IMAGE PROCESSING (RRY025)

One of the Exams in 2010/2011

1 NOISE REDUCTION [22 points]

(a) [2p] Describe the averaging filter as a tool for noise reduction, discuss its advantages and disadvantages, and explain for which type of noise it can be used.

(b) [4p] Same questions as in **(a)**, but for the median filter. Additional questions. The averaging and median filters are based on two important statistics: the mean and the median. What information do such statistics provide? Which one is more 'robust'? And in what sense?

(c) [6p] Same questions as in **(a)**, but for the Wiener filter, assuming that the image has NO distortions (the point spread function is a δ function). Further questions. Illustrate how to construct the Wiener filter using the power spectrum of the noisy image. Wiener filtering is often called 'optimal' filtering. What does 'optimal' mean? Explain!

(d) [5p] In certain applications of image/signal processing, it is not good to suppress the noise. It would be better instead *to reduce its intensity without changing its frequency content.* For example, if you are in a pub and someone phones you, it would be nice 'to decrease the volume of the background noise', without changing the volume of your voice! It would not be nice to remove the background noise entirely. The person that has phoned you could think that you are in a library or in a church. You don't want that! Don't you? :-) Assume for simplicity that the noise is Gaussian and white, and added to the signal. How would you reduce its intensity without changing its frequency content? How would you increase the signal-to-noise ratio by a factor of α? …Cool, isn't it? In which other applications of image/signal processing would this type of noise reduction be of interest?

(e) [5p] A common form of noise is the so-called power-law noise. Its Fourier power spectrum varies on average as $1/f^{\alpha}$, where f is the frequency and α determines the 'colour' of the noise. For example, if $\alpha = 1$ we have 'pink' noise, and if $\alpha = -1$ we have 'blue' noise ($\alpha = 0$ corresponds to the well-known white noise). Consider then these two types of noise: pink and blue. Show what their power spectra look like. In general, which type of noise is more difficult to remove properly, the pink or the blue one? And why? Show now what their wavelet coefficients look like. If D_n is the set of detail coefficients at level n, and σ_n is their standard deviation, what is the relation between σ_{n+1} and σ_n for pink noise? And for blue noise? If you have two images, one polluted by pink noise and the other polluted by blue noise, how would you de-noise them? Are you sure? Or maybe you should make some more assumptions about the noise …

2 MISCELLANEA [8 points]

(a) [5p] You have an analog image, which you would like to digitize and then pre-compress using a transform (remember: to pre-compress means to set a number of coefficients to zero). Your scanner is very cheap and, unfortunately, the digitized image is of bad quality. The user guide warns you, in fact, that the scanning process produces artifacts at the sampling scale and at a scale twice as large, and that such artifacts are oriented vertically. You don't have enough money to buy a better scanner, so you are forced to use your skills of image processer. Find a smart way to pre-compress the digitized image and get rid of the artifacts at the same time. NOTE: The original analog image is of high importance to you. So, when you pre-compress the digitized image, you don't want to loose any further information than that already artifacted by the scanning. What is the resulting compression factor?

(b) [3p] In your opinion, what is the most interesting topic of the course? Explain how important this topic is in the context of image processing, and how important it is for your studies/job.

HELP TO SOME OF THE QUESTIONS

1 Noise Reduction

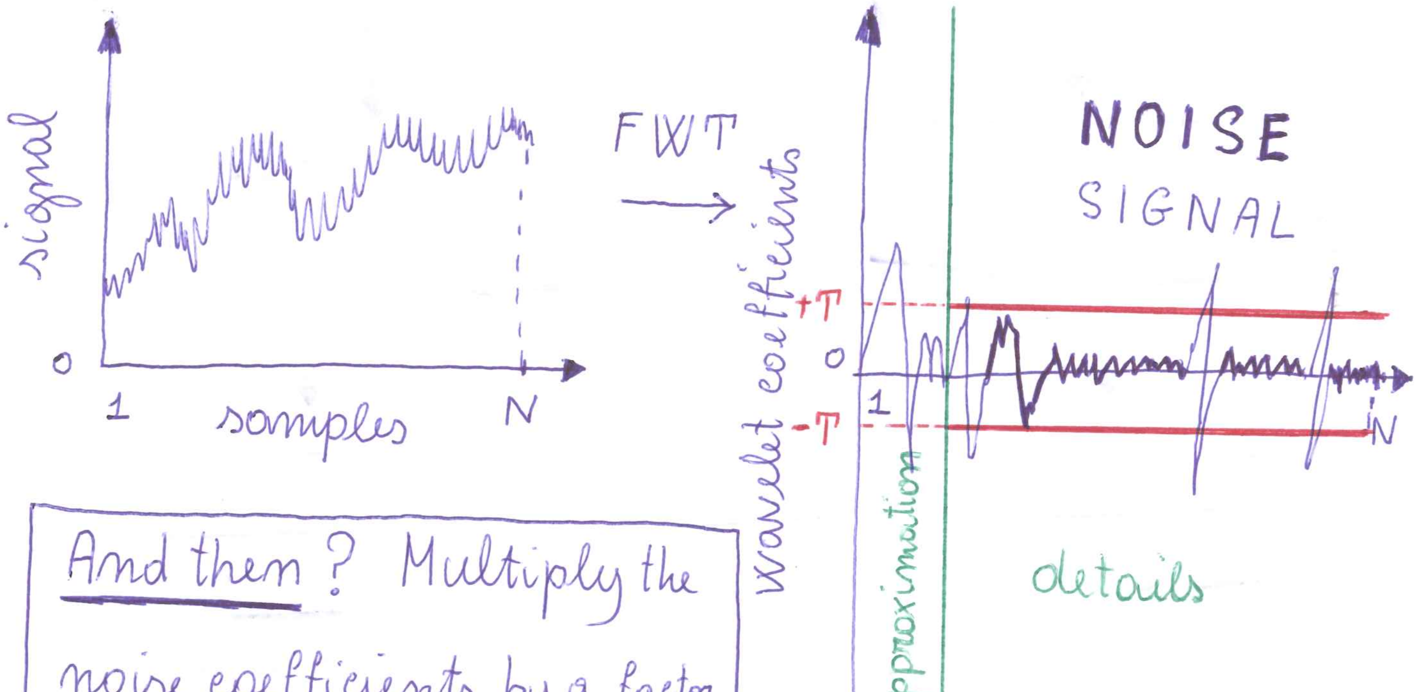
(c) ... assuming that the image has **NO** distortions!

(d) First of all, we should separate signal from noise.

In image space?... No!

In Fourier space?... Possible, but not so good...

In wavelet space?... YES, by thresholding!

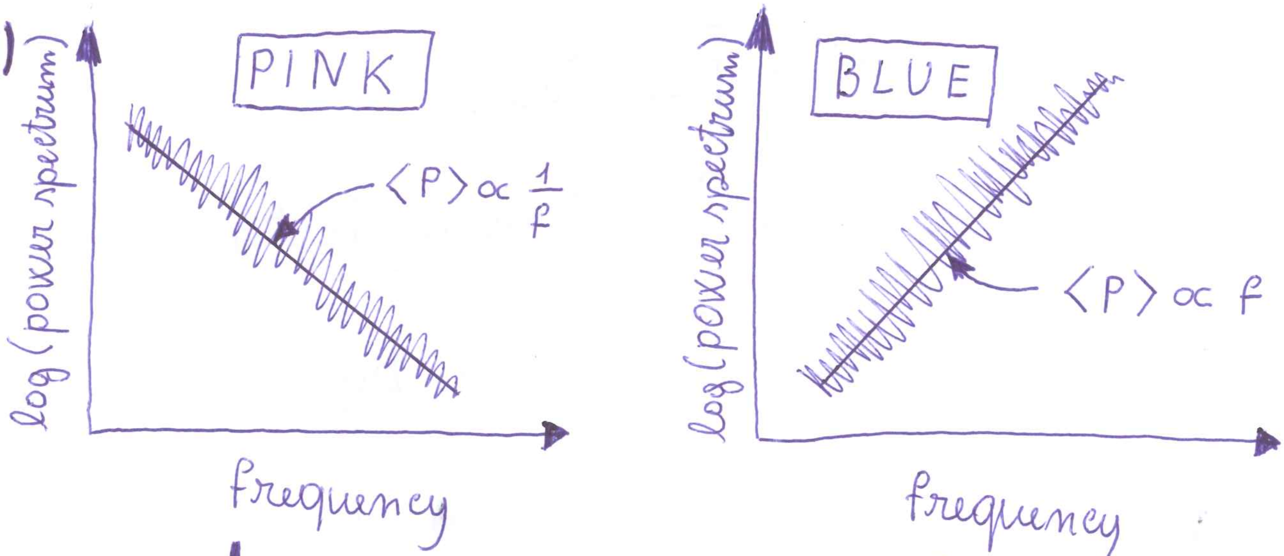


And then? Multiply the noise coefficients by a factor 1/2 and IFWT!

To think about:

- How can we determine the threshold?
  (The noise is Gaussian, white and additive ...)
- Why threshold only the detail coefficients?
  (Because the approximation ...)
- At which level should we FWT?
  (Size of the wavelet vs. size of the coarsest detail ...)

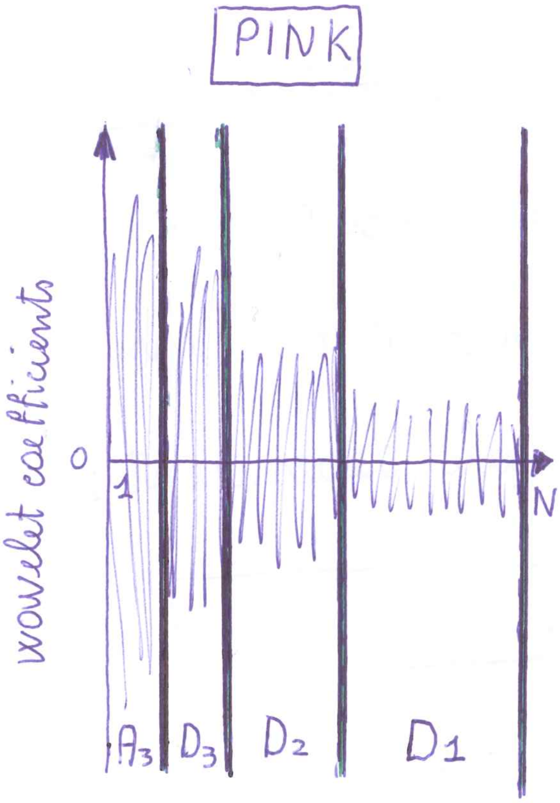
(e)

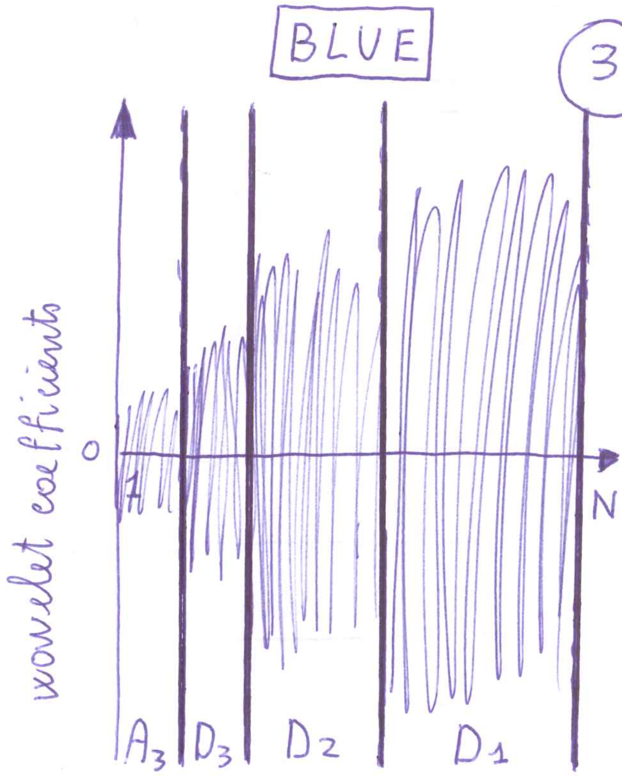


Pink	Blue
more power at <u>low</u> frequencies	more power at <u>high</u> frequencies

<u>Pink noise</u>: <u>strongly</u> coupled to the signal	The <u>signal</u>, in general, has more power at <u>low</u> frequencies	<u>Blue noise</u>: <u>weakly</u> coupled to the signal

Pink noise is more difficult to remove properly, in general.





Reflect now!

- ⟨Fourier power spectrum⟩ $\propto$

  mean square amplitude of the noise at <u>frequency</u> f

  $\propto 1/f^2$

- σ_n = standard deviation of D_n $\propto$

  <u>root</u> mean square amplitude of the noise at <u>scale</u>

  s_n ($= 2^{n-1} \times$ sampling scale)

- scale $\propto$ 1 / frequency

→ <u>Pink noise</u>:

$\sigma_{n+1} = \underbrace{\sqrt{2}}\ \sigma_n$

<u>Blue noise</u>: ←

$\sigma_{n+1} = \underbrace{1/\sqrt{2}}\ \sigma_n$

If the noise is Gaussian and additive, then we can remove it from an image by thresholding the detail coefficients. In contrast to the case of white noise, now the threshold is scale-dependent:

Pink noise:

$$T_{n+1} = \underbrace{\sqrt{2}}\, T_n$$

Blue noise:

$$T_{n+1} = \underbrace{1/\sqrt{2}}\, T_n$$

In both cases, T_1 can be determined as for white noise:

$$T_1 = \sqrt{2 \ln N}\, \underbrace{\sigma_1}$$

can be robustly estimated through the median absolute deviation of D_1

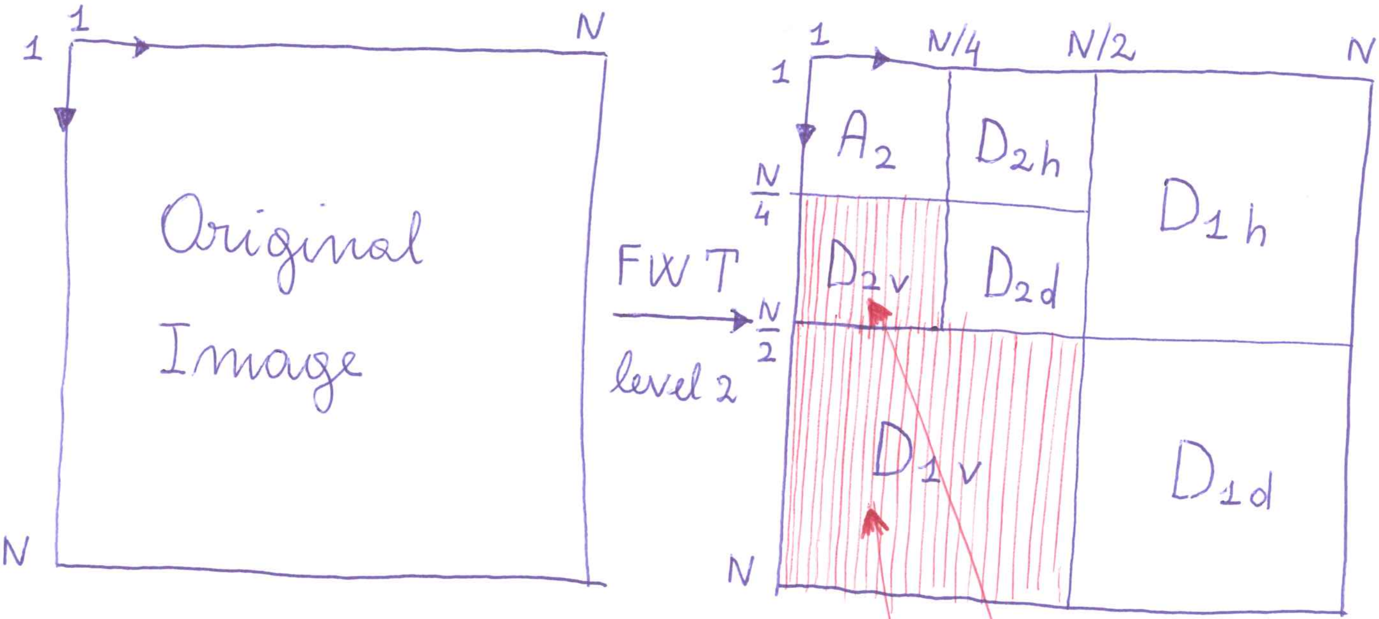
Further thinking:

- Pink vs. Blue ...
- And if the noise is not Gaussian? ...
- And if the noise is not additive? ...
- And if the colour of noise is not known? ...

2 Miscellanea

(a) Which transform is able to decompose an image at various scales and separate vertical features (from horizontal features, etc) ? The fast wavelet transform!

REMEMBER the FWT at level 2 of a house ...



- D_1 = detail coefficients at the sampling scale
- D_2 = detail coefficients at a scale twice as large
  - * h = horizontal
  - * v = vertical
  - * d = diagonal
- A_2 = approximation coefficients

→ The artifacts will appear here and here

How to pre-compress and get rid of the artifacts (6)

at the same time: set D_{1V} and D_{2V} to zero!

$$\underbrace{CF}_{\text{compression factor}} = \frac{\text{total number of wavelet coefficients}}{\text{number of wavelet coefficients that are not set to zero}}$$

$$= \frac{N^2}{N^2 - \left(\frac{N}{2}\right)^2 - \left(\frac{N}{4}\right)^2} = \frac{16}{11} \simeq 1.45$$