Image Processing
INT3404 20
Week 7:
Filtering in the frequency domain

Lecturer: Nguyen Thi Ngoc Diep, Ph.D.
Email: ngocdiep@vnu.edu.vn
Slide & code: https://github.com/chupibk/INT3404_20

1

Schedule

Week	Content	Homework
1	Introduction	Set up environments: Python 3, OpenCV 3, Numpy, Jupyter Notebook
2	Digital image – Point operations Contrast adjust – Combining images	HW1: adjust gamma to find the best contrast
3	Histogram - Histogram equalization – Histogram-based image classification	Self-study
4	Spatial filtering - Template matching	Self-study
5	Feature extraction Edge, Line, and Texture	Self-study
6	Morphological operations	HW2: Barcode detection → Require submission as mid-term test
7	Filtering in the Frequency domain Announcement of Final project topics	Final project registration
8	Color image processing	HW3: Conversion between color spaces, color image segmentation
9	Geometric transformations	Self-study
10	Noise and restoration	Self-study
11	Compression	Self-study
12	Final project presentation	Self-study
13	Final project presentation Class summarization	Self-study

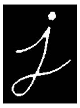
2

2

Recall week 6: 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 $A \bullet B = (A \oplus B) \ominus B$

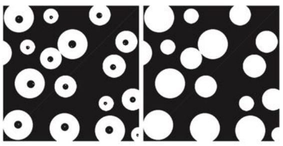
$= \left[\bigcup\{(B)_z \mid (B)_z \cap A = \varnothing\}\right]^c$

3

Some applications of morphology

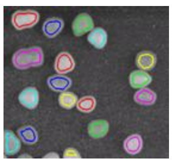
Background extraction

Hole filling



Object detection

Connected components



4

Final projects

- Work based on groups,
- One group: 4-5 members
- Problem set: (each group chooses one problem)
  - Create filtering effects (>=15)
  - Correspondence problem
  - Lane detection

5

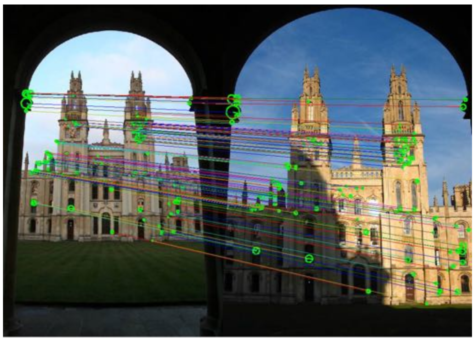
Filtering effect

- Examples: https://alvinalexander.com/design/gimp-catalog-filters-effects-examples-cheat-sheet/

6

Correspondence problem

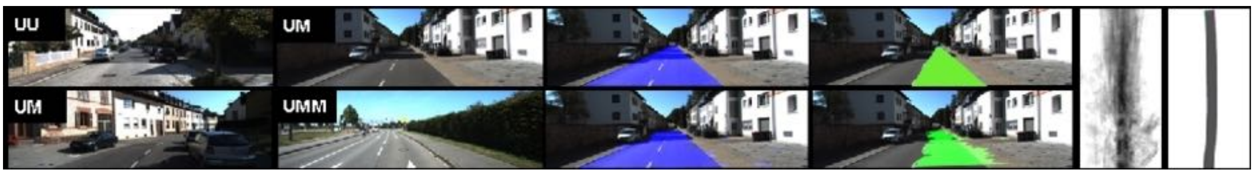
- Find corresponding points in images
  - For object detection
  - For creating panorama image
  - For combining images



7

Lane detection

- Dataset: http://www.cvlibs.net/datasets/kitti/eval_road.php



- Dataset: TuSimple

https://github.com/TuSimple/tusimple-benchmark/tree/master/doc/lane_detection

8

Schedule for Final projects

- Presentation time: week 12, 13
- Submission: source code, presentation slides

9

Week 7

Fourier transform

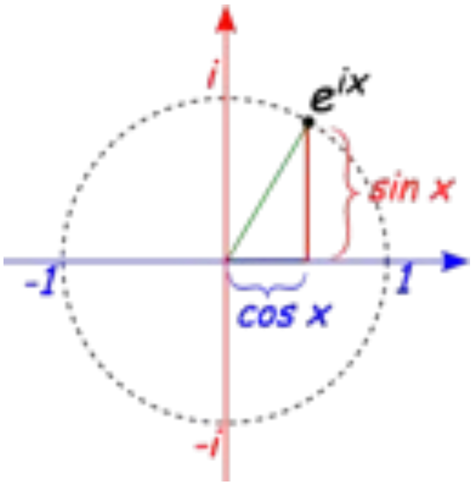
10

A path to understand Fourier transform

e^{ix} = cos x + i sin x	Euler's formula → ways to move in a circle
$e^{-2\pi i t}$	Time component
$e^{-2\pi i f t}$	Cycling frequency
$g(t)e^{-2\pi i f t}$	With the signal
$\boxed{\hat{g}(f) = \frac{1}{N}\sum_{k=1}^{N} g(t_k)e^{-2\pi i f t_k}}$	Defining the energy of the signal at a particular frequency

11

Euler's formula



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

https://www.mathsisfun.com/algebra/eulers-formula.html

12

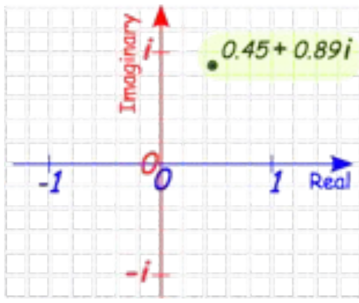
Example: when x = 1.1

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

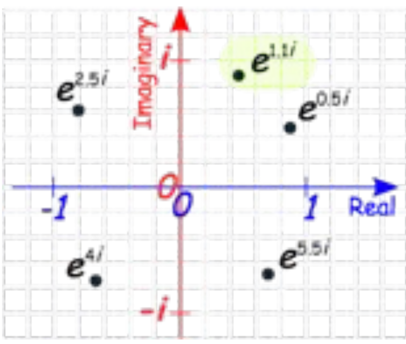
$e^{1.1i} = \cos 1.1 + i \sin 1.1$

$e^{1.1i} = 0.45 + 0.89\,i$ (to 2 decimals)

Note: we are using radians, not degrees.



13



14

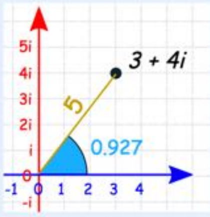
With radius of **r**

Example: the number **3 + 4i**

To turn **3 + 4i** into re^{ix} form we do a Cartesian to Polar conversion:

- $r = \sqrt{(3^2 + 4^2)} = \sqrt{(9+16)} = \sqrt{25} = \mathbf{5}$
- $x = \tan^{-1}(4/3) = \mathbf{0.927}$ (to 3 decimals)

So **3 + 4i** can also be $\mathbf{5e^{0.927\,i}}$

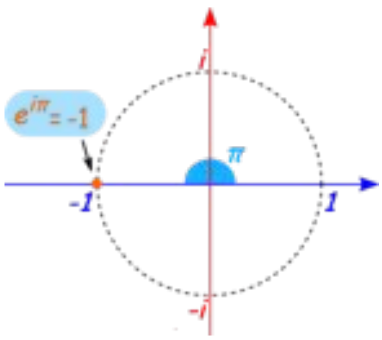


15

$e^{i\pi} = \cos\pi + i\sin\pi$

$e^{i\pi} = -1 + i \times 0$ (because $\cos\pi = -1$ and $\sin\pi = 0$)

$e^{i\pi} = -1$



16

Adding components

https://sites.northwestern.edu/elannesscohn/2019/07/30/developing-an-intuition-for-fourier-transforms/

17

Fourier transform

$$\hat{g}(f) = \frac{1}{N}\sum_{k=1}^{N} g(t_k)e^{-2\pi i f t_k}$$

To find the energy at a particular frequency, the signal is wrapped around a circle at the particular frequency and the points along the path are averaged.

18

Discrete Fourier transform

- 1D DFT

$$F_k = \sum_{n=0}^{N-1} f_n \cdot e^{-\frac{i2\pi}{N}kn}$$

- 2D DFT

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

Similarly for Inverse DFT

$$f_n = \frac{1}{N}\sum_{n=0}^{N-1} F_k \cdot e^{\frac{i2\pi}{N}kn}$$

19

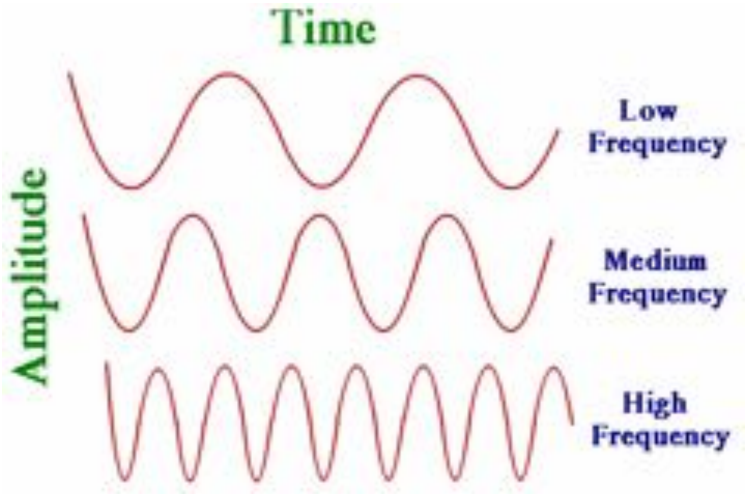
Computing DFT

- 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/

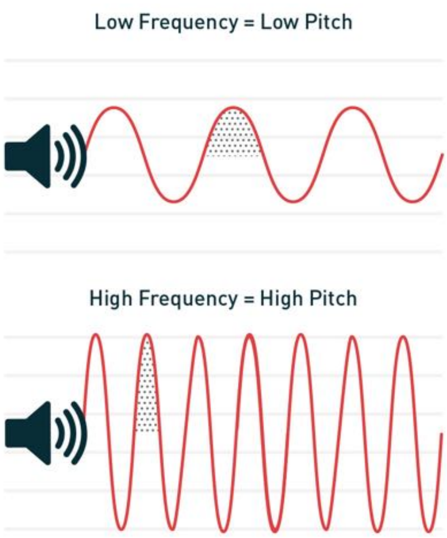
20

Low frequency, High frequency?



21

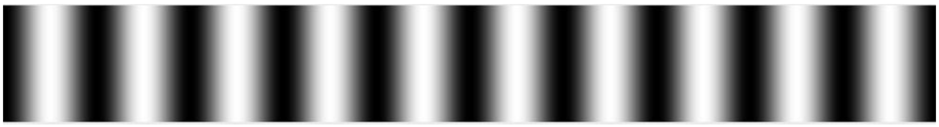
Low frequency, High frequency?



22

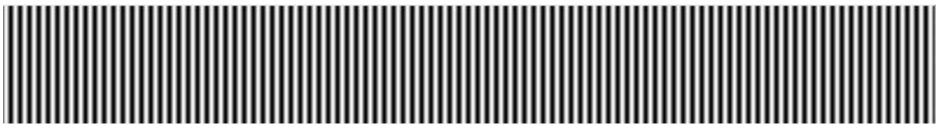
Low frequency, High frequency?

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



Smooth

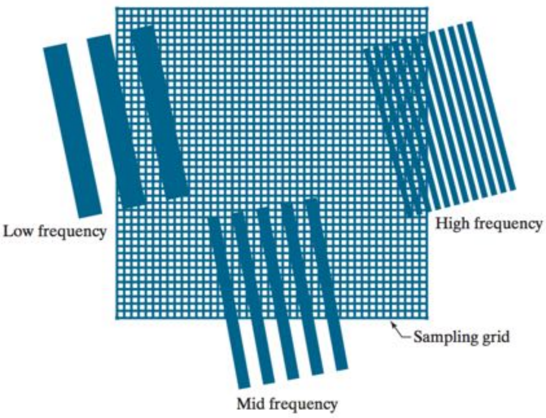
increase the frequency by a factor of 10, so that n = 100 repetitions
f = 100/800 = 1/8 = 0.125 repetitions/px



Finer details,
many edge

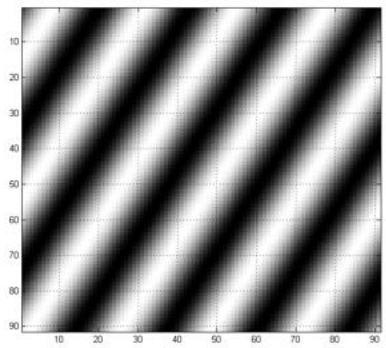
23

Low frequency, High frequency?

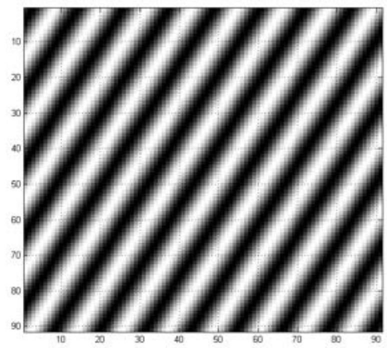


24

Low frequency, High frequency?



Low frequency



High frequency

25

26

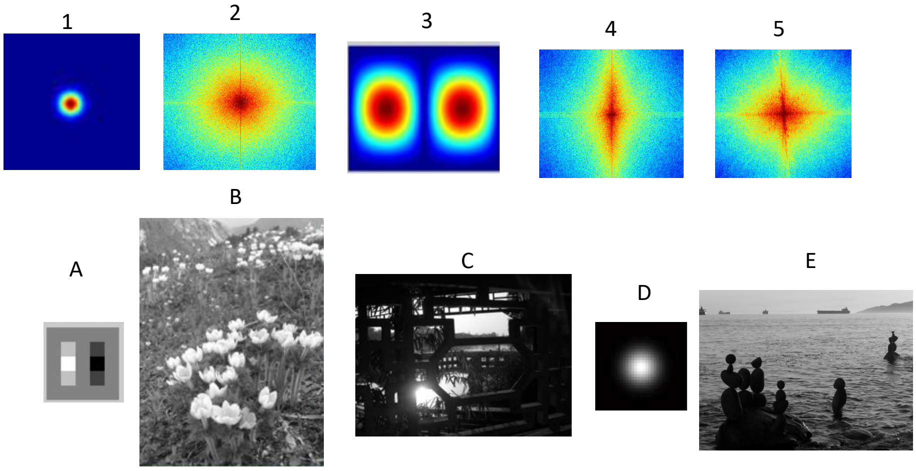
DFT demo

- <view code>

27

Practice question

Match the spatial domain image to the Fourier magnitude image



28

Properties of Fourier Transform

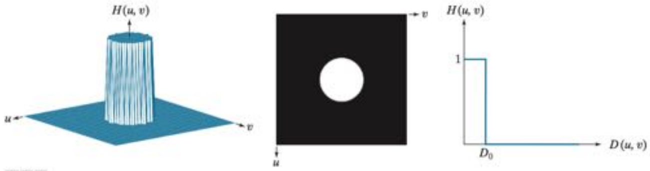
	Spatial Domain *(x)*	Frequency Domain *(u)*
Linearity	$c_1 f(x) + c_2 g(x)$	$c_1 F(u) + c_2 G(u)$
Convolution	$f(x) * g(x)$	$F(u)G(u)$
Scaling	$f(ax)$	$\frac{1}{\lvert a \rvert} F\left(\frac{u}{a}\right)$
Differentiation	$\frac{d^n f(x)}{dx^n}$	$(i2\pi u)^n F(u)$

29

Filtering in Frequency domain

30

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.

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

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

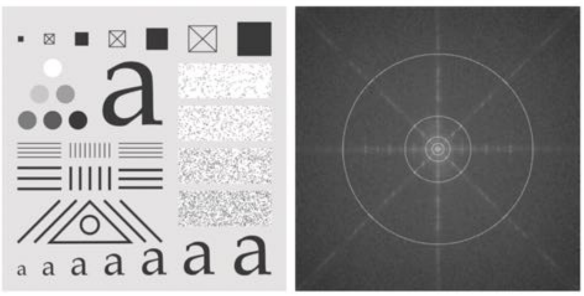
- Cut off high frequencies specified by a distance d_0.
  - Cannot be realized by electronic component → not practical

4/20/20 31

31

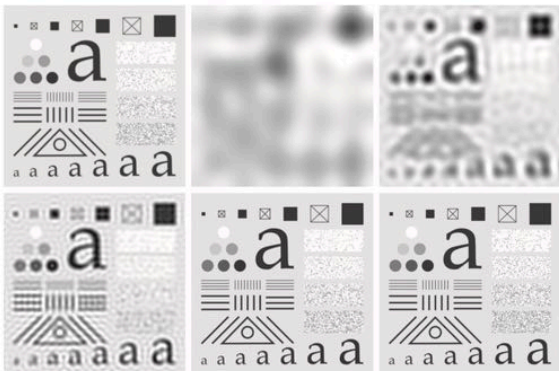
Ideal Lowpass filter

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).

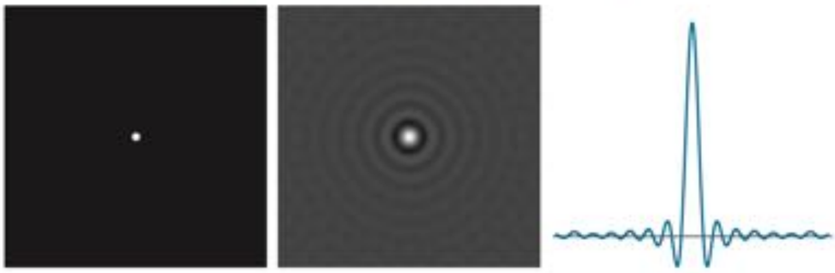
4/20/20 32

32

Ideal Lowpass filter

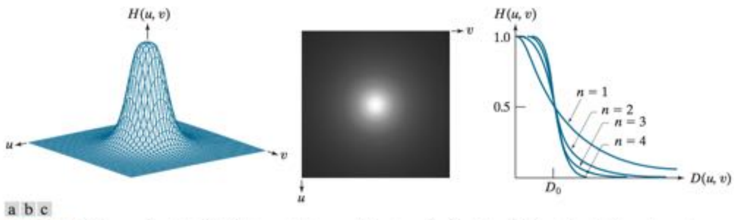
a b c

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).



33

Butterworth Filter



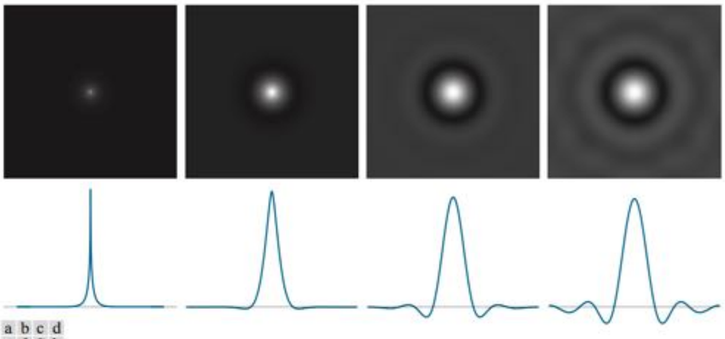
a b c

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.

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

34

Butterworth Filter

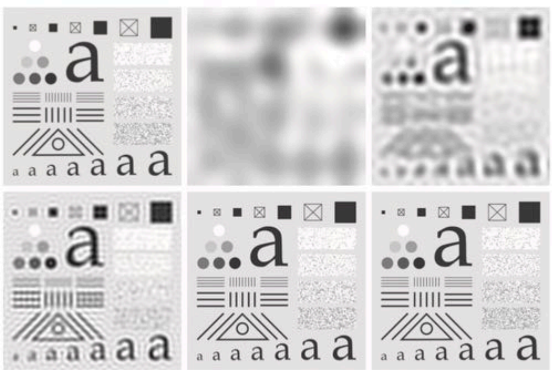


a b c d
e f g h

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.

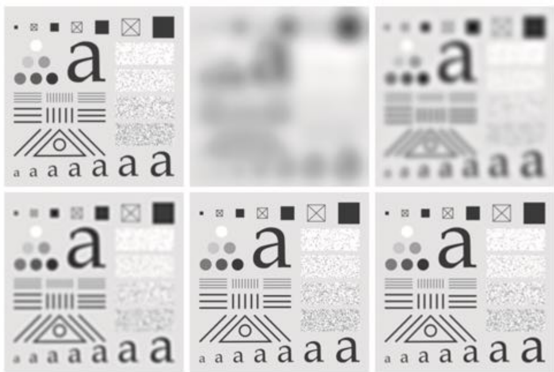
35

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).

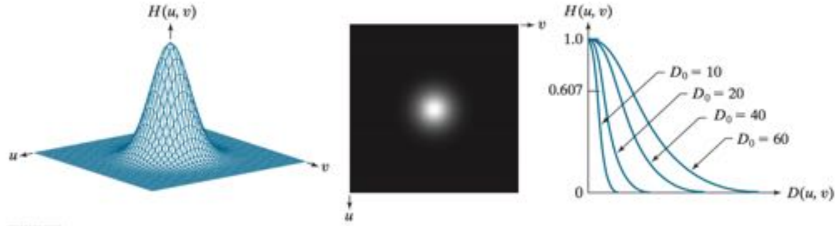


a b c
d e f

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.

36

Gaussian Filter



a b c

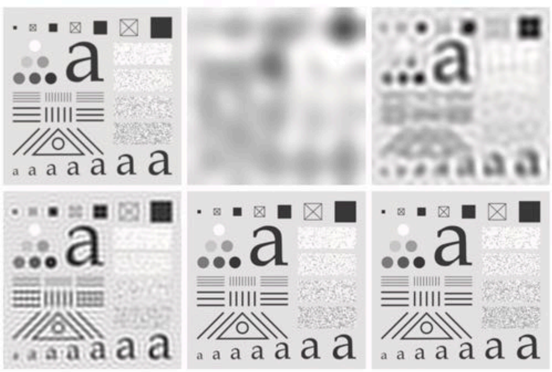
FIGURE 4.43 (a) Perspective plot of a GLPF transfer function. (b) Function displayed as an image. (c) Radial cross sections for various values of D_0.

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

4/20/20 37

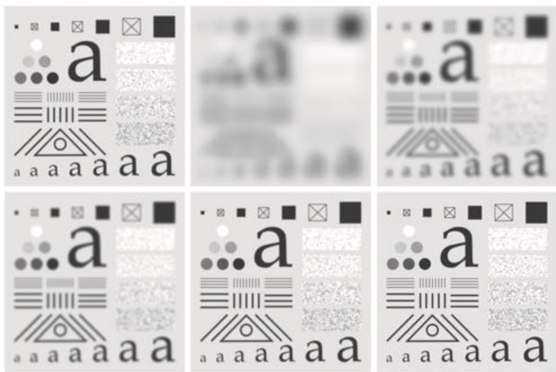
37

Ideal lowpass filter vs Guassian 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).



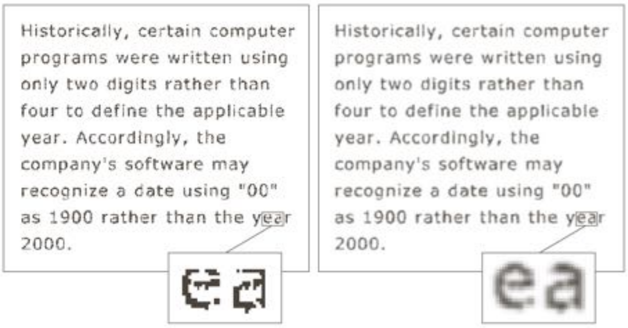
a b c
d e f

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.

4/20/20 38

38

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).

4/20/20 39

39

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).

40

Highpass filter

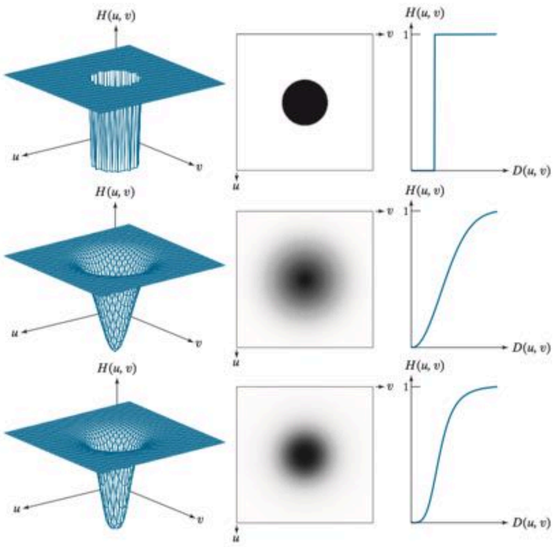


FIGURE 4.51 Top row: Perspective plot, image, and, radial cross section of an IHPF transfer function. Middle and bottom rows: The same sequence for GHPF and BHPF transfer functions. (The thin image borders were added for clarity. They are not part of the data.)

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

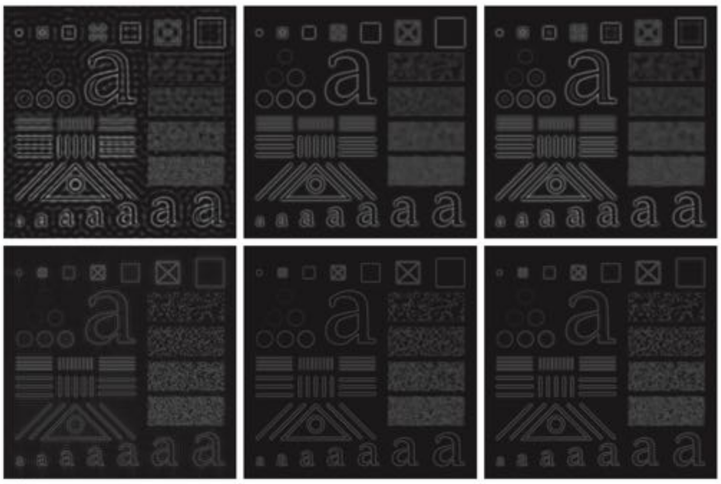
$$H(u,v) = 1 - e^{-D^2(u,v)/2D_0^2}$$

Butterworth

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

41

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 $D_0 = 160$.

42

43