# Image Denoising : Course 1

Maxence Gollier

October 2023

## 1 Exercices

### 1.1 Exercise 8.1

We have by definition for any $0 \leq i, j \leq \kappa^2$

$$\mathrm{Cov}(\tilde{P}_i, \tilde{P}_j) = \mathbb{E}[\tilde{P}_i \tilde{P}_j] - \mathbb{E}[\tilde{P}_i]\mathbb{E}[\tilde{P}_j].$$

Since $\tilde{P} = P + N$,

$$\mathrm{Cov}(\tilde{P}_i, \tilde{P}_j) = \mathbb{E}[(P_i + N_i)(P_j + N_j)] - \mathbb{E}[(P_i + N_i)]\mathbb{E}[P_j + N_j].$$

Since $\mathbb{E}[N] = 0$ and P,N are independant, we get

$$\mathrm{Cov}(\tilde{P}_i, \tilde{P}_j) = \mathbb{E}[P_i P_j] - \mathbb{E}[P_i]\mathbb{E}[P_j] + \mathbb{E}[N_i N_j].$$

This yields

$$C_{\tilde{P}} = C_P + \sigma^2 I.$$

Finally, concerning the mean, we simply have

$$\mathbb{E}[\tilde{P}] = \mathbb{E}[P + N] = \bar{P}.$$

### 1.2 Exercise 8.3

We first reformulate $\hat{P}_1$. Let $\tilde{G}_i$, $\tilde{\lambda}_i$ respectively be the eigenvectors and eigenvalues of $C_{\tilde{P}}$. Let $\tilde{G}, \tilde{\Lambda}$ respectively be the matrix with columns $\tilde{G}_i$ and the diagonal matrix containing $\tilde{\lambda}_i$. We have $C_{\tilde{P}} = \tilde{G}\tilde{\Lambda}\tilde{G}^{-1}$ and $\tilde{G}^{-1} = \tilde{G}^\top$. Then, we have

$$\begin{aligned}
\hat{P}_1 &= \bar{P} + (C_{\tilde{P}} - \sigma^2 I)C_{\tilde{P}}^{-1}(\tilde{P} - \bar{P}) \\
&= \bar{P} + \tilde{G}(\tilde{\Lambda} - \sigma^2 I)\tilde{\Lambda}^{-1}\tilde{G}^\top(\tilde{P} - \bar{P}) \\
&= \bar{P} + \sum_i \frac{\tilde{\lambda}_i - \sigma^2}{\tilde{\lambda}_i}\langle \tilde{P} - \bar{P}, \tilde{G}_i \rangle \tilde{G}_i
\end{aligned}$$

Similarly for $\hat{P}_2$, with $C_P^1 = G^1\tilde{\Lambda}^1(G^1)^{-1}$

$$\begin{aligned}\hat{P}_2 &= \bar{P}^1 + C_P^1(C_P^1 + \sigma^2 I)^{-1}(\tilde{P} - \bar{P}^1)\\ &= \bar{P}^1 + G^1\Lambda^1(\Lambda^1 + \sigma^2 I)^{-1}G^\top(\tilde{P} - \bar{P}^1)\\ &= \bar{P}^1 + \sum_i \frac{\lambda_i^1}{\lambda_i^1 + \sigma^2}\langle \tilde{P} - \bar{P}^1, G_i^1\rangle G_i^1\end{aligned}$$

Compared with the Wiener coefficients respectively

$$\alpha(i) = \max\{0, \frac{\langle \tilde{U}, G_i\rangle^2 - c\sigma^2}{\langle \tilde{U}, G_i\rangle^2}\}$$

$$\alpha(i) = \frac{\langle \hat{U}_1, G_i\rangle^2}{\langle \hat{U}_1, G_i\rangle^2 + \sigma^2}$$

We conclude that we can interpret the two step Bayesian method as the Wiener empirical and oracular method.

### 1.3 Exercise 8.4

We have by Bayes rule

$$\mathbb{P}(P)\mathbb{P}(\tilde{P}|P)\|P - \hat{P}\|^2 = \mathbb{P}(\tilde{P})\mathbb{P}(P|\tilde{P})\|P - \hat{P}\|^2$$

Then, by Fubini's theorem

$$\begin{aligned}MSE &= \int\int \mathbb{P}(P)\mathbb{P}[\tilde{P}|P]\|P - \hat{P}\|^2 d\tilde{P}dP\\ &= \int\int \mathbb{P}(\tilde{P})\mathbb{P}(P|\tilde{P})\|P - \hat{P}\|^2 dPd\tilde{P}\end{aligned}$$

### 1.4 Exercise 8.5

We have

$$\text{MMSE}(\tilde{P}) = \int \mathbb{P}(P|\tilde{P})(P - \hat{P})^2 dP$$

Differentiating with respect to $\hat{P}$ gives

$$\frac{\partial}{\partial \hat{P}}\text{MMSE} = 2\hat{P}\int \mathbb{P}(P|\tilde{P})dP - 2\int \mathbb{P}(P|\tilde{P})PdP = 2\hat{P} - 2\int \mathbb{P}(P|\tilde{P})PdP = 0.$$

Which implies

$$\hat{P} = \int \mathbb{P}(P|\tilde{P})PdP$$

Figure 1: Image used for patch self-similarity analysis

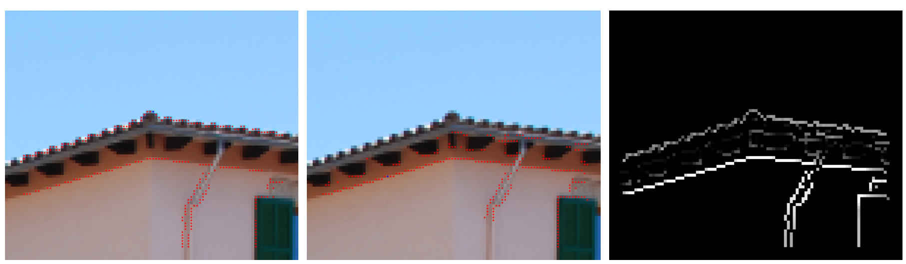
# 2 Experimental Report

## 2.1 Article 1: Exploring Patch Similarity in an Image

To explore patches self-similarity we will use the following image

We will explore two different patches with distinct natures : the first contains human-made paterns. The second one are natural leafs.

The reference patch for the first experiment can be seen on the left image of Figure 2.



(a) Closests to reference patch (in blue)
(b) Closests to the farthest from the reference patch
(c) histogram of all closests groups of closests

Figure 2

Clearly, we can see some structure in the histogram, namely, the edges of the parts of the house and the shadow. Now, we take a look at the different patches close to the reference patch on Figure 3.

We see that some are clearly similar but not all. One can see 4 different categories : the patches with the sky, the patches in light brown, darker brown and with a half green part. In this case, one can suppose that the number of closest patches is simply to high. Still, we carry the analysis with that choice of parameter, and compare it with a choice of smaller number of closest patches.

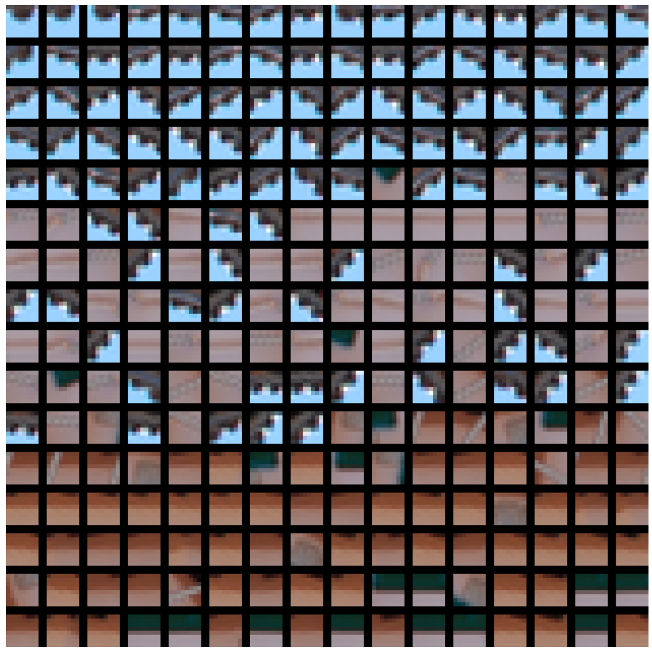
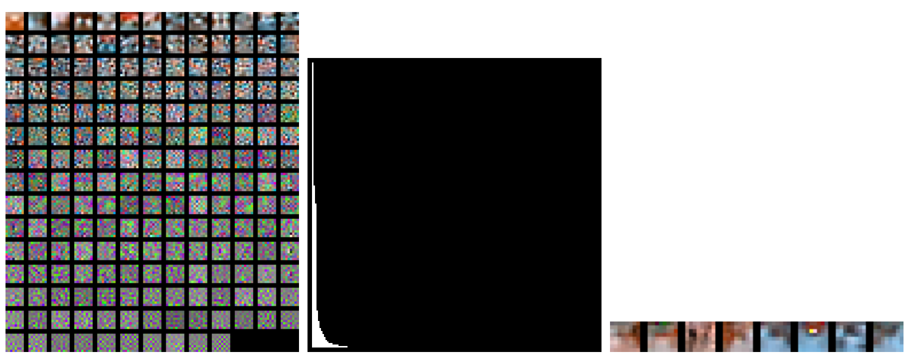


Figure 3: Closest patches to reference patches

Now, we take a look at the PCA.



(a) Basis of image patches (b) Eigenvalues (c) Simulated patches

Figure 4

There are about 25 dimensions needed to represent the set of patches. This implies that the texture are not sparse but may be due to the fact that to much neighboor patches are taken. Finally, we see that the Gaussianity has to be rejected on Figure 5

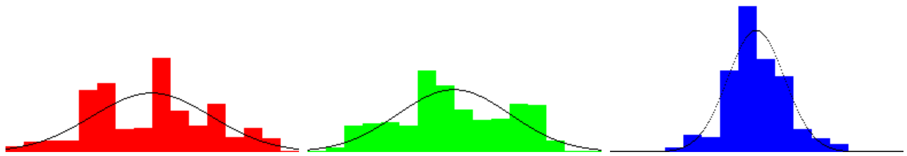
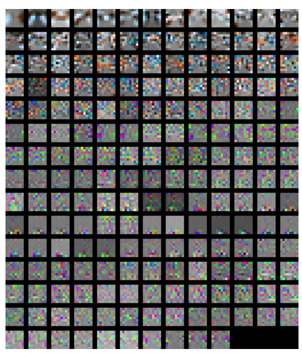


Figure 5: Histograms of PCA projections, Anderson-Darling pvalues are respectively from left to right $< 10^{-4}, < 10^{-4}, < 10^{-4}$.

Now, we investigate what happens when the number of closest patches are 64. We only show the effect on the PCA and the Gaussianity tests as we can see from Figure 3 that this is equivalent to choosing only the patches that contain sky parts.



(a) Basis of image patches

(b) Eigenvalues

(c) Simulated patches

Figure 6

Comparing the eigenvalues distribution from Figure 6 with these from Figure 4 we see that we need less dimensions to represent the patches. The main difference is with the Gaussians and the Anderson-Darling gaussianity test.

Figure 7 shows that the gaussianity of the set of patches can not be rejected anymore. This confirms that gaussianity of similar patches holds but depending on the number of similar patches we choose.

We now investigate self-similarity in the cases of leafs on Figure 1.

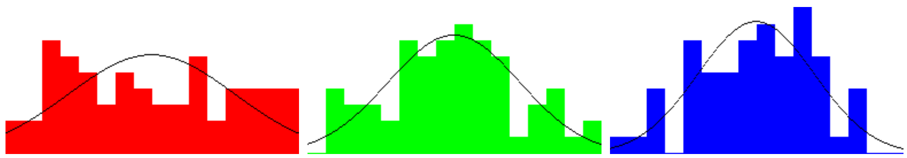


Figure 7: Histograms of PCA projections, Anderson-Darling pvalues are respectively from left to right $0.02, 0.78, 0.5$.

(a) Closests to reference patch (in blue)

(b) Closests to the farthest from the reference patch

(c) histogram of all closests groups of closests

Figure 8

The histogram at Figure 8 shows that the structure of the closest patches looks coherent with the structure of the tree. We take a look at the 255 closest patches at Figure 9.

A visual inspection shows that all patches look indeed quite similar, hence even natural textures do contain self similarity. We take a look at the PCA at Figure 11

Even more than for the last choice of patches, we now have a very high number of significative eigenvalues. This indicate that the patches are not sparse at all, this might be due to the fact that the patches from that region are highly textured hence, they are globally similar but have a lot of local differences between each other.

Finally, we see that gaussianity is not rejected even though the patches are not sparse.

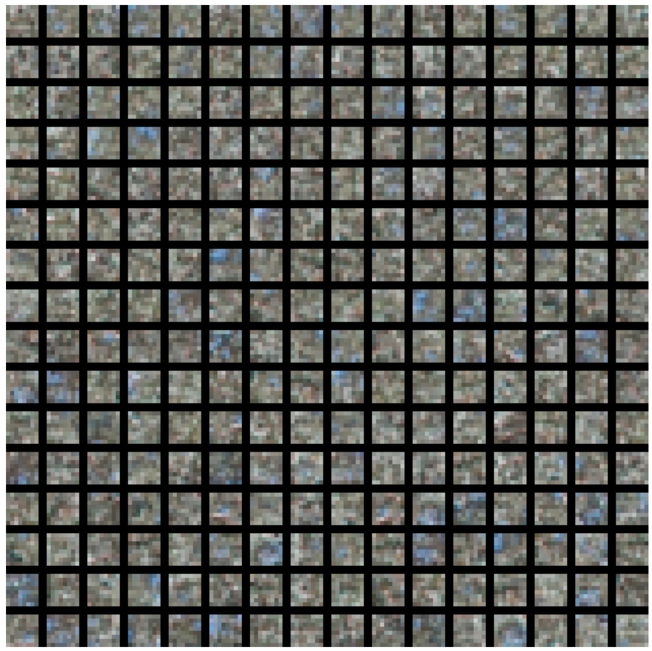
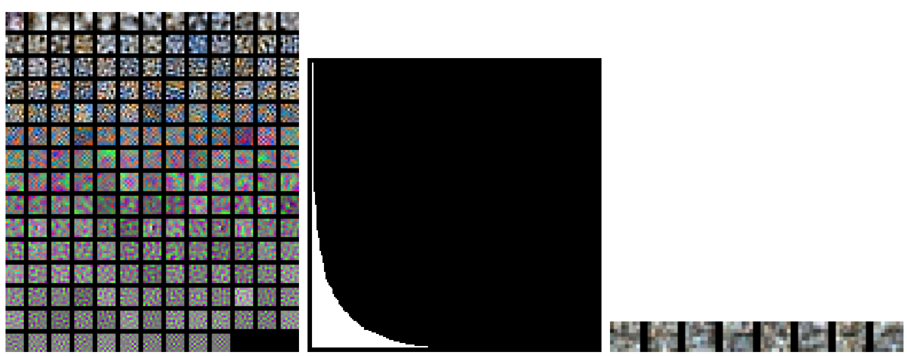


Figure 9: Closest patches to reference patches



(a) Basis of image patches (b) Eigenvalues (c) Simulated patches

Figure 10

## 2.2 Article 2: Implementation of the "Non-Local Bayes" (NL-Bayes) Image Denoising Algorithm

We will compare the NL-Bayes algorithm with the BM3D algorithm in theory and in practice.

In theory, both methods assume that patch self-similarity holds. Though they involve different approaches : using 3D transforms for BM3D, and a

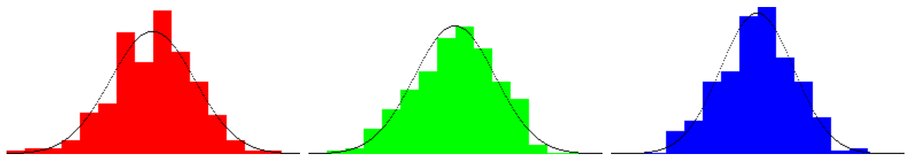


Figure 11: Histograms of PCA projections, Anderson-Darling pvalues are respectively from left to right $0.26, 0.05, 0.6$.

Bayesian approach for NL-Bayes, the steps from each can be easily compared as they both "try" to achieve the same objective. We use the Table 1 from the NL-Bayes article as reference to perform the theorical comparison.

- Step 1
  - **Preprocessing** : Both transform the RGB colors to an uncorelated color space.
  - **Grouping** : Both methods are identical for the patch creation except for the similarity threshold which NL-Bayes does not have.
  - **Collaborative Filtering** The BM3D now makes a 3D transform to perform hard-tresholding while the NL-Bayes method uses the first filter formula from the Bayesian approach. In this step, the Bayesian approach is far less intricated than BM3D.
  - **Aggregation** Identical
  - **Post processing** NL-Bayes transforms back to the RGB space.
- Step 2
  - **Grouping** The main difference is that the similarity threshold is still fixed for BM3D while it is adaptative according to the distances found. We note that the Bayesian approach now acts on all color channels while the BM3D still acts only on one color channel.
  - **Collaborative filtering** BM3D now estimates the Wiener filter based on the estimate from Step 1 and by applying DCT and Hadamard transforms. The bayesian approach uses the second filter formula. Again this looks less intricated for the bayesian approach.

In conclusion, we find that BM3D looks more straightforward (like we have already seen in the course, taking transforms and applying hard thresholding or Wiener filtering). On the other hand, once the Bayesian formulae have been found, the denoising processing is less intricated as we simply have to apply these formulae in the Bayesian methods.

We now take a look on how the methods compare in practice. The article already provides tables where it is seen that the NL-Bayesian achieves on average 0.5 more PSNR than BM3D for color images. We have seen that the treatment of the colors in both algorithm are quite similar but not exactly the same. In practice, the table shows that for grey images, NL-Bayesian only outperforms BM3D by 0.15 PSNR and some variation of BM3D, namely BM3D-SAPCAo outperforms NL-Bayes.