Lecture 10:

Recall:

- Mathematical formulation of image blur:

$$\underbrace{g}_{\text{Blurry image}} = \underbrace{f}_{\text{original image}} * h + \underbrace{n}_{\text{noise}} \Rightarrow \underbrace{G}_{\text{DFT of } g} = c\overbrace{F}^{\text{DFT of } f} \odot \underbrace{H}_{\text{DFT of } h} + \overbrace{N}^{\text{DFT of } n}$$

- Direct inverse filtering: $T(u,v) = \dfrac{1}{H(u,v) + \varepsilon \,\mathrm{sgn}(H(u,v))}$ (Replacing cH by H)
  (Boast up noise)
- Modified inverse filtering:

$$B(u,v) = \frac{1}{1 + \left(\frac{u^2+v^2}{D^2}\right)^n} \quad \text{and} \quad T(u,v) = \frac{B(u,v)}{H(u,v) + \varepsilon \,\mathrm{sgn}(H(u,v))}.$$

Method 3: Wiener filter

Let $T(u,v) = \dfrac{\overline{H(u,v)}}{|H(u,v)|^2 + \dfrac{S_n(u,v)}{S_f(u,v)}}$ where $S_n(u,v) = |N(u,v)|^2$

$S_f(u,v) = |F(u,v)|^2$

If $S_n(u,v)$ and $S_f(u,v)$ are not known, then we let $K = \dfrac{S_n(u,v)}{S_f(u,v)}$ to get:

$$T(u,v) = \frac{\overline{H(u,v)}}{|H(u,v)|^2 + K}$$

Let $\hat{F}(u,v) = T(u,v)\,G(u,v)$. Compute $\hat{f}(x,y)$ = inverse DFT of $\hat{F}(u,v)$.

In fact, the Wiener filter can be described as an inverse filtering as follows:

$$\hat{F}(u,v) = \left[\left(\frac{1}{H(u,v)}\right)\left(\frac{|H(u,v)|^2}{|H(u,v)|^2 + K}\right)\right] G(u,v)$$

Behave like "Modified inverse filtering"

≈ 0 if $H(u,v) \approx 0$ (if (u,v) far away from 0)

≈ 1 if $H(u,v)$ is large (if $(u,v) \approx (0,0)$)

What does Wiener filtering do mathematically?

We can show: Wiener filter minimizes the mean square error:

$$\varepsilon^2(\underset{\text{original}}{f}, \underset{\text{Restored}}{\hat{f}}) = \sum_{x=-\frac{N}{2}}^{\frac{N}{2}} \sum_{y=-\frac{N}{2}}^{\frac{N}{2}} \left| f(x,y) - \hat{f}(x,y) \right|^2$$

degradation

Observed

Let $g = h * f + n$ ← noise

original

Then, the restored image $\hat{f}(x,y)$ can be written as:

$$\hat{f}(x,y) = w(x,y) * g(x,y) \quad \text{for some } w(x,y)$$

Recall: $\hat{f}$ is obtained as follows:

Step 1: Let $\hat{F}(u,v) = \underbrace{W(u,v)}_{\text{Filter}} G(u,v)$ $\quad (G(u,v) = DFT(g)(u,v))$

Step 2: Compute iFT of $\hat{F}$ to get $\hat{f}$

$\therefore \hat{f} = w * g$ for some w. $\quad$ (or $\hat{F} = DFT(\hat{f}) = W \odot G$)

Thus, $\hat{f}$ denpends on W

We can regard $\varepsilon^2(\hat{f}, f)$ as a functional depending on W:

$$\varepsilon^2(\hat{f}, f) = \varepsilon^2(W)$$

We can find an optimal W that minimizes:

$$\varepsilon^2(W) = \| \hat{f}(W) - f \|_F^2$$

Under suitable condition (spatially correlated), the minimizer W is given by:

$$W(u,v) = \frac{\overline{H(u,v)}}{|H(u,v)|^2 + \frac{S_n(u,v)}{S_f(u,v)}}$$ where $S_n(u,v) = |N(u,v)|^2$, $S_f(u,v) = |F(u,v)|^2$

Method 4: Constrained least square filtering

Disadvantages of Wiener's filter:

① $|N(u,v)|^2$ and $|F(u,v)|^2$ must be known / guessed

② Constant estimation of ratio is not always suitable

Goal: Consider a least square minimization model.

Let $g = \underset{\text{degradation}}{h} * f + \underset{\text{noise}}{n}$

In matrix form, $\vec{g} = D\vec{f} + \vec{n}$ $\vec{g}, \vec{f}, \vec{n} \in \mathbb{R}^{N^2}$, $D \in M_{N^2 \times N^2}$

$\vec{g} = \mathcal{S}(g)$, $\vec{f} = \mathcal{S}(f)$, $\vec{n} = \mathcal{S}(n)$

$\mathcal{S}(g)$: Stacked image of g

D: transformation matrix of $h * f$ (or f)

Constrained least square problem:

Given $\vec{g}$, we need to find an estimation of $\vec{f}$ such that it minimizes:

$$E(f) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} |\Delta f(x,y)|^2 \quad \text{subject to the constraint:}$$

$$\|\vec{g} - D\vec{f}\|^2 = \varepsilon$$

What is Δf ?

In the discrete case, we can estimate:

$$\Delta f(x,y) \approx f(x+1,y) + f(x,y+1) + f(x-1,y) + f(x,y-1) - 4f(x,y)$$

Taylor expansion:

$$\frac{\partial^2 f}{\partial x^2}(x,y) \approx \frac{f(x+h,y) - 2f(x,y) + f(x-h,y)}{h^2} \qquad \xrightarrow{\text{Put } h=1} \qquad \Delta f(x,y) = \left(\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}\right)(x,y)$$

$$\frac{\partial^2 f}{\partial y^2}(x,y) \approx \frac{f(x,y+h) - 2f(x,y) + f(x,y-h)}{h^2}$$

$\therefore$ Δ is the Laplacian in the discrete case

Remark:

- More generally, $\Delta f = p * f$ ← discrete convolution

  where

$$p = \begin{array}{c@{}l} \begin{pmatrix} 0 & \cdots & & \cdots & 0 \\ \vdots & & 1 & & \vdots \\ & 1 & -4 & 1 & \\ \vdots & & 1 & & \vdots \\ 0 & & \cdots & & 0 \end{pmatrix} & \begin{matrix} \\ \\ x=0 \\ \\ \\ \end{matrix} \\ \quad\quad y=0 & \end{array}$$

- Minimizing $\displaystyle\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} \left|\Delta f(x,y)\right|^2$ is to denoise.

- Also, $\vec{g} = D\vec{f} + \vec{n} \Leftrightarrow \vec{g} - D\vec{f} = \vec{n} \Rightarrow \|\vec{g} - D\vec{f}\|^2 = \|\vec{n}\|^2 \approx \underbrace{\varepsilon}_{\text{noise level}}$

  $\therefore$ the constraint $\|\vec{g} - D\vec{f}\|^2$ is to solve the deblurring problem.

- $\displaystyle\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} |\Delta f(x,y)|^2$ Denoise

- $\|\vec{g} - D\vec{f}\|^2 = \varepsilon$ Deblur

<u>Remark:</u> $\underbrace{\|\vec{g} - D\vec{f}\|^2}_{= \|\vec{n}\|^2} = \varepsilon$ means we allow some fixed level of noise.

Let $\overrightarrow{p*f} = \mathcal{S}(p*f) = L\vec{f}$ transformation matrix representing the convolution with p.

Then: $E(\vec{f}) = (L\vec{f})^T(L\vec{f})$

We will prove:

<u>Theorem:</u> The constrained least square problem has the optimal solution in the spatial domain that satisfies:

$$(D^T D + \gamma L^T L)\vec{f} = D^T \vec{g}$$

for some suitable parameter γ.

In the frequency domain,

$$\hat{F}(u,v) := DFT(f)(u,v) = \frac{1}{N^2}\,\frac{\overline{H(u,v)}}{|H(u,v)|^2 + \gamma\,|P(u,v)|^2}\,G(u,v)$$

$\Big(H = DFT(h);\ G(u,v) = DFT(g);\ P(u,v) = DFT(p)$ where

$$p = \begin{pmatrix} 0 & \cdots & & \cdots & 0 \\ \vdots & & 1 & & \vdots \\ \vdots & 1 & -4 & 1 & \vdots \\ \vdots & & 1 & & \vdots \\ 0 & \cdots & & \cdots & 0 \end{pmatrix} \Big)$$

Remark: Constrained least square filtering:

$$T(u,v) = \frac{1}{N^2} \frac{\overline{H(u,v)}}{|H(u,v)|^2 + \gamma |P(u,v)|^2}$$

Let $\tilde{F}(u,v) = T(u,v)\, G(u,v)$

Compute Inverse DFT of $\tilde{F}(u,v)$.

Sketch of proof:

Recall: our problem is to minimize:

$$\vec{f}^T L^T L \vec{f} \quad \text{subject to} \quad \|\vec{g} - D\vec{f}\|^2 = \varepsilon$$

$$\|\vec{g} - D\vec{f}\|^2 = (\vec{g} - D\vec{f})^T (g - D\vec{f})$$

From calculus, the minimizer must satisfy:

$$\nabla \mathcal{L} \overset{\text{def}}{=} \nabla \left(\vec{f}^T L^T L \vec{f} + \lambda (\vec{g} - D\vec{f})^T (\vec{g} - D\vec{f}) \right) = 0 \quad \text{for}$$

where $\vec{f} = (f_1, f_2, \ldots, f_i, \ldots, f_{N^2})^T$ and λ is the Lagrange's multiplier.

Here, $\nabla \mathcal{L} = \left(\frac{\partial \mathcal{L}}{\partial f_1}, \frac{\partial \mathcal{L}}{\partial f_2}, \ldots, \frac{\partial \mathcal{L}}{\partial f_{N^2}} \right)^T$

Easy to check:

- $\nabla(\vec{f}^T \vec{a}) = \vec{a}$
- $\nabla(\vec{b}^T \vec{f}) = \vec{b}$
- $\nabla(\vec{f}^T A \vec{f}) = (A + A^T) \vec{f}$

$$\vec{f}^T \vec{a} = (f_1, f_2, \ldots, f_n) \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} = a_1 f_1 + a_2 f_2 + \ldots + a_n f_n$$

$$\frac{\partial \vec{f}^T \vec{a}}{\partial f_j} = a_j$$

$$\therefore \nabla(\vec{f}^T \vec{a}) \stackrel{\text{def}}{=} \left(\frac{\partial \vec{f}^T \vec{a}}{\partial f_1}, \ldots, \frac{\partial \vec{f}^T \vec{a}}{\partial f_n} \right)^T = (a_1, a_2, \ldots, a_n)^T$$

etc. . . .

$\therefore \quad D = 0 \Rightarrow (2L^TL)\vec{f} + \lambda(-D^T\vec{g} - D^T\vec{g} + 2D^TD\vec{f}) = 0$

$\Rightarrow (D^TD + \gamma L^TL)\vec{f} = D^T\vec{g}$ where $\gamma = \frac{1}{\lambda}$ and λ is the Lagrange's multiplier.

Parameter γ can be determined by direct substitution into the equation:

$$(\vec{g} - D\vec{f})^T(\vec{g} - D\vec{f}) = \varepsilon.$$

How about in the frequency (Fourier) domain?

Two important theorems

Notation: Let $\mathcal{O}$ be a linear transformation defined by:

$$\mathcal{O}(f) = k * f \quad \text{for all } f \in M_{N\times N}(\mathbb{R}), \text{ where}$$

$k \in M_{N\times N}(\mathbb{R})$.

Let $D \in M_{N^2 \times N^2}(\mathbb{R})$ be the transformation matrix representing $\mathcal{O}$.

That is, $\quad \mathcal{S}(\mathcal{O}(f)) = D\, \mathcal{S}(f)$. $\in \mathbb{R}^{N^2}$

Here, $\mathcal{S}$ is the stacking operator. $\mathcal{S}(I)$ is the vectorized image of I (1st col of I becomes first n entries of $\mathcal{S}(I)$, 2nd col of I becomes second n entries of $\mathcal{S}(I)$, …, etc)

Theorem 1: Let $\mathbb{K} = DFT(k)$. Then:

$$D = W \Lambda_D W^{-1} \quad \text{and} \quad D^T = W \overline{\Lambda}_D W^{-1} \quad \text{where}$$

$$\Lambda_D = N^2 \begin{pmatrix} \mathbb{K}(0,0) & & & & & & & & & \\ & \mathbb{K}(1,0) & & & & & & & & \\ & & \ddots & & & & & & & \\ & & & \mathbb{K}(N-1,0) & & & & & & \\ & & & & \mathbb{K}(0,1) & & & & & \\ & & & & & \ddots & & & & \\ & & & & & & \mathbb{K}(N-1,1) & & & \\ & & & & & & & \ddots & & \\ & & & & & & & & \mathbb{K}(0,N-1) & \\ & & & & & & & & & \ddots \\ & & & & & & & & & & \mathbb{K}(N-1,N-1) \end{pmatrix}$$

1st col of $\mathbb{K}$ becomes 1st n diagonal entries

2nd col of $\mathbb{K}$ becomes 2nd diagonal entries

for $W = W_N \otimes W_N$ where $W_N = \left(\frac{1}{\sqrt{N}} e^{\frac{2\pi j}{N} mn} \right)_{0 \le m,n \le N-1} \in M_{N \times N}(\mathbb{C})$