# TD2-Transformation

Professor: Marc Antonini (am@i3s.unice.fr)

Lab assistant: Anca Grapa (agrapa@i3s.unice.fr)

The purpose of this practical session is to perform an analysis-synthesis type of transformation of an image with Haar wavelets and Laplacian Pyramid.

## 1. Haar Wavelet

The Haar filter was proposed in 1909 by Alfréd Haar. It is the simplest orthogonal wavelet transform. Haar filter relies on averaging and differencing values of a matrix making a bid for producing another sparse or nearly sparse matrix.

A sparse matrix is a matrix in which a large portion of its entries are 0. A sparse matrix can be stored in an efficient manner, leading to smaller file sizes. Below you will find an example of the general pipeline for 1D signal analysis and synthesis.

Example of 1D signal analysis :

1. Input signal :
$$A = [88\ 88\ 89\ 90\ 92\ 94\ 96\ 97]$$
2. Group the coefficients :
$$A_g = [88\ 88]\ [89\ 90]\ [92\ 94]\ [96\ 97]$$
3. Compute the average of each group (**approximation coefficients**).
4. Compute half of the difference of each group (**detail coefficients**).
5. Store in a new vector :
$$A_1 = [88\ 89.5\ 93\ 96.5\ 0\ -0.5\ -1\ -0.5]$$

In order to perform a multiscale decomposition we iterate steps (2)-(4) using as an input the averaged values (the first half of the signal). For example, after a subsequent level of decomposition, the values of the signal should be the following :

$$A_2 = [88.75\ 94.75\ \ -0.75\ \ -1.75\ 0\ \ -0.5\ \ -1\ \ -0.5]$$

In the case of a 2D signal, we apply the steps to each row and column of the input image.

Example of 1D signal synthesis :

1. Input signal :

$$A_h = [88\ 89.5\ 93\ 96.5\ 0\ \ -0.5\ \ -1\ \ -0.5]$$

2. Upsample the average values and use the differences to reconstruct :

$$A_h = \left[\underbrace{88\ 88}_{\mp 0}\ \underbrace{89.5\ 89.5}_{\mp 0.5}\ \underbrace{93\ 93}_{\mp 1}\ \underbrace{96.5\ 96.5}_{\mp 0.5}\right]$$

3. Output signal identical to the original one :

$$\tilde{A} = [88\ 88\ 89\ 90\ 92\ 94\ 96\ 97] = A$$

**Requirement** : Using your development environment of choice, perform Haar wavelet transform (analysis-synthesis) of the 2D images you're provided with. The Figure 1 illustrates the first level of a Haar decomposition of an image during the analysis step. You can use the above 1-D example as a guideline, paying attention to apply the suggested pipeline to every row and column alternatively.
If we consider a square matrix $NxN$, the total number of decomposition levels for a Haar operator is equal to $\log_2(N) - 1$. Knowing that, perform the Haar transformation on multiple scales.

**Side note** : The above process could be implemented as a convolution between the input signal and a set of lowpass and highpass filters which result in averaging and differencing the values of each group. Similarly, for a 2D signal, we apply the low-pass and high-pass filter to each row and column of the input image.

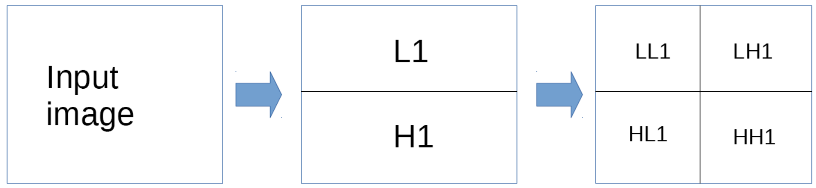


FIGURE 1 – Haar 2D decomposition on first level : columns and then rows. (We denote $L$ as corresponding to low-frequency, see approximation coefficients and $H$ as high-frequency, see detail coefficients).

# 2. Laplacian Pyramid

The Laplacian pyramid was introduced by Burt and Adelson in 1983. It is very similar to Gaussian pyramid but more efficient from the compression point of view.

Example of 2D signal analysis :

— Build a 2D Gaussian Filter $g(x, y)$ (e.g. using $3x3$, $5x5$ Gaussian convolution kernels)

— Convolve "lena.jpg" with the Gaussian filter, to obtain a smoothed (low-pass filtered) version, $\mathbf{G}$.

— Downsample ($2 \downarrow$) the rows and columns of $\mathbf{G}$ (e.g. keep only the odd indexes of both rows and columns), to reduce its size, and obtain $\mathbf{R}$, the next level of the Gaussian pyramid. In order to compute the Laplacian pyramid, you need to subtract every 2 sequential layers. To achieve this, you will upsample $\mathbf{R}$.

— Iterate steps (2)-(3) in order to build $\log_2(N)-1$ Laplacian decomposition layers, paying attention to use $\mathbf{R}$ obtained at previous level as the input for Gaussian filtering.

Example of 2D signal synthesis :

1. Upsample the last $R$ from your decomposition
2. Add the corresponding previous Laplacian decomposition and iterate the above process until reconstruction.