Image Processing

INT3404 1/ INT3404E 21

Giảng viên: TS. Nguyễn Thị Ngọc Diệp

Email: ngocdiep@vnu.edu.vn

1

Schedule

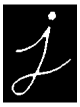
Tuần	Nội dung	Yêu cầu đối với sinh viên (ngoài việc đọc tài liệu tham khảo)
1	Giới thiệu môn học	Cài đặt môi trường: Python 3, OpenCV 3, Numpy, Jupyter Notebook
2	Ảnh số (Digital image) – Phép toán điểm (Point operations) Làm quen với OpenCV + Python	
3	Điều chỉnh độ tương phản (Contrast adjust)– Ghép ảnh (Combining images)	Làm bài tập 1: điều chỉnh gamma tìm contrast hợp lý
4	Histogram - Histogram equalization	Thực hành ở nhà
5	Phép lọc trong không gian điểm ảnh (linear processing filtering)	Thực hành ở nhà
6	Phép lọc trong không gian điểm ảnh cont. (linear processing filtering) Thực hành: Ứng dụng của histogram; Tìm ảnh mẫu (Template matching)	Bài tập mid-term
7	Trích rút đặc trưng của ảnh Cạnh (Edge) và đường (Line) và texture	Thực hành ở nhà
8	Các phép biến đổi hình thái (Morphological operations)	Làm bài tập 2: tìm barcode
9	Chuyển đổi không gian – Miền tần số – Phép lọc trên miền tần số Thông báo liên quan đồ án môn học	Đăng ký thực hiện đồ án môn học
10	Xử lý ảnh màu (Color digital image)	Làm bài tập 3: Chuyển đổi mô hình màu và thực hiện phân vùng
11	Các phép biến đổi hình học (Geometric transformations)	Thực hành ở nhà
12	Nhiễu – Mô hình nhiễu – Khôi phục ảnh (Noise and restoration)	Thực hành ở nhà
13	Nén ảnh (Compression)	Thực hành ở nhà
14	Hướng dẫn thực hiện đồ án môn học	Trình bày đồ án môn học
15	Hướng dẫn thực hiện đồ án môn học Tổng kết cuối kỳ	Trình bày đồ án môn học

2

Recall week 8: Morphological operations

translation $(B)_z = \{c \mid c = b + z, \text{ for } b \in B\}$

erosion $A \ominus B = \{z \mid (B)_z \subseteq A\}$



$\hat{B} = \{w \mid w = -b, \text{ for } b \in B\}$ reflection

$A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \varnothing\}$ **dilation**

Opening $A \circ B = (A \ominus B) \oplus B$

$$= \bigcup\{(B)_z \mid (B)_z \subseteq A\}$$

Closing $A \bullet B = (A \oplus B) \ominus B$

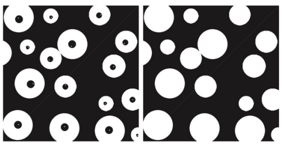
$$= \left[\bigcup\{(B)_z \mid (B)_z \cap A = \varnothing\}\right]^c$$

3

Some applications of morphology

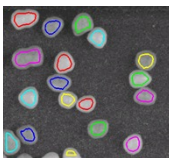
Background extraction

Hole filling



Object detection

Connected components



4

Final projects

- Work based on groups,
  - One group: 5-6 members
- Problem – each group defines their own task, NO overlapping
  - Early birds may win ☺
  - The registration form will start soon!
- YES: OpenCV [+ ML]
- NO: deep learning, image filtering/effect, sudoku, road segmentation, license plate, meter, MNIST

5

Schedule for Final projects

- Presentation time: week 14, 15
- Submission: source code, presentation slides
- Evaluation criteria:
  - Idea: 20%
  - Implementation: 30%
  - Evaluation method: 20%
  - Presentation: 30%

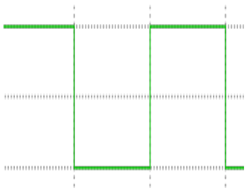
6

Week 9

Fourier transform

7

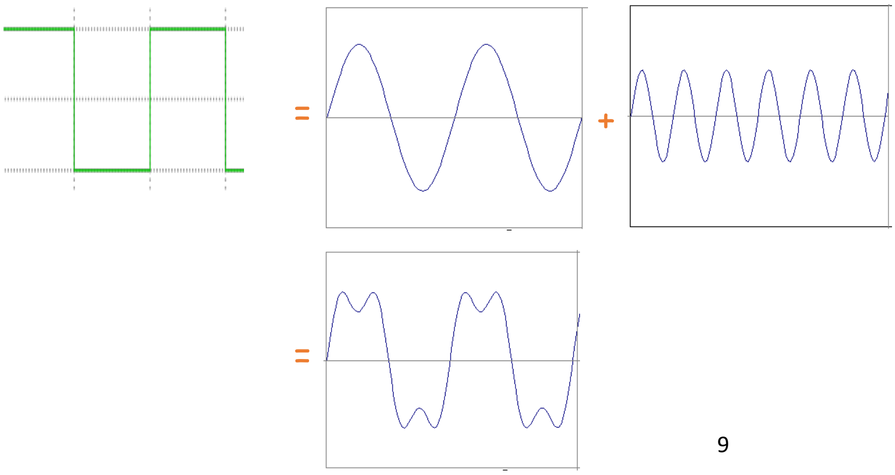
Frequency Spectra



4/5/23 8

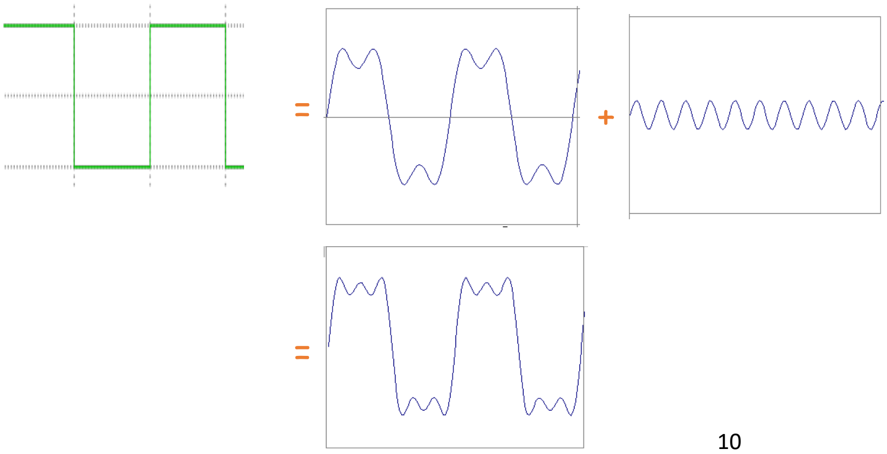
8

Frequency Spectra



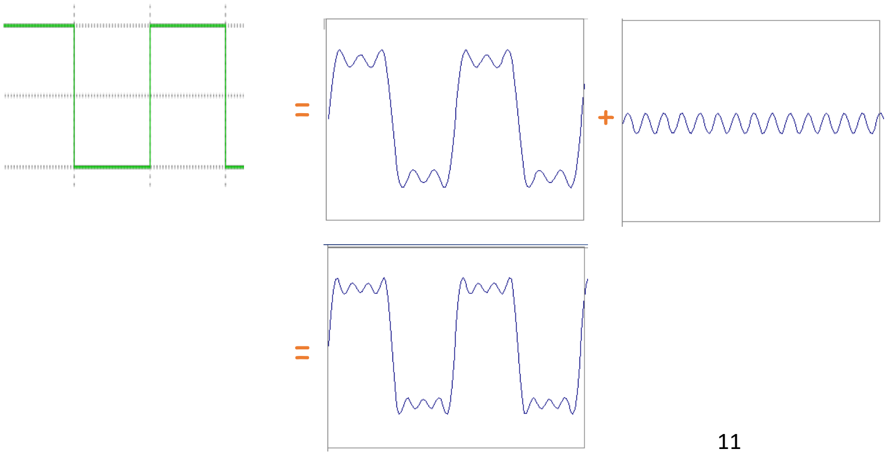
9

Frequency Spectra



10

Frequency Spectra

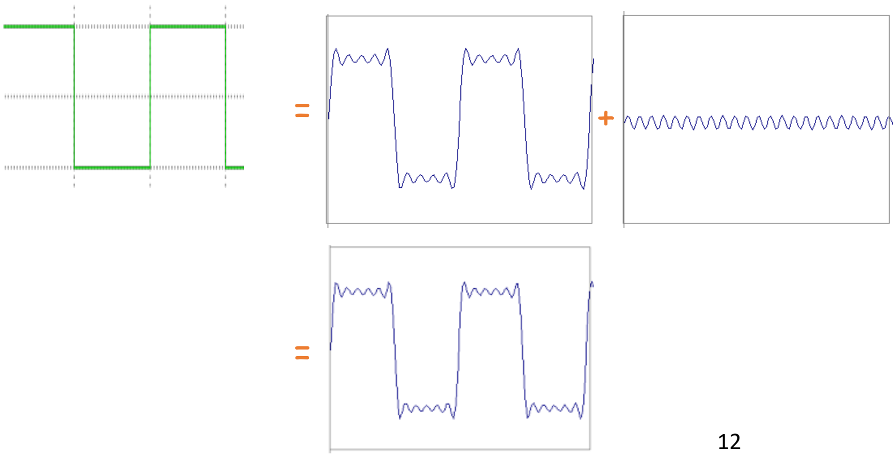


4/5/23

11

11

Frequency Spectra

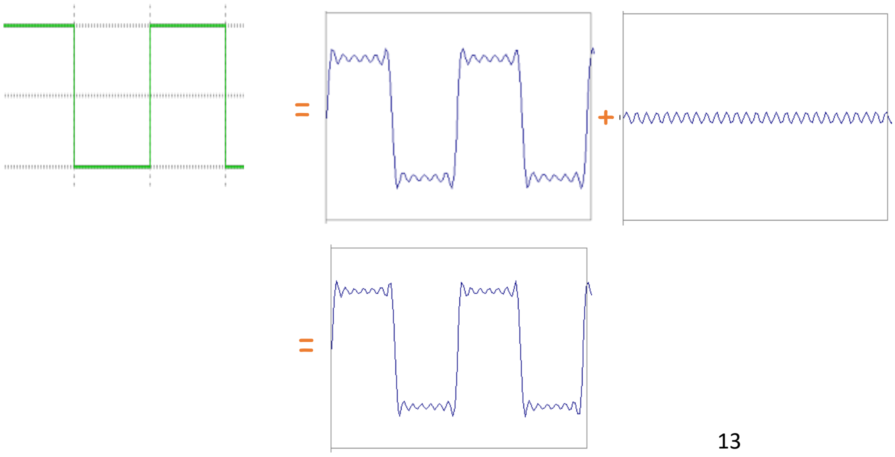


4/5/23

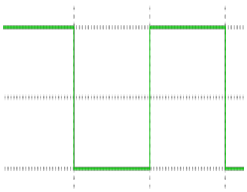
12

12

Frequency Spectra

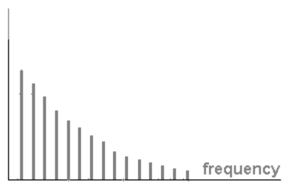


Frequency Spectra

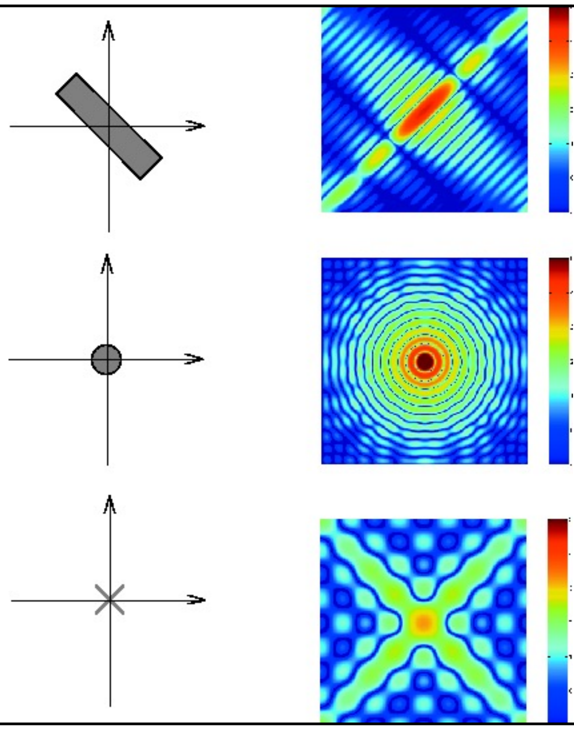


=

$$\boxed{A\sum_{k=1}^{\infty}\frac{1}{k}\sin(2\pi kt)}$$



Fourier transform of 2D signal



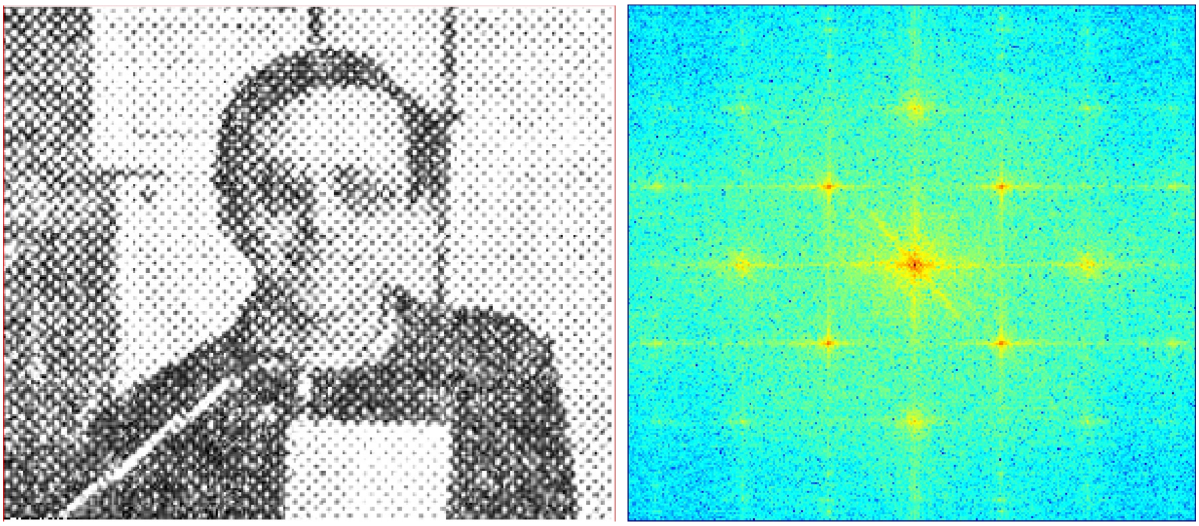
15

Why do we convert images to spectrum domain?

- For exposing image features not visible in spatial domain, e.g. periodic interferences
- For achieving more compact image representation (coding), e.g. JPEG, JPEG2000
- For designing digital filters
- For fast processing of images, e.g. digital filtering of images in spectrum domain

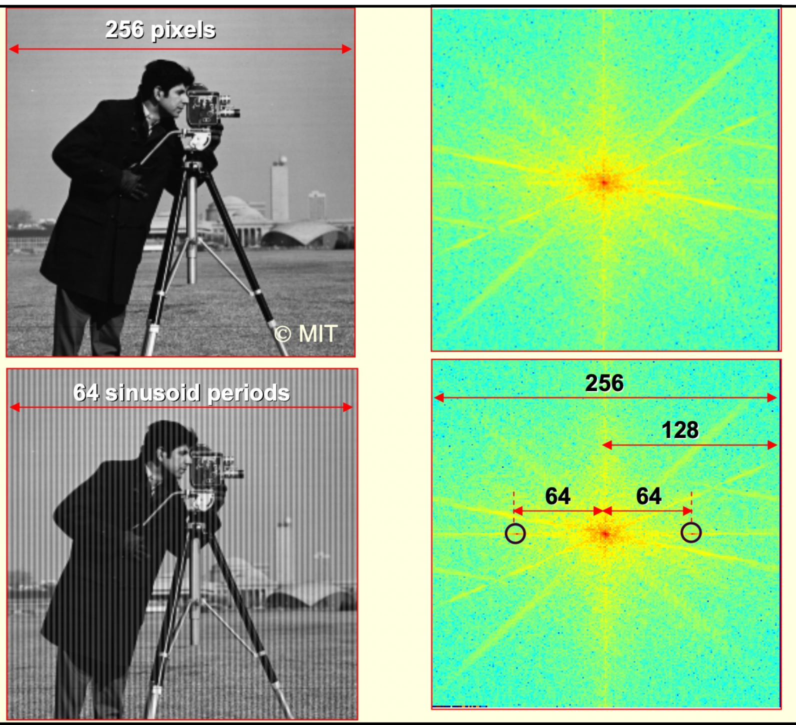
16

Periodic features of images



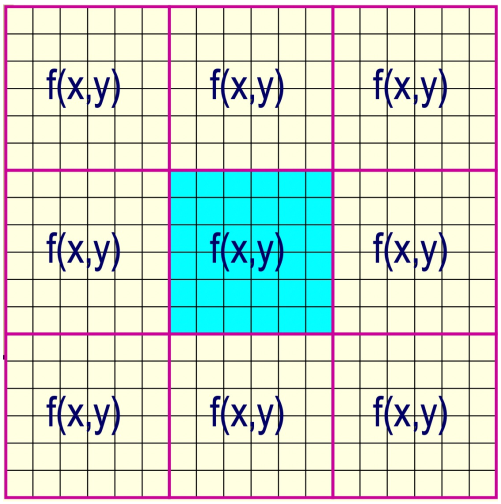
17

Periodic features



18

Fourier transform of images



It's assumed the transformed image is a periodic function of period (N, N)

19

1D Discrete Fourier transform

- Complex series: X_k, x_n

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N}kn} \qquad k = 0,\dots,N-1$$

$$x_n = \frac{1}{N}\sum_{k=0}^{N-1} X_k e^{\frac{2\pi i}{N}kn} \qquad n = 0,\dots,N-1$$

4/5/23 20

20

2D Discrete Fourier transform

- Discrete domain:

$$F[k,l] = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f[m,n]e^{-j2\pi\left(\frac{k}{M}m+\frac{l}{N}n\right)}$$

$$e^{ix} = \cos x + i\sin x$$

Inverse

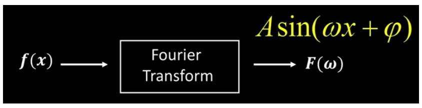
$$f[m,n] = \sum_{k=0}^{M-1}\sum_{l=0}^{N-1} F[k,l]e^{j2\pi\left(\frac{k}{M}m+\frac{l}{N}n\right)}$$

21

21

2D Fourier transform

- f – image
- F – frequency image.



For every ω from 0 to ∞ (actually $-\infty$ to ∞), $F(\omega)$ holds the amplitude A and phase f of the corresponding sinusoid

$$F(\omega) = R(\omega) + iI(\omega) \qquad A = \pm\sqrt{R(\omega)^2 + I(\omega)^2} \qquad \varphi = \tan^{-1}\frac{I(\omega)}{R(\omega)}$$

Which one is more informative, magnitude or phase?

22

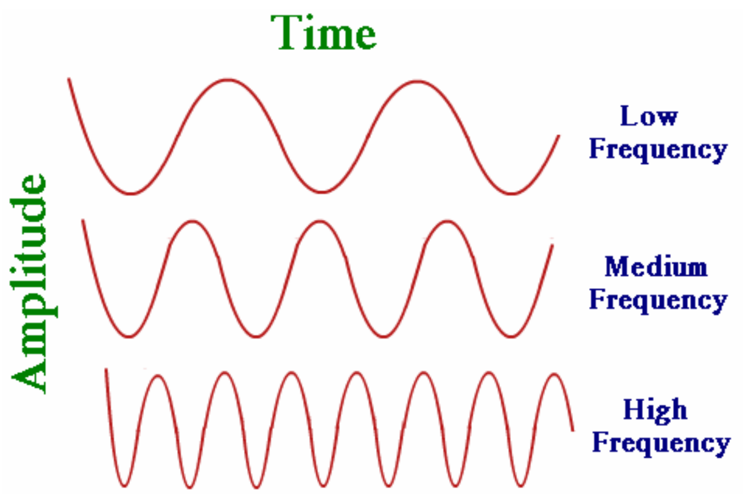
22

Computing DFT

- Slow version: Matrix multiplication
- Fast version: Fast Fourier Transform
  - Understanding the FFT algorithm (Cooley-Tukey algorithm)
  - https://jakevdp.github.io/blog/2013/08/28/understanding-the-fft/

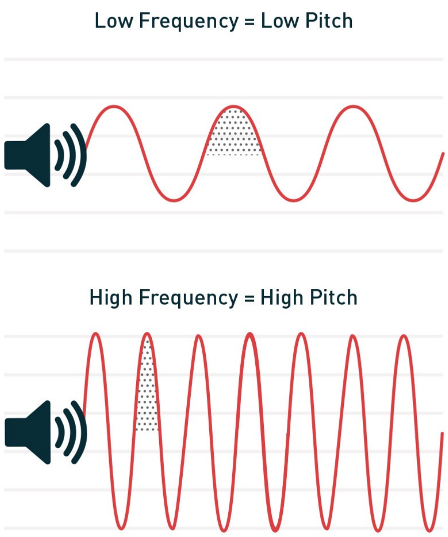
23

Low frequency, High frequency?



24

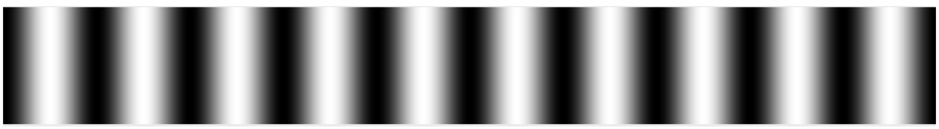
Low frequency, High frequency?



25

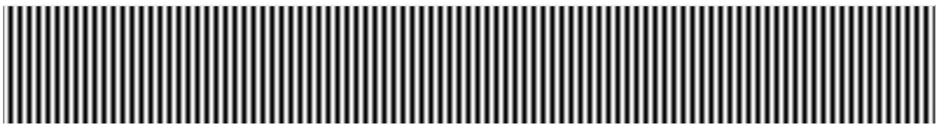
Low frequency, High frequency?

800px X 100px grayscale image
Generated using $I(x) = \sin(2\pi f x)$
where f = 10repetitions/800px = 0.0125 repetitions/px



Smooth

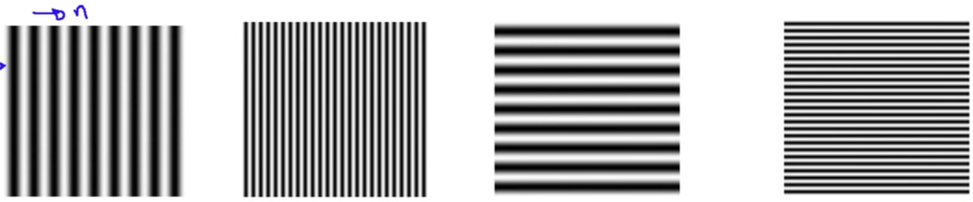
increase the frequency by a factor of 10, so that n = 100 repetitions
f = 100/800 = 1/8 = 0.125 repetitions/px



Finer details, many edge

26

Low frequency – High frequency

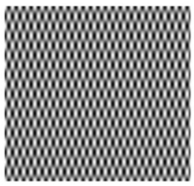


$\cos\left(\frac{18\pi}{256}n\right)$ $u=0,\ v=\frac{18\pi}{256}$

$\cos\left(\frac{50\pi}{256}n\right)$ $u=0,\ v=\frac{50\pi}{256}$

$\cos\left(\frac{18\pi}{256}m\right)$ $u=\frac{18\pi}{256},\ v=0$

$\cos\left(\frac{50\pi}{256}m\right)$ $u=\frac{50\pi}{256},\ v=0$

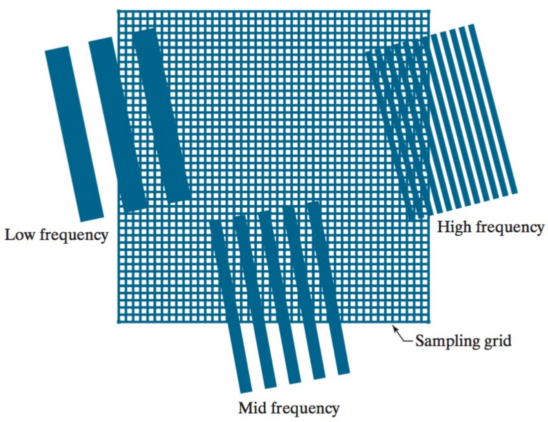


$\cos\left(\frac{50\pi}{256}n\right)\cos\left(\frac{18\pi}{256}m\right)$

$u=\frac{18\pi}{256},\ v=\frac{50\pi}{256}$

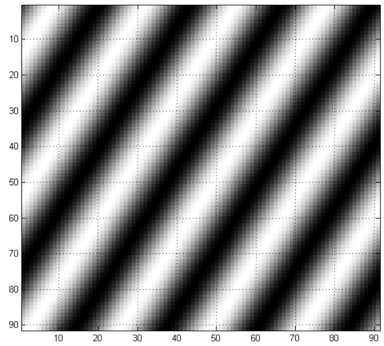
27

Low frequency, High frequency?

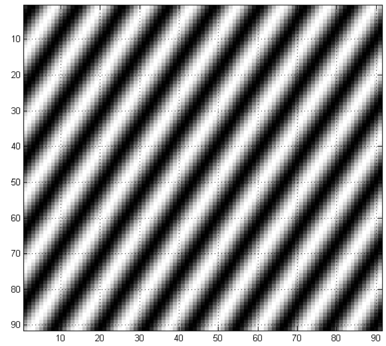


28

Low frequency, High frequency?



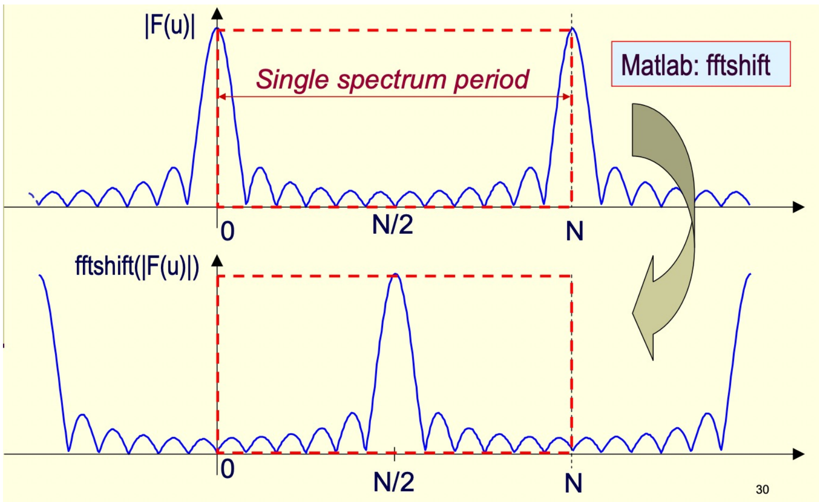
Low frequency



High frequency

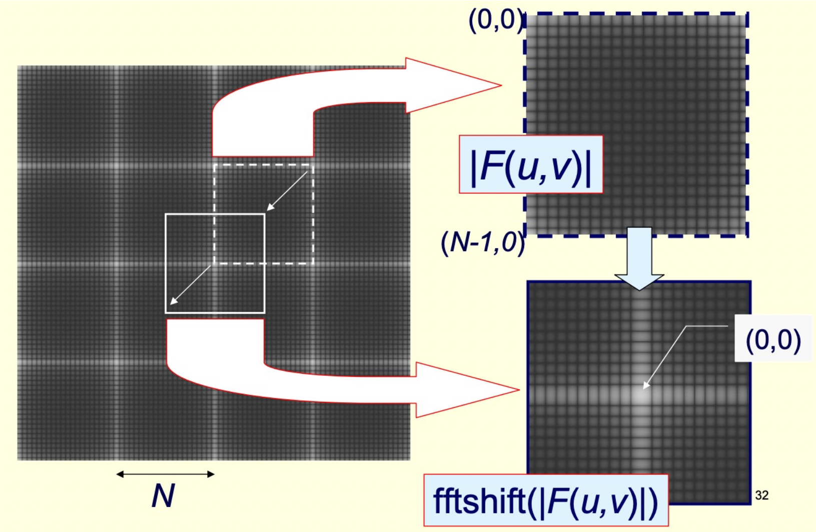
29

FFTshift



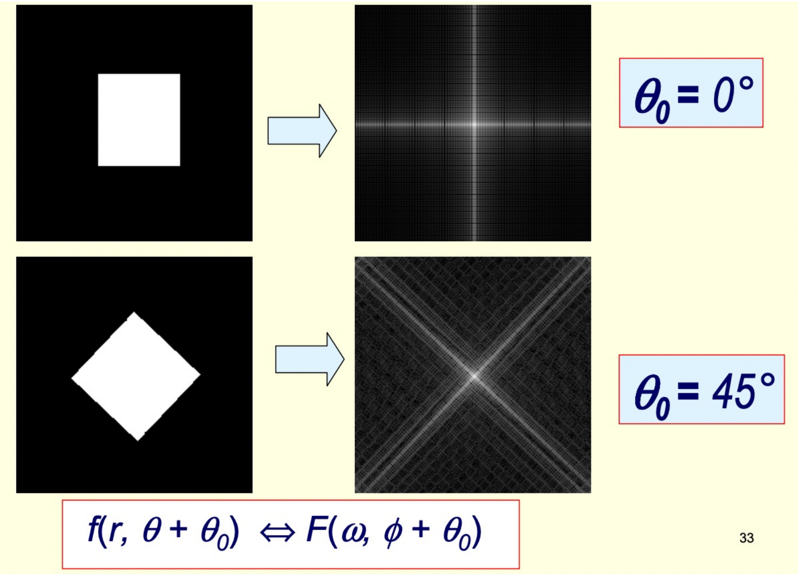
30

Translation in spectral domain



31

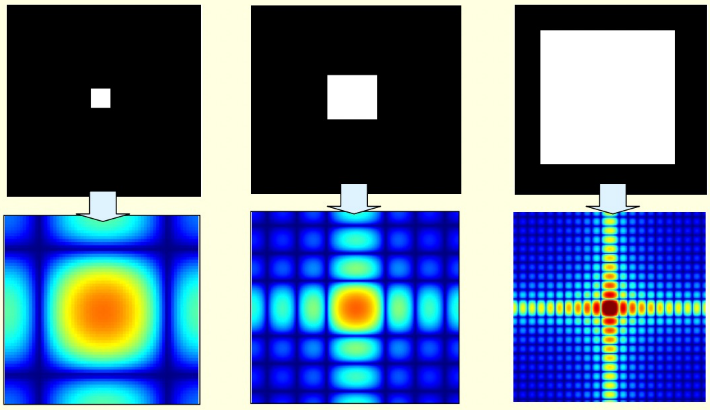
Rotation in spectral domain



32

Scaling

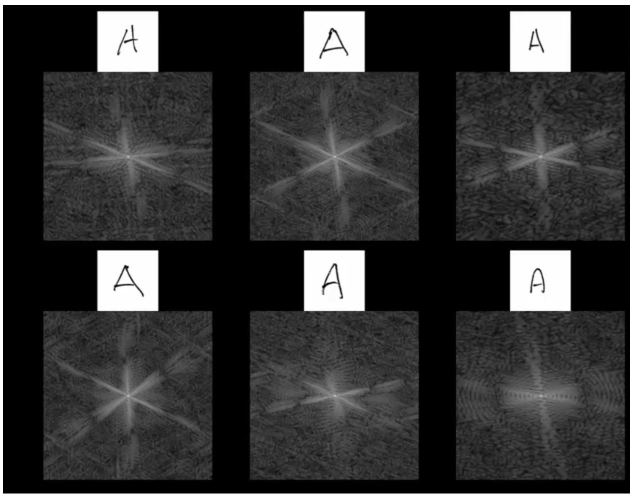
$$\mathfrak{F}\{f(ax,by)\} = |ab|^{-1} F(u/a,\ v/b) \qquad a,b \in R$$



35

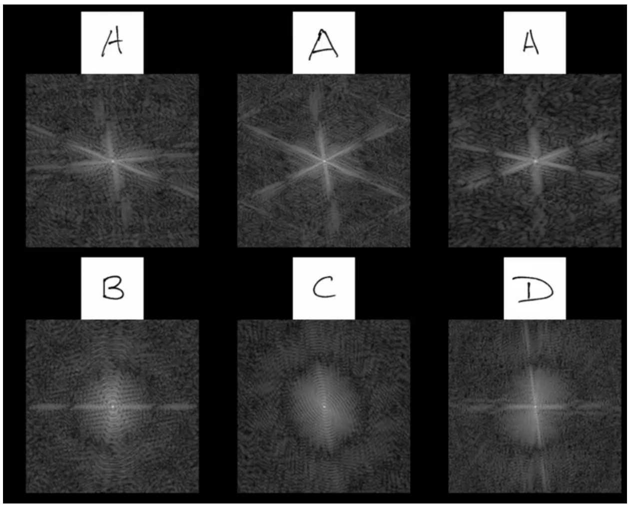
33

Fourier transform



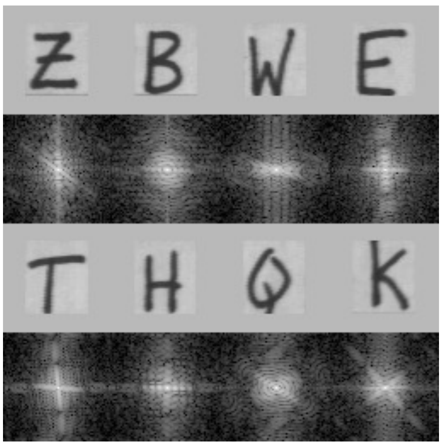
34

Fourier transform



35

Fourier transform



FT of each character symbol is different from others, especially at the low frequencies.

36

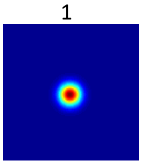
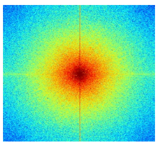
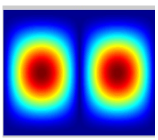
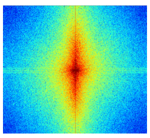
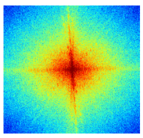
DFT demo

- <view code>

37

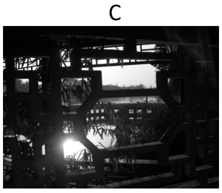
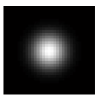
Practice question

Match the spatial domain image to the Fourier magnitude image

1

2

3

4

5


A

B

C

D

E


38

Properties of Fourier Transform

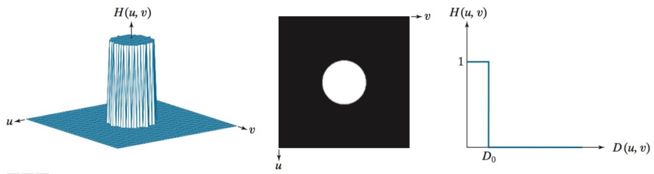
	Spatial Domain *(x)*	Frequency Domain *(u)*
Linearity	$c_1 f(x) + c_2 g(x)$	$c_1 F(u) + c_2 G(u)$
Convolution	$f(x) * g(x)$	$F(u)G(u)$
Scaling	$f(ax)$	$\frac{1}{\lvert a \rvert} F\left(\frac{u}{a}\right)$
Differentiation	$\frac{d^n f(x)}{dx^n}$	$(i2\pi u)^n F(u)$

39

Filtering in Frequency domain

40

Ideal Lowpass filter



a b c

FIGURE 4.39 (a) Perspective plot of an ideal lowpass-filter transfer function. (b) Function displayed as an image. (c) Radial cross section.

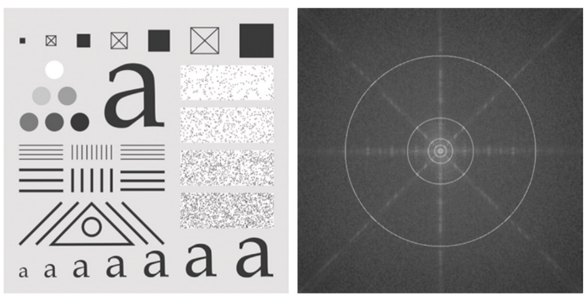
$$H(u,v) = \begin{cases} 1 & \text{if } D(u,v) \le D_0 \\ 0 & \text{if } D(u,v) > D_0 \end{cases}$$

$$D(u,v) = \left[(u - P/2)^2 + (v - Q/2)^2\right]^{1/2}$$

- Cut off high frequencies specified by a distance d_0.
  - Cannot be realized by electronic component → not practical

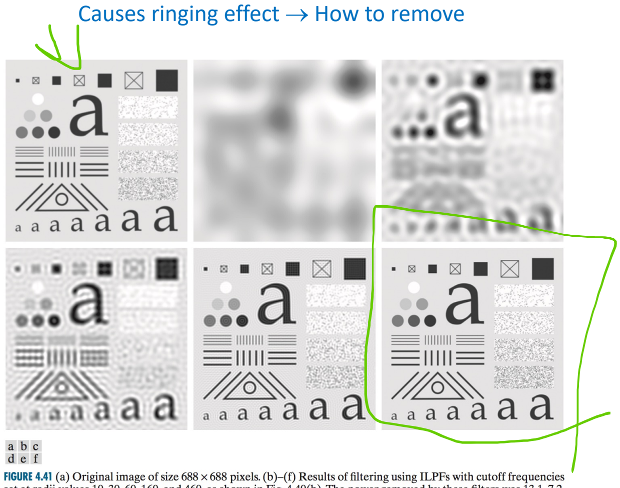
Ideal Lowpass filter

Causes ringing effect → How to remove



a b

FIGURE 4.40 (a) Test pattern of size 688 × 688 pixels, and (b) its spectrum. The spectrum is double the image size as a result of padding, but is shown half size to fit. The circles have radii of 10, 30, 60, 160, and 460 pixels with respect to the full-size spectrum. The radii enclose 86.9, 92.8, 95.1, 97.6, and 99.4% of the padded image power, respectively.



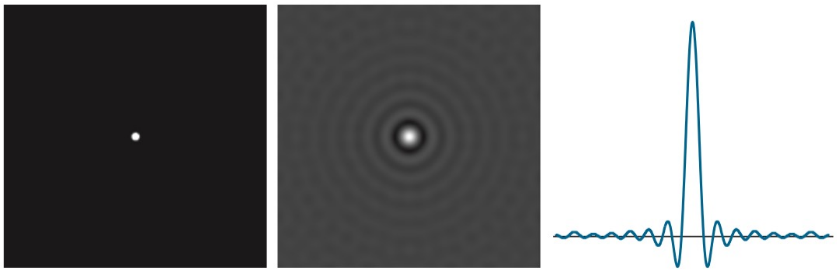
a b c
d e f

FIGURE 4.41 (a) Original image of size 688 × 688 pixels. (b)–(f) Results of filtering using ILPFs with cutoff frequencies set at radii values 10, 30, 60, 160, and 460, as shown in Fig. 4.40(b). The power removed by these filters was 13.1, 7.2, 4.9, 2.4, and 0.6% of the total, respectively. We used mirror padding to avoid the black borders characteristic of zero padding, as illustrated in Fig. 4.31(c).

Ideal Lowpass filter

a b c

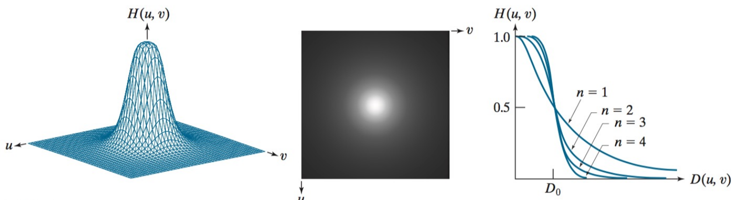
FIGURE 4.42
(a) Frequency domain ILPF transfer function.
(b) Corresponding spatial domain kernel function.
(c) Intensity profile of a horizontal line through the center of (b).



4/5/23 43

43

Butterworth Filter



a b c

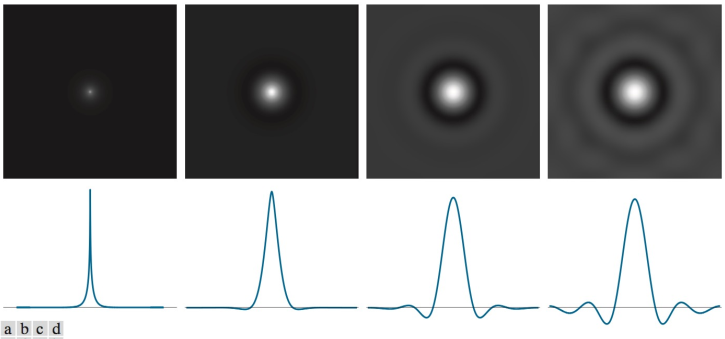
FIGURE 4.45 (a) Perspective plot of a Butterworth lowpass-filter transfer function. (b) Function displayed as an image. (c) Radial cross sections of BLPFs of orders 1 through 4.

$$H(u,v) = \frac{1}{1 + \left[D(u,v)/D_0\right]^{2n}}$$

4/5/23 44

44

Butterworth Filter



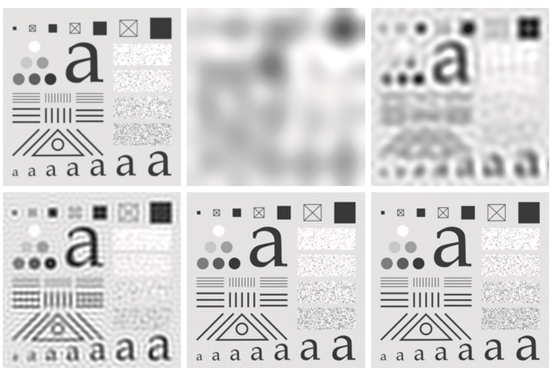
a b c d
e f g h

FIGURE 4.47 (a)–(d) Spatial representations (i.e., spatial kernels) corresponding to BLPF transfer functions of size 1000 × 1000 pixels, cut-off frequency of 5, and order 1, 2, 5, and 20, respectively. (e)–(h) Corresponding intensity profiles through the center of the filter functions.

4/5/23 45

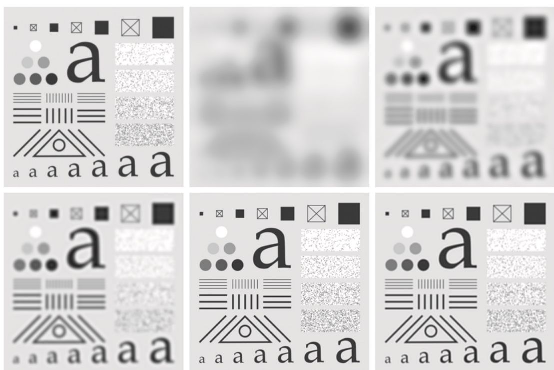
45

Ideal lowpass filter vs Butterworth Filter



a b c
d e f

FIGURE 4.41 (a) Original image of size 688 × 688 pixels. (b)–(f) Results of filtering using ILPFs with cutoff frequencies set at radii values 10, 30, 60, 160, and 460, as shown in Fig. 4.40(b). The power removed by these filters was 13.1, 7.2, 4.9, 2.4, and 0.6% of the total, respectively. We used mirror padding to avoid the black borders characteristic of zero padding, as illustrated in Fig. 4.31(c).



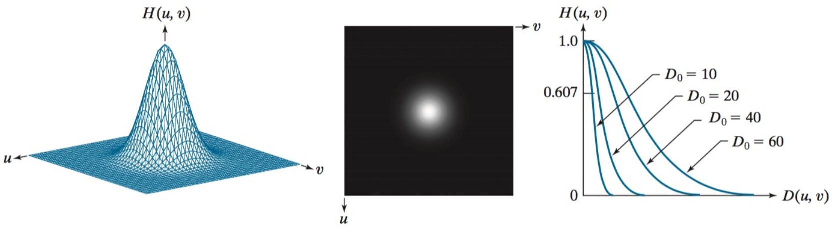
a b c
d e f

FIGURE 4.46 (a) Original image of size 688 × 688 pixels. (b)–(f) Results of filtering using BLPFs with cutoff frequencies at the radii shown in Fig. 4.40 and $n = 2.25$. Compare with Figs. 4.41 and 4.44. We used mirror padding to avoid the black borders characteristic of zero padding.

4/5/23 46

46

Gaussian Filter

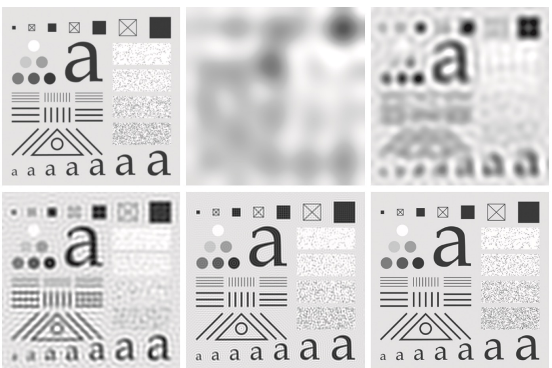


a b c

FIGURE 4.43 (a) Perspective plot of a GLPF transfer function. (b) Function displayed as an image. (c) Radial cross sections for various values of D_0.

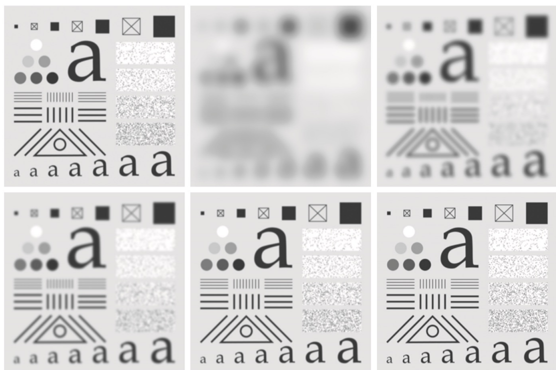
$$H(u,v) = e^{-D^2(u,v)/2D_0^2}$$

Ideal lowpass filter vs Gaussian Lowpass Filter



a b c
d e f

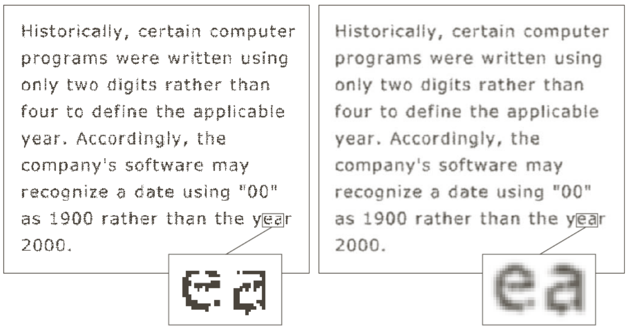
FIGURE 4.41 (a) Original image of size 688 × 688 pixels. (b)–(f) Results of filtering using ILPFs with cutoff frequencies set at radii values 10, 30, 60, 160, and 460, as shown in Fig. 4.40(b). The power removed by these filters was 13.1, 7.2, 4.9, 2.4, and 0.6% of the total, respectively. We used mirror padding to avoid the black borders characteristic of zero padding, as illustrated in Fig. 4.31(c).



a b c
d e f

FIGURE 4.44 (a) Original image of size 688 × 688 pixels. (b)–(f) Results of filtering using GLPFs with cutoff frequencies at the radii shown in Fig. 4.40. Compare with Fig. 4.41. We used mirror padding to avoid the black borders characteristic of zero padding.

Gaussian Lowpass Filter



a b

FIGURE 4.49 (a) Sample text of low resolution (note broken characters in magnified view). (b) Result of filtering with a GLPF (broken character segments were joined).

49

Gaussian Lowpass filter example

a b c

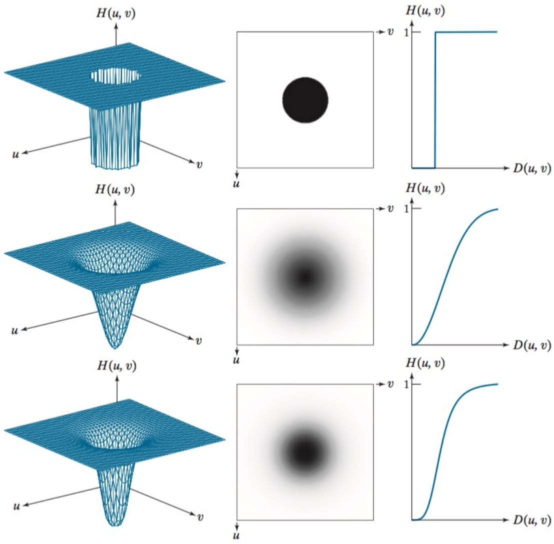
FIGURE 4.49 (a) Original 785×732 image. (b) Result of filtering using a GLPF with $D_0 = 150$. (c) Result of filtering using a GLPF with $D_0 = 130$. Note the reduction in fine skin lines in the magnified sections in (b) and (c).

50

Highpass filter

a b c
d e f
g h i

FIGURE 4.51 Top row: Perspective plot, image, and, radial cross section of an IHPF transfer function. Middle and bottom rows: The same sequence for GHPF and BHPF transfer functions. (The thin image borders were added for clarity. They are not part of the data.)



Ideal

$$H(u,v) = \begin{cases} 0 & \text{if } D(u,v) \le D_0 \\ 1 & \text{if } D(u,v) > D_0 \end{cases}$$

Gaussian

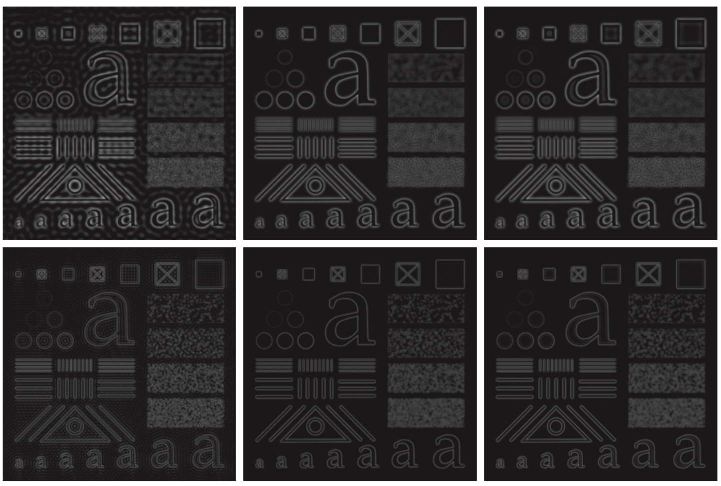
$$H(u,v) = 1 - e^{-D^2(u,v)/2D_0^2}$$

Butterworth

$$H(u,v) = \frac{1}{1 + \left[D_0/D(u,v)\right]^{2n}}$$

51

Highpass filter



a b c
d e f

FIGURE 4.53 Top row: The image from Fig. 4.40(a) filtered with IHPF, GHPF, and BHPF transfer functions using $D_0 = 60$ in all cases ($n = 2$ for the BHPF). Second row: Same sequence, but using $D_0 = 160$.

52

Reference

- http://mstrzel.eletel.p.lodz.pl/mstrzel/pattern_rec/fft_ang.pdf

53