Wiener Filter

Mathematical Models and Methods for Image Processing

Edoardo Peretti

February 27th 2025

Empirical Wiener Filter

Let $\hat{\boldsymbol{y}}^{HT}$ be the hard threshold estimate, with DCT coefficients:

$$\hat{\boldsymbol{x}}^{HT} = D^T \hat{\boldsymbol{y}}^{HT}$$

The empirical Wiener filter attenuates the DCT coefficients as:

$$\hat{x}_i^{Wie} = \frac{\left(\hat{x}_i^{HT}\right)^2}{\left(\hat{x}_i^{HT}\right)^2 + \sigma^2} x_i$$

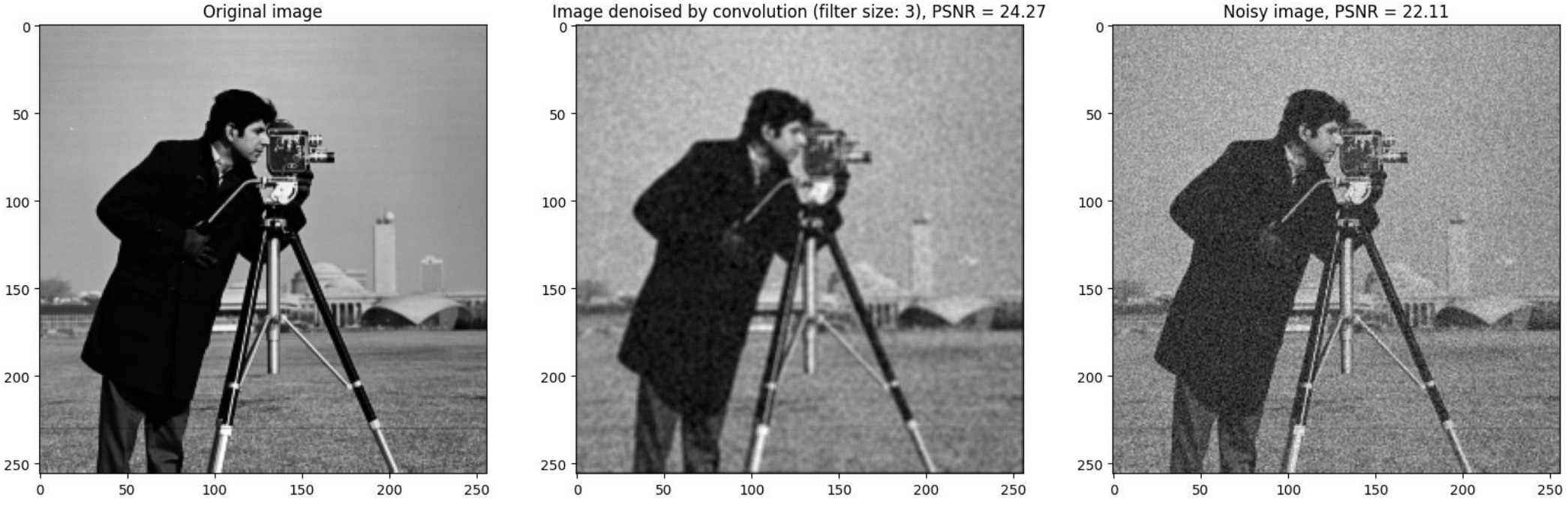
The empirical Wiener estimate is thus:

$$\hat{\boldsymbol{y}}^{Wie} = D\, \hat{\boldsymbol{x}}^{Wie}$$

Assignments

Assignment 1– Baseline denoising

1. Synthetically generate a noisy image (AWGN model)
2. Implement the noise level estimation method
    1. compare the robust vs non-robust standard deviation estimator
3. Perform denoising by convolution
    1. what happens when you change the kernel size?

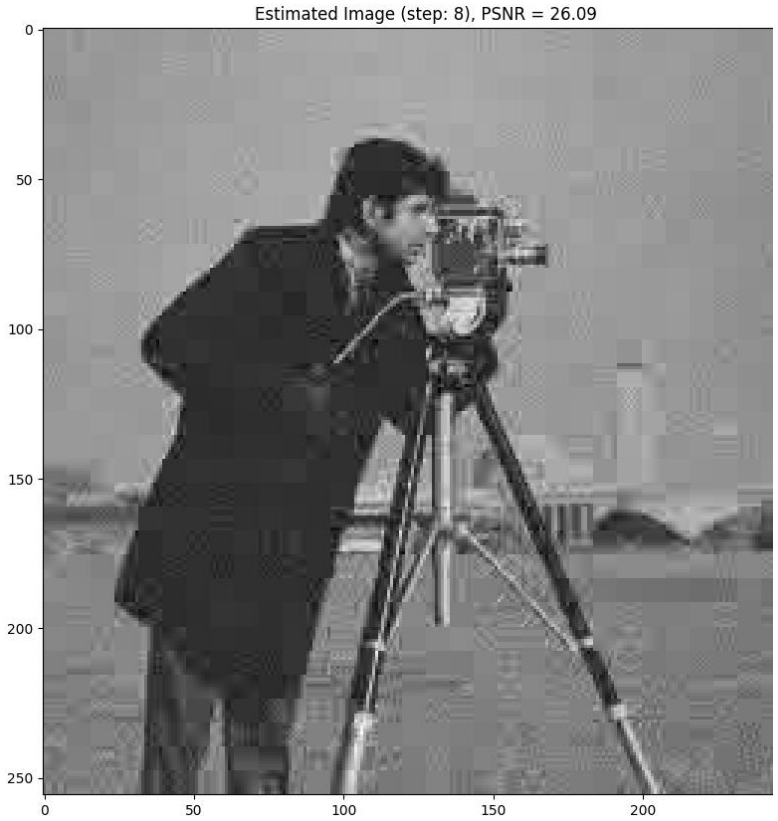


Assignment 2 – Sliding DCT denoising

1. Implement the sliding-DCT denoising using
   - no aggregation (operate on non-overlapping tiles)
   - aggregation using uniform weights
   - aggregation using weights inversely proportional to patch sparsity in DCT domain.
2. Test the three algorithms on both checkerboard and cameraman image
3. Test how much the choice of the threshold τ influences the denoising performance. Observe the resulting image when:
   - $\tau \ll 3\sigma$
   - $\tau \gg 3\sigma$

This is very important to understand how important is the choice of the threshold

DCT Denoising – Expected results



Uniform weights

Sparsity-aware weights

Assignment 3 – Wiener filter

1. Compute a first estimate using DCT denoising with Hard Thresholding
2. Use this estimate to perform denoising via Wiener Filtering
   - Start with $STEP = \sqrt{M}$ (non overlapping patches, as in JPEG)
   - Decrease the $STEP$ and perform aggregation