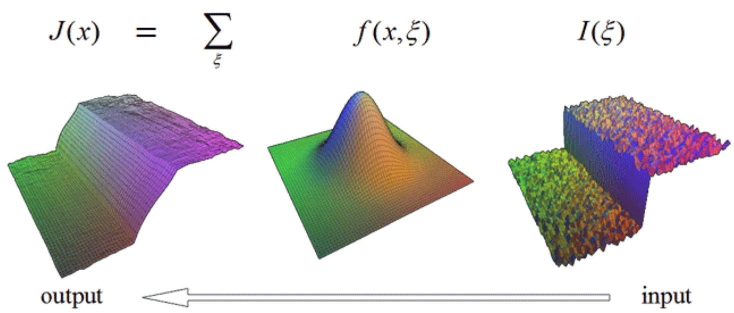


Figure 2: Gaussian filtering a step (image from Shang Shou, Communication and Multimedia Laboratory,Taiwan)

1.2 Median Filter

Average filters tend to blur images and consequently to remove information on contours since they correspond to strong variations of the intensity function that are reduced by the filter. In order to limit this effect, the median does not average over the neighborhood but takes instead the median value over this neighborhood.

Example over the following 3x3 neighborhood:

$$I = \begin{bmatrix} 2 & 12 & 12 \\ 2 & 12 & 60 \\ 2 & 2 & 12 \end{bmatrix}$$

The central pixel value will be replaced here by the median value which is 12.

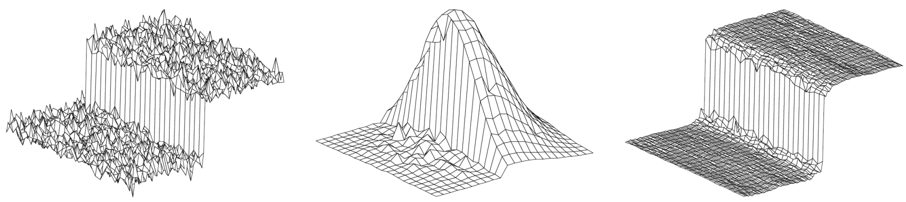
☞ Non linear filter !

☞ Very good at suppressing impulse noise (despeckle filter).

- ☞ Preserve the contour information and can be applied iteratively, in contrast to average filters that progressively remove contours when applied iteratively.
- ☞ Remove anyway thin contours.

1.3 Bilateral Filter

With the objective to improve local image filtering, and in particular to preserve edges, numerous other filters have been proposed. Of particular interest is the bilateral filter introduced by Tomasi and Manduchi in 1998. It is a non linear Gaussian filter that tend to preserved edges by weighting pixel contributions with respect to their similarity to the central pixel. Hence, the average is mainly performed over pixels with a similar appearance in the neighborhood.



(a) (b) (c)

Figure 3: Bilateral filter (image taken from Tomasi and Manduchi ICCV'98): (a) A gray level step perturbed with Gaussian noise; (b) Bilateral filter coefficients h for a 23x23 neighborhood; (c) The step in (a) after filtering with h.

Recall the local filtering of I into I':

$$I'(i,j) = \sum_{(m,n)\in\mathcal{N}} h(m,n)\; I(i-m, j-n),$$

with:

$$\sum_{(m,n)\in\mathcal{N}} h(m,n) = 1.$$

Let $k()$ and $l()$ be two Gaussian kernels:

$$k(d) = \exp -\frac{1}{2}(\frac{d}{\sigma_k})^2,\; l(d) = \exp -\frac{1}{2}(\frac{d}{\sigma_l})^2.$$

Then the coefficients of the bilateral filter h for the pixel (i,j) are:

$$h(m,n) = \frac{1}{c} k(\sqrt{m^2+n^2})\; l(\sqrt{(I(i,j) - I(i-m, j-n))^2}$$

with:

$$c = \sum_{(m,n)\in\mathcal{N}} k(\sqrt{m^2+n^2})\, l(\sqrt{(I(i,j) - I(i-m, j-n))^2}$$

- ☞ The kernel l reduces the influence of pixels having intensities different from the central pixel (i, j) (i.e. $h(m, n)$ small in that case)
- ☞ Contours are preserved since only pixels with similar appearance impact the weighted average.

Figure 4: Original image and its noisy version.

Figure 5: Local filtering: 3x3 Gaussian filter, 1 and 2 iterations.

Figure 6: Local filtering: 5x5 Gaussian filter, 5x5 bilateral filter.

Figure 7: Local filtering: 3x3 median filter, 1 and 2 iterations.

2 Image Improvement

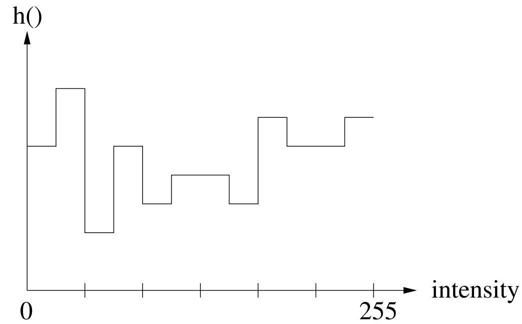
Image improvement consists in modifying the visual properties of an image with the aim of easing its interpretation by humans. Examples include increasing the contrast, sharpening regions through intensity modifications, etc. To this purpose, histograms are frequently used to manipulate intensities.

2.1 Histograms

The histogram of an image is a function $h(x)$ that associates to each intensity value x the number of pixels in the image having this intensity value.

Algorithm

```
int row,col,rowmax,colmax;
int count[GREYMAX];
for(row=0;row<rowmax;row++)
 for(col=0;col<colmax;col++)
   h[image[row][col]]++;
```



☞ For a color image with 3 components per pixel, the histogram that associates to each color the number of corresponding pixels is a 3 dimensional surface within a 4 dimensional space. In practice, 3 mono-dimensional histograms, one per component, are usually considered.

Figure 8: Color histograms.

☞ The histogram can be normalized in which case it gives an estimation of the probability density $p(i)$ for each intensity:

$$p(i) = hist(i) / \sum_{j=0}^{Max} hist(j),$$

$$\sum_{i=0}^{Max} p(i) = 1.$$

☞ The range of intensity values $[i_{min}, i_{max}]$ of an image is called the dynamic of the image. This dynamic can be changed through transformations as explained below.

2.2 Histogram Modifications

In order to modify the image properties (contrast in most applications), a general strategy is to associate to each intensity in the image a new value. This transformation function that applies to intensity modifies the image histogram and can be defined with respect to this modification.

We consider the transformations T of the form:

$$i' = T(i),$$

where $i, i' \in [0, Max]$ and Max is the maximum intensity value, i.e. 255 in general. This transformation T is usually assumed to be such that:

- $T(i)$ is monotonic (often increasing) on the intensity interval. This constraint garanties that the intensity order in the image is preserved after transformation.
- $0 \leq T(i) \leq Max$ for $0 \leq i \leq Max$ to enforce the new image to be consistent with the intensity range.
- The inverse transformation satisfies the two above conditions.

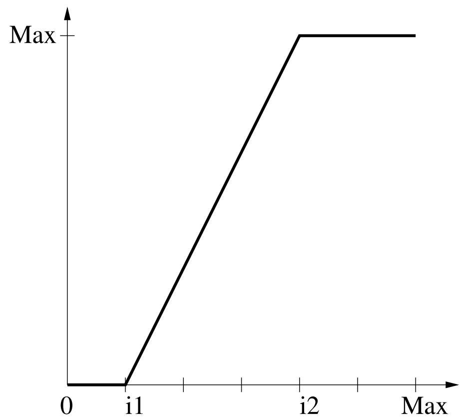
A few illustrative examples of histogram modifications follow.

2.3 Linear Transformations

Linear histogram modifications are numerous and diverse. They can stress an area with a specific intensity or modify the intensity value distribution. An example of such transformation is the histogram stretching.

The histogram stretching (also called normalization) consists in modifying the intensity range (the dynamic) so that the transformed image presents a larger dynamic. If $[i1, i2]$ is the current dynamic of the image, the linear transformation T that maximizes the dynamic is then:

$$T(i) = Max\ (i - i1)/(i2 - i1).$$



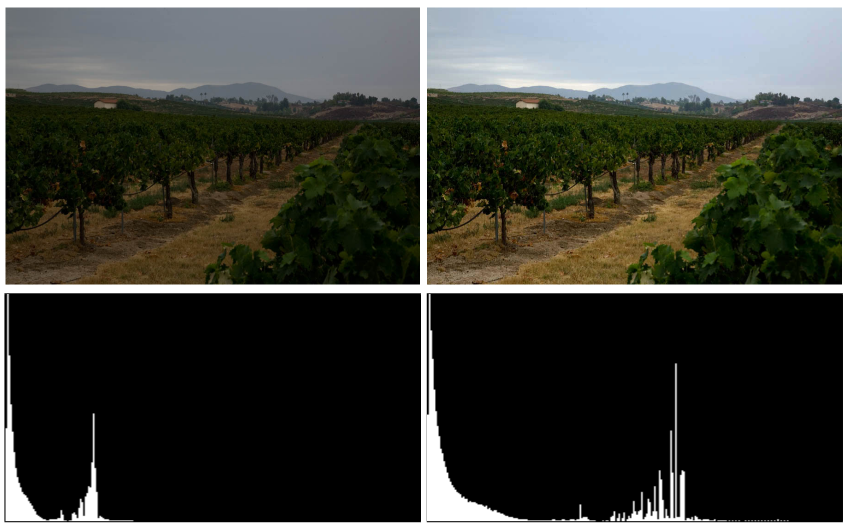


Figure 9: Histogram stretching.

Other linear histogram transformations:

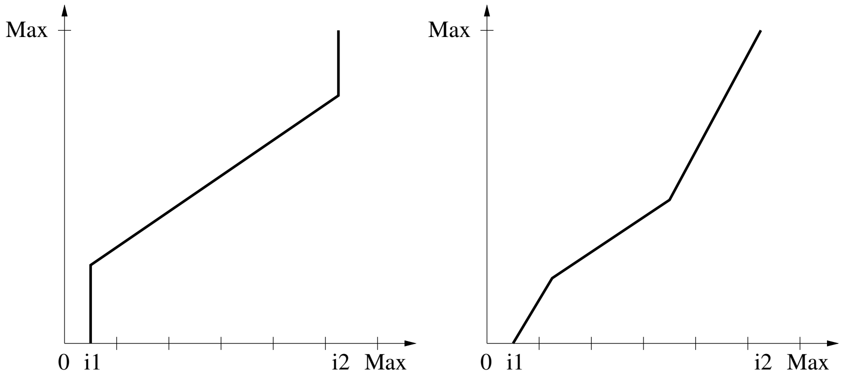
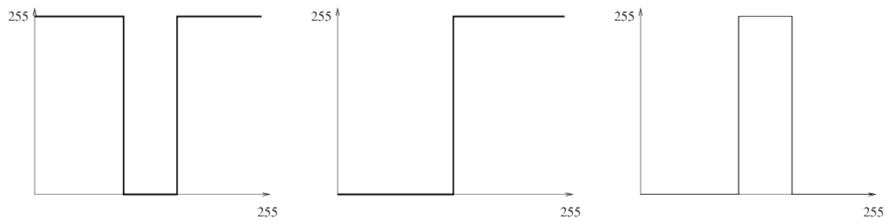


Figure 10: Histogram compression and piewise linear adjustment.

2.4 Exercise

(a) (b) (c)



(1) (2) (3)

Figure 11: Histogram modifications.

- Each tiger image (a), (b) and (c) above is a modification of the original tiger image on the left with respect to one of the histogram transformations (1), (2) or (3). Find the correct association between these images and the transformations.

2.5 Non-linear Transformation

A more sophisticated and widely used transformation to improve the intensity distribution in an image is the histogram equalization. The idea is to transform the intensity so that the histogram becomes as flat as possible, hence yielding a more uniform distribution of intensities. Information theory says that it maximizes then the image entropy and consequently leads to a more informative image.

Assume that the intensity distribution is uniform, i.e. $p(i) = cst$, then the cumulative distribution $F(i) = \sum_{j=0}^{i} p(j)$ is a line with a constant slope (the red line in figure 2.5 below). In practice this cumulative function is arbitrary (though increasing) and the equalization consists in projecting the current F onto the optimal F^* one. For each intensity value i, the new value i^* is then obtained by projecting the point associted to i onto the optimal cumulative distribution (the arrow in the figure 2.5) and taking the corresponding intensity i^*.

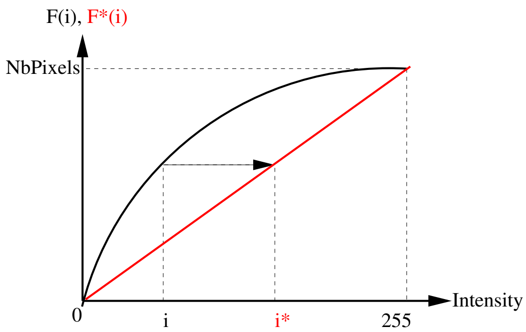


Figure 12: Histogram equalization: F is the current cumulative distribution function of intensities i and F^* is the targeted one.

In detail, the slope of the optimal cumulative distribution F^* writes:

$$slope = \frac{NbPixels}{Max} = \frac{\sum_{i=0}^{Max} hist(i)}{Max}.$$

Thus:

$$i^* = T(i) = \frac{F(i)}{slope} = Max \frac{\sum_{j=0}^{i} hist(j)}{\sum_{j=0}^{Max} hist(j)},$$

- ☞ Note that due to the digitalization, the resulting histogram is not flat.
- ☞ For an image globally bright, the equalization will increase the dynamic of the dark part of the image against the bright part dynamic, and vice versa.

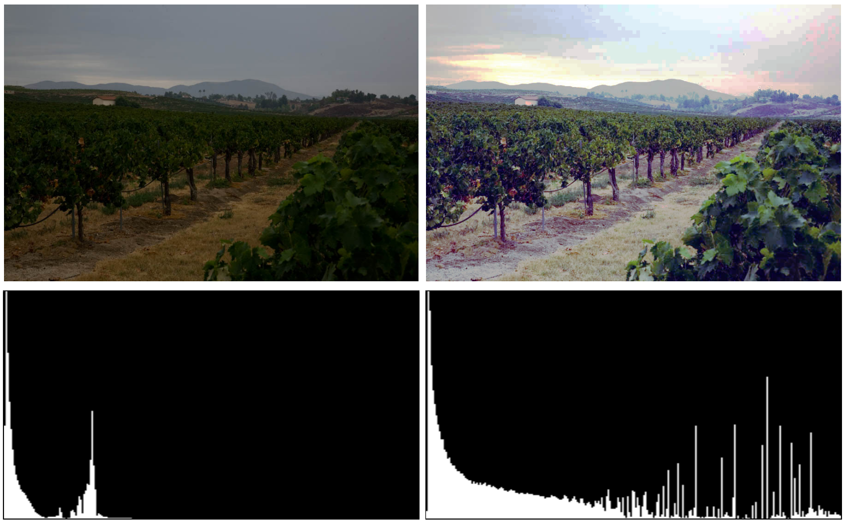


Figure 13: Histogram equalization.

☞ it allows to make image comparison on a similar basis.

☞ Equalization can be performed per region in the image and is called adaptive in that case.