ARAVIND
20191020124

DIP - Lecture 8

Bilateral filtering
Linearity
Intro to freq. domain processing

→ Laplacian filter sum of elements must be zero

→ Max spatial filter (non-linear)
- ↳ Thinning of region (erode edges
- ↳ Removes pepper noise

→ Min spatial filter (non-linear)
- ↳ Thickening of region (dilate)
- ↳ removes salt noise

→ Median spatial filter (non-linear) → Non-linear as max, min can't be mapped as linear
- ↳ Preserves edges
- ↳ Less susceptible to noise
- ↳ Removes salt & pepper noise
- ↳ Also called rank/order statistic filter

Mean	(vs)	Gaussian	(vs)	Median
→ Blurs image → Removes simple noise → No details are preserved		→ Blurs image → Preserves details only for small σ		→ Preserves some details → Good at removing strong noise

Edge Preserving Filtering

→ Edges → Smooth only along edges
→ Smooth regions: Smoothing isotropically

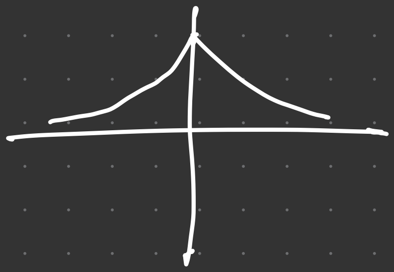
Bilateral filter:

Change in intensity sharply shows presence of edge.
↳ We need to preserve it

Gaussian Weights:

$$W_s(p) = e^{-\left(\frac{(u-p)^2}{2\sigma_s^2}\right)}$$

→ Exponentially decaying quad. term



→ Smoothes edges due to wave function as it doesn't take into account intensity but only takes geometric weights into account

Photometric Weights:

$$W_r(p) = e^{-\left(\frac{(I(u)-I(p))^2}{2\sigma_r^2}\right)}$$

↳ Understand notion of edge as diff will be smaller

↳ Depends on pixel intensity values

Combining both,

$$W_{bi}(p) = W_s(p) \times W_r(p)$$

Denoise ← → Edge-Preserving

$$I'(u) = \frac{\sum_{p \in N(u)} e^{-\frac{\|u-p\|^2}{2\sigma_s^2}} e^{-\frac{|I(u)-I(p)|}{2\sigma_r^2}} I(p)}{\sum_{p \in N(u)} e^{-\frac{\|u-p\|^2}{2\sigma_s^2}} e^{-\frac{|I(u)-I(p)|}{2\sigma_r^2}}}$$

→ Normalisation

$$BF[I]_p = \underbrace{\frac{1}{W_p}}_{\text{Normalisation}} \sum_{q \in S} \underbrace{G_{\sigma_s}(\|p-q\|)}_{\text{Space}} \underbrace{G_{\sigma_r}(|I_p - I_q|)}_{\text{Range}} I_q$$

If $\sigma_r = \infty$, only denoising happens ⇒ Gaussian blurring.

⇓ σ_r should not be too high

Linear Spatial Filters:

$$g(i,j) = \frac{\sum_{k,l} f(k,l)\, w(i,j,k,l)}{\sum_{k,l} w(i,j,k,l)}$$

In 2D, space = domain

domain kernel:

$$d(i,j,k,l) = \exp\left(-\frac{(i-k)^2 + (j-l)^2}{2\sigma_d^2}\right)$$

Geometric

data-dependent range kernel (for edges)

$$r(i,j,k,l) = \exp\left(\frac{-\|f(i,j) - f(k,l)\|^2}{2\sigma_r^2}\right)$$

Photometric

Final equations

$$w(i,j,k,l) = \exp\left(\frac{-(i-k)^2 + (j-l)^2}{2\sigma_d^2} - \frac{\|f(i,j) - f(k,l)\|^2}{2\sigma_r^2}\right)$$

Iterating bilateral filters:

→ Generates more piece-wise flat
→ Often not preferred
→ Looks more cartoonish

Effect of noise on derivatives

↳ Derivatives amplify noise
↳ Used to focus of high frequency

⇒ Solutions:
Smoothen first

Laplacian of Gaussian:

↳ Noise suppression as we

↳ Second order derivatives will give proper zero crossing

$$g(x) = e^{\frac{-x^2}{2\sigma^2}}$$

$$g'(x) = \frac{-x}{\sigma^2} e^{\frac{-x^2}{2\sigma^2}}$$

$$g''(x) = \left[\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2}\right] e^{\frac{-x^2}{2\sigma^2}}$$

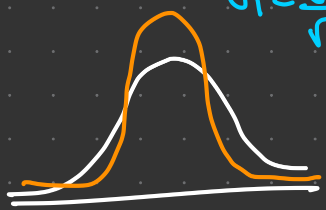
Difference of Gaussian

→ Band pass

$\nabla^2 G_\sigma = G_{\sigma_1} - G_{\sigma_2}$

Best when

$\sigma_1 = \frac{\sigma}{\sqrt{2}}, \sigma_2 = \sqrt{2}\sigma$ [General]



Linear Spatial filtering:

⇒ Scaling & Additivity

$$T[a x_1(n) + b x_2(n)] = a T[x_1(n)] + b T[x_2(n)]$$

Convolution/Linear filters:

Smoothing (average, gaussian)

Edge filters (prewitt, sobel, laplacian)

Cross-correlation:

Operation of sliding kernel/filter across image and compute SOP.

$$H \circ I(x,y) = \sum_{i=-N}^{N} \sum_{j=-N}^{N} H(i,j) \cdot I(x-i, y-j)$$

Convolution:

Rotating kernel, filter by 180° (flip rows then columns), slide kernel and find SoP

$$H * I(x,y) = \sum_{i=-N}^{N} \sum_{j=-N}^{N} H(i,j) \cdot I(x-i, y-j)$$

Both are same only when kernel is symmetric

Two paradigms → Transform Domain

$\boxed{\text{INP}} \rightarrow \boxed{\text{Transform}} \rightarrow \boxed{\text{Process}} \rightarrow \boxed{\text{Inv. Transform}} \rightarrow \boxed{\text{Output}}$

↙ Spatial domain

Periodic → Frequency of occurrence

$$x(t) = A\cos(\underbrace{\omega t}_{\text{Angular freq}}) = A\cos(2\pi f t) = A\cos\left(\frac{2\pi}{\underbrace{T}_{\text{Fund. Period}}} t\right)$$

Periodic images

Spatial periods: Minimum # of pixels b/w two identical patterns in periodic image

$$\nu_{max} = \frac{1}{\text{Minimal period}}$$

→ Fast repetition in space implies distance of points in freq. domain increases