COL783: Digital Image Analysis

Assignment-2
Part-1
a.

Nearest neighbour interpolation

Let original image be $f[m,n] = \sum_{m,n} f\,\delta(x-m\Delta T)\,\delta(y-n\Delta T)$

$$\mathbb{F}\{f[m,n]\} = F(u,v)$$

$$= \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f[m,n]\, e^{-2\pi j\left(\frac{u m}{M} + \frac{v n}{N}\right)}$$

where $v \in \left[-\frac{N}{2}, \frac{N}{2}\right]$

$u \in \left[-\frac{M}{2}, \frac{M}{2}\right]$

By Nearest neighbour resizing

$$f'[p,q] = f\left[\lfloor p/k \rfloor, \lfloor q/k \rfloor\right]$$

where k is the scaling factor

$$f'(x',y') = \sum_{m=0}^{\lfloor \frac{M}{k} \rfloor - 1}\sum_{n=0}^{\lfloor \frac{N}{k} \rfloor - 1} f[m,n]\,\mathrm{rect}\left(\frac{x'}{k} - m\right)\mathrm{rect}\left(\frac{y'}{k} - n\right)$$

where rect( ) is the unit box function

$$\mathbb{F}\left\{\mathrm{rect}\left(x/k\right)\mathrm{rect}\left(y/k\right)\right\} = k^2 \,\mathrm{sinc}\left(\frac{k\omega}{2\pi M}\right)\mathrm{sinc}\left(\frac{k v}{2\pi N}\right)$$

$$\int_{-1/2}^{1/2} 1\, e^{-2\pi i\left(\frac{ux}{M} + \frac{vy}{N}\right)} dx$$

$$\Rightarrow F'(u', v') = \sum_{p=0}^{M'-1} \sum_{q=0}^{N'-1} f'[p,q]\, e^{-2\pi i\left(\frac{u'p}{M'} + \frac{v'q}{N'}\right)}$$

where $M' = \left\lfloor \frac{M}{k} \right\rfloor$

$N' = \left\lfloor \frac{N}{k} \right\rfloor$

$$= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f[m,n] \sum_{p=km}^{k(m+1)-1} \sum_{q=kn}^{k(n+1)-1} e^{-2\pi i\left(\frac{u'p}{M'} + \frac{v'q}{N'}\right)}$$

rect() = 1 for $0 < \frac{x' - km}{k} < 1 \Rightarrow km < x' < k(m+1)$

$$\sum_{p=km}^{k(m+1)-1} e^{-\frac{iu'p}{M'}} = e^{-\frac{iu'}{M'}(km)} \cdot \frac{1 - e^{-i\frac{u'}{M'}k}}{1 - e^{-\frac{iu'}{M'}}}$$

$$\Rightarrow F'(u', v') = \sum_{m,n} f[m,n]\, e^{-j\left(\frac{u'}{M'}km + \frac{v'}{N'}kn\right)} \frac{\left(1 - e^{-j\frac{u'}{M'}k}\right)\left(1 - e^{-j\frac{v'}{N'}k}\right)}{\left(1 - e^{-j\,u'/M'}\right)\left(1 - e^{-j\,v'/N'}\right)}$$

b. Only the frequencies that lie within the new Nyquist limit after resizing can be passed before resampling/resizing.

New Nyquist frequency in horizontal direction: $\mu_{max} = 1/2\Delta X' = k/2\Delta X$

since new $\Delta X' = \Delta X/k$

And likewise for vertical direction. We assume dx and dy to be 1 (i.e. 1 pixel per length)

We used elliptical disk for the kernel in the frequency domain:

Here nu_x ranges from -W//2 to W//2 and nu_y from -H//2 to H//2

```
if (nu_y/(h*f_y_nyq))**2+(nu_x/(w*f_x_nyq))**2<=1:
    H[nu_y+h//2, nu_x+w//2]=1
```

Corresponding kernel in spatial domain:

Suppose we have a sampled image $f[m,n]$, which can be represented as a sum of impulses in the continuous domain:

$$f_s(x,y) = \sum_{m,n} f[m,n]\,\delta(x-m)\,\delta(y-n)$$

Resampling}

If we resample (zoom or shrink) by a factor k:

$$x' = km, \quad y' = kn$$

and define the resampled image as

$$f'[m,n] = f_s(x'/k, y'/k)$$

then in the frequency domain, this corresponds to scaling the frequencies:

$$F'(u',v') = \frac{1}{k^2}\,F\left(\frac{u'}{k},\frac{v'}{k}\right)$$

Aliasing Considerations

- If $k<1$ (shrinking), high frequencies in F may exceed the new Nyquist frequency and fold into lower frequencies (aliasing).

- The new Nyquist frequency after downsampling is $\frac{0.5}{k}$ cycles/sample.

Hence, we must pre-filter f to remove all frequencies beyond this:

$$|u| \leq \frac{1}{2k}, \quad |v| \leq \frac{1}{2k}$$

Frequency-Domain Filter

The ideal low-pass (box) filter in 2D is

H(u,v) = 1 if $|u| \leq \frac{1}{2k}, \quad |v| \leq \frac{1}{2k}$ and 0 otherwise

Spatial-Domain Kernel: The corresponding spatial kernel is the inverse Fourier transform of $H(u, v)$:

$$h(x, y) = \operatorname{sinc}\left(\frac{x}{k}\right) \operatorname{sinc}\left(\frac{y}{k}\right)$$

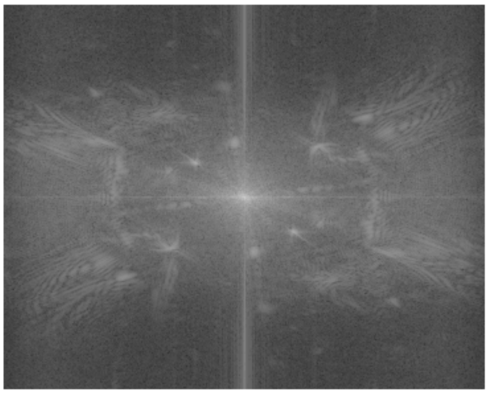
where $\operatorname{sinc}(t) = \frac{\sin(\pi t)}{\pi t}$.

[c]. Here, the outputs are zoomed back to original size using nearest neighbour interpolation.

For k=0.5

Original Image

FFT Spectrum (Original)



Bilinear Resized After Fourier Transform Filter

Bilinear Resized (Directly from Input)

FFT Magnitude Spectrum after Resize

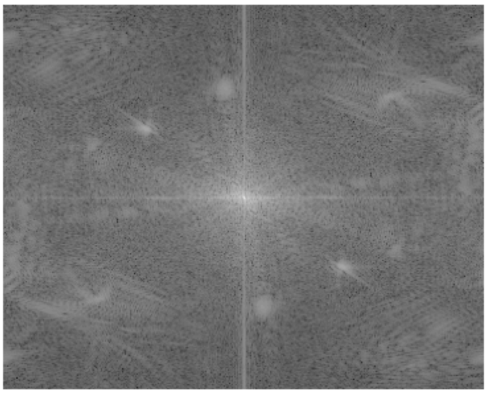
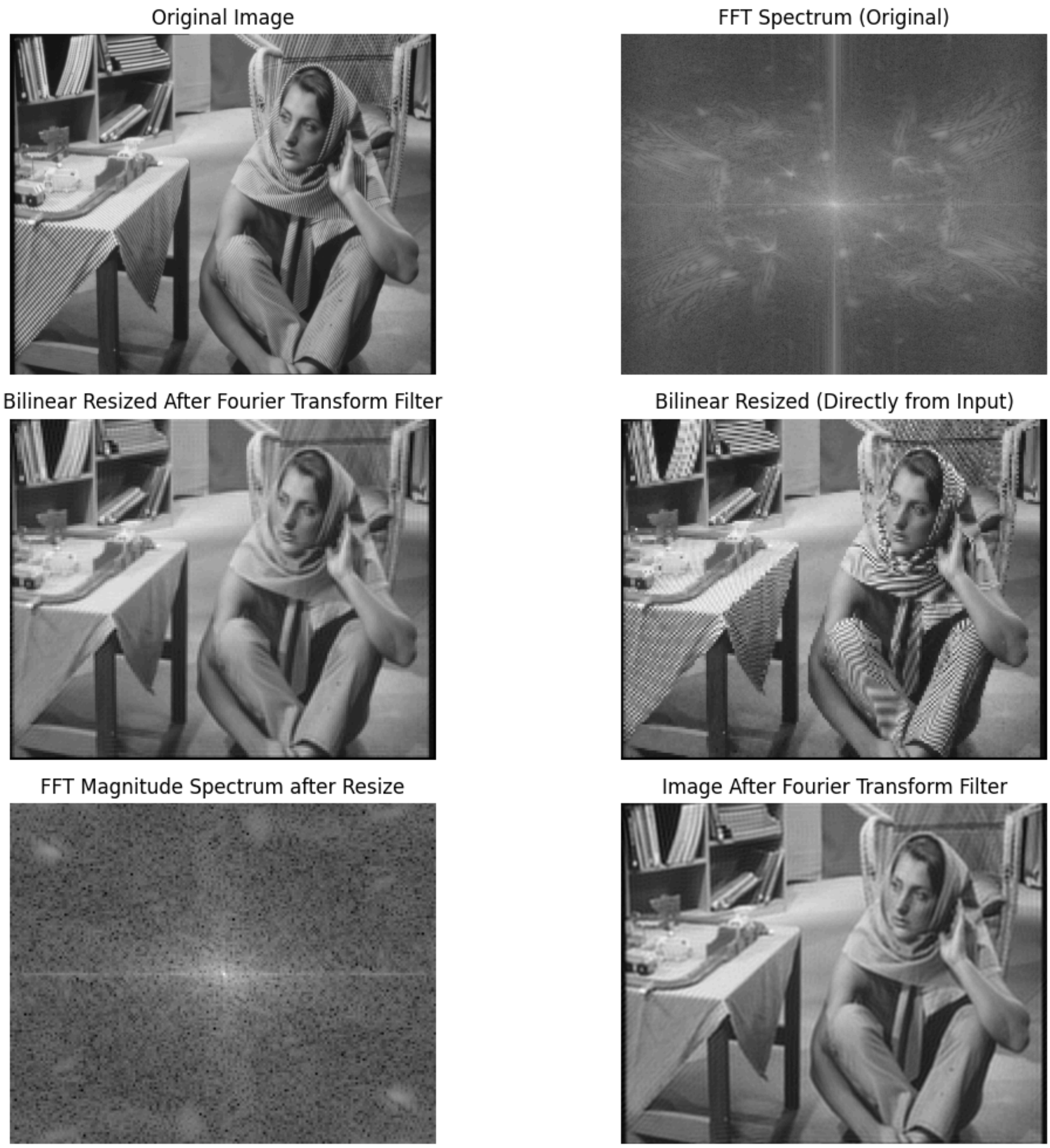


Image After Fourier Transform Filter

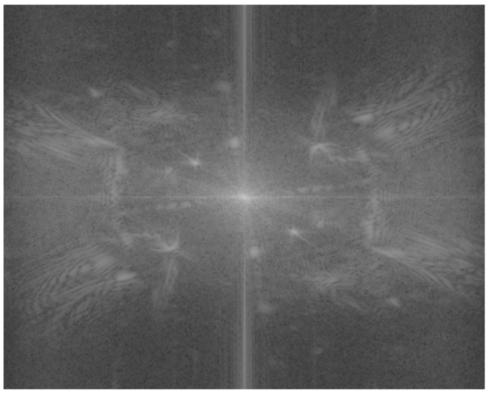
For k=0.25


Original Image
FFT Spectrum (Original)
Bilinear Resized After Fourier Transform Filter
Bilinear Resized (Directly from Input)
FFT Magnitude Spectrum after Resize
Image After Fourier Transform Filter

For k=0.125

Original Image

FFT Spectrum (Original)



Bilinear Resized After Fourier Transform Filter

Bilinear Resized (Directly from Input)

FFT Magnitude Spectrum after Resize

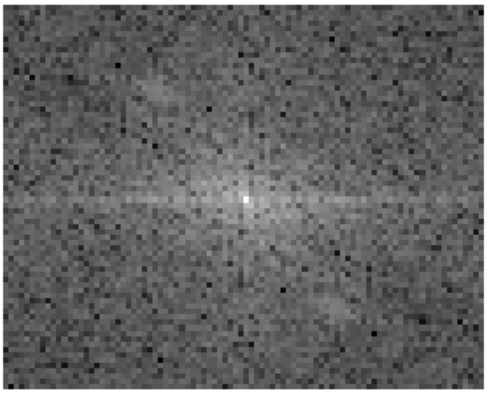


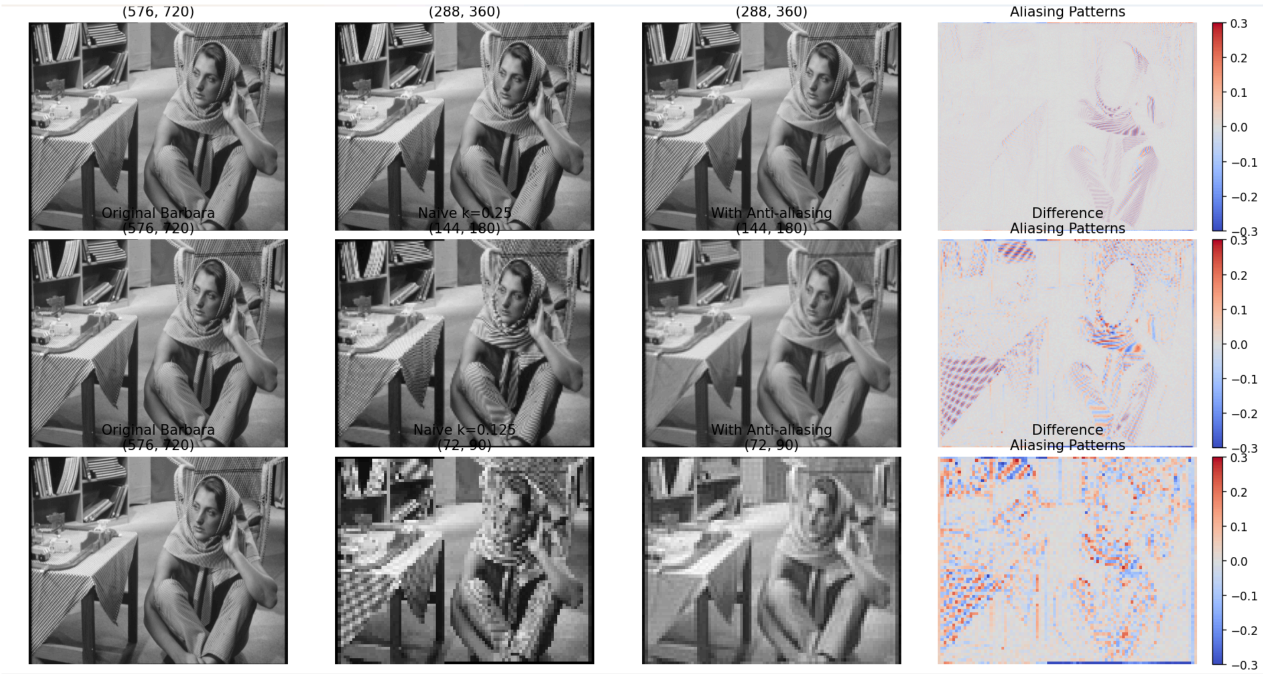
Image After Fourier Transform Filter

Results:-
Loaded image with OpenCV: (576, 720)
Using image with shape: (576, 720)

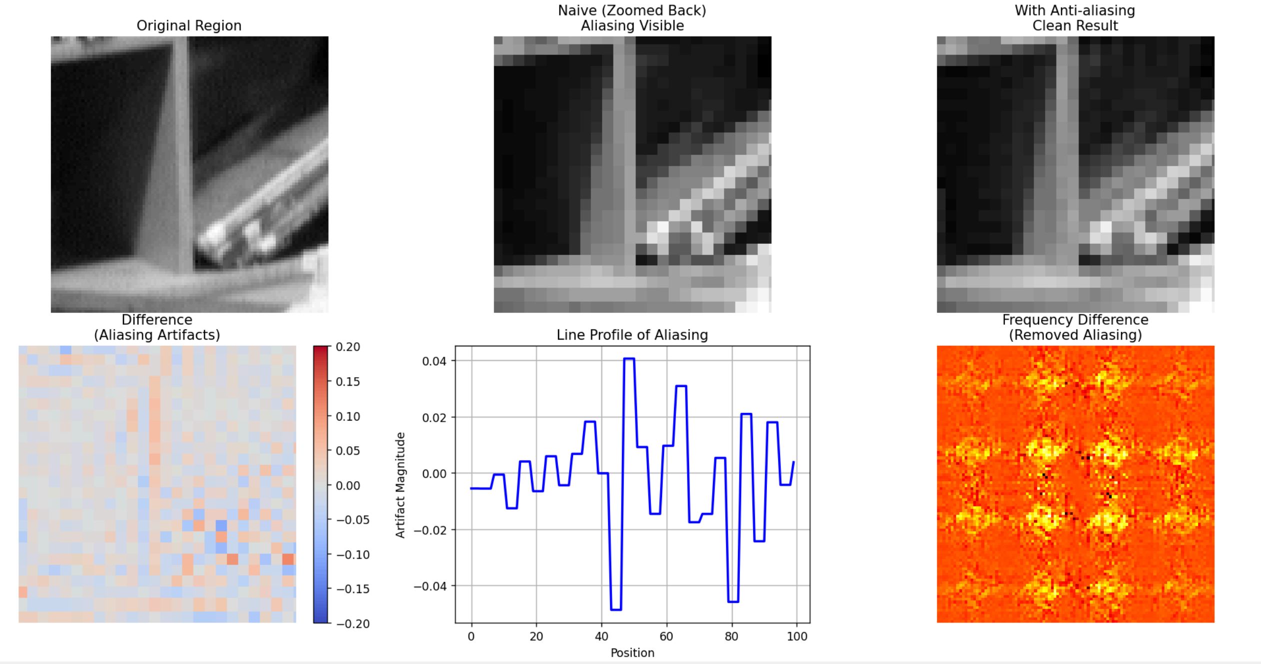
--- Testing k = 0.5 ---
Naive method - MSE: 0.003217, HF content: 0.022705
Proper method - MSE: 0.003397, HF content: 0.017598
Aliasing reduction: -5.6%

--- Testing k = 0.25 ---
Naive method - MSE: 0.007016, HF content: 0.020040
Proper method - MSE: 0.005602, HF content: 0.013340
Aliasing reduction: 20.1%

--- Testing k = 0.125 ---
Naive method - MSE: 0.011831, HF content: 0.014412
Proper method - MSE: 0.008360, HF content: 0.010851
Aliasing reduction: 29.3%



--- Detailed Analysis for k=0.25 —



Quantitative Results:

----

k	MSE Naive	MSE Proper	Reduction %	HF Naive	HF Proper
0.5	0.003217	0.003397	-5.6	0.022705	0.017598
0.25	0.007016	0.005602	20.1	0.020040	0.013340
0.125	0.011831	0.008360	29.3	0.014412	0.010851

Key Observations for Barbara Image:

1. Aliasing appears as moiré patterns in textured regions (scarf, cloth)
2. Anti-aliasing filtering preserves fine details while removing artifacts
3. The reduction in MSE shows significant improvement in image quality
4. High-frequency content is more accurately represented with filtering
5. The ideal lowpass filter with cutoff k/2 effectively prevents aliasing

Sampling and Scaling

A discrete image f[m,n] represents samples of a continuous scene f(x,y) spaced by ΔX,ΔY.

Scaling the image by a factor k changes the sample spacing to

$$\Delta X' = \frac{\Delta X}{k}, \quad \Delta Y' = \frac{\Delta Y}{k}.$$

- k>1 → zooming (interpolation only)
- 0<k<1 → shrinking (requires pre-filtering)

According to the **Sampling Theorem**, a band-limited signal can be recovered if sampled at a rate greater than twice its highest frequency:

$$\frac{1}{\Delta X'} > 2\mu_{\max}.$$

When k<1, the new sampling rate is lower, so frequencies above k/2 of the normalized Nyquist rate must be suppressed by a **low-pass (anti-aliasing) filter**.

Frequency-Domain Interpretation

$$f(x,y) * h(x,y) \quad \longleftrightarrow \quad F(u,v)\, H(u,v)$$

Thus, filtering in the spatial domain corresponds to multiplication in the frequency domain.
To prevent aliasing before subsampling, H(u,v) should ideally be an **ideal low-pass filter** defined as:

$$H(u,v) = \begin{cases} 1, & \sqrt{u^2+v^2} \le \frac{k}{2}, \\ 0, & \text{otherwise.} \end{cases}$$

Its inverse transform is a **sinc-type kernel.**

Implementation Summary

1 Naive Resizing

- Implemented bilinear or nearest-neighbor interpolation only.
- No frequency filtering; hence, high-frequency components beyond the new Nyquist limit remain, producing **aliasing artifacts** such as Moiré patterns.
- Implemented using mesh-based coordinate mapping with *numpy*.

2 Proper Resizing with Anti-Aliasing

- Before downsampling (k<1k<1k<1), the image is transformed to the **frequency domain** via 2D FFT:

  $$F(u,v)=FFT2\{f(x,y)\}.$$
- An **ideal circular low-pass filter** of cutoff k/2k/2k/2 is applied:

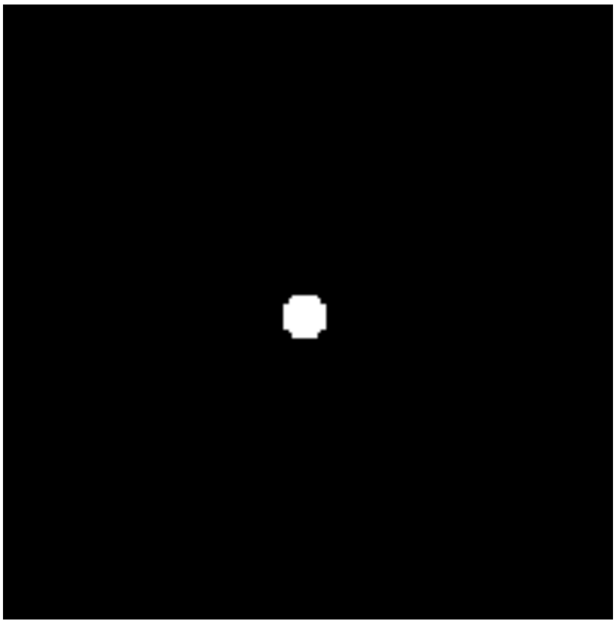
  $$F'(u,v)=F(u,v)\,H(u,v).$$
- The filtered spectrum is inverse-transformed using IFFT, yielding a band-limited version ready for safe downsampling.
- Implemented in Python using *numpy.fft* and visualized with *matplotlib*.

3 Experimental Setup

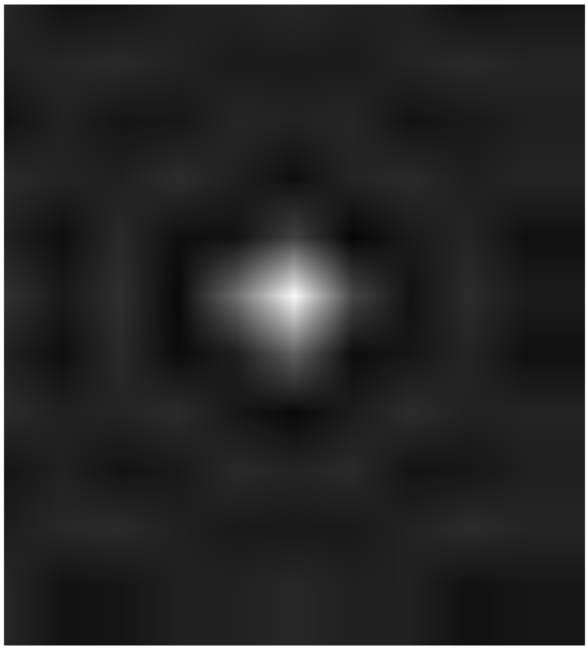
- Test image: *barbara.bmp* (or synthetic high-frequency image substitute).
- Scaling factors: k=1/2, 1/4, 1/8.
- Each result was also zoomed back to the original size for comparison and quantitative analysis (MSE, high-frequency content).

2. a. Psf values = [10, 20, 50]

Point image



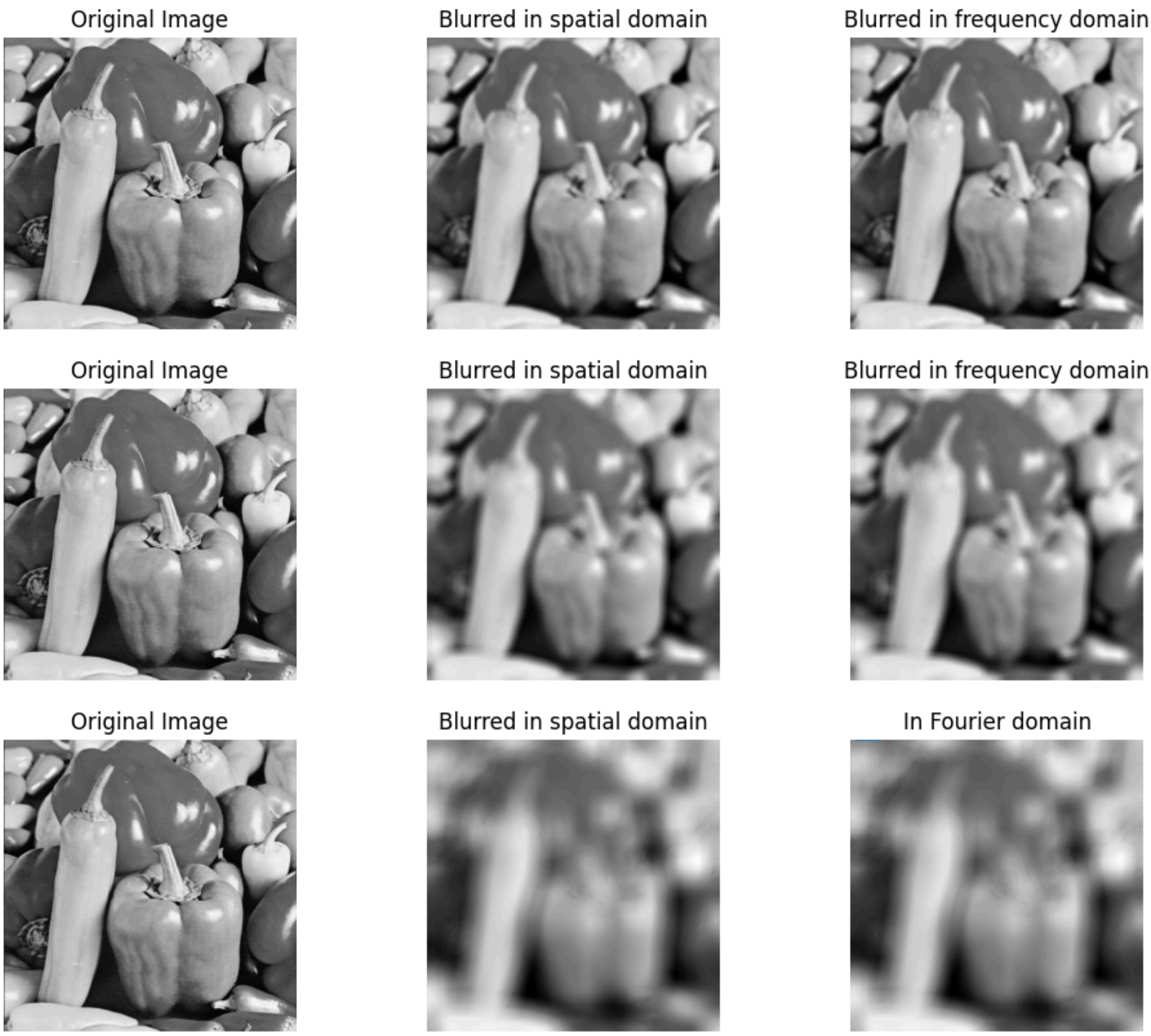
After blurring



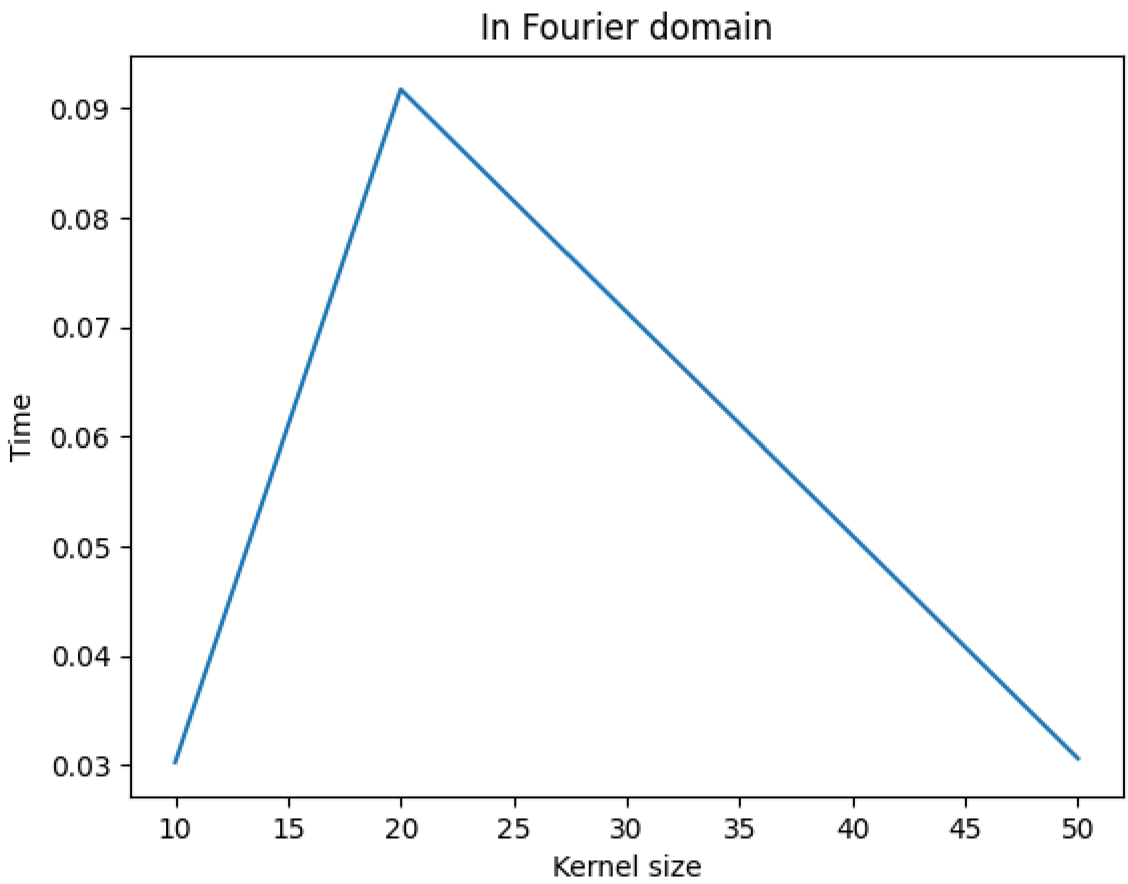
b. (ii)

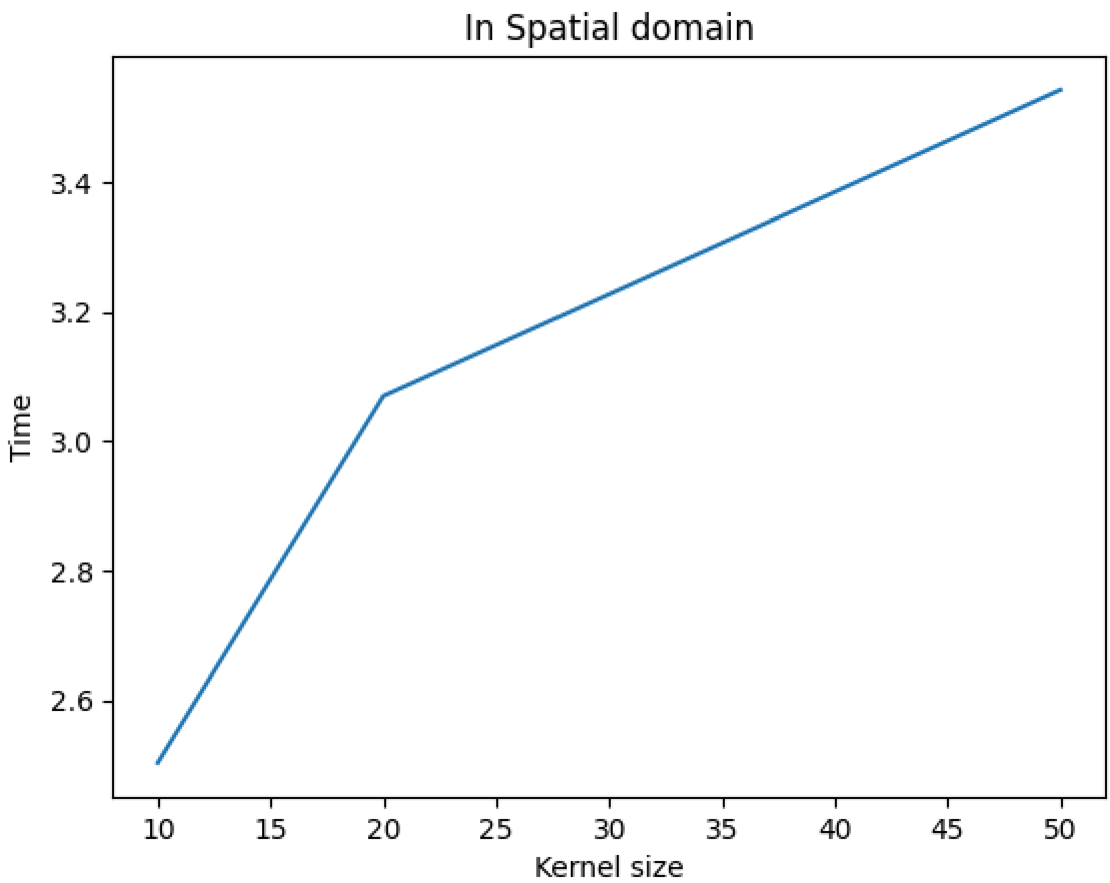
Average pixel intensities difference=

```
1.93998740035929
1.332597188914199
0.7110189591301395
```


Original Image
Blurred in spatial domain
Blurred in frequency domain
Original Image
Blurred in spatial domain
Blurred in frequency domain
Original Image
Blurred in spatial domain
In Fourier domain

c.


In Fourier domain
0.09
0.08
0.07
0.06
0.05
0.04
0.03
Time
10
15
20
25
30
35
40
45
50
Kernel size

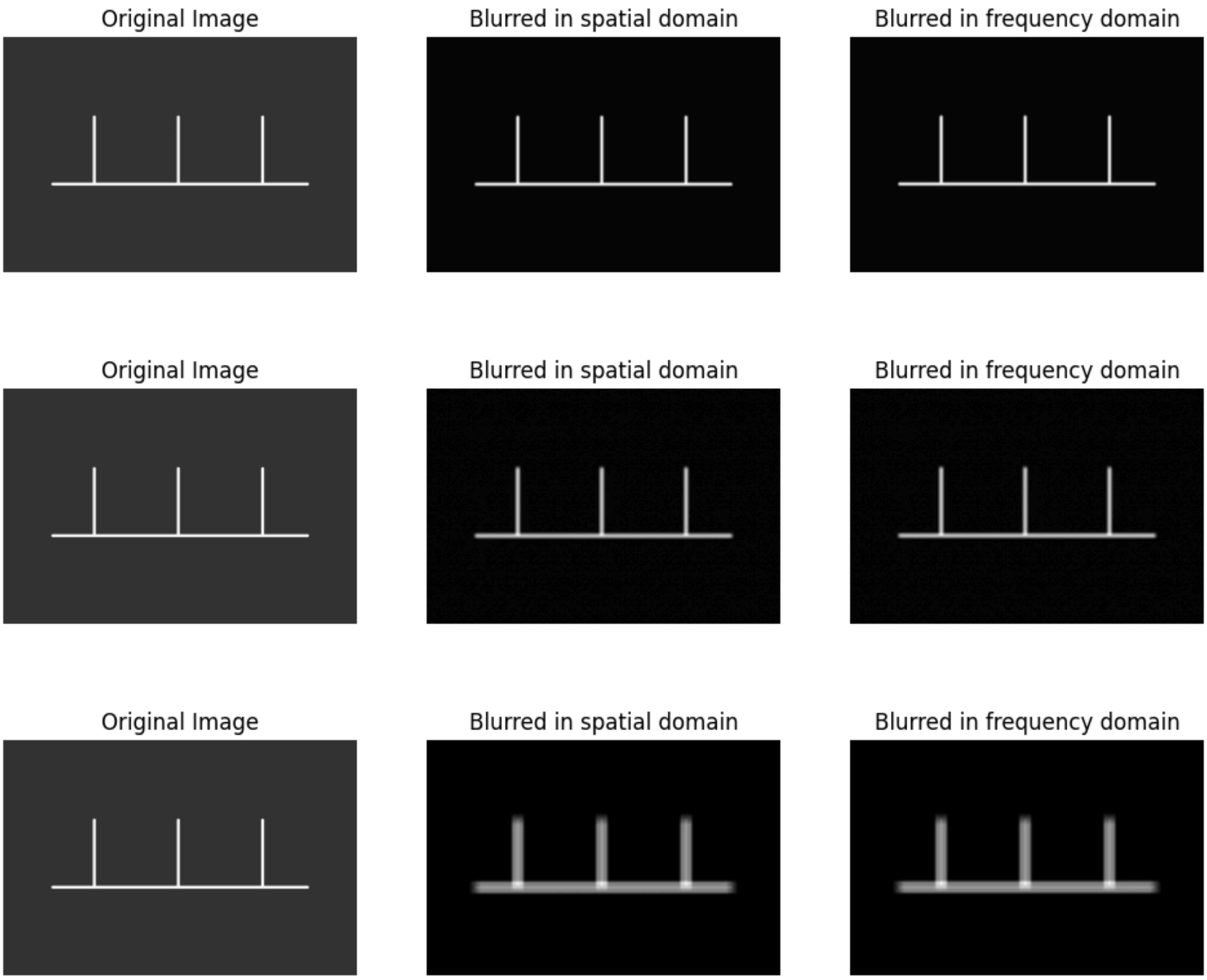

In Spatial domain
3.4
3.2
3.0
2.8
2.6
Time
10
15
20
25
30
35
40
45
50
Kernel size

c. The frequency domain transformation takes significantly less time compared to spatial domain transformation.

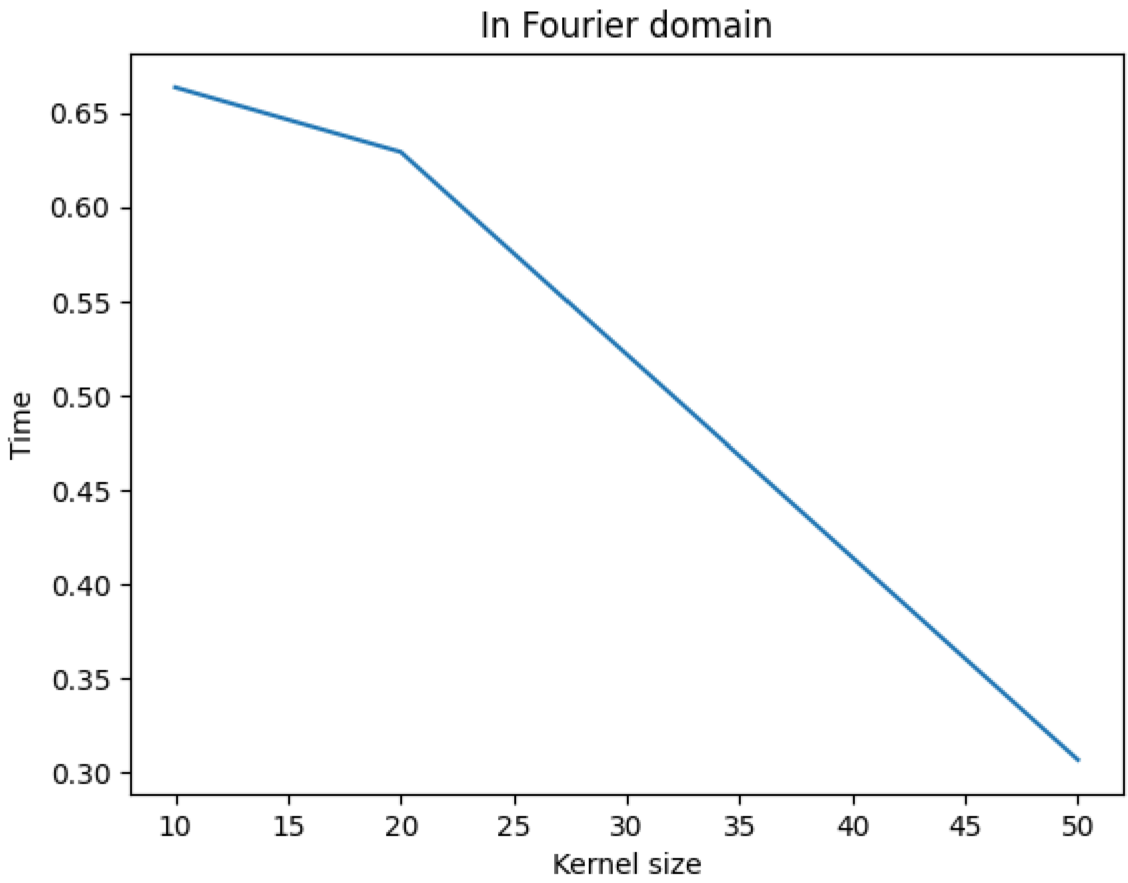
Average intensity difference:

```
0.35021864146097537
0.22283373508618512
0.10023862655466376
```

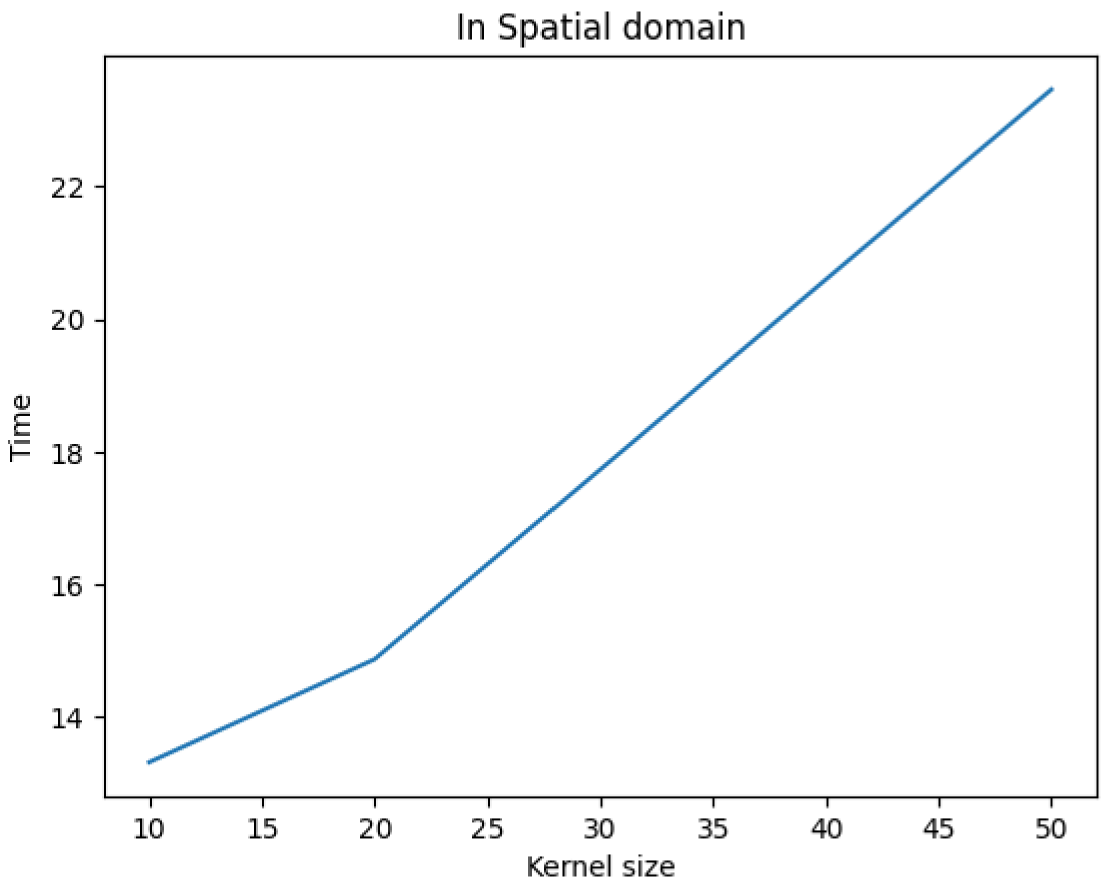
(i)



d.



In Fourier domain
0.65
0.60
0.55
0.50
0.45
0.40
0.35
0.30
Time
10
15
20
25
30
35
40
45
50
Kernel size


In Spatial domain
Time
Kernel size
14
16
18
20
22
10
15
20
25
30
35
40
45
50

Part-2

3.

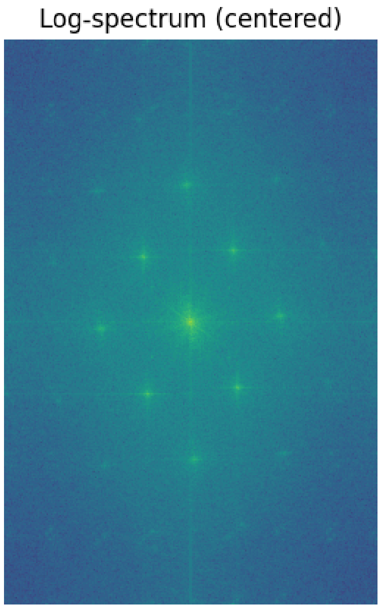
Ideal filter: $H(u, v) = \{1 \; D(u, v) \le D0, \; 0 \text{ otherwise}\}$

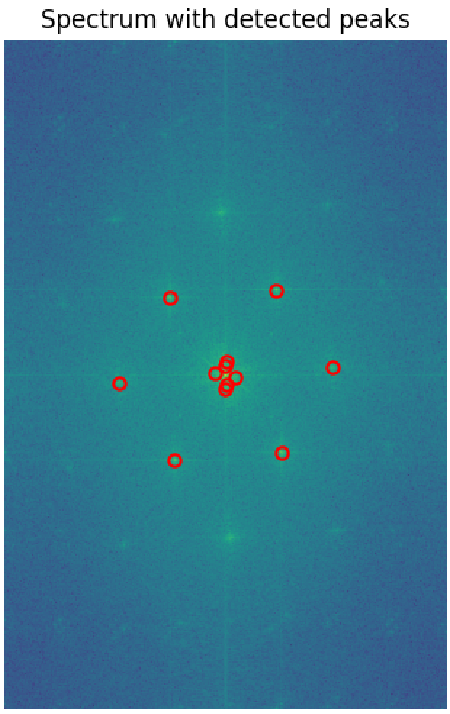
• $D(u, v)$ = distance from DC component

D0: cutoff frequency

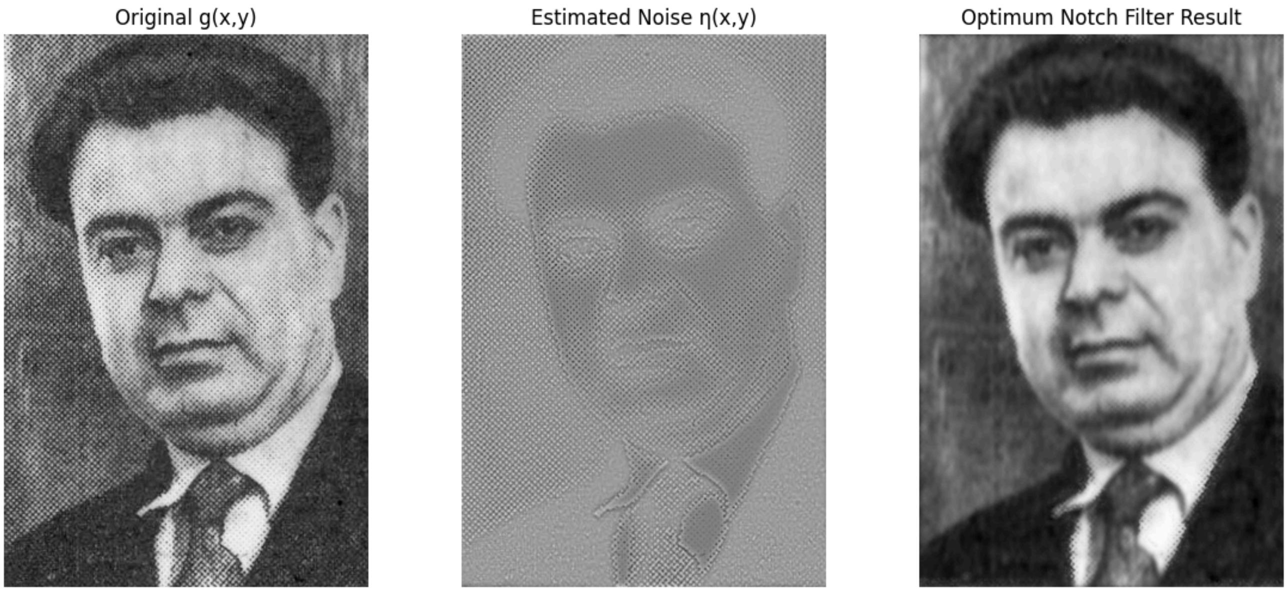
Gaussian low-pass filter $H(u, v) = \exp(-1/2 \, (D(u, v)/D0)^2)$

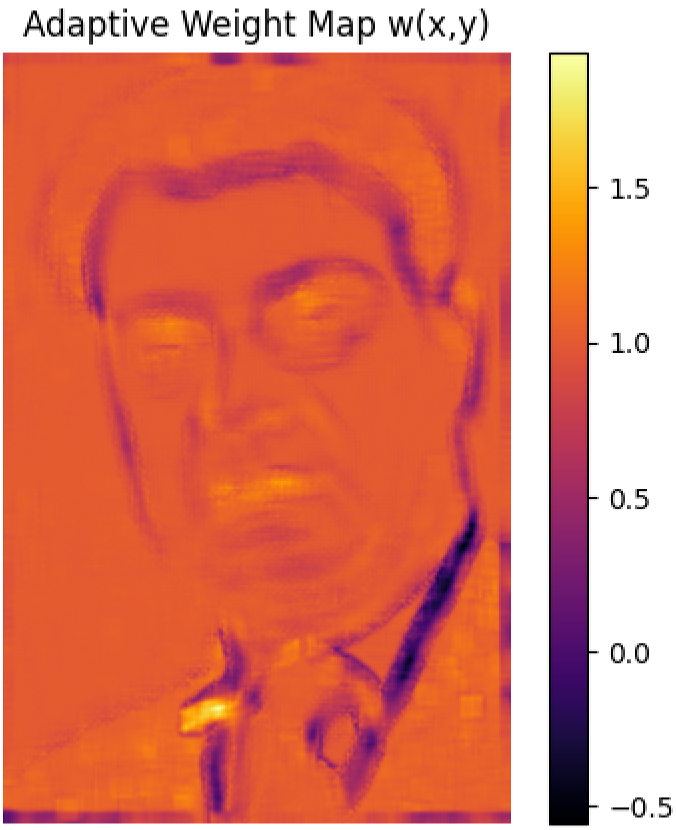
image shape: (748, 493)





Window size = 21





cov(g,η): min=-0.0029, max=0.0559
var(η): min=0.0005, max=0.0569
w: mean=0.9564, min=-0.5564, max=1.9355

d. Optimum notch filtering:

For each pixel (x, y), consider m×n neighbourhood, find scalar w such that
$\hat{f} = g - w\eta$ is optimal, according to some metric
$\hat{f}(\mathcal{N}) = g(\mathcal{N}) - w\eta(\mathcal{N})$
One choice: choose w to minimize variance of $\hat{f}(\mathcal{N})$
$\text{var}(\hat{f}) = \text{var}(g) + w2\ \text{var}(\eta) - 2w\ \text{cov}(g, \eta)$
$\Rightarrow w = \text{cov}(g, \eta)/\text{var}(\eta)$
Equivalent to choosing w so that $\hat{f}(\mathcal{N})$
has no correlation with $\eta(\mathcal{N})$

```
image shape: (718, 1303)
```

There's an **inverse relationship** between spatial spacing and frequency spacing:

f=l/d

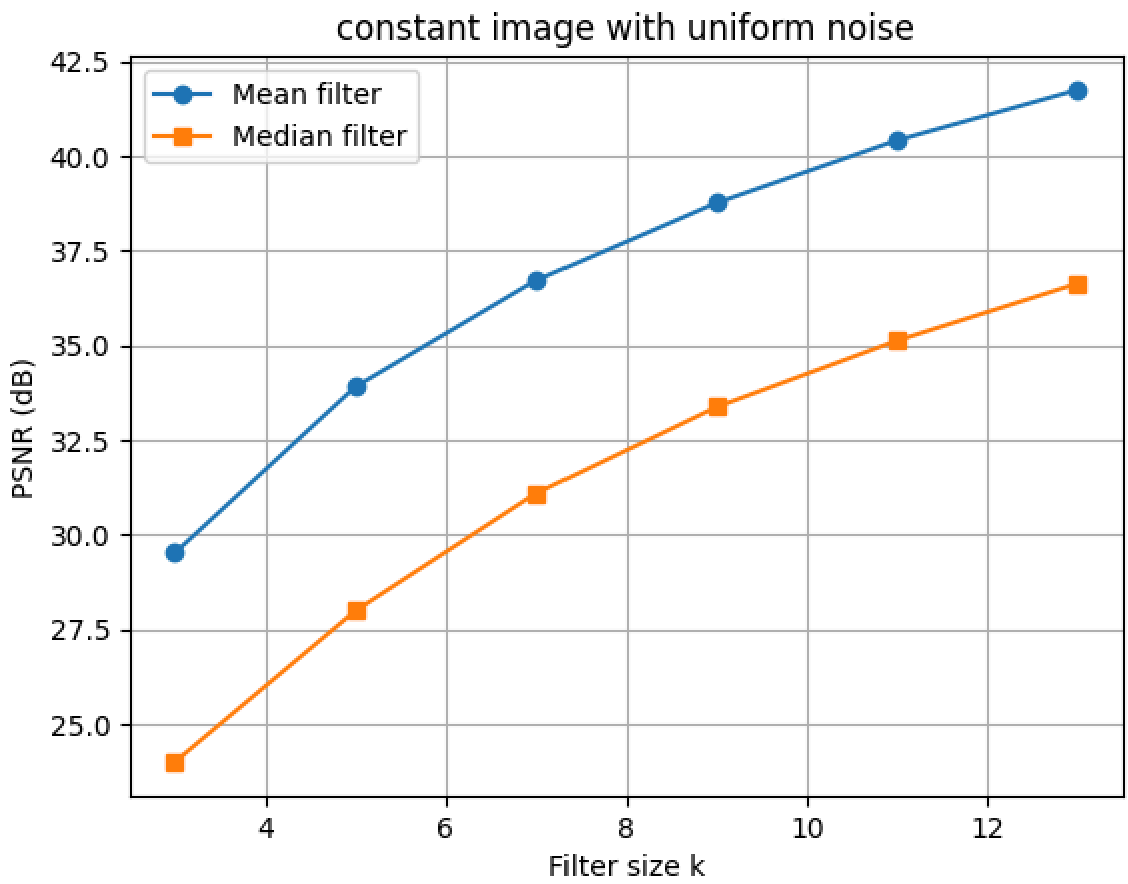
where

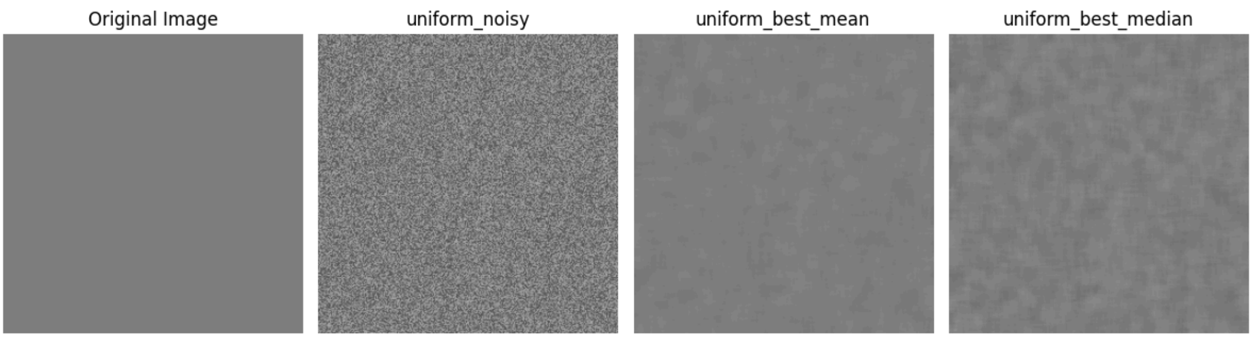
- f= spatial frequency (in cycles per unit distance),
- d = distance between dots (dot spacing).

So:

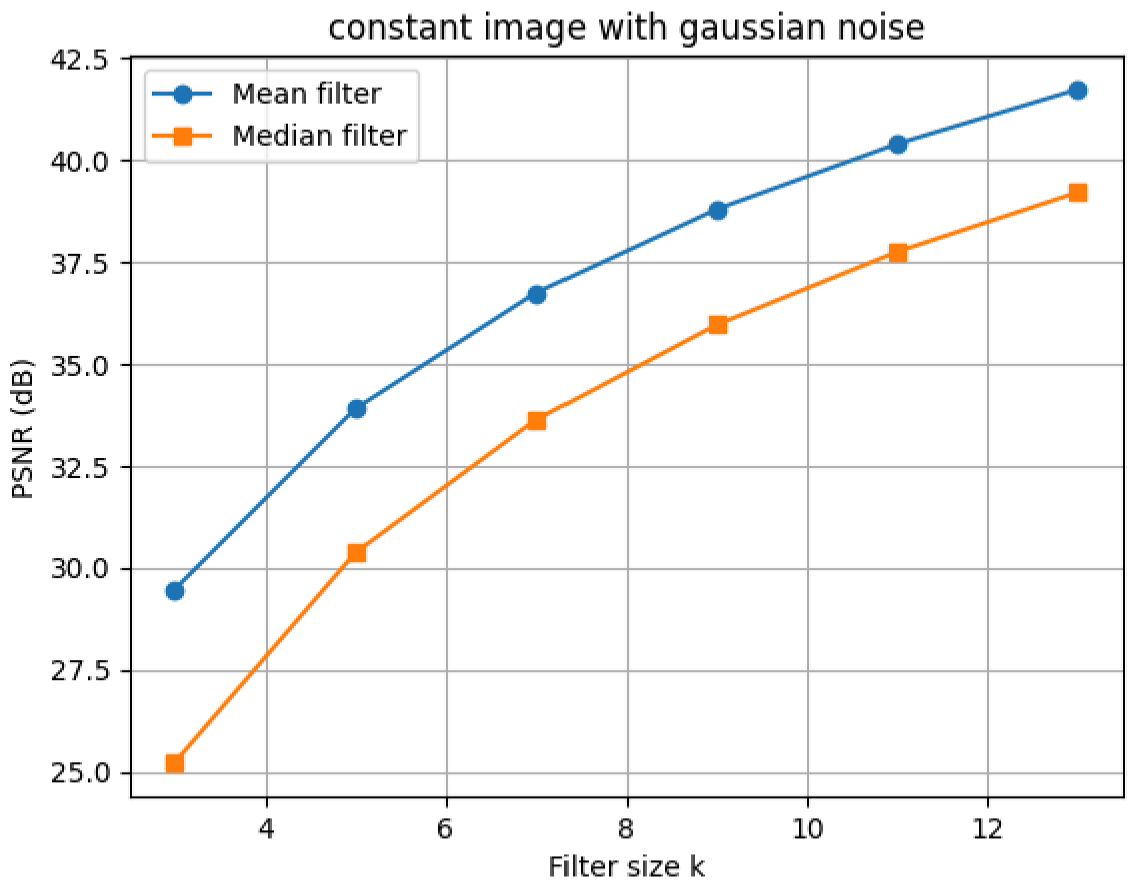
- **If dots are closely spaced (small d)**, the **spikes move farther out** from the DC center — higher frequencies.
- **If dots are far apart (large d)**, the **spikes appear closer** to the center — lower frequencies.

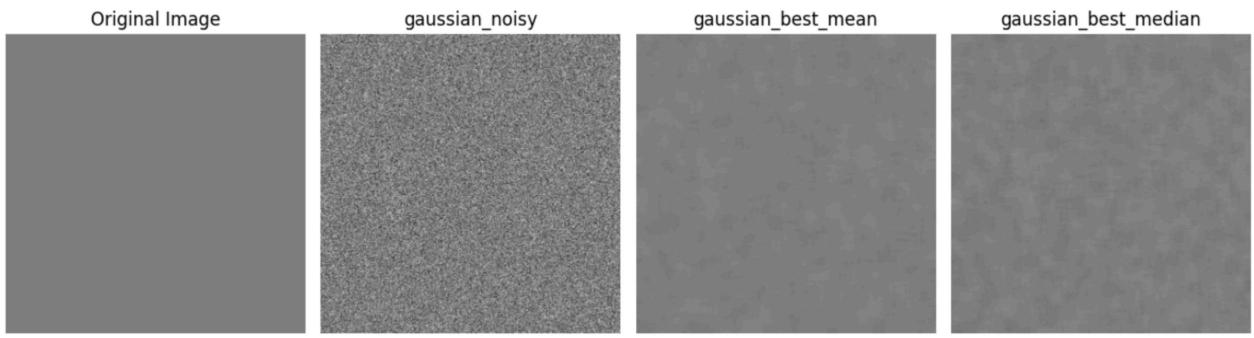
4. constant - uniform noisy PSNR: 20.01 dB



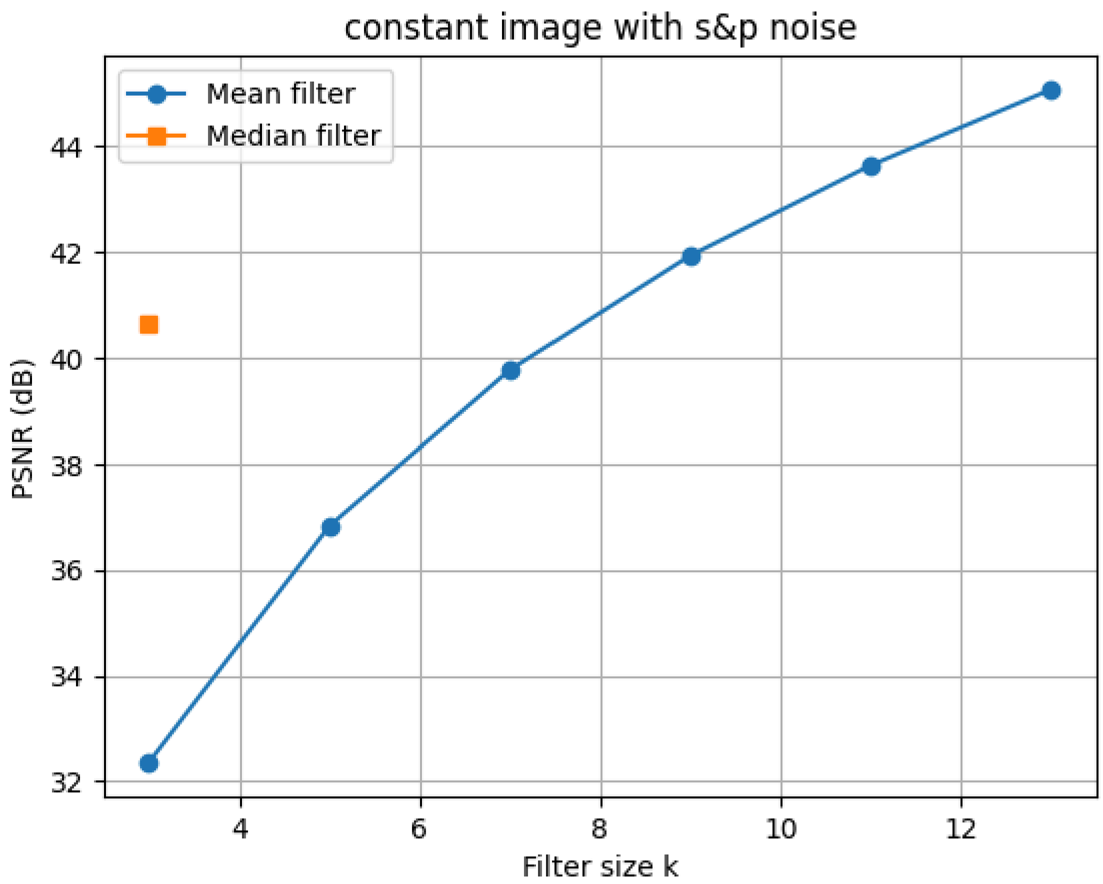


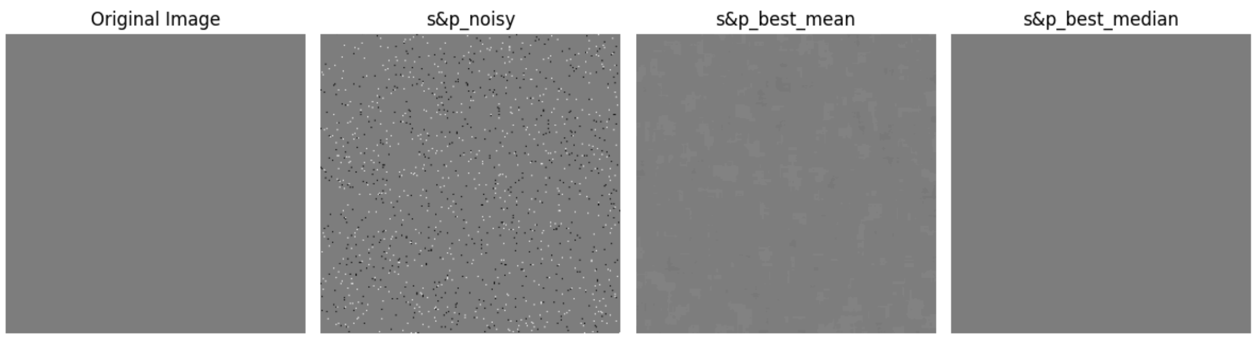
constant - gaussian noisy PSNR: 20.00 dB



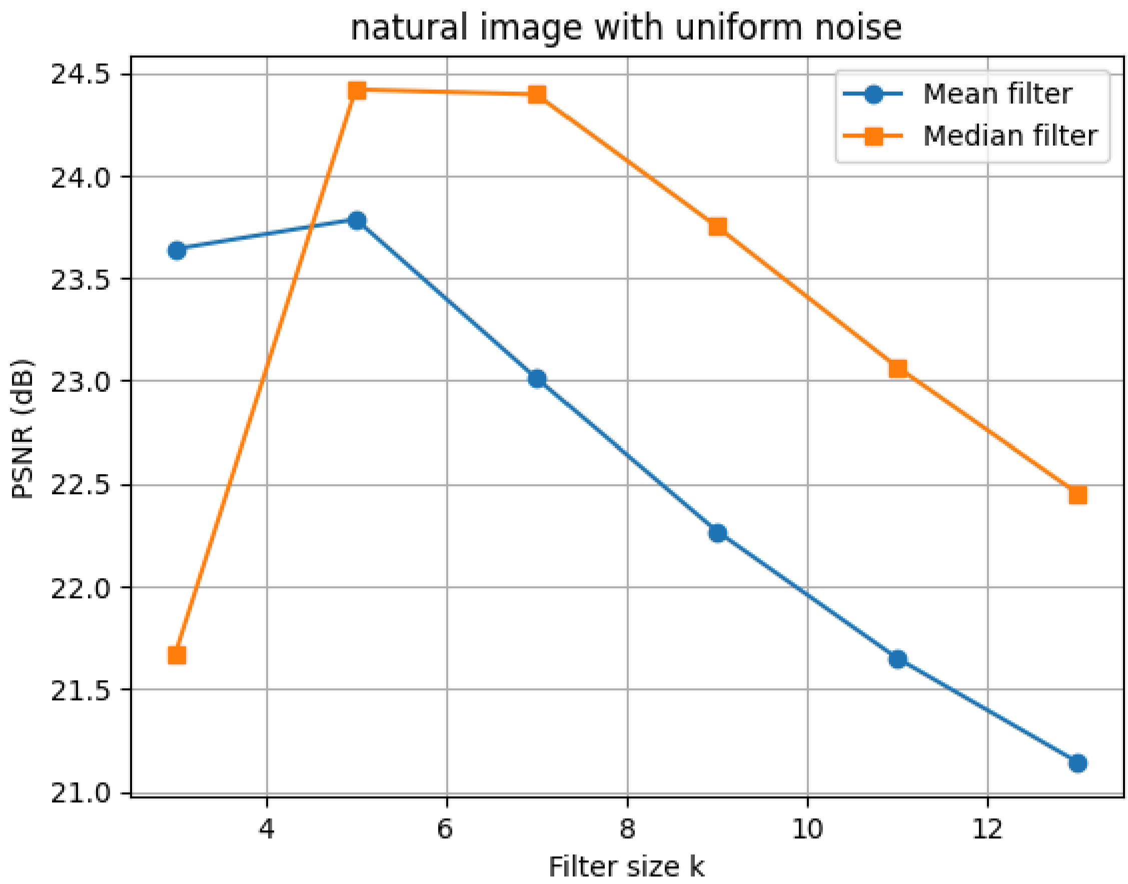


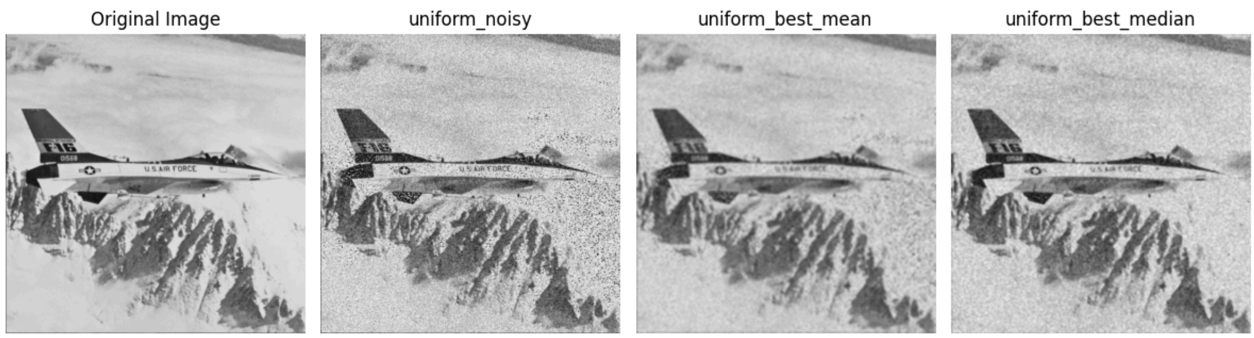
constant - s&p noisy PSNR: 22.88 dB



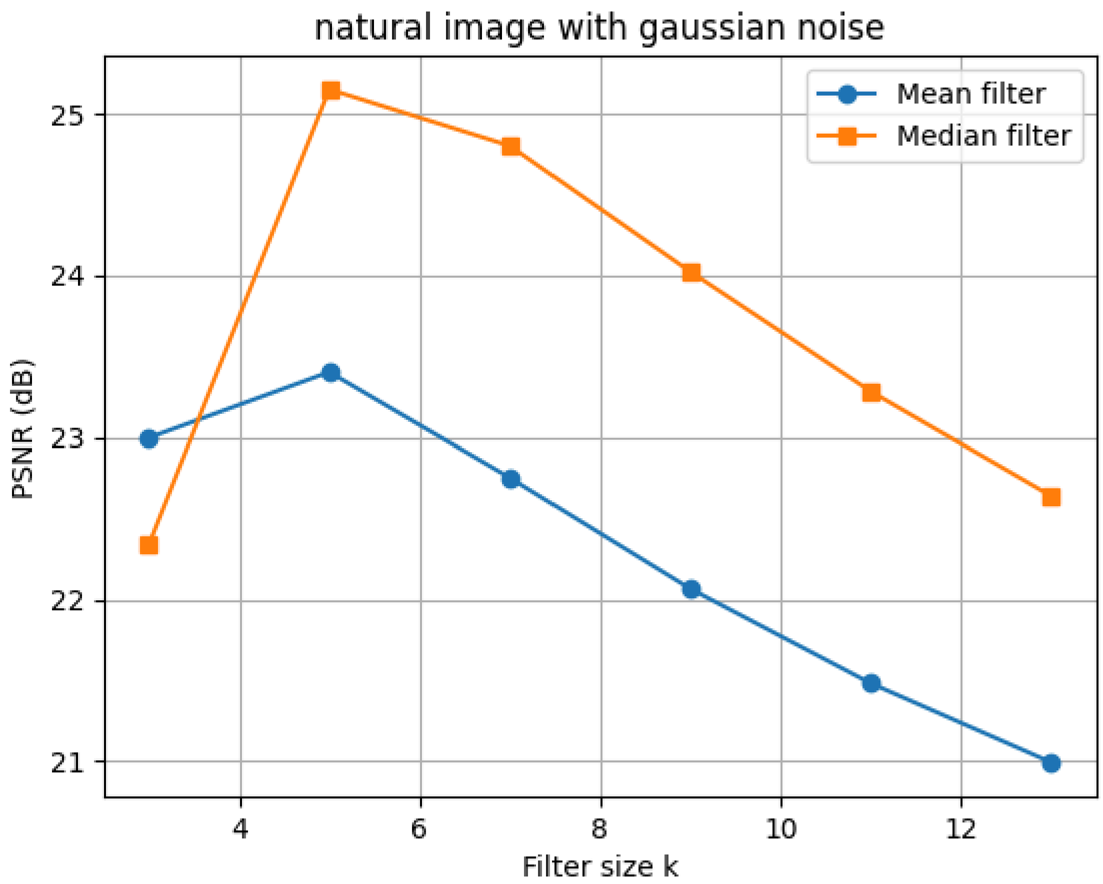


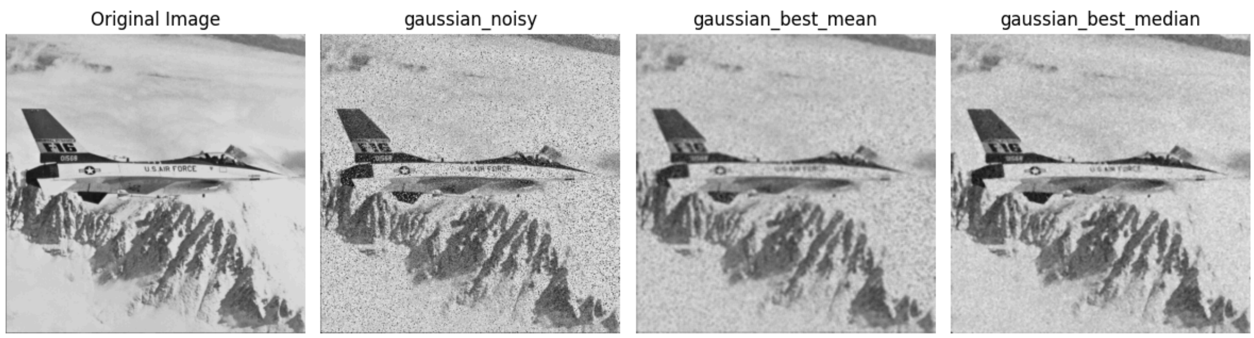
natural - uniform noisy PSNR: 17.04 dB



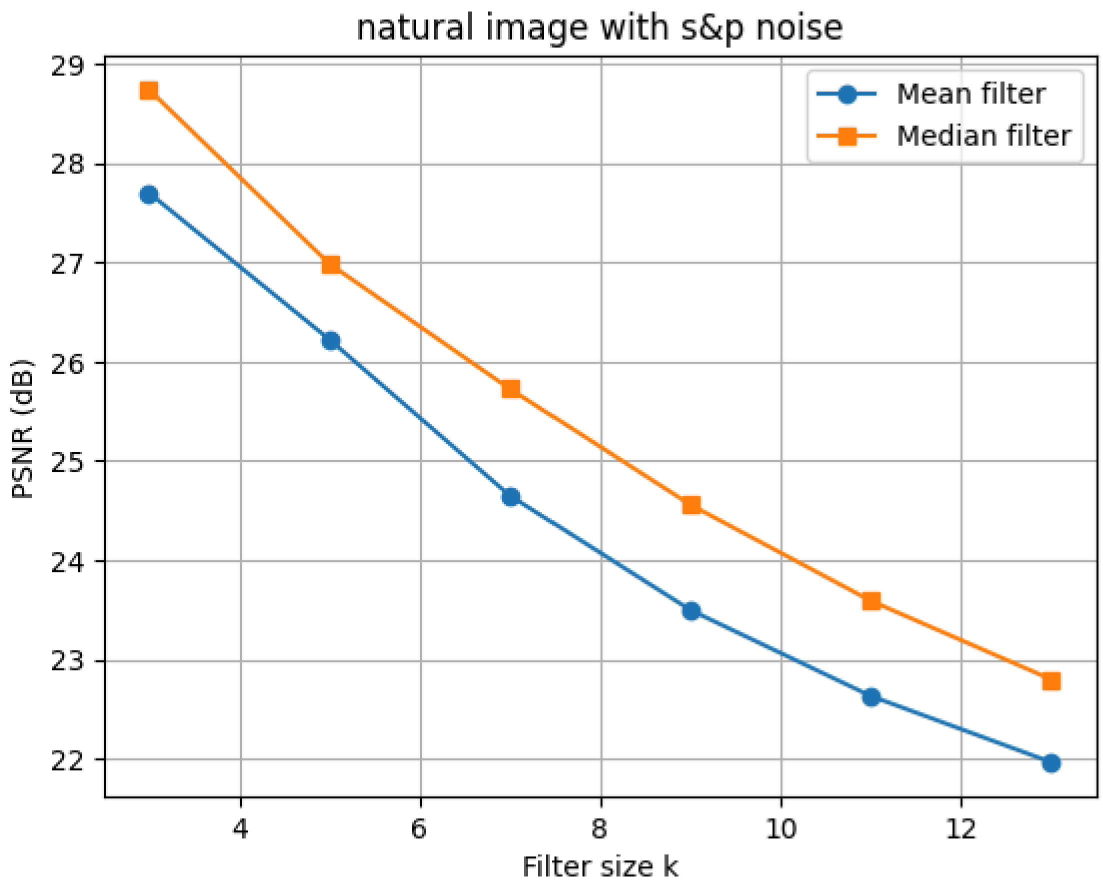


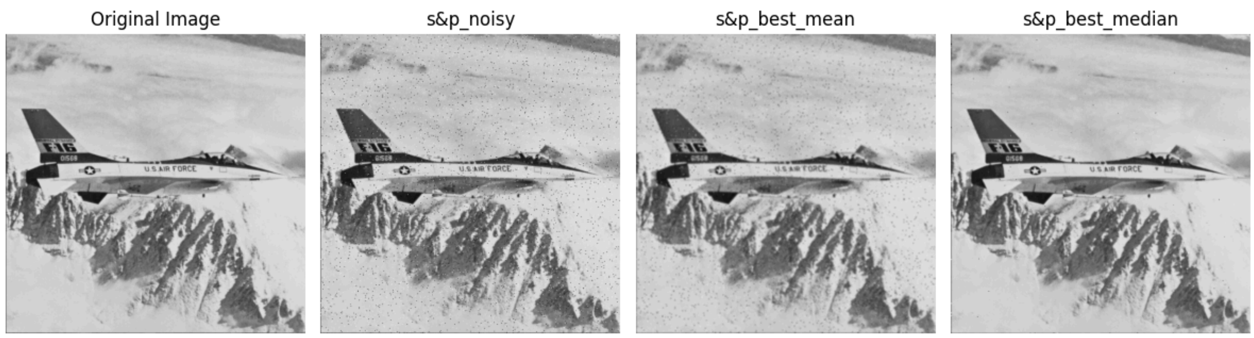
natural - gaussian noisy PSNR: 16.07 dB





natural - s&p noisy PSNR: 21.78 dB





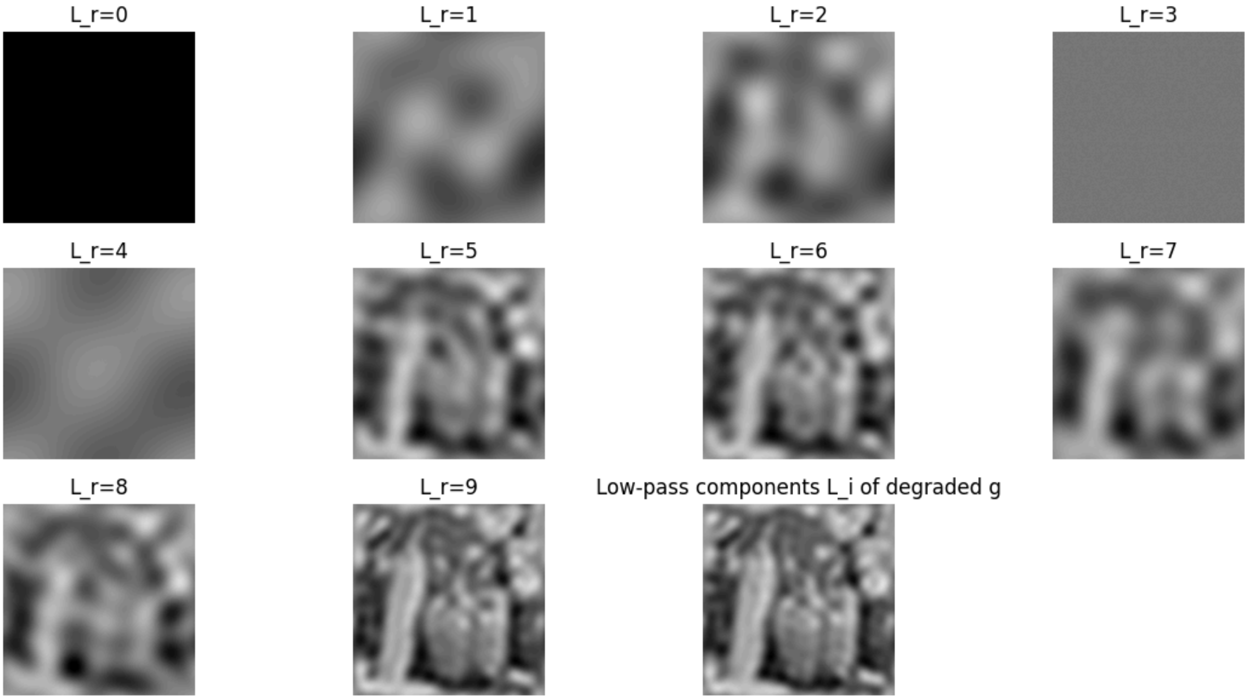
Filtering is a bias–variance trade-off:

- Noise → high variance, low bias
- Filtering → reduces variance (noise), increases bias (blur)

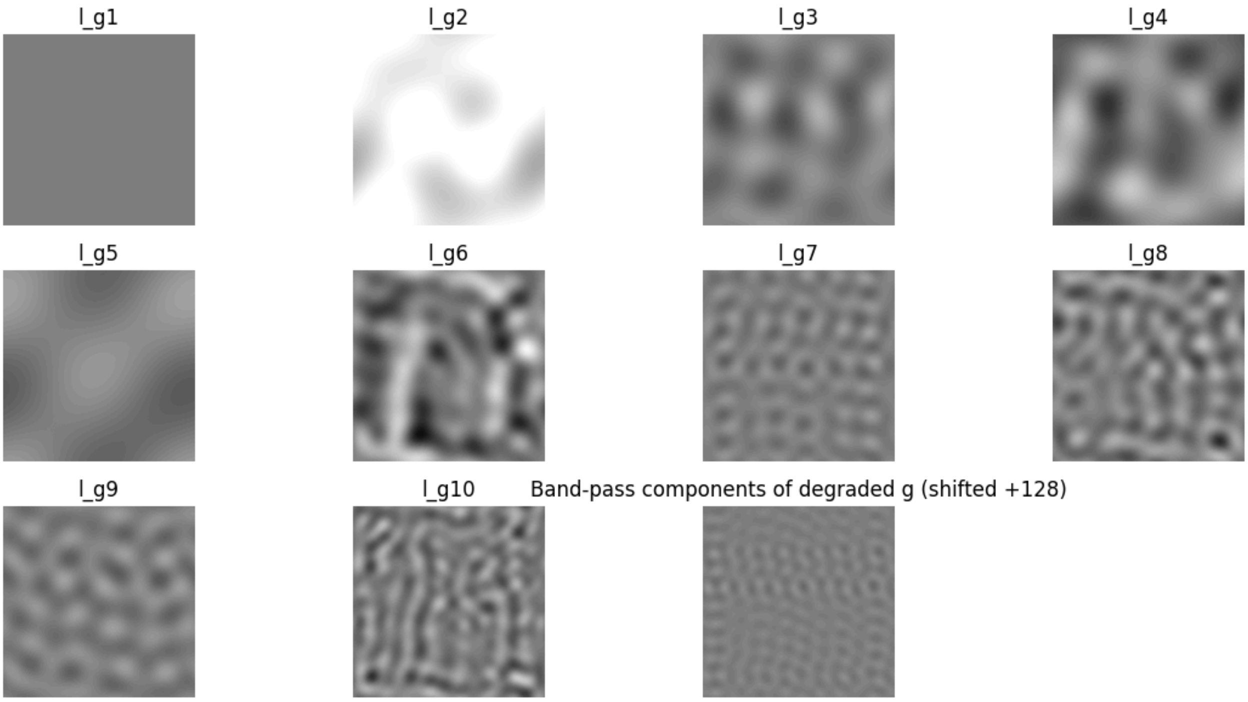
At first, adding a filter (increasing kernel size) reduces variance — so PSNR improves.
But beyond a certain point, the filter introduces bias — destroying real image details — and PSNR drops.

5. Degraded PSNR (w.r.t. blurred_large): 20.067831551134244

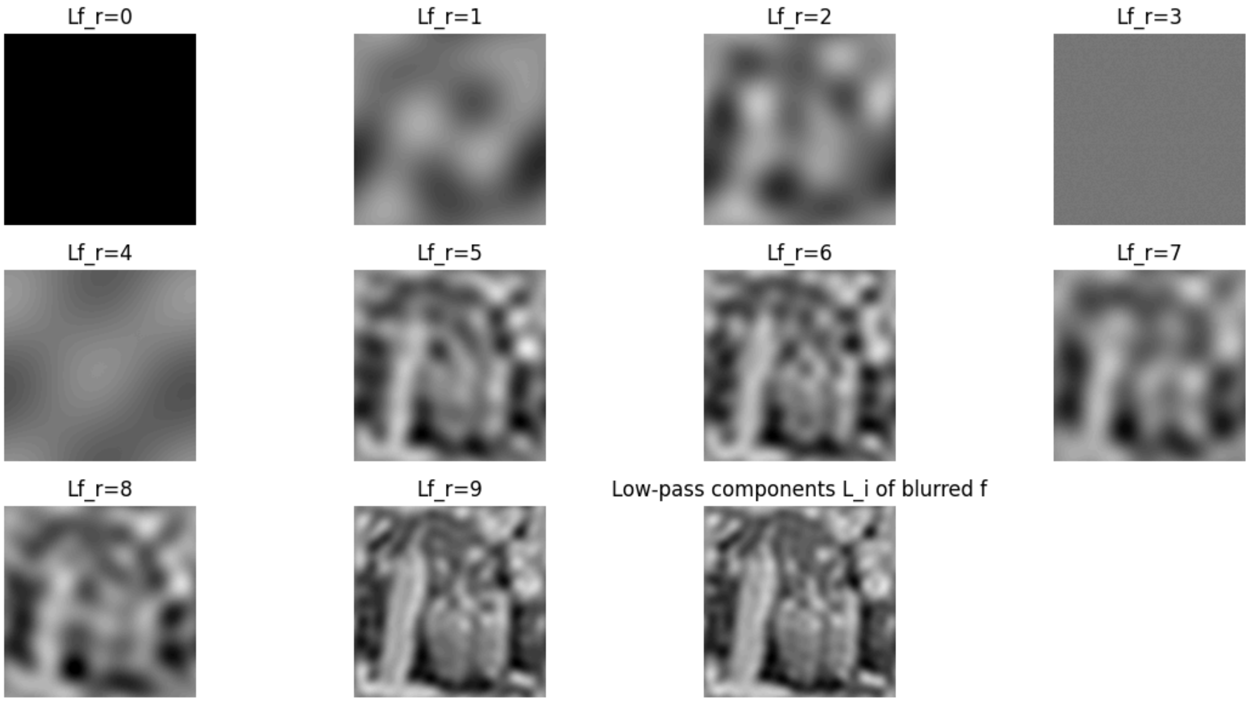
a. Showing low-pass images L_i (degraded g):



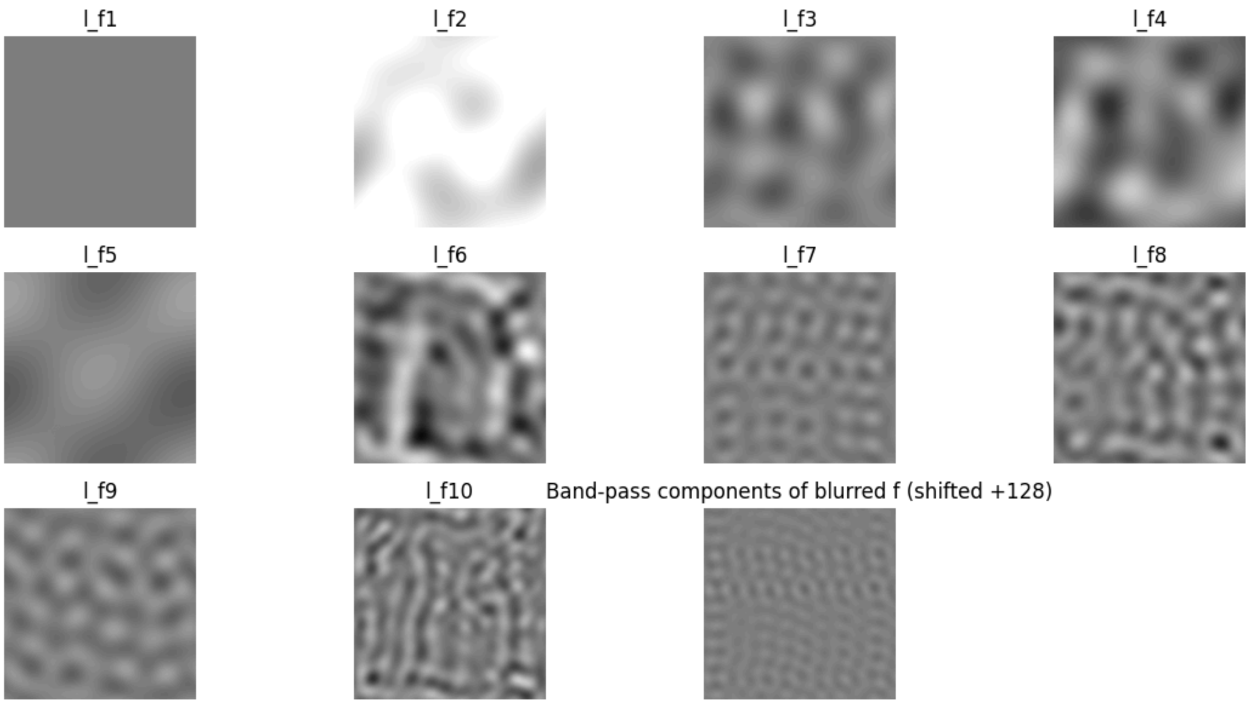
Showing band-pass images l_i = L_i - L_{i-1} (degraded g) with +128 added:



Showing low-pass images L_i of blurred (clean) f:



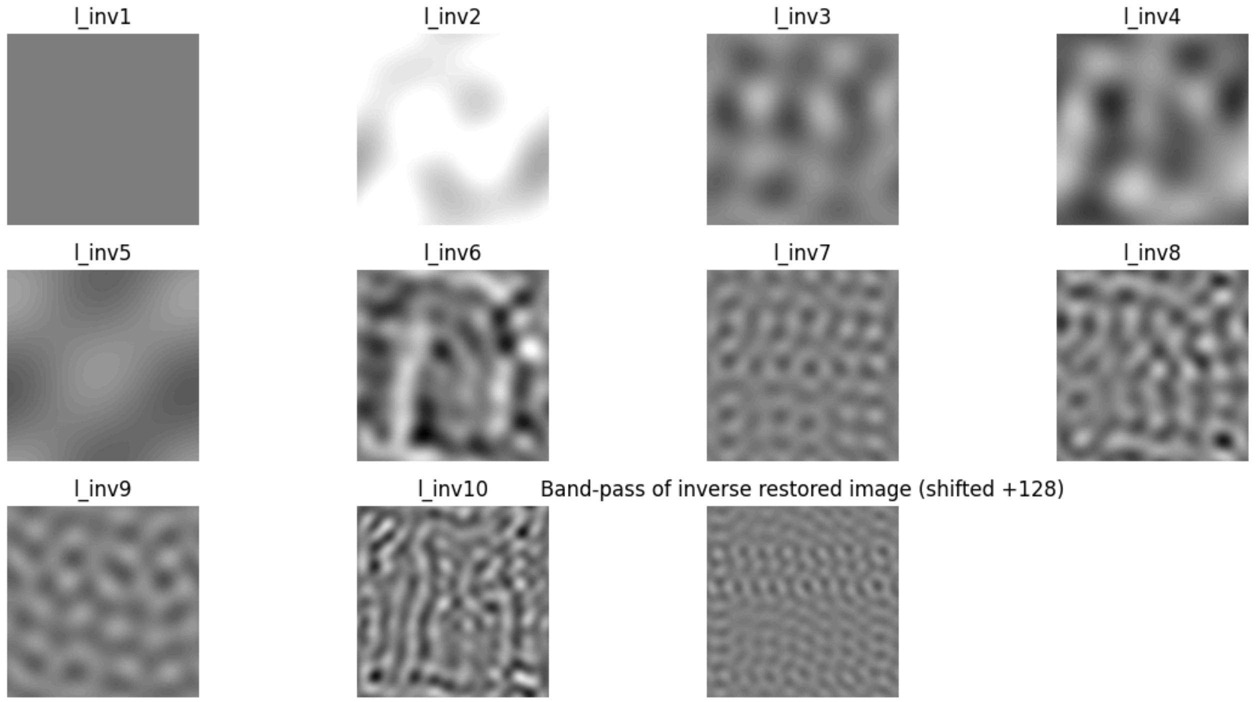
Showing band-pass images l_i of blurred (clean) f (with +128):



b. Inverse filter: $F^{\wedge} = G/H = GH^*/|H|^2$

PSNR of naive inverse filter reconstruction w.r.t. blurred (clean) f: 16.231612731826942

Band-pass of inverse restored image (with +128):



Inverse filter reconstruction

Inverse filter gives poorly restored image.

Estimated sigma_n^2 = 650.9073, sigma_f^2 = 2147.2844, K_est = 3.031304e-01

c. Wiener filter: F^ = WG = F = $\frac{H^*G}{HH^*+\lambda S_f/S_n}$

Plancherel's theorem

$$\int_{-\infty}^{\infty} |f(t)|^2 dt = \int_{-\infty}^{\infty} |F(\omega)|^2 d\omega$$

for 2-D signal $f(m, n)$ and its DFT $F(u,v)$

$$\sum_{m,n} |f(m,n)|^2 = \frac{1}{MN} \sum_{u,v} |F(u,v)|^2$$

where $M \times N$ is image size

Total energy in space domain =
Total energy in frequency domain

Hence, we use Plancherel's theorem to derive the relation between power spectrum and variance.

In space domain,

Total noise energy,

$$\frac{1}{MN}\sum_{m,n} \mathbb{E}\left(\eta(m,n)^2\right) - \underbrace{\cancel{\mathbb{E}\left(\eta(m,n)\right)^2}}_{\to 0} = \sigma_\eta^2 \text{ or}$$

for each (m,n) $\mathbb{E}\left(\eta(m,n)^2\right) - \cancel{\mathbb{E}\left(\eta(m,n)\right)^2} \to 0 = \sigma_\eta^2(m,n)$

If $\eta(m,n)$ is not uniform for all (m,n)

$$\Rightarrow \sum_{m,n} \mathbb{E}\left[|\eta(m,n)|^2\right] = \sum_{m,n}\left[\sigma_\eta(m,n)\right]^2$$

$$|||$$

By Plancherel's theorem

$$\mathbb{E}\sum_{m,n}|\eta(m,n)|^2 = \frac{1}{MN}\mathbb{E}\sum_{u,v}|N(u,v)|^2$$

$$= \frac{1}{MN}\sum_{u,v}\underbrace{\mathbb{E}(|N(u,v)|^2)}_{S_\eta}$$

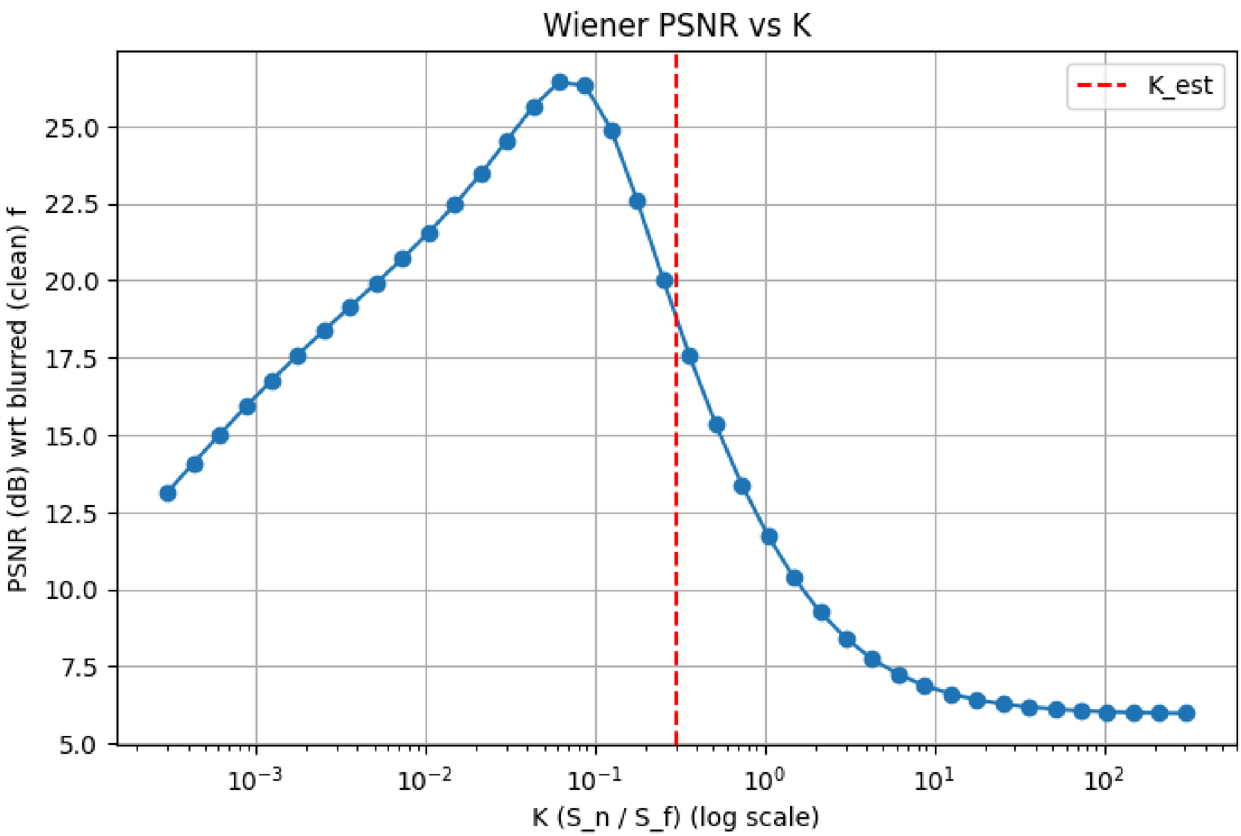
$$\therefore \frac{1}{MN}\sum_{u,v} S_\eta(u,v) = \sum_{m,n}\left[\sigma_\eta(m,n)\right]^2$$

If $\sigma_\eta(m,n)$ is constant, $S_\eta(u,v)$ is also constant $\Rightarrow$

$$\frac{1}{\cancel{MN}}\cancel{MN}\,S_\eta(u,v) = MN\,\sigma_\eta^2 \Rightarrow S_\eta(u,v) = MN\,\sigma_\eta^2$$

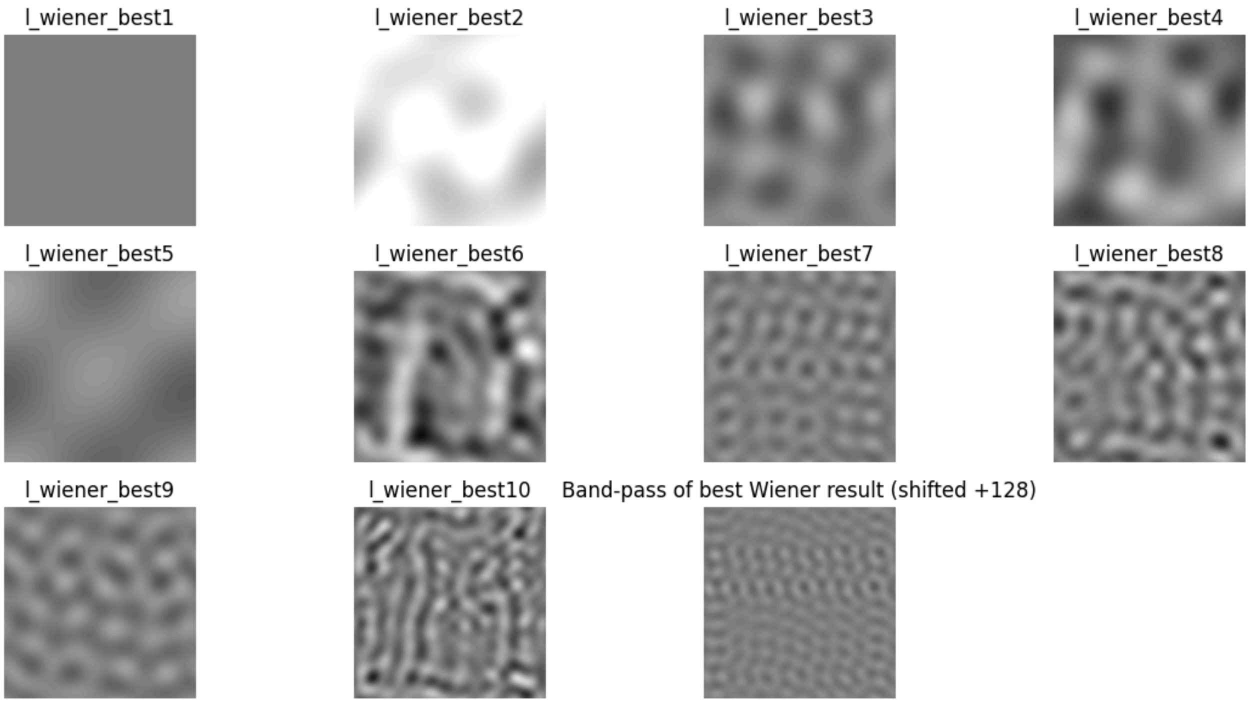
Similarly, $S_f = \sigma_f^2 MN$

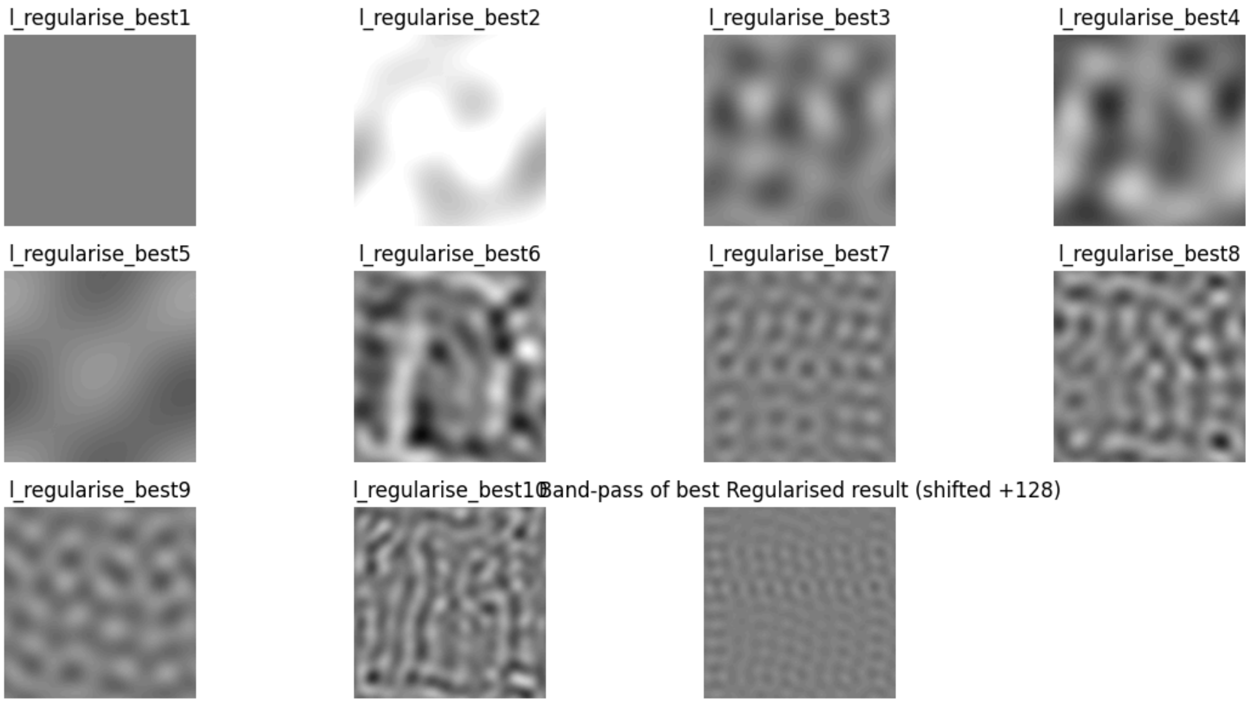
=> F^ = WG = F = $\frac{H^*G}{HH^* + \lambda\sigma_f^2/\sigma_n^2}$



Best K: 0.0615633 Best PSNR: 26.420075676552113

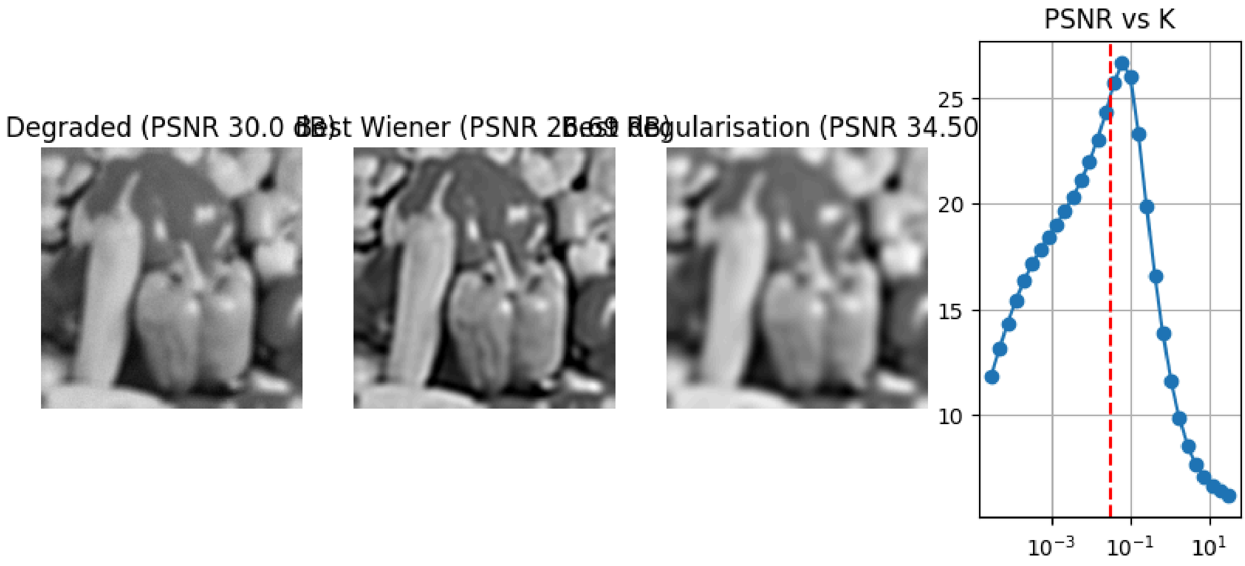
Wiener best (K=6.16e-02)





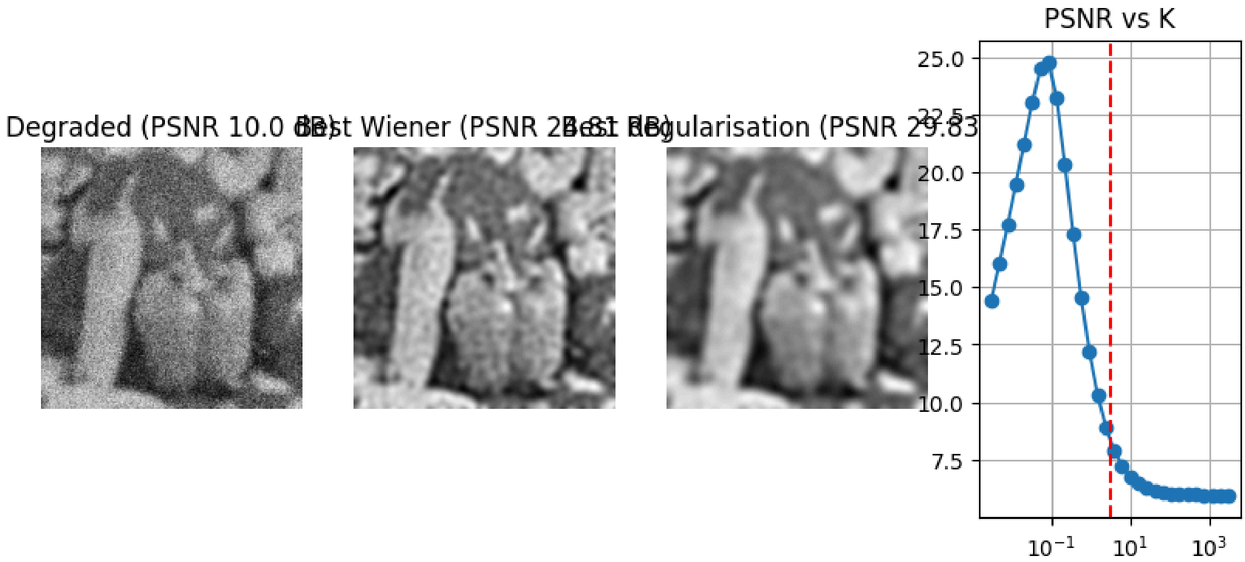
Target PSNR 30.0 dB: sigma_n2=65.2546, K_est=3.0389e-02

PSNR sweep best K=6.21e-02, best PSNR=26.688 dB



Target PSNR 10.0 dB: sigma_n2=6500.9453, K_est=3.0275e+00

PSNR sweep best K=8.50e-02, best PSNR=24.807 dB



f. Here,

$$R(\hat{f}) = \sum_{x,y}\left((\hat{f}(x+1,y) - \hat{f}(x))^2 + (\hat{f}(x,y+1) - \hat{f}(x,y))^2\right)$$

$$\& \text{ IF}\left(\hat{f}_{x+1,y} - \hat{f}_{x,y}\right) = \sum_{y=0}^{N-1}\sum_{x=0}^{M-1} \hat{f}(x+1,y)\, e^{-2\pi i\left(\frac{ux}{M}+\frac{vy}{N}\right)}$$

$$- F(u,v)$$

$$\sum_{y=0}\left\{\sum_{x=0}\left\{\hat{f}(x,y)\, e^{-2\pi i\left(\frac{ux}{M}+\frac{vy}{N}\right)}\right\} - \hat{f}(0,y)\, e^{-2\pi i \frac{vy}{N}}\right\}$$

$$- F(u,v)$$

Let $x' = x+1 \; : \; x = x'-1$

$$\Rightarrow \sum_{y=0}^{N-1}\sum_{x'=1}^{M-\cancel{1+1}} \hat{f}(x',y)\, e^{-2\pi i\left(\frac{u(x'-1)}{M}+\frac{vy}{N}\right)}$$

$$- \underbrace{\sum_{y=0}^{N}\sum_{x'=1}^{M} \hat{f}(x'-1,y)\, e^{-2\pi i\left(\frac{u(x'-1)}{M}+\frac{vy}{N}\right)}}_{F(u,v)} \Big)$$

$$= e^{2\pi i \frac{u}{M}} \sum_{y=0}^{N-1} \underbrace{\sum_{x'=1}^{M-1+1} \hat{f}(x',y)\, e^{-2\pi i\left(\frac{ux'}{M}+\frac{vy}{N}\right)}}_{\sum_{x'=0}^{M-1} \hat{f}(x',y)\, e^{-2\pi i\left(\frac{ux'}{M}+\frac{vy}{N}\right)} - \cancel{\hat{f}(0,y)\, e}^{-2\pi i \frac{vy}{N}} + \cancel{\hat{f}(M,y)}\, \underbrace{e^{-2\pi i}}_{1}{}^{-2\pi i \frac{vy}{N}}} - F(u,v)$$

By DFT periodicity,

$$\hat{f}(M, y) \equiv \hat{f}(0, y)$$

$$\Rightarrow \mathbb{F}(\hat{f}_{x+1,y} - \hat{f}_{x,y}) = (e^{2\pi i u/M} - 1)\hat{F}(u, v)$$

similarly,

$$\mathbb{F}(\hat{f}_{x,y+1} - \hat{f}_{x,y}) = (e^{2\pi i v/N} - 1)F(u, v)$$

By Plancherel's theorem

$$\Rightarrow R(\hat{f}) = \sum_{x,y} (\hat{f}_{x,y+1} - \hat{f}_{x,y})^2 + (\hat{f}_{x+1,y} - \hat{f}_{x,y})^2$$

$$= \frac{1}{MN} \sum_{u,v} |\hat{F}(u,v)|^2 \left(|e^{+2\pi i u/M} - 1|^2 + |e^{2\pi i v/N} - 1|^2 \right)$$

$$\cancel{e^{4\pi i}} \quad \underbrace{\left(\cos 2\pi i \frac{u}{M} - 1\right)}_{-2\sin^2 \pi i \frac{u}{M}} + \sin 2\pi i \frac{u}{M} j$$

$$= \frac{1}{MN} \sum_{u,v} |\hat{F}(u,v)|^2 \left(4\sin^2\left(\frac{\pi u}{M}\right) + 4\sin^2\left(\frac{\pi v}{N}\right) \right)$$

Since

$$\sum_{x,y} |\hat{f}(x+1,y) - \hat{f}(x,y)|^2 = \frac{1}{MN} \sum_{u,v} |\hat{F}(u,v)(e^{-j2\pi u/M} - 1)|^2$$

$$= \frac{1}{MN} \sum_{u,v} |\mathbb{F}\{\hat{f}(x+1,y) - \hat{f}(x,y)\}|^2$$

We have to minimise:

$$E(\hat{F}) = \sum_{u,v} \underbrace{|G - H\hat{F}|^2}_{(G-H\hat{F})(G^* - \hat{F}^* H^*)} + \lambda R$$

λ absorb MN in λ

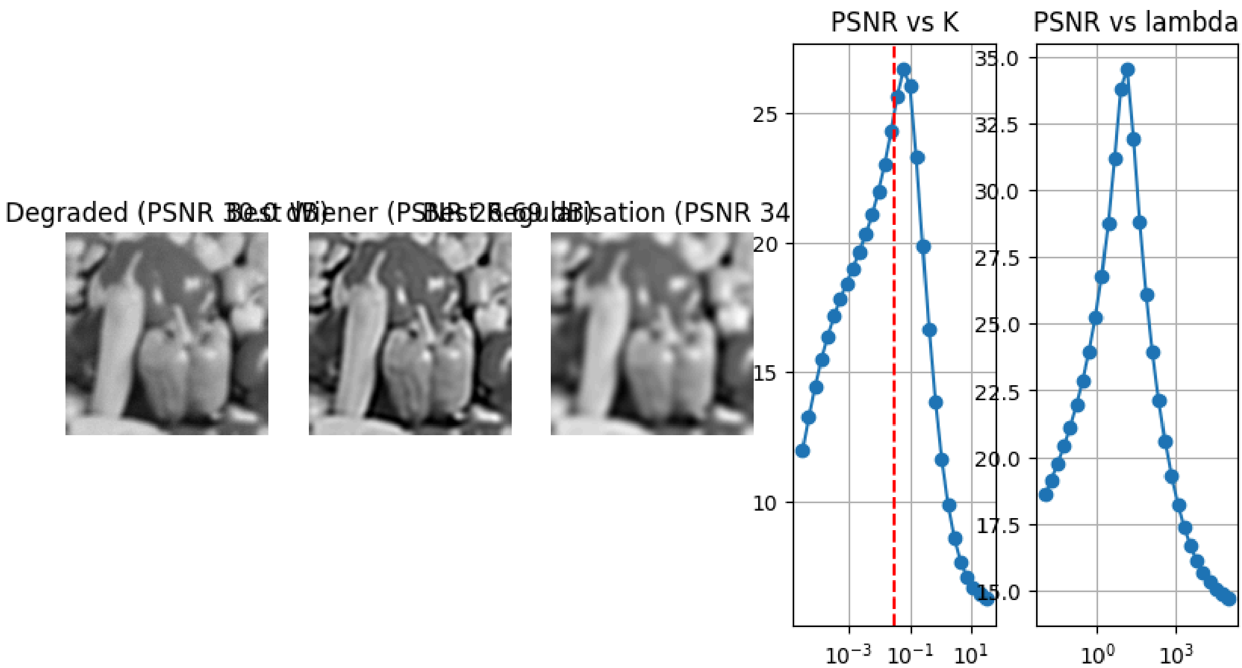
$$= \sum_{u,v} |G|^2 + |H|^2 \underbrace{|\hat{F}|^2}_{\hat{F}\cdot\hat{F}^*} - H\hat{F}G^* - \hat{F}^* G H^* + \lambda \underbrace{|\hat{F}|^2}_{\hat{F}\cdot\hat{F}^*} (D_x^2 + D_y^2)$$

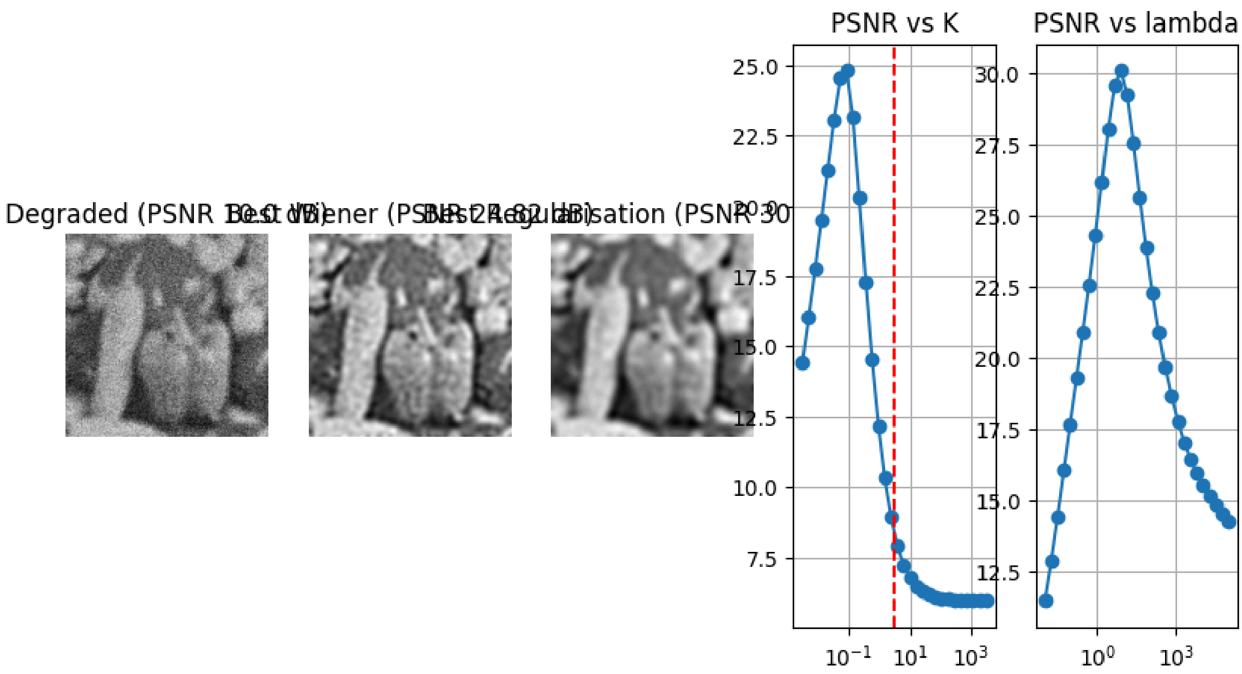
For each separate terms:

To min: $\frac{\partial E}{\partial \hat{F}^*} = 0 \Rightarrow |H|^2 \hat{F} - \cancel{HG^*} H^* G + \lambda \hat{F}(D_x^2 + D_y^2) = 0$

$$\Rightarrow \hat{F} = \frac{H^* G}{|H|^2 + \lambda(D_x^2 + D_y^2)}$$

Therefore, restored F = $\frac{H^*G}{HH^* + \lambda G}$ where G = $4(\sin^2(\pi u/M) + \sin^2(\pi v/N))$





Best lambda: 14.251026703029993 Best PSNR: 34.16081583524833

Regularised best (lambda=1.43e+01)