Chapter 5

Image pre-processing

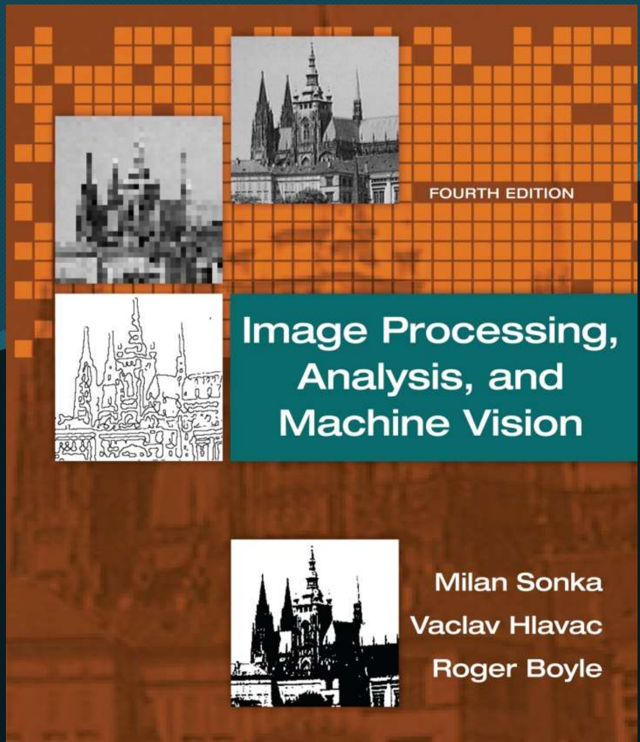
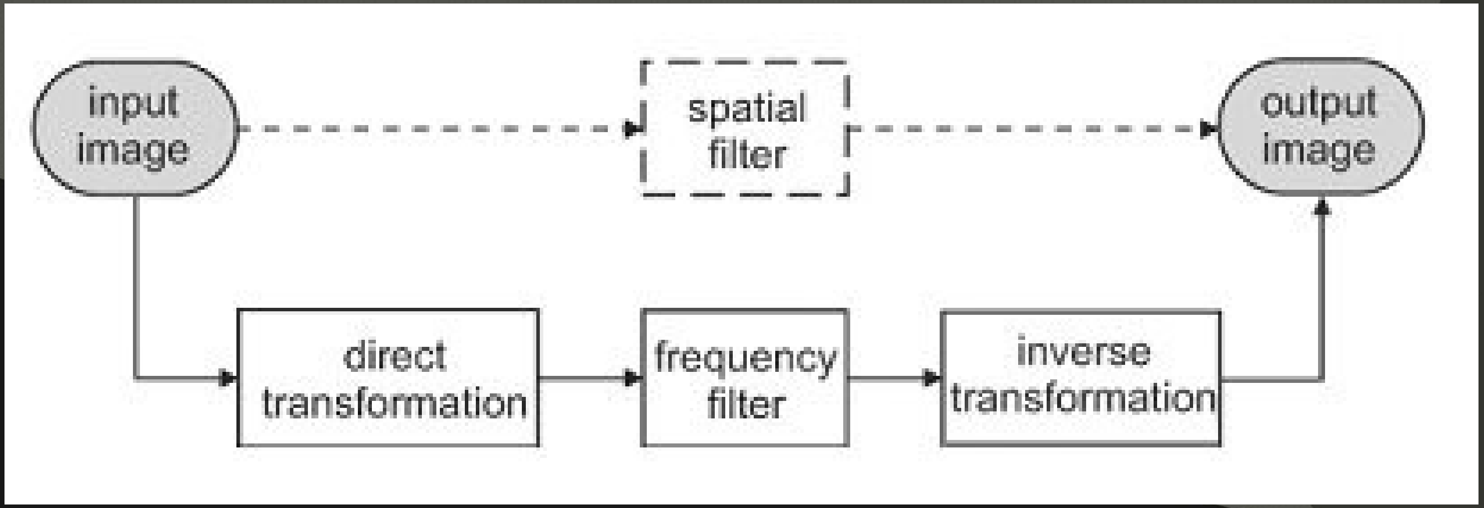


Image pre-processing

- Pixel brightness transformations
- Geometric transformations
- Local pre-processing
  - Image smoothing, edge detection, line detection, corner detection, and region detection.
- Image restoration

Introduction to linear integral transforms

- Image filtering (Chapter 3)
  - An application of a linear integral transform in image processing
  - Filtering can be performed in either spatial or frequency domains.
  - There is one-to-one mapping between the spatial and frequency domains.
  - For linear operations, these two ways should provide equivalent results.



Local pre-processing in the frequency domain

- Spatial frequency filtering
  - Assume that f is an input image and F is its Fourier transform.
  - A convolution filter h can be represented by its Fourier transform H.
  - The Fourier transform of the filter output after an image f has been convolved with the filter h can be computed in the frequency domain

$$G = F .* H$$

where $.*$ represents an element-by-element multiplication of matrices F and H (not matrix multiplication).

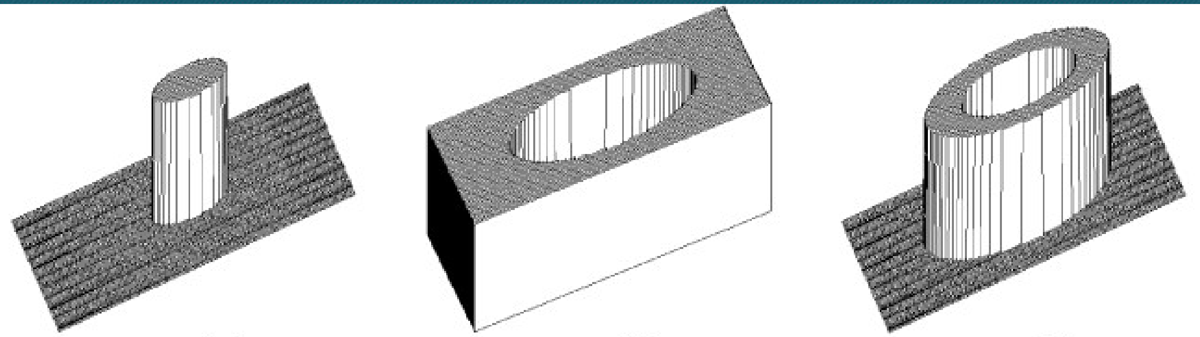
  - The filtered image g can be obtained by applying the inverse Fourier transform to G (equation 3.28).

$$g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G(u, v) e^{2\pi i(xu+yv)} \, du dv \qquad (3.28)$$

Local pre-processing in the frequency domain

- Some basic examples of spatial frequency filtering
  - Linear low-pass frequency filters
  - Linear high-pass frequency filters
  - Linear band-pass frequency filters

(http://www.ejectamenta.com/Imaging-Experiments/fourierimagefiltering.html)

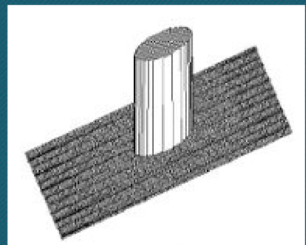


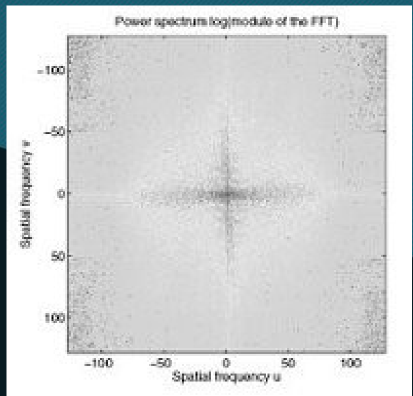
(a) (b) (c)

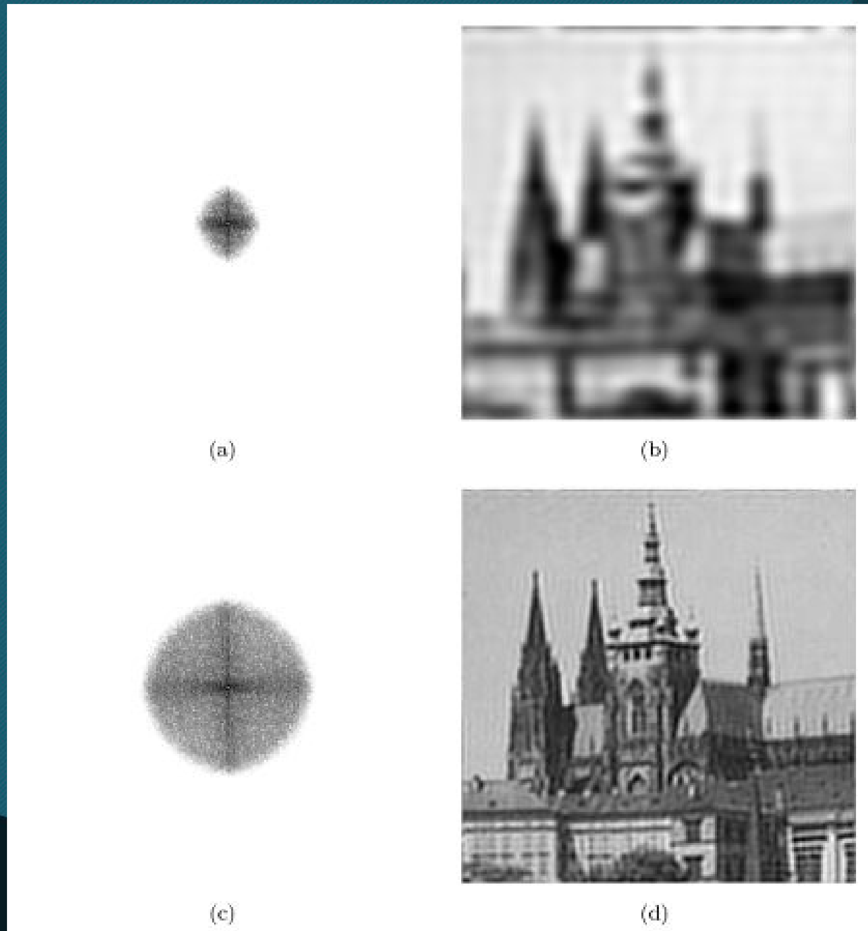
Figure 5.24: Frequency filters displayed in 3D. (a) Low-pass filter. (b) High-pass filter. (c) Band-pass filter. © *Cengage Learning 2015.*

Local pre-processing in the frequency domain

- Low-pass frequency-domain filtering
  - Noise can be suppressed.
  - A blurred image

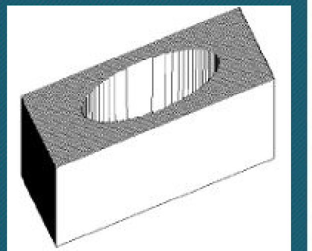


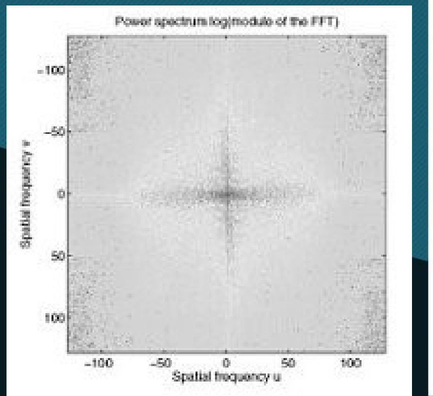


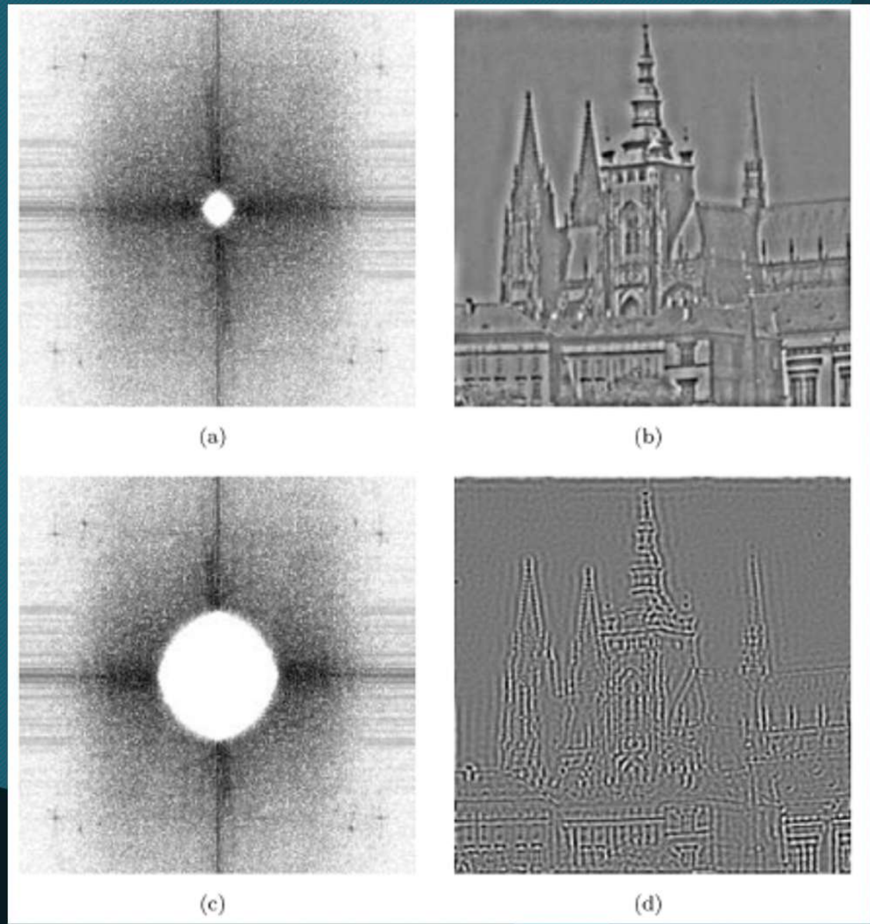


Local pre-processing in the frequency domain

- High-pass frequency-domain filtering
  - Edge can be enhanced.

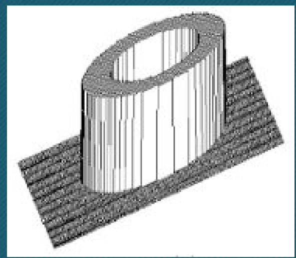


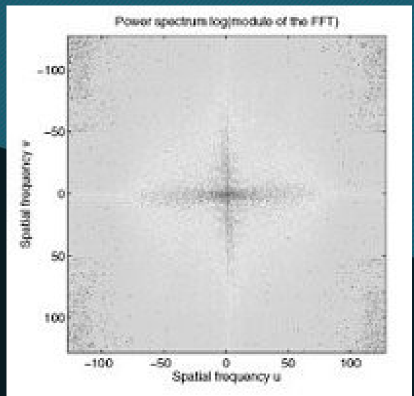


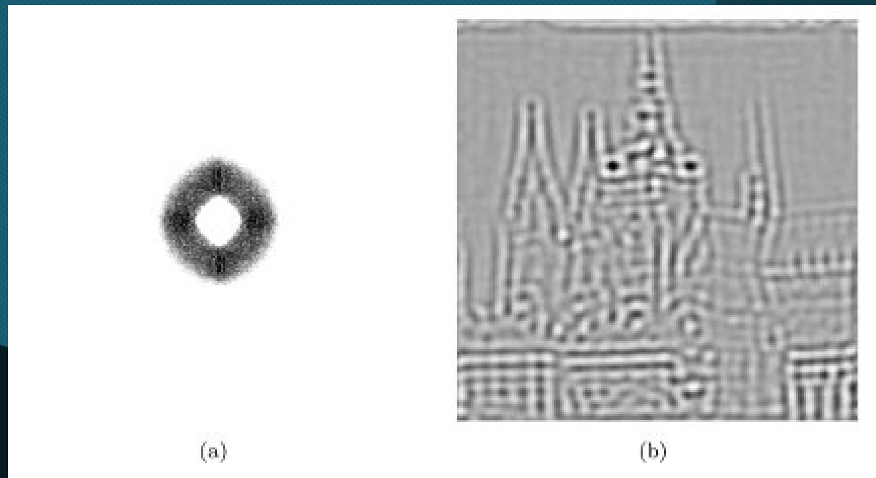


Local pre-processing in the frequency domain

- Bend-pass frequency-domain filtering

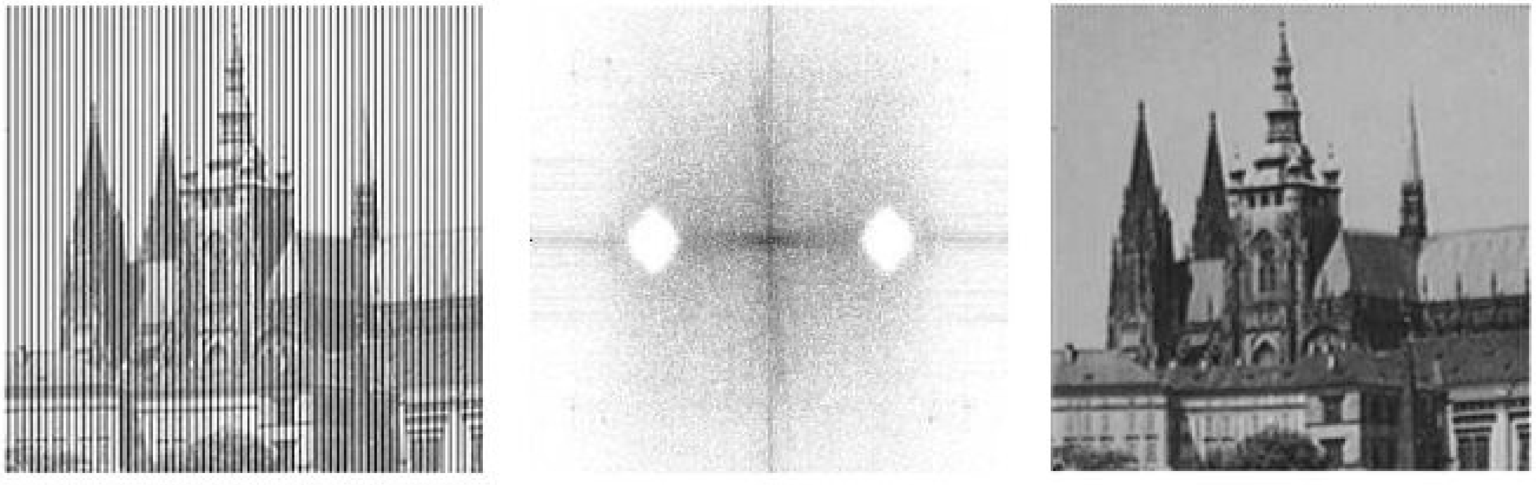






Local pre-processing in the frequency domain

- Periodic noise removal
  - (a) Noisy image
  - (b) Image spectrum used for image reconstruction
    - The areas of frequencies corresponding with periodic vertical lines are filtered out.
  - (c) Filtered image



(a) (b) (c)

Local pre-processing in the frequency domain

- Filters which prove useful for filtering in the frequency domain
  - Gaussian filter
  - Butterworth filter
- Choose an isotropic (等向的) filter for simplicity

$$D(u,v) = D(r) = \sqrt{u^2 + v^2}$$

and let D_0 be a parameter of the filter called the cut-off frequency.

PS. An isotropic filter treats all axes equally.

A cutoff frequency is a boundary in a system's frequency response at which energy flowing through the system begins to be reduced rather than passing through.

(截止頻率(Cutoff frequency)是指一個系統的輸出信號能量開始大幅下降的邊界頻率。)

Local pre-processing in the frequency domain

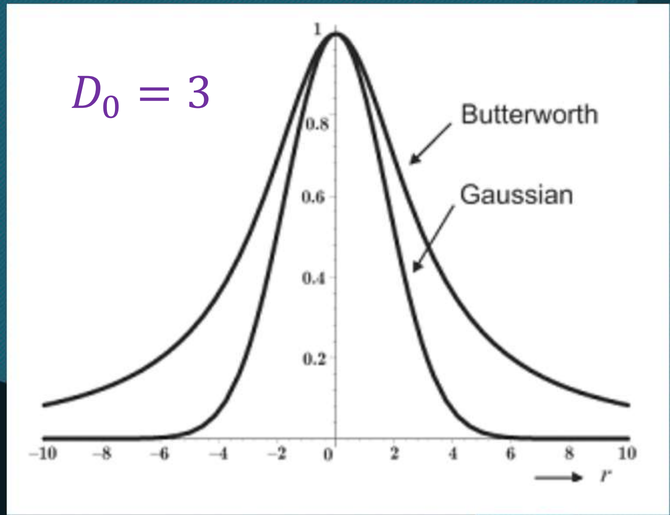
- The Gaussian low-pass filter
  - The cut-off frequency D_0 coincides (一致) with the dispersion σ.
  - The Fourier spectrum of a low-pass Gaussian filter G_{low} is

$$G_{low}(u,v) = \exp\left(-\frac{1}{2}\left(\frac{D(u,v)}{D_0}\right)^2\right)$$

- The Butterworth low-pass filter

$$B_{low}(u,v) = \frac{1}{1+\left(\frac{D(u,v)}{D_0}\right)^n}$$

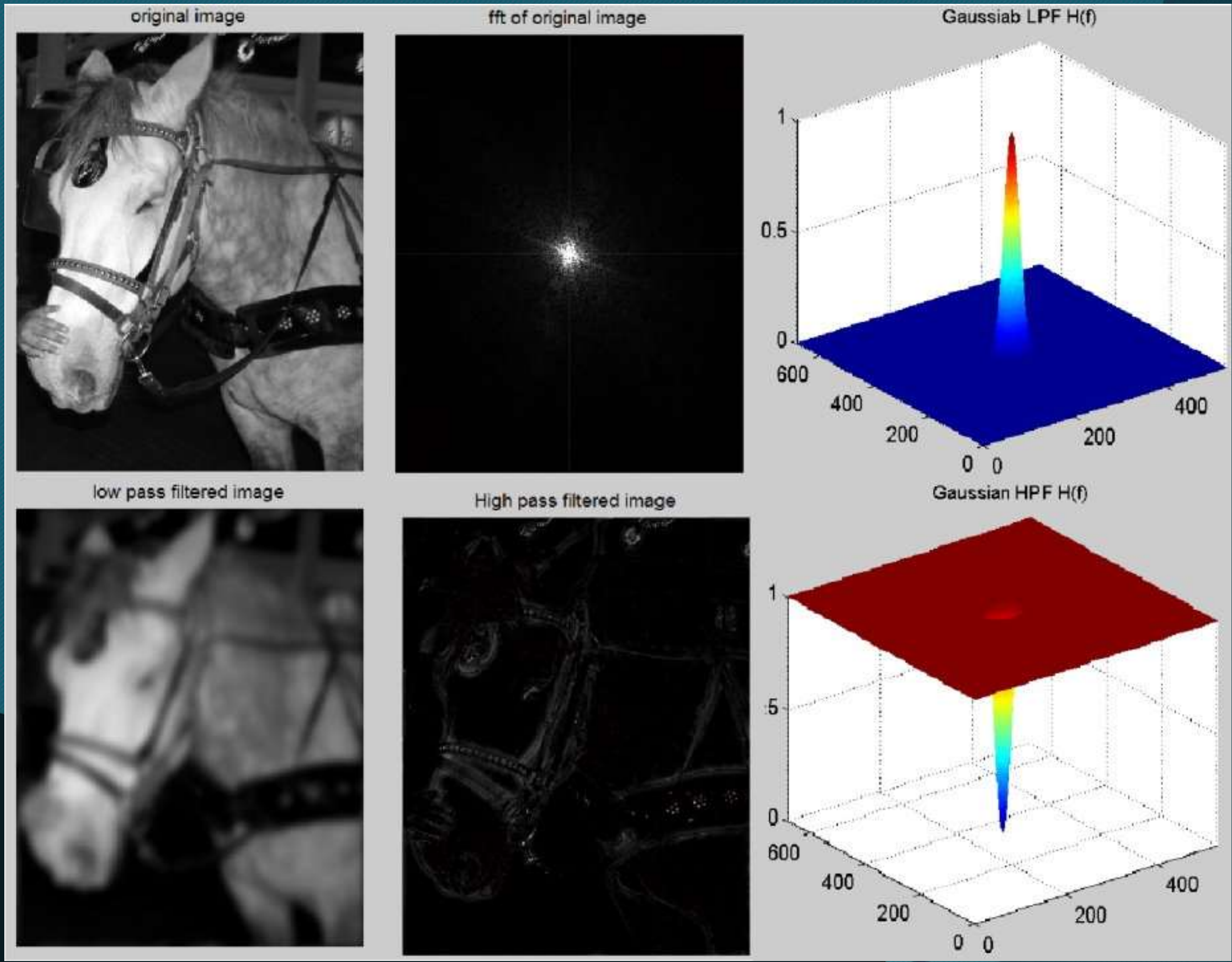
The usually Butterworth filter degree is $n = 2$.



Local pre-processing in the frequency domain

- **High-pass filter**
  - The high-pass filter is created easily from the low-pass filter.
  - If the Fourier frequency spectrum of a low-pass filter is H_{low}, the high-pass filter can be created by just flipping it vertically

$$H_{high} = 1 - H_{low}$$



https://www.mathworks.com/matlabcentral/mlc-downloads/downloads/submissions/46812/versions/1/screenshot.jpg

Local pre-processing in the frequency domain

- Homomorphic (同態的) filtering
  - Another useful pre-processing technique operating in the frequency domain
  - Homomorphic filtering is used to remove multiplicative noise.

    PS: In signal processing, the term multiplicative noise (乘性雜訊) refers to an unwanted random signal that gets multiplied into some relevant signal during capture, transmission, or other processing. An important example is the speckle noise (斑點雜訊) commonly observed in radar imagery.

  - The assumption is the image function $f(x, y)$ can be factorized as a product of two independent multiplicative components in each pixel.

$$f(x, y) = i(x, y)r(x, y)$$

where $i(x, y)$ is the illumination and $r(x, y)$ is the reflectance

Local pre-processing in the frequency domain

- Homomorphic filtering

$$f(x,y) = i(x,y)r(x,y)$$

Apply a logarithmic transform to the input image

$$z(x,y) = \log f(x,y) = \log i(x,y) + \log r(x,y)$$

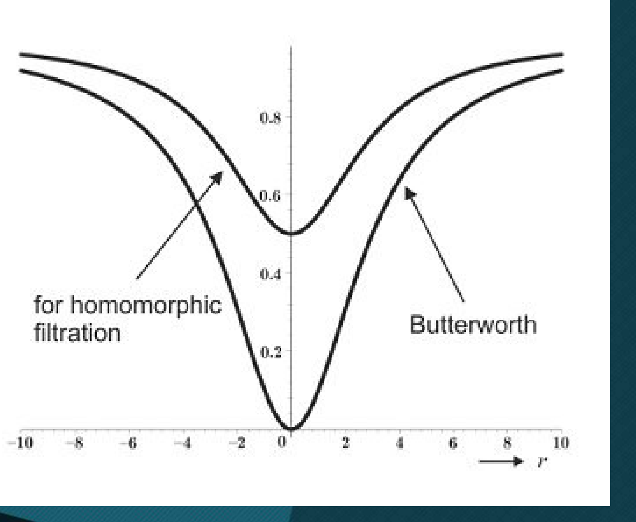
If the image is converted to Fourier space then its additive components remain additive due to the linearity of the Fourier transform.

$$Z(u,v) = I(u,v) + R(u,v)$$

Assume that the Fourier spectrum $Z(u,v)$ is filtered by the filter $H(u,v)$ and the spectrum $S(u,v)$ is the result.

$$S = H.*Z = H.*I + H.*R$$

We use a high-pass filter in the log domain to remove the low-frequency illumination component while preserving the high-frequency reflectance component.
http://blogs.mathworks.com/steve/2013/06/25/homomorphic-filtering-part-1/



Homomorphic filtering
https://zh.wikipedia.org/wiki/%E5%90%8C%E6%85%8B%E6%BF%BE%E6%B3%A2

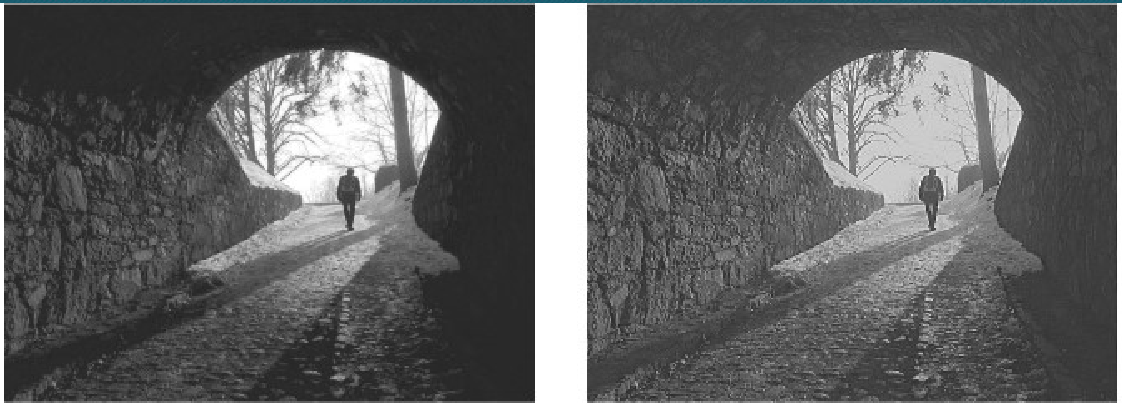
Homomorphic filtering

- Having the filtered spectrum $S(u, v)$, we can return to spatial coordinates using the inverse Fourier transform.

$$s(x, y) = \mathcal{F}^{-1} S(u, v)$$

The result image $g(x, y)$ filtered by the homomorphic filter is given by

$$g(x, y) = \exp(s(x, y))$$



(a) (b)

Figure 5.31: Illustration of homomorphic filtering. (a) Original image. (b) Homomorphic filtering. *Courtesy of T. Svoboda, Czech Technical University, Prague.*

Homomorphic filtering

1. 一張影像可以表示為照度(illumination)分量和反射(reflectance)分量的乘積。
   在時域上這兩者不可分離，但是經傅立葉轉換後在頻域中則可以線性分離。
2. 由於照度變化較小，可以視為影像低頻成份；而反射相對變化較大，可視為高頻成份。
   https://zh.wikipedia.org/wiki/%E5%90%8C%E6%85%8B%E6%BF%BE%E6%B3%A2


原图


结果


原图


结果

http://blog.csdn.net/bluecol/article/details/45788803

Line detection by local pre-processing operators

- Line finding operators
  - Line finding operators aim to find very thin curves in the image.
    - It is assumed that curves do not bend sharply.
    - Such curves and straight lines are called lines.
    - We assumed that the width of the lines is approximately one or two pixels.
    - For examples, some 3×3 convolution kernels are

$$h_1 = \begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix} h_2 = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix} h_3 = \begin{bmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{bmatrix} \ldots$$

    - Suggestion: Thresholded edges are usually wider than one pixel, and line thinning techniques may give a better result.

Edge Reinforcement with Thinning

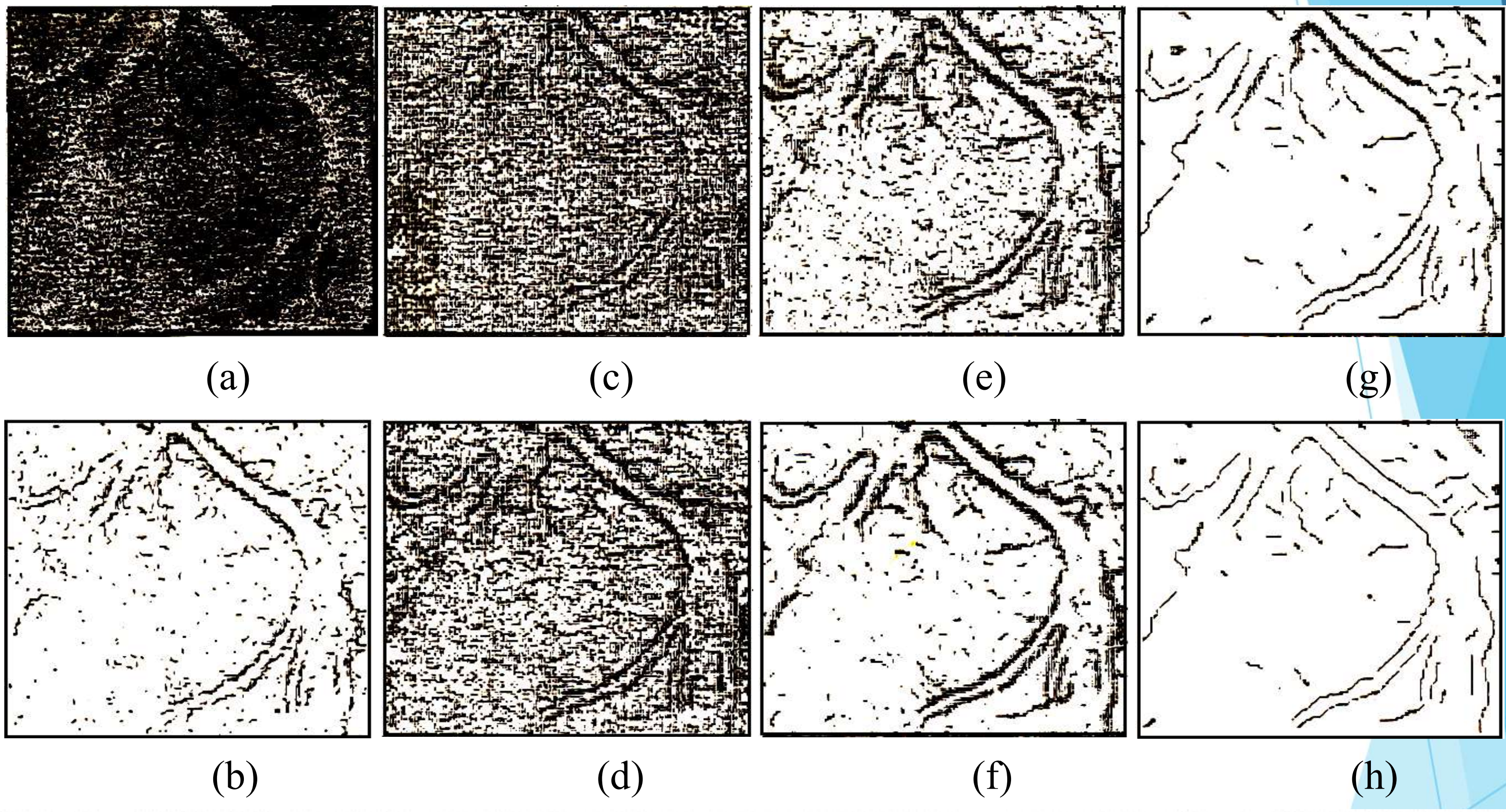


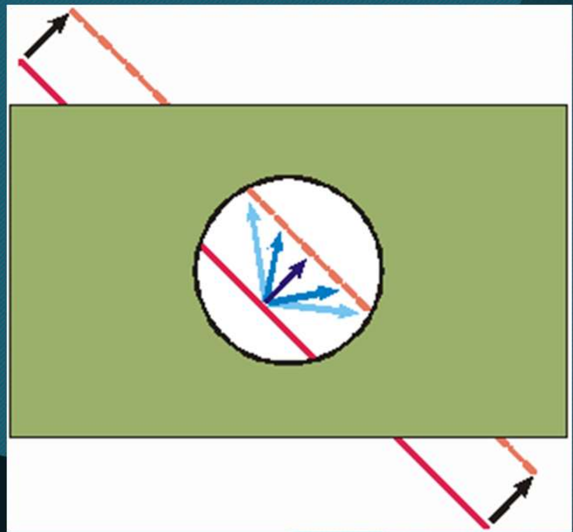
Fig. 7. (a) 132 × 132 portion of an angiographic image (in the LAO projection) of the region near the circumflex branch of the left coronary artery with white noise added to emulate an even poorer quality acquisition; (b) results of running the Sobel operator on (a) while using a high edge magnitude threshold; (c) initial labeling using a low-magnitude-theshold Sobel operator on the noisy scene in (a); (d), (e) results after iterations 2 and 5 of parametrized relaxation labeling without thinning; (f)–(h) results of iterations 5, 30, and 200 with thinning included (note that these are overall iterations 6 through 205).

Detection of corners

- Corners
  - One kind of interest points for the correspondence problem.
  - Corners in images can be located using local detector.
  - Interest points are obtained by thresholding the result of the corner detector.
  - Corners serves better than lines when the correspondence problem is to be solved.

Why?

The aperture problem. (孔徑問題)



Detection of corners

- Corners
  - Edge detectors themselves are not stable at corner.
  - This is natural as the gradient at the tip of the corner is ambiguous.
  - Figure 5.32 shows a sharp corner.
  - Near the corner there is a discontinuity in the gradient direction, and this observation is used in corner detectors.

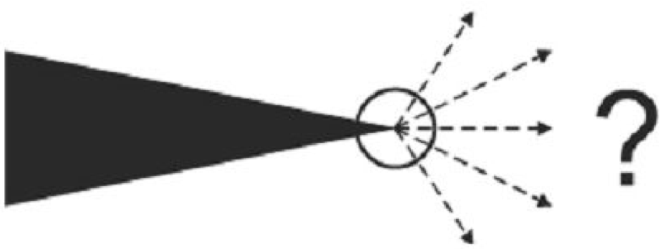


Figure 5.32: Ambiguity of edge detector at the corner tip. © *Cengage Learning 2015.*

Detection of corners

- Corners
  - A corner in an image can be defined as a pixel in whose immediate neighborhood there are two dominant(明顯的), different edge directions.
  - Corner detectors are not usually very robust. One can detect corners using two or more images.
  - Corner detector
    - Moravce detector
    - Zuniga-Haralick operator
    - Harris corner detector

Detection of corners

- Moravce corner detector
  - Moravce detector is maximal in pixels with high contrast, which are corners and sharp edges.
  - The Moravec operator MO is given by

$$\mathrm{MO}(i,j) = \frac{1}{8} \sum_{k=i-1}^{i+1} \sum_{l=j-1}^{j+1} |f(k,l) - f(i,j)|$$

Detection of corners

- Zuniga-Haralick operators
  - Zuniga-Haralick operators are based on the facet model.
  - The image function f is approximated in the neighborhood of the pixel (i, j) by a cubic polynomial with coefficients c_k.

$$f(i,j) = c_1 + c_2 x + c_3 y + c_4 x^2 + c_5 xy + c_6 y^2 \\ + c_7 x^3 + c_8 x^2 y + c_9 x y^2 + c_{10} y^3$$

  - The Zuniga-Haralick operators ZH is given by

$$\mathrm{ZH}(i,j) = \frac{-2(c_2^2 c_6 + c_2 c_3 c_5 + c_3^2 c_4)}{(c_2^2 + c_3^2)^{3/2}}$$

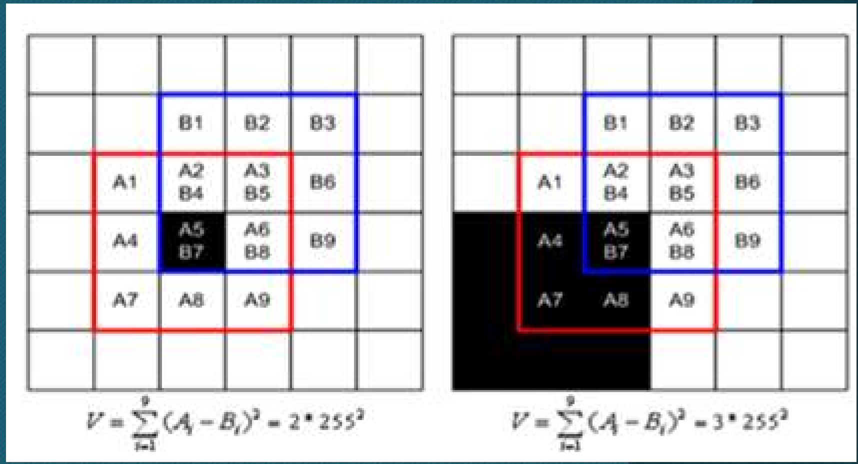
Detection of corners

- Harris corner detector
  - Harris corner detector improved upon Moravec's by considering the differential of the corner score (sum of square difference).
  - Consider a 2D gray-scale image f.
    - An image patch $W \in f$ is taken and is shifted by $\Delta x, \Delta y$.
    - The sum of square differences S between values of the image f given by the patch W and its shifted variant by $\Delta x, \Delta y$ is given by

$$S_w(\Delta x, \Delta y) = \sum_{x_i \in W} \sum_{y_i \in W} \left(f(x_i, y_i) - f(x_i - \Delta x, y_i - \Delta y)\right)^2$$

    - A corner must have a high response of $S_w(\Delta x, \Delta y)$ for all $\Delta x, \Delta y$.

http://blog.csdn.net/jwh_bupt/article/details/7628665



Detection of corners

- Harris corner detector
  - If the shifted image patch is approximated by the first-order Taylor expansion

$$f(x_i - \Delta x, y_i - \Delta y) \approx f(x_i, y_i) + \left[\frac{\partial f(x_i, y_i)}{\partial x} \quad \frac{\partial f(x_i, y_i)}{\partial y}\right]\begin{bmatrix}\Delta x \\ \Delta y\end{bmatrix}$$

Then the minimum of $S_w(\Delta x, \Delta y)$ can be obtained analytically.

$$S_w(\Delta x, \Delta y) = \sum_{x_i \in W} \sum_{y_i \in W} \left(f(x_i, y_i) - f(x_i - \Delta x, y_i - \Delta y)\right)^2$$

$$= \sum_{x_i \in W} \sum_{y_i \in W} \left(f(x_i, y_i) - \left(f(x_i, y_i) + \left[\frac{\partial f(x_i, y_i)}{\partial x} \quad \frac{\partial f(x_i, y_i)}{\partial y}\right]\begin{bmatrix}\Delta x \\ \Delta y\end{bmatrix}\right)\right)^2$$

$$= \sum_{x_i \in W} \sum_{y_i \in W} \left(-\left[\frac{\partial f(x_i, y_i)}{\partial x} \quad \frac{\partial f(x_i, y_i)}{\partial y}\right]\begin{bmatrix}\Delta x \\ \Delta y\end{bmatrix}\right)^2$$

Detection of corners

- Harris corner detector
  - Since that $\mathbf{u}^2 = \mathbf{u}^T\mathbf{u}$

$$S_w(\Delta x, \Delta y) = \sum_{x_i \in W} \sum_{y_i \in W} \left(- \begin{bmatrix} \frac{\partial f(x_i, y_i)}{\partial x} & \frac{\partial f(x_i, y_i)}{\partial y} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \right)^2$$

$$= \sum_{x_i \in W} \sum_{y_i \in W} [\Delta x \quad \Delta y] \left(\begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} \end{bmatrix} \right) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

$$= [\Delta x \quad \Delta y] \left(\sum_{x_i \in W} \sum_{y_i \in W} \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} \end{bmatrix} \right) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

$$= [\Delta x \quad \Delta y] A_w(x, y) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

Detection of corners

- Harris corner detector
  - The **Harris matrix** $A_w(x,y)$ represents one half the second derivative of the image patch W around the point $(x,y)=(0,0)$.
  - A is

$$A(x,y)=2\cdot\begin{bmatrix}\sum_{x_i\in W}\sum_{y_i\in W}\left(\frac{\partial f(x_i,y_i)}{\partial x}\right)^2 & \sum_{x_i\in W}\sum_{y_i\in W}\frac{\partial f(x_i,y_i)}{\partial x}\frac{\partial f(x_i,y_i)}{\partial y}\\ \sum_{x_i\in W}\sum_{y_i\in W}\frac{\partial f(x_i,y_i)}{\partial x}\frac{\partial f(x_i,y_i)}{\partial y} & \sum_{x_i\in W}\sum_{y_i\in W}\left(\frac{\partial f(x_i,y_i)}{\partial y}\right)^2\end{bmatrix}$$

  - **The Harris matrix A is symmetric and positive semi-definite.**

https://en.wikipedia.org/wiki/Positive-definite_matrix

Detection of corners

- Harris corner detector
  - Let λ_1, λ_2 be the eigenvalues of A, three distinct cases can occur:
    - Both eigenvalues are small.
      - Image f is flat at the examined pixel.
      - There is no edges or corners in this location.
    - One eigenvalue is small and the second large.
      - The local neighborhood is ridge(山脊)-shaped.
      - Significant change of f occurs if a small movement is made perpendicularly to the ridge.
    - Both eigenvalues are rather large.
      - A small shift in any direction causes significant change of f.
      - A corner has been found.

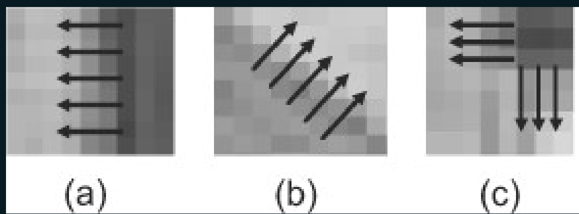


Figure 5.33: The Harris corner detector according to eigenvalues of the local structure matrix. (a), (b) Ridge detected, no corner at this position. (c) Corner detected. © *Cengage Learning 2015.*

Eigenvalue and Eigenvector

- Given a square matrix A, the eigenvalue can be obtained by $\det(A - \lambda I) = \mathbf{0}.$
- For example
  - The matrix $\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$ defines a linear transformation of the real plane.
  - The eigenvalues of matrix A are given by

$$\det \begin{bmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{bmatrix} = (2-\lambda)^2 - 1 = 0.$$

  - The roots of this equation are λ = 1 and λ = 3.
  - Considering first the eigenvalue λ = 3, we have $\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 3 \begin{bmatrix} x \\ y \end{bmatrix}.$
  - After matrix-multiplication $\begin{bmatrix} 2x+y \\ x+2y \end{bmatrix} = 3 \begin{bmatrix} x \\ y \end{bmatrix}.$
    - This matrix equation represents a system of two linear equations

      $2x + y = 3x$ and $x + 2y = 3y$.
    - Both the equations reduce to the single linear equation $x = y$.
    - To find an eigenvector, we are free to choose any value for x,

      so by picking x=1 and setting y=x, we find the eigenvector to be $\begin{bmatrix} 1 \\ 1 \end{bmatrix}.$

Detection of corners

- Harris corner detector
  - Harris suggested that exact eigenvalue computation can be avoided by calculating the response function.

$$R(A) = \det(A) - \kappa \operatorname{trace}^2(A)$$

(1) If $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ then $\det(A) = ad - bc$ and $\operatorname{trace}(A) = a + d$.

(2) Moreover, κ is a parameter, values from 0.04 to 0.15.

Detection of corners

- Examples of Harris corner detector

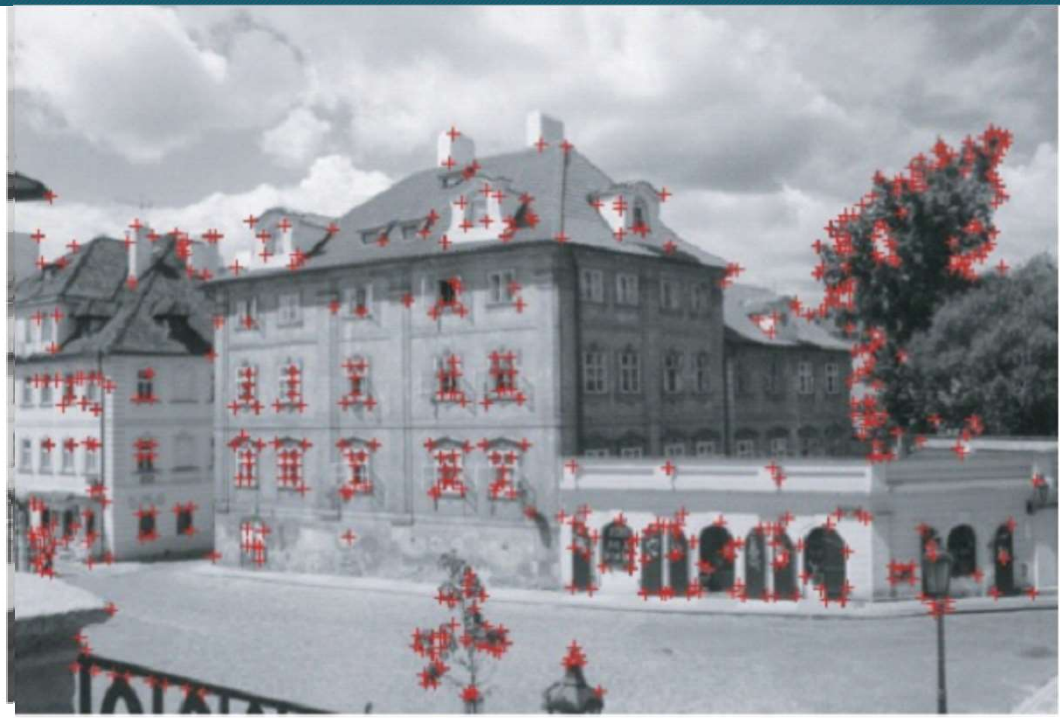


Figure 5.34: Red crosses mark Harris corners. *Courtesy of M. Urban, Czech Technical University, Prague. A color version of this figure may be seen in the color inset—Plate 7.*

Harris corner detector

Algorithm 5.5 Harris corner detector

1. Filter the image with a Gaussian.
2. Estimate intensity gradient in two perpendicular directions for each pixel, $\frac{\partial f(x,y)}{\partial x}, \frac{\partial f(x,y)}{\partial y}$. This is performed by twice using a 1D convolution with the kernel approximating the derivative.
3. For each pixel and a given neighborhood window:
   - Calculate the local structure matrix A.
   - Evaluate the response function $R(A)$.
4. Choose the best candidates for corners by selecting a threshold on the response function and perform non-maximal suppression.

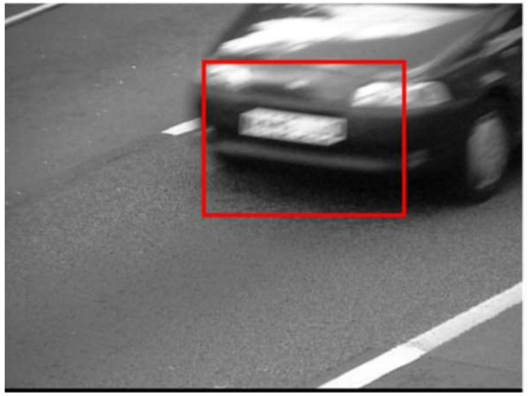
Image restoration

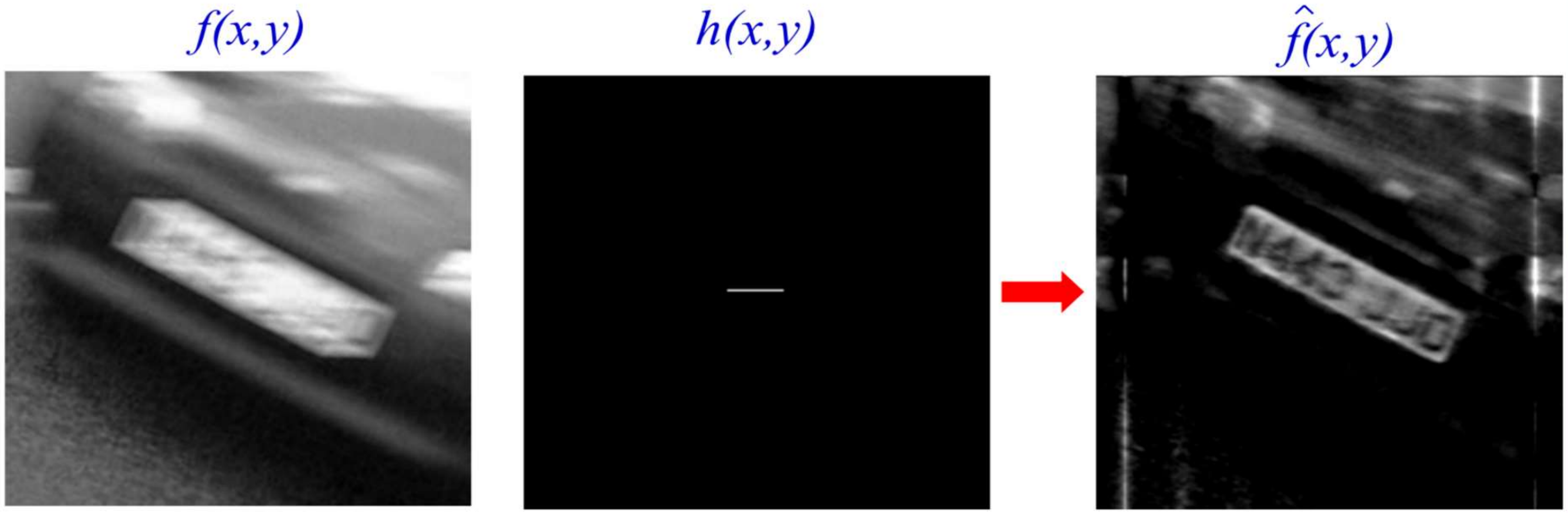
- Pre-processing methods that aim to suppress degradation using knowledge about its nature are called image restoration. (影像還原;影像復原)
- Most image restoration methods are based on convolution applied globally to the whole image.
- Image restoration techniques can be classified as deterministic or stochastic.
  - Deterministic (決定論的) methods are applicable to images with little noise and a known degradation function.
  - Stochastic (隨機的) techniques try to find the best restoration according to a particular statistical criterion, e.g., a least-square method.

Image restoration

- Three typical degradations with a simple function
  - Relative constant speed movement of the object with respect to the camera
  - Wrong lens focus
  - Atmospheric turbulence (大氣亂流)
- In most practical cases, there is insufficient knowledge about the degradation, and it must be estimated and modeled.
  - A priori knowledge: known in advance or obtained before restoration
    - For example, the parameters of a device (camera)
  - A posteriori knowledge: obtained by analyzing the degraded image
    - For example, the interest points in the image

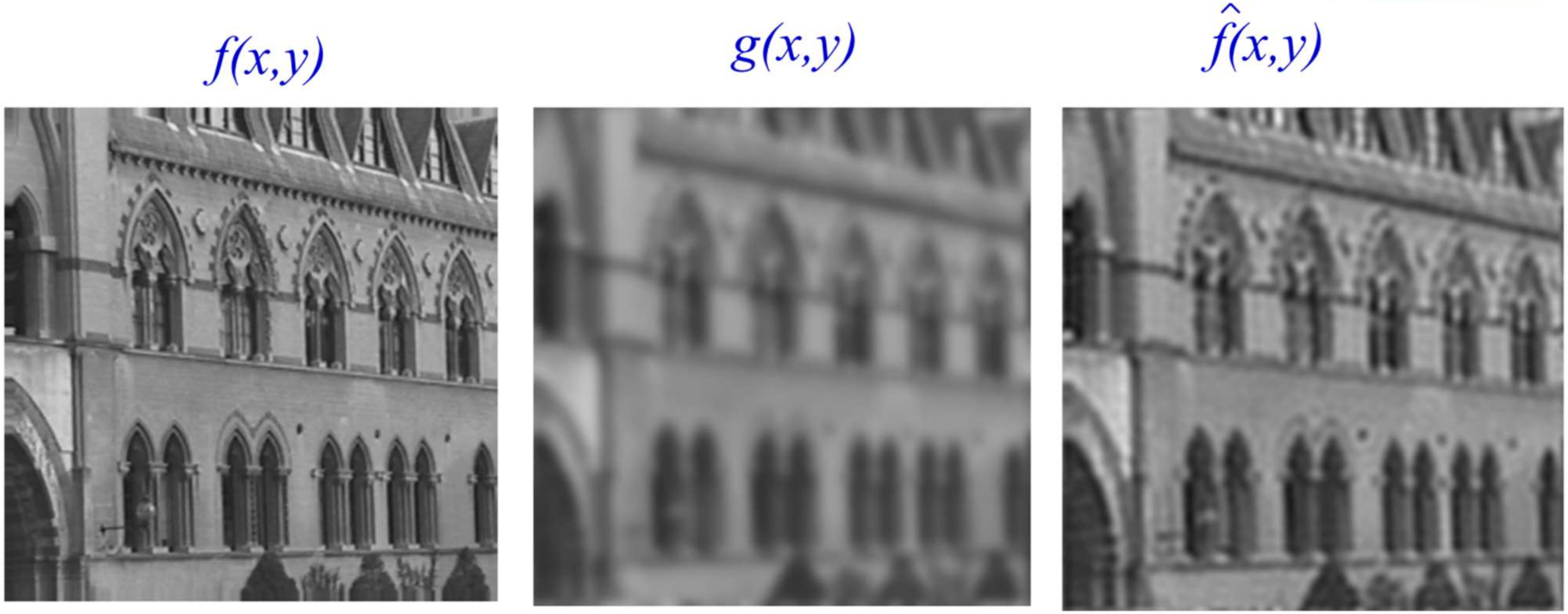
An example—Object moving





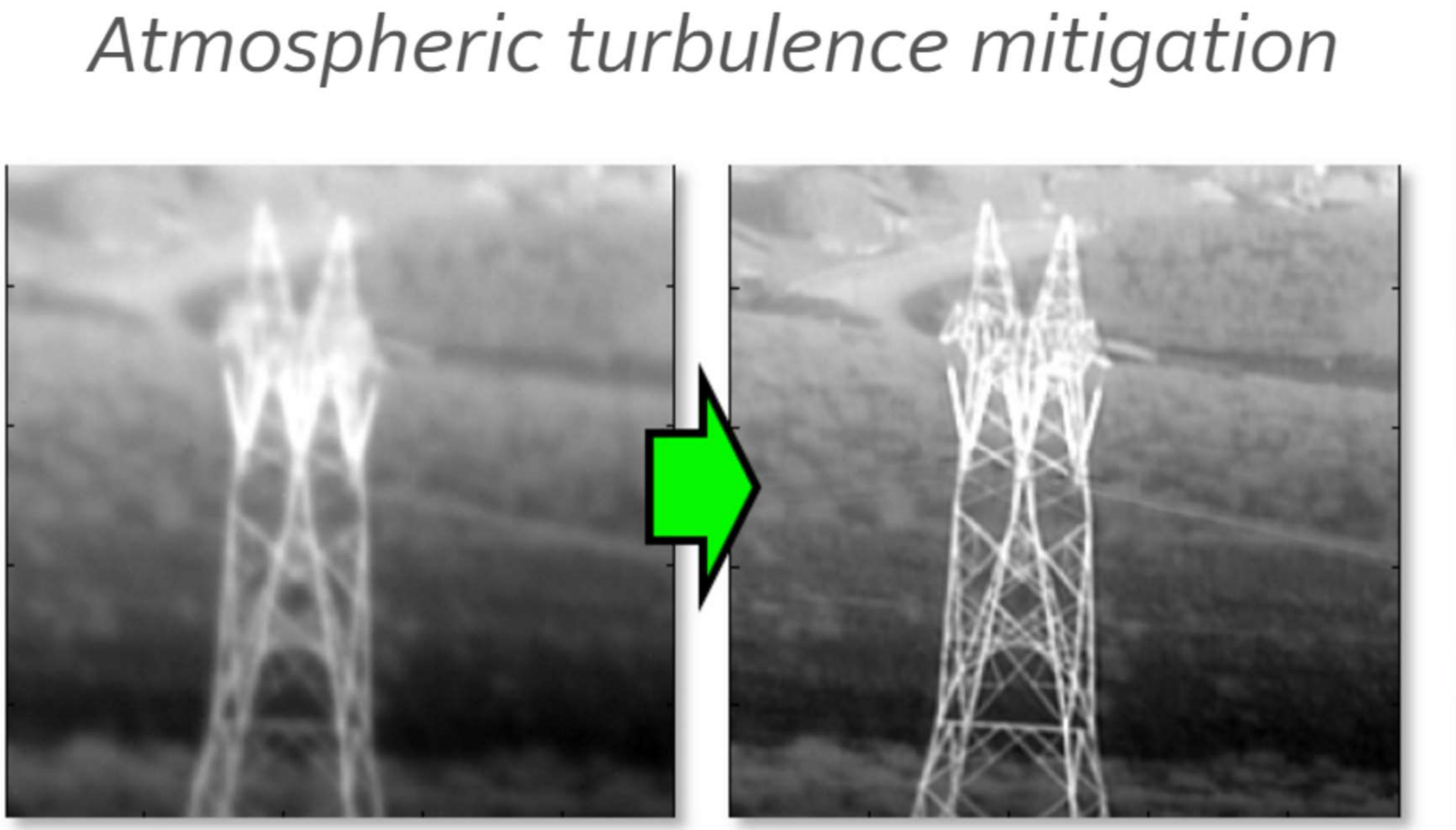
http://www.robots.ox.ac.uk/~az/lectures/ia/lect3.pdf

An example—Wrong lens focus



http://www.robots.ox.ac.uk/~az/lectures/ia/lect3.pdf

An example—Atmospheric turbulence



https://www.udayton.edu/engineering/research/signal-and-image-processing-lab/research/index.php

Image restoration

- A degraded image g can arise from the original image f by

$$g(i,j) = s\left(\int\int_{(a,b)\in\mathcal{O}} f(a,b)h(a,b,i,j)\mathrm{d}a\mathrm{d}b\right) + \nu(i,j)$$

where s is some non-linear function and ν describes the noise.

  - It can be simplified by neglecting the non-linearity and assuming that the function h is invariant with respect to position in the image.

$$g(i,j) = (f * h)(i,j) + \nu(i,j)$$

  - If the noise is not significant in this equation, the restoration equates to inverse convolution.
  - If noise is not negligible, then the inverse convolution is solved as an overdetermined system of linear equations.

Image restoration

- Degradations that are easy to restore
  - Since $g(i,j) = (f * h)(i,j)$, in the Fourier domain,

$$G = HF$$

- Case 1: Relative motion of camera and object
  - Suppose V is the constant speed in the direction of the x axis, T be the shutter open time. The Fourier transform $H(u,v)$ of the degradation caused in time T is give by [Rosenfeld and Kak, 1982]

$$H(u,v) = \frac{\sin(\pi V T u)}{\pi V u}$$

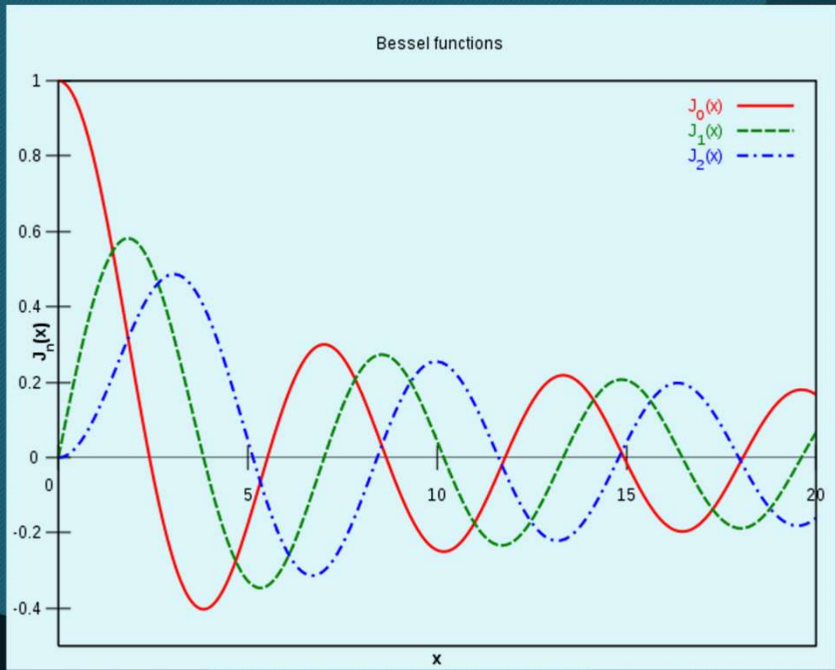
Image restoration

- Case 2: Wrong lens focus
  - Image smoothing caused by imperfect focus of a thin lens can be described by the function [Born and Wolf, 1969]

$$H(u, v) = \frac{J_1(ar)}{ar}$$

where J_1 is the Bessel function of the first order, $r^2 = u^2 + v^2$, and a is the displacement.

  - The model is not space invariant.



由Alessio Damato - own work本向量圖形使用gnuplot創作。https://commons.wikimedia.org/w/index.php?curid=365944

Image restoration

- Case 3: Atmospheric turbulence
  - Atmospheric turbulence is degradation that needs to be restored in remote sensing (遙測) and astronomy (天文學).
  - One mathematical model [Hufnagel and Stanley, 1964] is

$$H(u,v) = e^{-c\left(u^2+v^2\right)^{5/6}}$$

where c is a constant that depends on the type of turbulence which is usually found experimentally.

Image restoration

- Inverse filtering

$$g(i,j) = (f * h)(i,j) + \nu(i,j)$$

  - Inverse filtering assumes that
    - degradation was caused by a linear function $h(i,j)$
    - consider the additive noise ν as another source of degradation
    - additive noise ν is independent of the signal
  - After applying the Fourier transform

$$G(u,v) = F(u,v)H(u,v) + N(u,v)$$

  - The original image F (its Fourier transform to be exact)

$$F(u,v) = G(u,v)H^{-1}(u,v) - N(u,v)H^{-1}(u,v)$$

Image restoration

- Wiener (least mean square) filtering
  - Wiener filtering attempts to take account of noise properties by incorporating a priori knowledge in the image restoration formula.
  - Restoration by the Wiener filter gives an estimate $\hat{f}$ of the original image f with minimal mean square error.

$$e^2 = \mathcal{E}\left\{\left(f(i,j) - \hat{f}(i,j)\right)^2\right\}$$

where $\mathcal{E}$ denotes the mean operator.

  - Denote the Fourier transform of the Wiener filter by H_W.
  - The estimate $\hat{F}$ of the Fourier transform F of the original image f can be obtained as

$$\hat{F}(u,v) = H_W(u,v)G(u,v)$$

Image restoration

- Wiener (least mean square) filtering
  - The estimate $\hat{F}$ of the Fourier transform F of the original image f can be obtained as

$$\hat{F}(u,v) = H_W(u,v)G(u,v)$$

  - H_W is not derived here, but may be found elsewhere [Gonzalez and Woods, 1992] as

$$H_W(u,v) = \frac{H^*(u,v)}{|H(u,v)|^2 + \left[S_{\nu\nu}(u,v)/S_{ff}(u,v)\right]^2}$$

where H: the transform function of the degradation.

$*$: the complex conjugate

$S_{\nu\nu}$: the spectral density of the noise

S_{ff}: the spectral density of the undegraded image

https://en.wikipedia.org/wiki/Spectral_density

http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/VELDHUIZEN/node15.html

Image restoration

- An restoration example of an image that was degraded by 5 pixels motion in the direction of the x axis

(a) (b)

Figure 5.36: Restoration of motion blur using Wiener filtering. *Courtesy of P. Kohout, Criminalistic Institute, Prague.*