MSc Laboratory Experiment
Discrete Cosine Transform vs Wavelet Transform

1 Preliminaries

The main aim of this experiment is to make the student familiar with some typical image processing applications and also to give him/her an idea of why wavelets are better than other commonly used bases such as the Discrete Cosine Transform.

All the numerical experiments will be performed on the grayscale image 'Cameraman'. Please check with the lab staff how to access this file. To run the experiments, students can use either the Matlab wavelet toolbox or the free software 'Wavelab802', or can write any piece of software themselves.

To read the image 'Cameraman' in Matlab, use the command A=imread('Cameraman.gif') and to view the image use the commands 'Imagesc' and 'Colormap'. Notice that Matlab treats an image as a matrix A whose value at a certain location (i, j) indicates the luminance of the image pixel at that location.

2 The Discrete Cosine Transform (DCT)

The previous image compression standard JPEG was based on the Discrete Cosine Transform.

Given a column vector X of size N. The DCT coefficients of X are $Y = UX$ where the element $u_{i,j}$ of the DCT matrix U is given by:

$$u_{i,j} = \begin{cases} \sqrt{\frac{1}{N}} \cos(\frac{\pi}{N}(i-1)(j-\frac{1}{2})) & i = 1 \\ \\ \sqrt{\frac{2}{N}} \cos(\frac{\pi}{N}(i-1)(j-\frac{1}{2})) & i = 2, ..., N \end{cases}$$

In JPEG the image is divided into blocks of 8×8 pixels and the 2-D version of the DCT is applied on each block. Notice that the 2-D version of the DCT is nothing else than the 1-D DCT transform applied to columns first and rows then.

Exercise 1 *Write a Matlab programme that, given the input image, computes the 2-D DCT transform on each block of size* 8×8. *Display the transformed image.*

Write a second programme that inverts the DCT and make sure that you are actually able to reconstruct the original image from the transformed one exactly.

3 The Wavelet Transform (WT)

The standard 2-D wavelet transform simply consists of two 1-D transforms applied along the rows first and the column then of the image. A block diagram of the standard 2-D wavelet transform is illustrated in Figure 1. The filter L_0 is usually a low pass filter while H_0 is high pass. The process is usually iterated on the 'low-low' pass version of the image. The resulting transformed image after three iterations is shown in Figure 2.

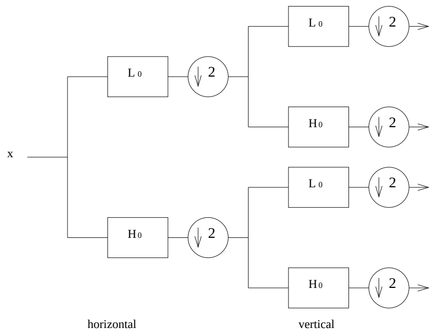

Figure 1: Separable wavelet transform: block diagram

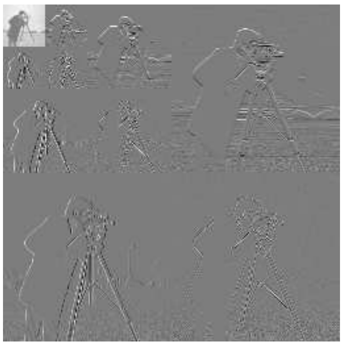

Figure 2: Three iterations of the separable wavelet transform

Exercise 2 *Write a programme that computes l levels of the 2-D WT on Cameraman, where l can be chosen arbitrarily. Use as wavelet filters either 'Haar' or Daubechies with two vanishing moments (i.e., two zeros at π, this is known as 'db2' in the wavelet toolbox). Make sure, if necessary, that periodic extension is used to deal with border effects (Use the command 'Dwtmode' in the wavelet toolbox).*

Display the result for $l = 2$ and $l = 4$.

Exercise 3 *Write a programme that computes the 2-D Inverse Wavelet Transform (IWT). Make sure that you can actually reconstruct Cameraman exactly.*

4 Non-linear Approximation (NLA)

Consider a signal $f(t) \in L_2(\mathbb{R})$ and an orthogonal basis $\varphi_i(t)$, $i \in \mathbb{Z}$. We can write $f(t)$ as

$$f(t) = \sum_{i=-\infty}^{\infty} \langle f, \varphi_i \rangle \varphi_i(t).$$

Now, assume that we select only N elements of $\{\varphi_i\}$ to represent f, we have that the approximated version of f is given by

$$\hat{f}_N(t) = \sum_{i \in I_N} \langle f, \varphi_i \rangle \varphi_i(t)$$

where I_N is the set of selected indices. The squared error due to this approximation is given by

$$\|f(t) - \hat{f}_N(t)\|^2 = \sum_{i \notin I_N} |\langle f, \varphi_i \rangle|^2$$

Clearly, to minimize this error, the indices in I_N must correspond to the N vectors having the largest inner product amplitudes $|\langle f, \varphi_i \rangle|$. This form of approximation is called non-linear since it depends on the input signal.

Now, assume that we have access to two different orthogonal bases $\{\varphi_i\}$ and $\{g_i\}$, one way to compare these two bases is by checking which one provides a better NLA of a signal f for different values of N. Clearly a basis with better approximation properties is more suited for compression.

Exercise 4 *Compare the non-linear approximation properties of the DCT and WT.*

Compute the 8×8 DCT and the 2-D WT (with 6 decomposition levels) of Cameraman. Choose different values of N (i.e., $N = 100$, $N = 200$ ect.). For each chosen value of N, keep only the N largest absolute values of the transformed coefficients. That is, set to zero all the wavelet or discrete cosine coefficients except for the N largest ones. Reconstruct the image.

Compare the PSNR for the two cases. Notice that $PSNR = 10\log_{10}\frac{255^2}{MSE}$ where $MSE = \frac{1}{M^2}\sum_{i,j}(x_{i,j} - \hat{x}_{i,j})^2$ and M^2 is the number of pixels in the image while $x_{i,j}$ indicates the value of the pixel at that location. Draw a plot whose x-axis is given by N and the y-axis by the values of the PSNR for the two cases. Is the WT performing better than the DCT?

Exercise 5 *Repeat the previous experiment but choose different wavelet bases and try different numbers of decomposition levels. Try to justify the results you obtain.*

5 Denoising

The term denoising usually refers to the removal of noise from a corrupted signal. In the typical scenario, the original discrete signal $x[n]$ has been corrupted by additive Gaussian noise and one observes $y[n] = x[n] + e[n]$ where $e[n]$ are i.i.d. zero mean Gaussian variable with variance σ^2. One way to remove the noise is by computing the wavelet or cosine transform and shrinking the wavelet or cosine coefficients which are below a certain threshold. That is, all the coefficients whose amplitude is below a threshold T are set to zero.

Exercise 6 *Using the Matlab command 'randn', add Gaussian noise to the image Cameraman. Consider three possible noise variances $\sigma^2 = 10$, $\sigma^2 = 100$, $\sigma^2 = 400$. Display the three noisy images. For each choice of the variance, compute the PSNR between the original image x and the noisy one y.*

The choice of the threshold T is usually very critical. An empirical rule says that a good value is $T = 3\sigma$.

Exercise 7 *Compute the WT (6 decomposition levels and Daubechies with two vanishing moments) and the DCT of each of the three noisy images of the previous exercise. Set to zero the transformed coefficients which are below the threshold T and reconstruct the images. Compute the PSNR between the original image x and the reconstructed one. Compare the performance of the WT and DCT.*