Lecture 11:

Image sharpening in the frequency domain

Goal: Enhance image so that it shows more obvious edges.

Method 1: Laplacian masking

Recall that : $\Delta f(x,y) = \dfrac{\partial^2 f}{\partial x^2} + \dfrac{\partial^2 f}{\partial y^2}$.

In the discrete case, $\Delta f(x,y) \approx f(x+1,y) + f(x,y+1) + f(x,y-1) + f(x-1,y) - 4f(x,y)$

or $\Delta f \approx p * f$ where $p = \begin{pmatrix} & 1 & \\ 1 & -4 & 1 \\ & 1 & \end{pmatrix}$

We can observe that $-\Delta f$ captures the edges of the image (leaving other region zero)

$\therefore$ Shapen image $= f + \overbrace{(-\Delta f)}^{\text{add more edges}}$

In the frequency domain: $\mathrm{DFT}(g) = \mathrm{DFT}(f) - \mathrm{DFT}(\overset{p*f}{\overset{\shortparallel}{\Delta f}})$

$$\mathrm{DFT}(g)(u,v) = \mathrm{DFT}(f)(u,v) - c\,\mathrm{DFT}(p)(u,v)\,\mathrm{DFT}(f)(u,v)$$

$\therefore \mathrm{DFT}(g)(u,v) = [1 - \underset{c\,\mathrm{DFT}(p)}{\underset{\shortparallel}{H_{laplacian}(u,v)}}]\,\mathrm{DFT}(f)(u,v)$

Method 2: Unsharp masking

Idea: Add high-frequency component

Definition: Let f = input image (blurry)

Let f_{smooth} = smoother image (using mean filter / Gaussian filter etc)

Define a sharper image as:

$$g(x,y) = f(x,y) + k(f(x,y) - f_{smooth}(x,y))$$

When $k=1$, the method is called unsharp masking.

When $k>1$, the method is called highboost filtering.

In the frequency domain, let $DFT(f_{smooth})(u,v) = \underbrace{H_{LP}}_{\text{Low-pass filter}}(u,v)\, DFT(f)(u,v)$

Then: $DFT(g)(u,v) = [1 + k(1 - H_{LP}(u,v)]\, DFT(f)(u,v)$

Image denoising in the spatial domain

Definition: Linear filter = modify pixel value by a linear combination of pixel values of local neighbourhood.

Example 1: Let f be an $N \times N$ image. Extend the image periodically. Modify f to $\tilde{f}$ by:

$$\tilde{f}(x, y) = f(x, y) + 3f(x-1, y) + 2f(x+1, y).$$

This is a linear filter.

Example 2: Define

$$\tilde{f}(x, y) = \frac{1}{4}\left(f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)\right)$$

This is also a linear filter.

<u>Recall</u>: The discrete convolution is defined as:

$$I * H(u,v) = \sum_{m=-M}^{M} \sum_{n=-N}^{N} I(u-m, v-n) H(m,n)$$

(Linear combination of pixel values around (u, v))

Therefore, **Linear filter is equivalent to a discrete convolution**.

Geometric illustration

$$\begin{tikzcd}[column sep=huge, row sep=large]
I \arrow[d, "* H"] \arrow[dr, phantom, "\cong"] & \hat{I} \arrow[d, "\times \hat{H}"] \\
\tilde{I} \arrow[r, shift left, "DFT"] & \hat{\tilde{I}} \arrow[l, shift left, "iDFT"]
\end{tikzcd}$$

(spatial) (frequency)

Commonly used filter (linear)

- **Mean filter:**

$$H = \frac{1}{9} \begin{array}{c} \\ \\ \\ \\ \end{array} \begin{array}{cc} \begin{array}{ccc} -1 & 0 & 1 \end{array} & \\ \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} & \begin{array}{c} -1 \\ 0 \\ 1 \end{array} \end{array}$$

(Here, we only write down the entries of the matrix for indices $-1 \leq k, l \leq 1$ for simplicity. All other matrix entries are equal to 0.)

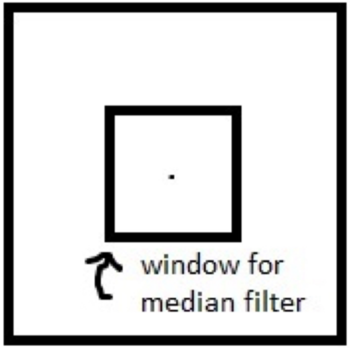
This is called the *mean filtering with window size* 3×3.

- **Gaussian filter:** The entries of H are given by the Gaussian function $g(r) = exp\left(-\frac{r^2}{2\sigma^2}\right)$, where $r = \sqrt{x^2 + y^2}$.

$$H(x,y) = exp\left(-\frac{(x^2+y^2)}{2\sigma^2}\right)$$

Non-linear spatial filter

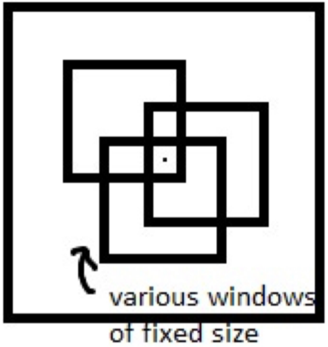
- **Median filter**



Take a window with center at pixel (x_0, y_0). Update the pixel value at (x_0, y_0) from $I(x_0, y_0)$ to $\tilde{I}(x_0, y_0) = \text{median}(I \text{ within the window})$

Example 4: If pixel values within a window is $0, 0, 1, 2, 3, 7, 8, 9, 9$, then the pixel value is updated as 3 (median).

- **Edge-preserving filter**

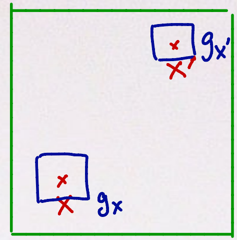


- **Step 1:** Consider all windows with certain size around pixel (x_0, y_0) (not necessarily be centered at (x_0, y_0));
- **Step 2:** Select a window with minimal variance;
- **Step 3:** Do a linear filter (mean filter, Gaussian filter and so on).

- <u>Non-local mean filter</u>

Let g be a $N \times N$ image.

$\left.\begin{array}{l} X = (x, y) \\ X' = (x', y') \end{array}\right\}$ Two pixels.



Define: $S_X = \{(x+s, y+t) : -a \le s, t \le a\}$; $S_{X'} = \{(x'+s, y'+t) : -a \le s, t \le a\}$

Define: $g_X = g|_{S_X}$ and $g_{X'} = g|_{S_{X'}}$.

$\underbrace{(2a+1)}_{m} \times \underbrace{(2a+1)}_{m}$ image

Let $\tilde{g}_X$ = smoothed image of g_X by Gaussian smoothing

$\tilde{g}_{X'}$ = smoothed image of $g_{X'}$ by Gaussian smoothing.

Define the weight: $w(X, X') = e^{\frac{-\|\tilde{g}_X - \tilde{g}_{X'}\|_F^2}{h^2}}$

noise level parameter

(small when X and X' are far away)

far away in term of small images

$$\boxed{\begin{array}{l} \text{Non-local mean filter of } g: \\ \hat{g}(X) = \dfrac{\sum_{X' \in \text{image domain}} w(X, X')\, g(X')}{\sum_{X' \in \text{image domain}} w(X, X')} \end{array}}$$

Image denoising using energy minimization

Let g be a noisy image corrupted by additive noise n.

Then: $g(x,y) = \underbrace{f(x,y)}_{\text{Clean image}} + \underbrace{n(x,y)}_{\text{noise}}$

Recall: Laplacian masking : $g = f - \Delta f$ (Obtain a (non-smooth) sharp image from a smooth image)

Conversely, to get a smooth image f from a non-smooth image g, we can solve the PDE for f : $-\Delta f + f = g$

unknown (f) known (g)

- In the discrete case, the PDE can be approximated (discretized) to get:

$$f(x,y) = g(x,y) + [f(x+1,y) + f(x,y+1) + f(x-1,y) + f(x,y-1) - 4f(x,y)]$$

for all (x,y) (Linear System)

Solved by

Direct method (Big linear system)

Iterative method

- $-\Delta f + f = g$ can also be solved in the frequency domain:

$$DFT(f) = DFT(g + \underbrace{\Delta f}_{p * f})$$

$$\therefore DFT(f)(u,v) = DFT(g)(u,v) + c\,DFT(p)(u,v)\,DFT(f)(u,v)$$

$$\Leftrightarrow DFT(f)(u,v) = \left[\frac{1}{1 - c\,DFT(p)(u,v)}\right] DFT(g)(u,v)$$

$\downarrow$ inverse DFT

$f(x,y)$!!

Consider: $E_{discrete}(f) = \sum_{x=1}^{N}\sum_{y=1}^{N}\left(f(x,y) - g(x,y)\right)^2 + \sum_{x=1}^{N}\sum_{y=1}^{N}\left[\left(f(x+1,y) - f(x,y)\right)^2 + \left(f(x,y+1) - f(x,y)\right)^2\right.$

Suppose f is a minimizer of $E_{discrete}$. Then, for each (x,y),

$E_{discrete}$ depends on $f(x,y)$ for each (x,y)

$$\frac{\partial E_{discrete}}{\partial f(x,y)} = 0.$$

$$= 2\left(f(x,y) - g(x,y)\right) + 2\left(f(x+1,y) - f(x,y)\right)(-1) + 2\left(f(x,y+1) - f(x,y)\right)(-1) + 2\left(f(x,y) - f(x-1,y)\right) + 2\left(f(x,y) - f(x,y-1)\right)$$

By simplification, we get:

$$f(x,y) = g(x,y) + \left[f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y)\right]$$

The continuous version of $E_{discrete}$ can be written as:

$$E(f) = \iint \left(f(x,y) - g(x,y)\right)^2 + \iint \left[\underbrace{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}_{|\nabla f|^2}\right] dx\,dy$$

Remark:

- Solving $f = g + \Delta f$ is equivalent to energy minimization
- The first term in Ediscrete is called the fidelity term. Aim to find f that is close to g.
- The second term is called the regularization term. Aim to enhance smoothness.

$$E_{discrete}(f) = \boxed{\sum_{x=1}^{N}\sum_{y=1}^{N}\left(f(x,y) - g(x,y)\right)^2} + \boxed{\sum_{x=1}^{N}\sum_{y=1}^{N}\left(\overbrace{f(x+1,y) - f(x,y)}^{\left(\frac{\partial f}{\partial x}\right)^2 \approx}\right)^2 + \left(\underbrace{f(x,y+1) - f(x,y)}_{\approx \left(\frac{\partial f}{\partial y}\right)^2}\right)^2}$$

fidelity term

given a noisy g

= regularization term

input noisy

1. $$\int \big(f(x,y) - \underset{\text{input}}{g(x,y)}\big)^2 + \int \left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2$$

$$\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2 = |\nabla f|^2$$

2. $$\int \big(f(x,y) - g(x,y)\big)^2 + \int |\nabla f|^1$$ (TV-denoising /ROF)