

## Global Image Properties

Global image properties refer to an image as a whole rather than components. Computation of global image properties is often required for image enhancement, preceding image analysis.

We treat

- empirical mean and variance
- histograms
- projections
- cross-sections
- frequency spectrum

1

## Empirical Mean and Variance

Empirical mean = average of all pixels of an image

$$\bar{g} = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} g_{mn} \quad \text{with } M \times N \text{ image size}$$

Simplified notation: 
$$\bar{g} = \frac{1}{K} \sum_{k=0}^{K-1} g_k$$

Incremental computation: 
$$\bar{g}_0 = 0 \quad \bar{g}_k = \frac{\bar{g}_{k-1}(k-1) + g_k}{k} \quad k = 2 \dots K$$

Empirical variance = average of squared deviation of all pixels from mean

$$\sigma^2 = \frac{1}{K} \sum_{k=1}^K (g_k - \bar{g})^2 = \frac{1}{K} \sum_{k=1}^K g_k^2 - \bar{g}^2$$

Incremental computation:

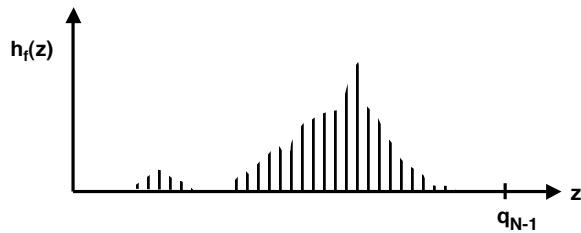
$$\sigma_0^2 = 0 \quad \sigma_k^2 = \frac{(\sigma_{k-1}^2 + \bar{g}_{k-1}^2)(k-1) + g_k^2}{k} - \left( \frac{\bar{g}_{k-1}(k-1) + g_k}{k} \right)^2 \quad k = 2 \dots K$$

2

## Greyvalue Histograms

A greyvalue histogram  $h_f(z)$  of an image  $f$  provides the frequency of greyvalues  $z$  in the image.

The histogram of an image with  $N$  quantization levels is represented by a 1D array mit  $N$  elements.

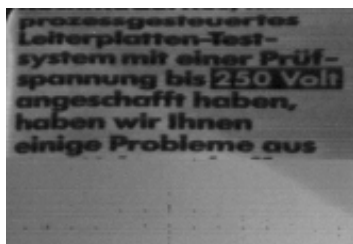


A greyvalue histogram describes discrete values, a greyvalue distribution describes continuous values.

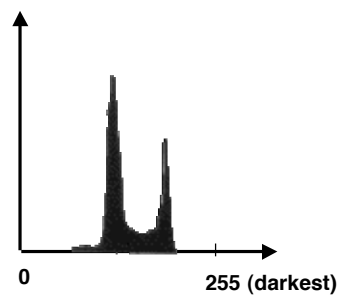
3

## Example of Greyvalue Histogram

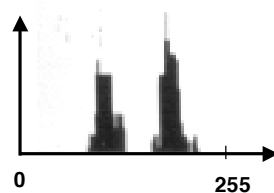
image



histogram



A histogram can be "sharpened" by discounting pixels at edges (more about edges later):



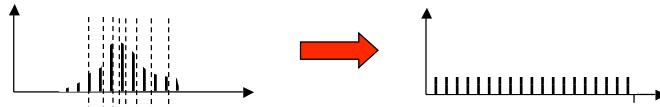
4

## Histogram Modification

Greyvalues may be remapped into new greyvalues to

- facilitate image analysis
- improve subjective image quality

**Example:** Histogram equalization



1. Cut histogram into N stripes of equal area (N = new number of greyvalues)
2. Assign new greyvalues to consecutive stripes

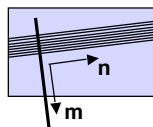


Examples show improved resolution of image parts with most frequent greyvalues (road surface)

5

## Projections

A projection of greyvalues in an image is the sum of all greyvalues orthogonal to a base line:

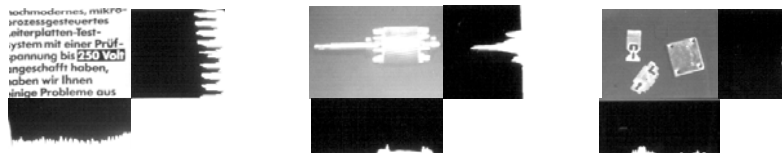


$$p_m = \sum_n g_{mn}$$

Often used:

"row profile" = row vector of all (normalized) column sums

"column profile" = column vector of all (normalized) row sums

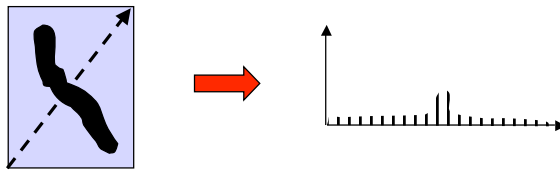


6

## Cross-sections

A cross-section of a greyvalue image is a vector of all pixels along a straight line through the image.

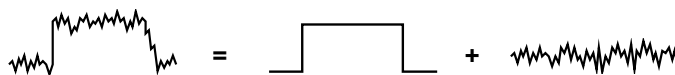
- fast test for localizing objects
- commonly taken along a row or column



7

## Noise

Deviations from an ideal image can often be modelled as additive noise:



Typical properties:

- mean 0, variance  $\sigma^2 > 0$
- spatially uncorrelated:  $E[r_{ij} r_{mn}] = 0$  for  $ij \neq mn$
- temporally uncorrelated:  $E[r_{ij,t1} r_{ij,t2}] = 0$  for  $t1 \neq t2$

$E[x]$  is  
"expected  
value" of  $x$

- Gaussian probability density:  $p(r) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{r^2}{2\sigma^2}}$

Noise arises from analog signal generation (e.g. amplification) and transmission.

There are several other noise models other than additive noise.

8

## Noise Removal by Averaging

Principle:  $\hat{r}_k = \frac{1}{K} \sum_{k=1}^K r_k \Rightarrow 0$  sample mean approaches density mean

There are basically 2 ways to "average out" noise:

- temporal averaging if several samples  $g_{ij,t}$  of the same pixel but at different times  $t = 1 \dots T$  are available
- spatial averaging if  $g_{mn} \approx g_{ij}$  for all pixels  $g_{mn}$  in a region around  $g_{ij}$

How effective is averaging of K greyvalues?

$\hat{r}_k = \frac{1}{K} \sum_{k=1}^K r_k$  is random variable with mean and variance depending on K

$$E[\hat{r}_k] = \frac{1}{K} \sum_{k=1}^K E[r_k] = 0 \quad \text{mean}$$

$$E[(\hat{r}_k - E[\hat{r}_k])^2] = E[\hat{r}_k^2] = E\left[\frac{1}{K^2} \left(\sum_{k=1}^K r_k\right)^2\right] = \frac{1}{K^2} \sum_{k=1}^K E[r_k^2] = \frac{\sigma^2}{K} \quad \text{variance}$$

Example: In order cut the standard deviation  $\sigma$  in half, 4 values have to averaged

9

## Example of Averaging



intensity averaging with  
5 x 5 mask

$$\frac{1}{25} \begin{array}{|c|c|c|c|c|} \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline \end{array}$$

10

## Simple Smoothing Operations

### 1. Averaging

$$\hat{g}_{ij} = \frac{1}{|D|} \sum_{g_{mn} \in D} g_{mn} \quad D \text{ is region around } g_{ij}$$

Example of 3-by-3 region D

	ij	

### 2. Removal of outliers

$$\hat{g}_{ij} = \begin{cases} \frac{1}{|D|} \sum_{g_{mn} \in D} g_{mn} & \text{if } \left| g_{ij} - \frac{1}{|D|} \sum_{g_{mn} \in D} g_{mn} \right| \geq S \\ g_{ij} & \end{cases} \quad S \text{ is threshold}$$

### 3. Weighted average

$$\hat{g}_{ij} = \frac{1}{\sum w_k} \sum_{g_k \in D} w_k g_k \quad w_k = \text{weights in } D$$

Example of weights in 3-by-3 region

1	2	1
2	3	2
1	2	1

Note that these operations are heuristics and not well founded!

11

## Bimodal Averaging

To avoid averaging across edges, assume bimodal greyvalue distribution and select average value of modality with largest population.

1. Determine  $\bar{g}_D = \frac{1}{|D|} \sum_{g_{mn} \in D} g_{mn}$

2.  $A = \{g_k \text{ with } g_k \geq \bar{g}_D\}$      $B = \{g_k \text{ with } g_k < \bar{g}_D\}$

$$3. \hat{g}_D = \begin{cases} \frac{1}{|A|} \sum_{g_k \in A} g_k & \text{if } |A| \geq |B| \\ \frac{1}{|B|} \sum_{g_k \in B} g_k & \text{otherwise} \end{cases}$$

Example:

B	11	14	15
	13	12	25
	15	19	26
		A	

$\bar{g}_D = 16,7 \rightarrow A, B \rightarrow \hat{g}_D = 13$

12

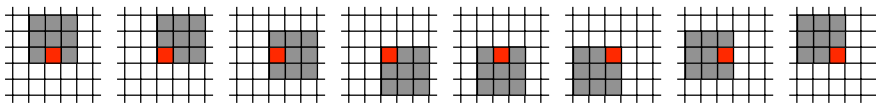
## Averaging with Rotating Mask

Replace center pixel by average over pixels from the most homogeneous subset taken from the neighbourhood of center pixel.

Measure for (lack of) homogeneity is dispersion  $\sigma^2$  (= empirical variance) of the greyscale values of a region D:

$$\bar{g}_{ij} = \frac{1}{|D|} \sum_{g_{mn} \in D} g_{mn} \quad \sigma_{ij}^2 = \frac{1}{|D|} \sum_{g_{mn} \in D} (g_{mn} - \bar{g}_{ij})^2$$

Possible rotated masks in 5 x 5 neighbourhood of center pixel:



**Algorithm:**

1. Consider each pixel  $g_{ij}$
2. Calculate dispersion in mask for all rotated positions of mask
3. Choose mask with minimum dispersion
4. Assign average greyvalue of chosen mask to  $g_{ij}$

13

## Median Filter

Median of a distribution  $P(x)$ :  $x_m$  such that  $P(x < x_m) = 1/2$

**Median Filter:**

$$\hat{g}_{ij} = \max a \text{ with } g_k \in D \text{ and } |\{g_k < a\}| < \frac{|D|}{2}$$

1. Sort pixels in D according to greyvalue
2. Choose greyvalue in middle position

**Example:**

11	14	15
13	12	25
15	19	26



11  
 12  
 13  
 14  
15  
 15  
 19  
 25  
 26



greyvalue of center pixel of region is set to 15

Median Filter reduces influence of outliers in either direction!

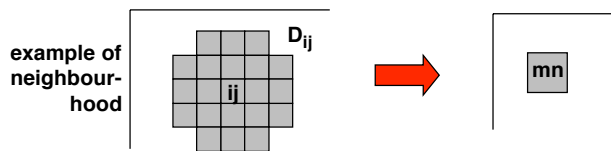
14

## Local Neighbourhood Operations

Many useful image transformations may be defined as an instance of a local neighbourhood operation:

Generate a new image with pixels  $\hat{g}_{mn}$  by applying operator  $f$  to all pixels  $g_{ij}$  of an image

$$\hat{g}_{mn} = f(g_1, g_2, \dots, g_K) \quad g_1, g_2, \dots, g_K \in D_{ij}$$



Pixel indices  $i, j$  may be incremented by steps larger than 1 to obtain reduced new image.

15

## Example of Sharpening



intensity sharpening  
with 3 x 3 mask

-1	-1	-1
-1	9	-1
-1	-1	-1

"unsharp masking" =  
subtraction of blurred image

$$\hat{g}_{ij} = g_{ij} - \frac{1}{|D|} \sum_{g_{mn} \in D} g_{mn}$$



16

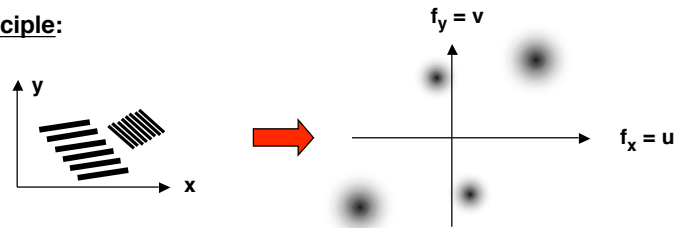


## Spectral Image Properties

An image function may be considered a sum of spatially sinusoidal components of different frequencies.

The frequency spectrum indicates the magnitudes of the spatial frequencies contained in an image.

Principle:



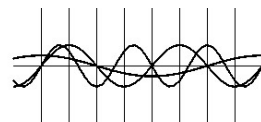
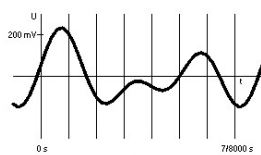
Important qualitative properties of spectral information:

- spectral information is independent of image locations
- sharp edges give rise to high frequencies
- noise (= disturbances of image signal) is often high-frequency

17

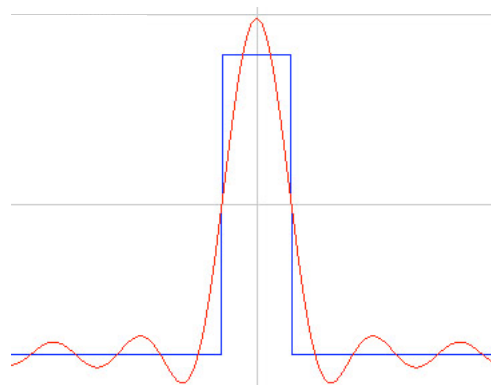
## Illustration of 1-D Fourier Series Expansion

**original waveform**



**sinusoidal components  
add up to original waveform**

**approximation of a rectangular pulse  
with 1 ... 5 sinusoidal components**



Online demonstration of Fourier Series approximations at <http://www.jhu.edu/~signals/fourier2/>

18

## Discrete Fourier Transform (DFT)

Computes image representation as a sum of sinusoids.

Discrete Fourier Transform:

$$G_{uv} = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} g_{mn} e^{-2\pi i \left( \frac{mu}{M} + \frac{nv}{N} \right)}$$

for  $u = 0 \dots M-1, v = 0 \dots N-1$

Inverse Discrete Fourier Transform:

$$g_{mn} = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} G_{uv} e^{2\pi i \left( \frac{mu}{M} + \frac{nv}{N} \right)}$$

for  $m = 0 \dots M-1, n = 0 \dots N-1$

Notation for computing the Fourier Transform:

$$G_{uv} = F\{g_{mn}\}$$

$$g_{mn} = F^{-1}\{G_{uv}\}$$

Transform is based on periodicity assumption

⇒ periodic continuation may cause boundary effects



19

## Basic Properties of DFT

- Linearity:  $F\{a g_{mn} + b g_{mn}\} = a F\{g_{mn}\} + b F\{g_{mn}\}$
- Symmetry:  $G_{-u,-v} = G_{uv}$  for real  $g_{mn}$  (such as images)

In general, the Fourier transform is a complex function with a real and an imaginary part:

$$G_{uv} = R_{uv} + i I_{uv}$$

Euler's formula:

$$r e^{iz} = r \cos(z) + r i \sin(z)$$

$$|G_{uv}| = \sqrt{R_{uv}^2 + I_{uv}^2}$$

frequency spectrum or amplitude spectrum

$$P_{uv} = |G_{uv}|^2 = R_{uv}^2 + I_{uv}^2$$

power spectrum or spectral density

$$\Phi_{uv} = \tan^{-1} \left( \frac{I_{uv}}{R_{uv}} \right)$$

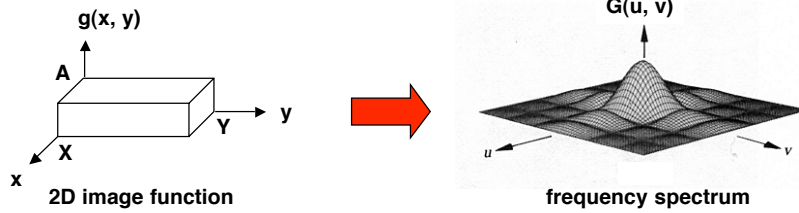
phase spectrum

Recommended reading:

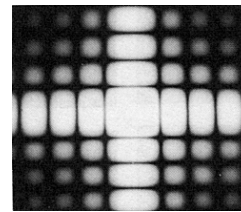
Gonzalez/Wintz  
Digital Image Processing  
Addison Wesley 87

20

## Illustrative Example of Fourier Transform



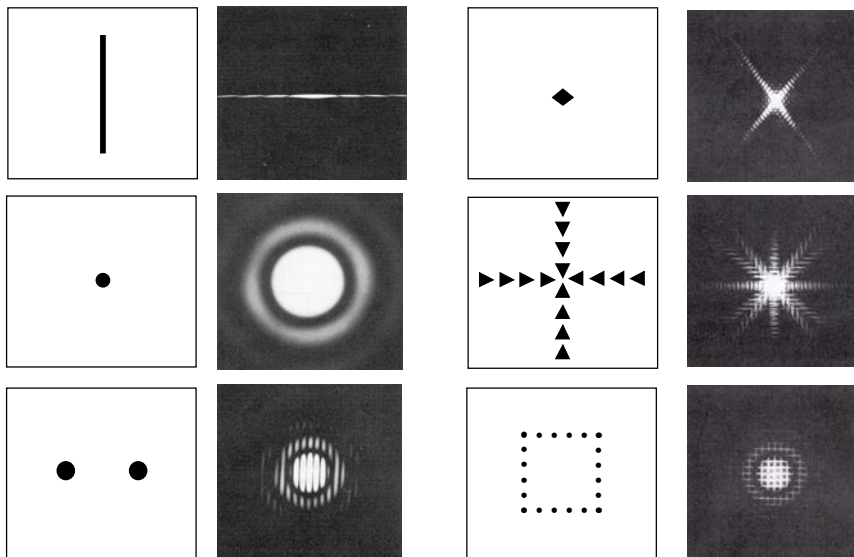
Note that large spectral amplitudes occur in directions vertical to prominent edges of the image function



frequency spectrum as an intensity function

21

## Examples of Fourier Transform Pairs



22

## Fast Fourier Transform (FFT)

Ordinary DFT needs  $\sim(MN)^2$  operations for an  $M \times N$  image.

Example:  $M = N = 1024$ ,  $10^{-12}$  sec/operation  $\Rightarrow$  1,1 sec

FFT is based on recursive decomposition of  $g_{mn}$  into subsequences.  
 $\Rightarrow$  multiple use of partial results  $\Rightarrow \sim MN \log_2(MN)$  operations  
 Same example needs only 0.000021 sec

Decomposition principle for 1D Fourier transform:

$$G_r = \frac{1}{N} \sum_{n=0}^{N-1} g_n e^{-2\pi i r \frac{n}{N}} \quad \{g_n\} = \begin{cases} \{g_n^{(1)}\} = \{g_{2n}\} \\ \{g_n^{(2)}\} = \{g_{2n+1}\} \end{cases} \quad n = 0 \dots N/2-1$$

$$G_r = \frac{1}{N} \sum_{n=0}^{N/2-1} \left\{ g_n^{(1)} e^{-2\pi i r \frac{2n}{N}} + g_n^{(2)} e^{-2\pi i r \frac{(2n+1)}{N}} \right\} \quad r = 0 \dots N-1$$

$$G_r = G_r^{(1)} + e^{-2\pi i \frac{r}{N}} G_r^{(2)} \quad r = 0 \dots N/2-1$$

$$G_{r+N/2} = G_r^{(1)} - e^{-2\pi i \frac{r}{N}} G_r^{(2)}$$

All  $G_r$  may be computed by  $2(N/2)^2$  instead of  $(N)^2$  operations!

23

## Convolution

Convolution is an important operation for describing and analyzing linear operations, e.g. filtering.

Definition of 2D convolution for continuous signals:

$$g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(r, s) h(x-r, y-s) dr ds = f(x, y) * h(x, y)$$

Convolution in the spatial domain is dual to multiplication in the frequency domain:

$$F\{f(x, y) * h(x, y)\} = F(u, v) H(u, v)$$

$$F\{f(x, y) h(x, y)\} = F(u, v) * H(u, v)$$

$H$  can be interpreted as attenuating or amplifying the frequencies of  $F$ .

$\Rightarrow$  Convolution describes filtering in the spatial domain.

24

## Filtering in the Frequency Domain

A filter transforms a signal by modifying its spectrum.

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

- F Fourier transform of the signal
- H frequency transfer function of the filter
- G modified Fourier transform of signal

- low-pass filter      *low frequencies pass, high frequencies are attenuated or removed*
- high-pass filter     *high frequencies pass, low frequencies are attenuated or removed*
- band-pass filter     *frequencies within a frequency band pass, other frequencies below or above are attenuated or removed*

Often (but not always) the noise part of an image is high-frequency and the signal part is low-frequency. Low-pass filtering then improves the signal-to-noise ratio.

25

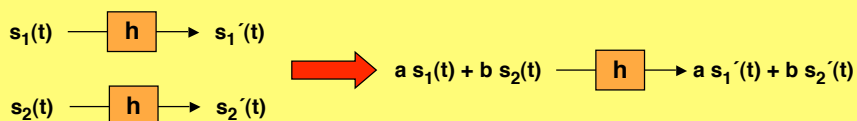
## Filtering in the Spatial Domain

Filtering in the spatial domain is described by convolution.

$$g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(r, s) h(x-r, y-s) dr ds = f(x, y) * h(x, y)$$

Commonly used description for the effect of technical components in linear signal theory:

$$s'(t) = \int_{-\infty}^{+\infty} h(r) s(t-r) dr$$



An impulse  $\delta$  as input generates the filter function  $h(x, y)$  as output:

$$h(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(r, s) \delta(x-r, y-s) dr ds = h(x, y) * \delta(x, y)$$

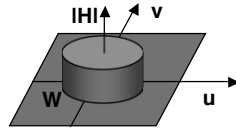
$h(x, y)$  is often called "impulse response"

26

## Low-pass Filters

### Ideal low-pass filter

All frequencies above  $W$  are removed



$$|H(u, v)| = \begin{cases} 1 & \text{for } \sqrt{u^2 + v^2} \leq W \\ 0 & \text{otherwise} \end{cases}$$

Note that the filter function  $h(x, y)$  is rotation symmetric and

$$h(r) \sim \sin 2\pi W r / (2\pi W r) \quad \text{with } r^2 = x^2 + y^2$$

=> impuls-shaped input structures may produce ring-like structures as output

### Gaussian filter

A Gaussian filter has an optimally smooth boundary, both in the frequency and the spatial domain. It is important for several advanced image analysis methods, e.g. generating multiscale images.

$$H(u, v) = e^{-\frac{1}{2}(u^2 + v^2)\sigma^2} \quad h(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2} \frac{x^2 + y^2}{\sigma^2}}$$

27

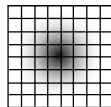
## Discrete Filters

For periodic discrete 2D signals (e.g. discrete images), the convolution operator which describes filtering is

$$g_{ij} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_{mn} h_{-m, j-n}$$

Each pixel  $g_{ij}$  of the filtered image is the sum of the products of the original image with the mirror filter  $h_{-m, -n}$  placed at location  $ij$ .

Example:



$h_{mn} = h_{-m, -n}$  is a bell-shaped function

The filtering effect is a smoothing operation by weighted local averaging.

The choice of weights of a local filter - the convolution mask - may influence the properties of the output image in important ways, e.g. with regard to remaining noise, blurred edges, artificial structures, preserved or discarded information.

28

## Matrix Notation for Discrete Filters

The convolution operation  $g_{ij} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_{mn} h_{i-m, j-n}$   
 may be expressed as matrix multiplication  $\underline{g} = \mathbf{H} \underline{f}$ .

Vectors  $\underline{g}$  and  $\underline{f}$  are obtained by stacking rows (or columns) onto each other:

$$\underline{g}^T = [g_{00} \ g_{01} \ \dots \ g_{0 \ N-1} \ g_{10} \ g_{11} \ \dots \ g_{1 \ N-1} \ \dots \ g_{M-1 \ 0} \ g_{M-1 \ 1} \ \dots \ g_{M-1 \ N-1}]$$

$$\underline{f}^T = [f_{00} \ f_{01} \ \dots \ f_{0 \ N-1} \ f_{10} \ f_{11} \ \dots \ f_{1 \ N-1} \ \dots \ f_{M-1 \ 0} \ f_{M-1 \ 1} \ \dots \ f_{M-1 \ N-1}]$$

The filter matrix  $\mathbf{H}$  is obtained by constructing a matrix  $H_j$  for each row  $j$  of  $h_{ij}$ :

$$H_j = \begin{bmatrix} h_{j \ 0} & h_{j \ N-1} & h_{j \ N-2} & \dots & h_{j \ 1} \\ h_{j \ 1} & h_{j \ 0} & h_{j \ N-1} & \dots & h_{j \ 2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{j \ N-1} & h_{j \ N-2} & h_{j \ N-3} & \dots & h_{j \ 0} \end{bmatrix}$$

$$\mathbf{H} = \begin{bmatrix} H_0 & H_{M-1} & H_{M-2} & \dots & H_1 \\ H_1 & H_0 & H_{M-1} & \dots & H_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ H_{M-1} & H_{M-2} & H_{M-3} & \dots & H_0 \end{bmatrix}$$

29

## Avoiding Wrap-around Errors

Wrap-around errors result from filter responses due to the periodic continuation of image and filter.

To avoid wrap-around errors, image and filter have to be extended by zeros.

A x B    original image size  
 C x D    original filter size  
 M x N    extended image and filter size

To avoid wrap-around error:

$$M \geq A + C - 1$$

$$N \geq B + D - 1$$

Example:



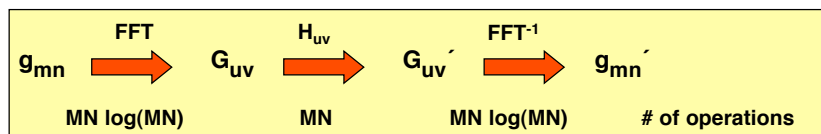
30

## Convolution Using the FFT

Convolution in the spatial domain may be performed more efficiently using the FFT.

$$g'_{ij} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} g_{mn} h_{i-m, j-n} \quad (MN)^2 \text{ operations needed}$$

Using the FFT and filtering in the frequency domain:



Example with  $M = N = 512$ :

- straight convolution needs  $\sim 10^{10}$  operations
- convolution using the FFT needs  $\sim 10^7$  operations

31

## Convolution and Correlation

The crosscorrelation function of 2 stationary stochastic processes  $f$  and  $h$  is:

$$g(x, y) = \iint_{-\infty}^{\infty} f(r, s) h(r - x, s - y) dr ds = f(x, y) \circ h(x, y) = f(x, y) * h(-x, -y)$$

Compare with convolution: filter function is not mirrored!

Correlation using Fourier Transform:

$$F\{f(x, y) \circ h(x, y)\} = F^*(u, v) H(u, v) \quad F^*, f^* \text{ are complex conjugates}$$

$$F\{f^*(x, y) h(x, y)\} = F(u, v) \circ H(u, v)$$

Correlation is particularly important for matching problems, e.g. matching an image with a template.

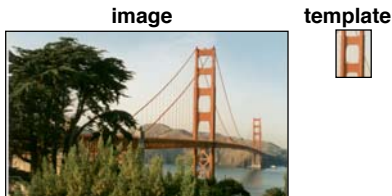
Correlation may be computed more efficiently by using the FFT.

32



## Correlation and Matching

Matching a template with an image:



- find degree of match for all locations of template
- find location of best match

For (periodic) discrete images, crosscorrelation at  $(i, j)$  is

$$c_{ij} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_{mn} h_{m-i, n-j}$$

Compare with Euclidean distance between  $f$  and  $h$  at location  $(i, j)$ :

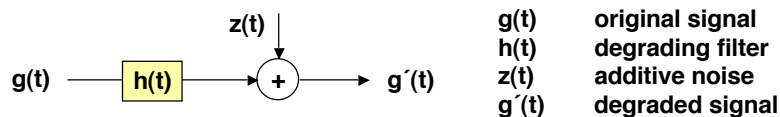
$$d_{ij} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} (f_{mn} - h_{m-i, n-j})^2 = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} (f_{mn})^2 - 2 \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_{mn} h_{m-i, n-j} + \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} (h_{m-i, n-j})^2$$

Since image energy and template energy are constant, correlation measures distance

33

## Principle of Image Restoration

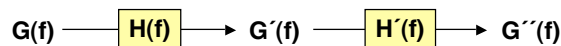
Typical degradation model of a continuous 1-dimensional signal:



How can one process  $g'(t)$  to obtain a  $g''(t)$  which best approximates  $g(t)$ ?



Note that a perfect restoration  $g''(t) = g(t)$  may not be possible even if  $z(t) = 0$ .



The ideal restoring filter  $H'(f) = 1/H(f)$  may not exist because of zeros of  $H(f)$ .

34

## Image Restoration by Minimizing the MSE

Degradation in matrix notation:  $\mathbf{g}' = \mathbf{H} \mathbf{g} + \mathbf{z}$

Restored signal  $\mathbf{g}''$  must minimize the mean square error  $J(\mathbf{g}'')$  of the remaining difference:

$$\min \|\mathbf{g}' - \mathbf{H}\mathbf{g}''\|^2$$

$$\delta J(\mathbf{g}'') / \delta \mathbf{g}'' = 0 = -2\mathbf{H}^T(\mathbf{g}' - \mathbf{H}\mathbf{g}'')$$

$$\mathbf{g}'' = \underbrace{(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T}_{\text{pseudoinverse of H}} \mathbf{g}'$$

If  $M = N$  and hence  $\mathbf{H}$  is a square matrix, and if  $\mathbf{H}^{-1}$  exists, we can simplify:

$$\mathbf{g}'' = \mathbf{H}^{-1}\mathbf{g}'$$

The matrix  $\mathbf{H}^{-1}$  gives a perfect restoration if  $\mathbf{z} = \mathbf{0}$ .