ECE 285 – Assignment #5

Writing part

---

1 Exercise 1 (Derivatives and convolutions)

Recall that the convolution product of two real functions f and g reads as

$$(f * g)(t) = \int_{-\infty}^{\infty} f(u)g(t-u)\mathrm{d}u \ .$$

Assume that f and g are continuously differentiable and that the above integral always exists and is finite.

1. What would be the necessary extra condition on f and g to have
$$f' * g = f * g' \ ?$$
2. Consider g being a Gaussian kernel. What does this condition imply on f?
3. Same question but for the exponential kernel.

Practical part

---

2 Exercise 2 (NL-means filter)

The NL-means filter is an extension of the bilateral filter, where instead of averaging values of pixels with similar values, the values of pixels centered on similar patches are averaged. The NL-means reads as:

$$x_{i,j}^{(\text{NL-means})} = \frac{1}{Z} \sum_{k=-s_1}^{s_1} \sum_{l=-s_2}^{s_2} \varphi\left(\frac{1}{P} \sum_{u=-p_1}^{p_1} \sum_{v=-p_2}^{p_2} (y_{i+k+u,j+l+v} - y_{i+u,j+v})^2 \right) y_{i+k,j+l} \tag{1}$$

$$\text{with} \quad Z = \sum_{k=-s_1}^{s_1} \sum_{l=-s_2}^{s_2} \varphi\left(\frac{1}{P} \sum_{u=-p_1}^{p_1} \sum_{v=-p_2}^{p_2} (y_{i+k+u,j+l+v} - y_{i+u,j+v})^2 \right) \ , \tag{2}$$

where $(-p_1, p_1) \times (-p_2, p_2)$ is the domain of a patch and $P = (2p_1 + 1) \times (2p_2 + 1)$ is its size. We will choose the following kernel function

$$\varphi(\alpha) = \exp\left(-\frac{\max(\alpha - 2\sigma^2, 0)}{16h\sigma^2/P} \right) \ . \tag{3}$$

1. Copy and paste `imbilateral_naive.m` to a new file `imnlmeans_naive.m`, change the function into

```
function x = imnlmeans_naive(y, sig, s1, s2, p1, p2, h)
```

and implement (except around boundaries) the NL-means filter with six loops as

```
x = zeros(n1, n2);
Z = zeros(n1, n2);
for i = (s1+p1+1):(n1-s1-p1)
    for j = (s2+p2+1):(n2-s2-p2)
        for k = -s1:s1
            for l = -s2:s2
                dist2 = 0;
                for u = -p1:p1
                    for v = -p2:p2
                        % complete
x = x ./ Z;
x(Z == 0) = 0;
```

2. Download `assignment5.zip` and extract the data:

   - `assignment5/zebra.png`

   Create a script `test_imnlmeans.m` that loads the image `x = castle` and creates `y` by adding additive white Gaussian noise of standard deviation $\sigma = 20$. Test your function on `y` with $s_1 = s_2 = 10$, $p_1 = p_2 = 3$ and $h = 1$. Zoom on the results using `linkaxes` to check that your function is consistent with the following ones:

(a) Original (b) Noisy (c) NL-means (42s)

3. Copy and paste `imbilateral.m` to a new file `imnlmeans.m`, change the function into

   ```
   function x = imnlmeans(y, sig, s1, s2, p1, p2, h, boundary)
   ```

   and implement the NL-means filter (including around boundaries) with two loops only.

   Hint: you just need to add a single line calling the function `imconvolve_spatial`.

4. Complete `test_imnlmeans.m` to test your new function. Compare the results and check that your new function is about 7 times faster (42s $\rightarrow$ 6s).

5. What is the complexity of both approaches?

6. Increase the noise level, and play with the search window sizes s_1 and s_2 and filtering parameter h.

3 Exercise 3 (Block-wise NL-means filter)

The block-wise NL-means is a variant of NL-means, where instead of averaging values of pixels with similar patches, similar patches are first averaged together:

$$\boldsymbol{x}_{i,j}^{\mathrm{BNLM}} = \frac{1}{Z} \sum_{k=-s_1}^{s_1} \sum_{l=-s_2}^{s_2} \varphi\left(\frac{1}{P}\|\boldsymbol{y}_{i+k,j+l} - \boldsymbol{y}_{i,j}\|_2^2\right) \boldsymbol{y}_{i+k,j+l} \tag{4}$$

where $\boldsymbol{x}_{i,j} = (x_{i+a,j+b})_{-p_1 \leqslant a \leqslant p_1, -p_2 \leqslant b \leqslant p_2}$ denotes the patch of x at location i, j. Next all restored patches are projected into the image domain at their original locations by averaging as

$$x_{i,j}^{\text{BNLM}} = \frac{1}{P} \sum_{a=-p_1}^{p_1} \sum_{b=-p_2}^{p_2} (\boldsymbol{x}_{i+a,j+b}^{\text{BNLM}})_{-a,-b}. \tag{5}$$

1. Plug these two equations together, and show that

$$x_{i,j}^{\text{BNLM}} = \frac{1}{Z} \sum_{k=-s_1}^{s_1} \sum_{l=-s_2}^{s_2} \sum_{a=-p_1}^{p_1} \sum_{b=-p_2}^{p_2} \varphi\left(\frac{1}{P} \sum_{u=-p_1}^{p_1} \sum_{v=-p_2}^{p_2} (y_{i+k+u+a,j+l+v+b} - y_{i+u+a,j+v+b})^2\right) y_{i+k,j+l} \tag{6}$$

$$\text{with} \quad Z = \sum_{k=-s_1}^{s_1} \sum_{l=-s_2}^{s_2} \sum_{a=-p_1}^{p_1} \sum_{b=-p_2}^{p_2} \varphi\left(\frac{1}{P} \sum_{u=-p_1}^{p_1} \sum_{v=-p_2}^{p_2} (y_{i+k+u+a,j+l+v+b} - y_{i+u+a,j+v+b})^2\right).$$

   Hint: note that $(\boldsymbol{y}_{i,j})_{-a,-b} = y_{i-a,j-b}$.

2. The naive implementation of eq. (6) will require 8 loops and would be extremely slow. Copy and paste `imnlmeans.m` to a new file `imblocknlmeans.m`, change the function into

```
function x = imblocknlmeans(y, sig, s1, s2, p1, p2, h, boundary)
```

   and implement the block-wise NL-means with two loops only.

   Hint: again, you just need to add a single line calling the function `imconvolve_spatial`.

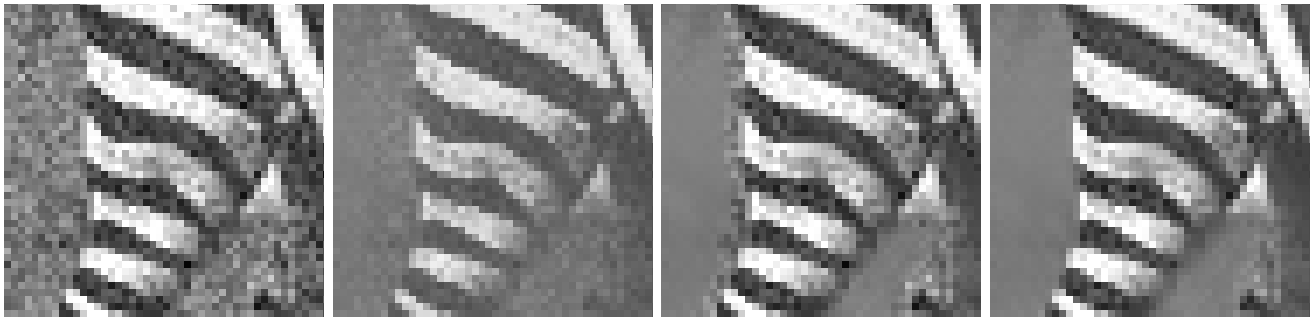
3. A very classical (but controversial) way to compare the quality of restoration techniques is to use the PSNR defined for images with values ranging in $[0, 255]$ as

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\frac{1}{n}\|\hat{x} - x\|_2^2} \tag{7}$$

   where x is the noise-free image and $\hat{x}$ the estimate obtained from y. The PSNR measures in decibels (dB) the quality of the restoration: the higher the better. Implement it in `impsnr.m`:

```
function p = impsnr(x, y)
```

4. Create a script `test_imblocknlmeans.m` to test your new function on the corrupted version `y` of `x = zebra` with additive white Gaussian noise of standard deviation $\sigma = 20$. Choose $s_1 = s_2 = 10$, $p_1 = p_2 = 3$ and $h = 1$, and compare the quality (PSNR) and the execution times of `imbilateral`, `imnlmeans` and `imblocknlmeans`. Check that your results are consistent with the following ones:



(a) Noisy (b) Bilateral (1.1s, 24.4dB) (c) NL-means (5.7s, 27.0dB) (d) BNLM (9.6s, 27.2dB)

5. What is the advantage of the block-wise NL-means compared to the standard NL-means?

6. What is the complexity of the block-wise NL-means?