Karlsruhe Institute of Technology

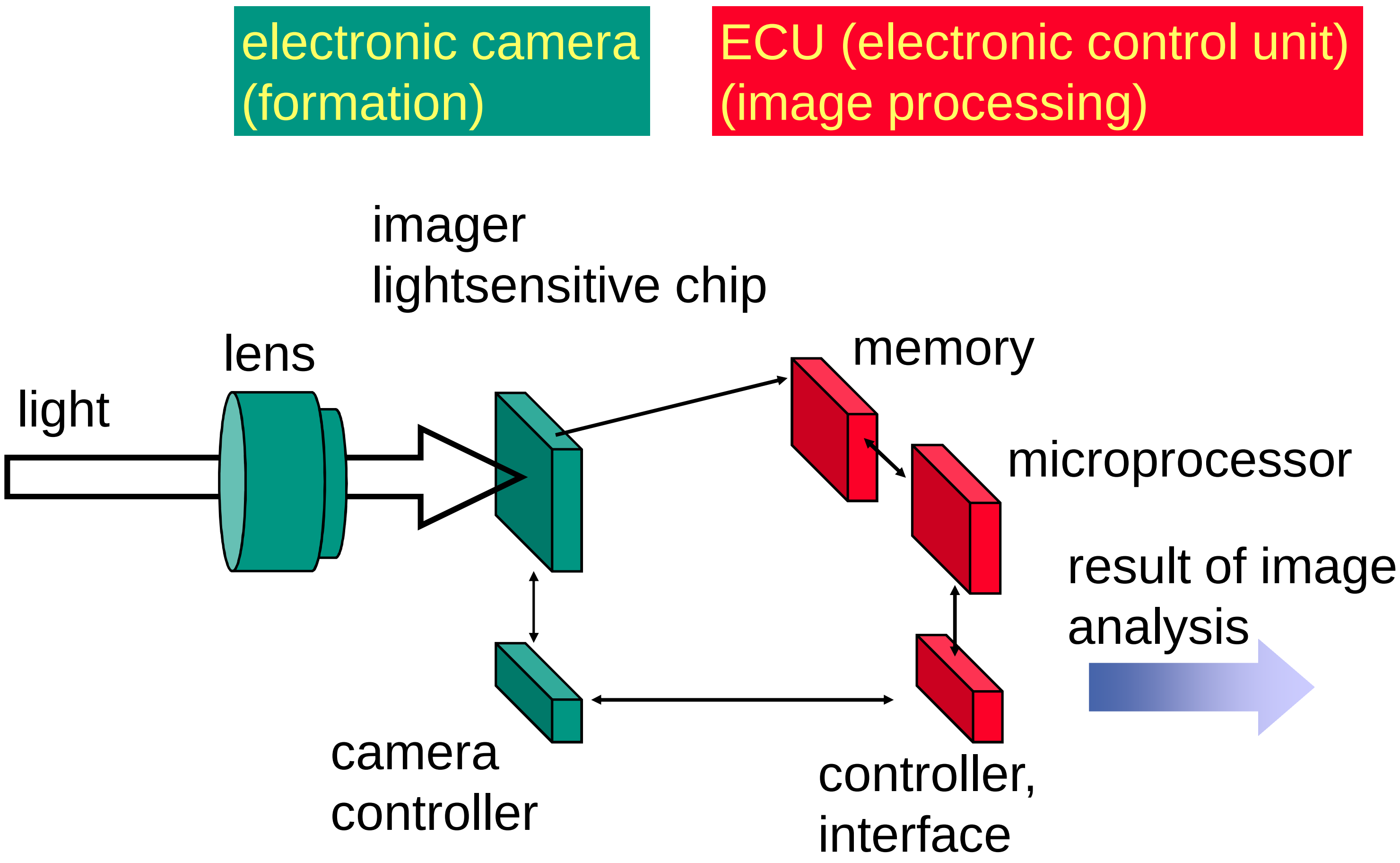
Machine Vision

Chapter 2: Image Preprocessing

Dr. Martin Lauer

Institut für Mess-
und Regelungstechnik

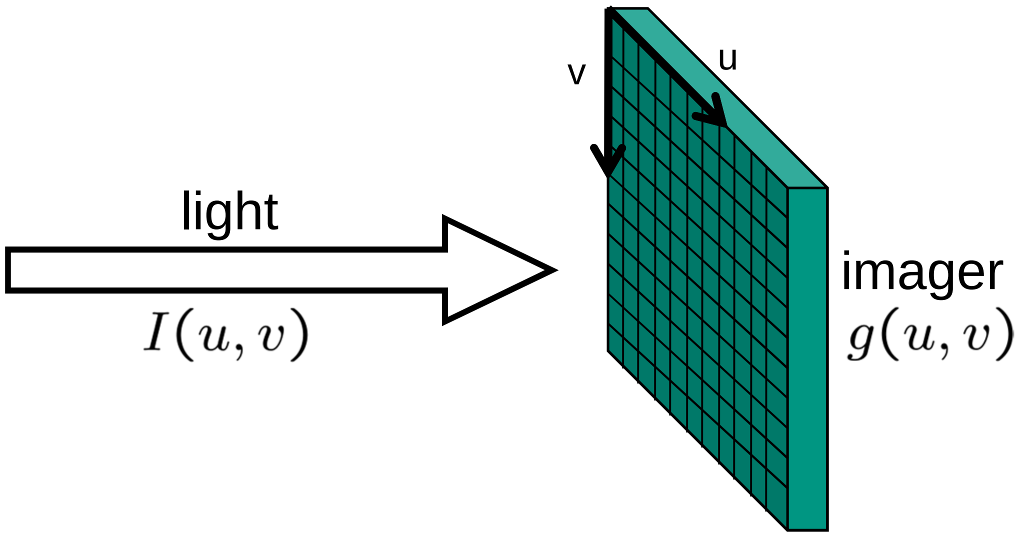
Image Formation and Analysis



Imager

- Process of image formation:
  - incident light intensity:
    $I : \mathbb{R}^2 \rightarrow \mathbb{R}$
  - output of imager:
    $g : \{0, \ldots, w-1\} \times \{0, \ldots, h-1\} \rightarrow \{0, \ldots, g_{max}\}$

    w, h : image width, height

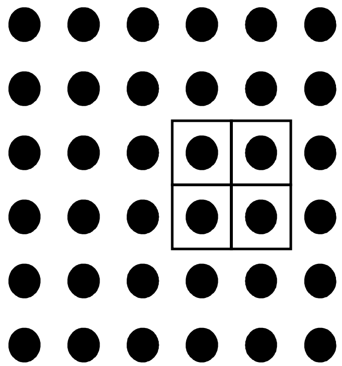


Imager

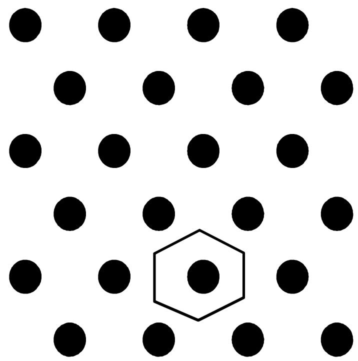
- Process of image formation:
  - **sampling**
    evaluate light intensity on a regular grid of points
  - **quantization**
    map continuous signals to discrete values (natural numbers)
  - **blur and noise**
  - **color**
    will be discussed later. Here: only light intensity/grey level images

Sampling

- 2D grids used for sampling



rectangular



hexagonal

- electronic cameras: rectangular, equidistant grids
- biology: hexagonal grids with varying resolution

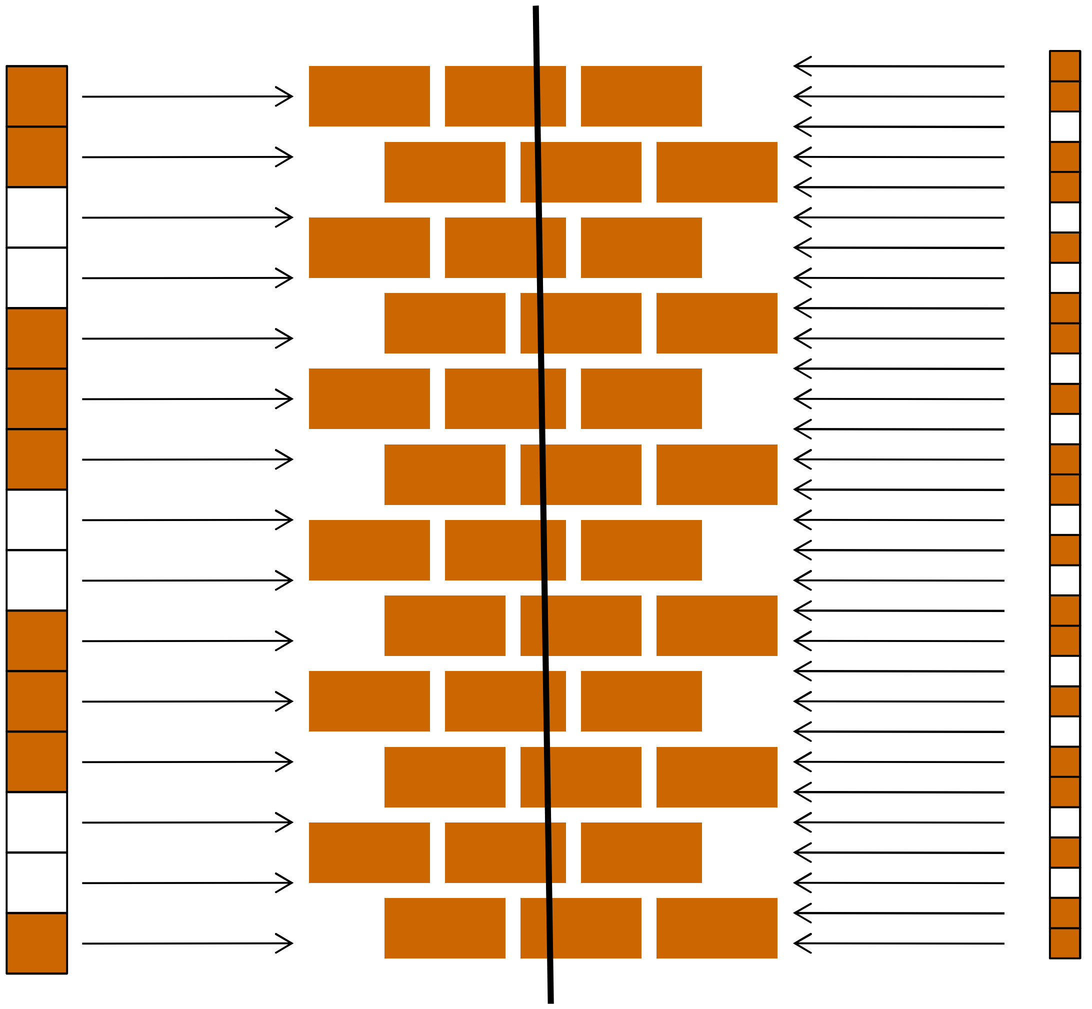
Sampling: Moiré Patterns

- Moiré patterns
  - sampling might cause artifacts

original picture

picture with Moiré pattern

Sampling: Moiré Patterns



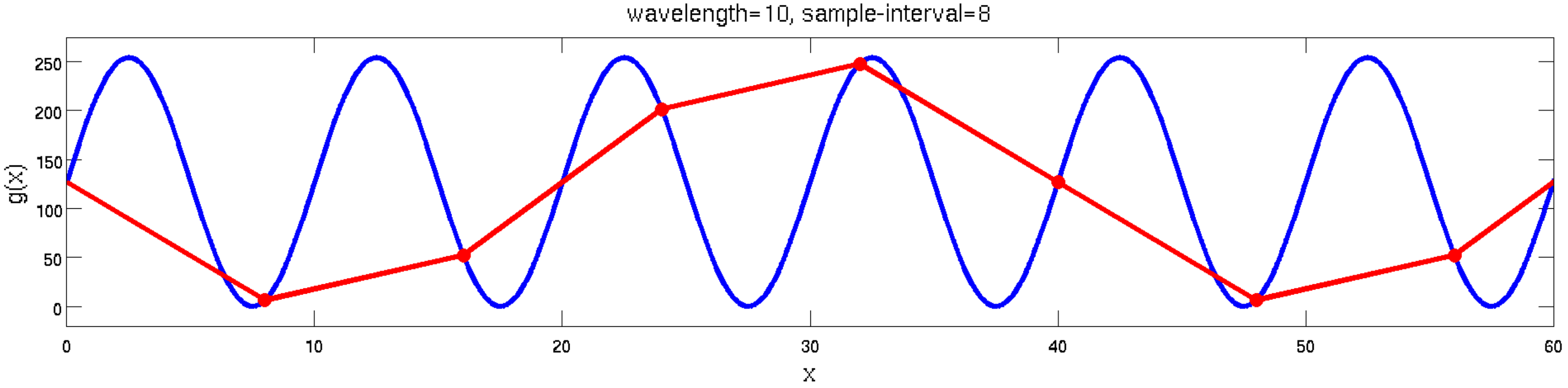
sparse sampling → 4 layers

wall of bricks, 12 layers

dense sampling → 12 layers

Sampling: Moiré Patterns cont.

- 1D-example of Moiré patterns:



The occurrence of Moiré patterns depends on the sampling rate compared to the maximal frequency of the signal (image)

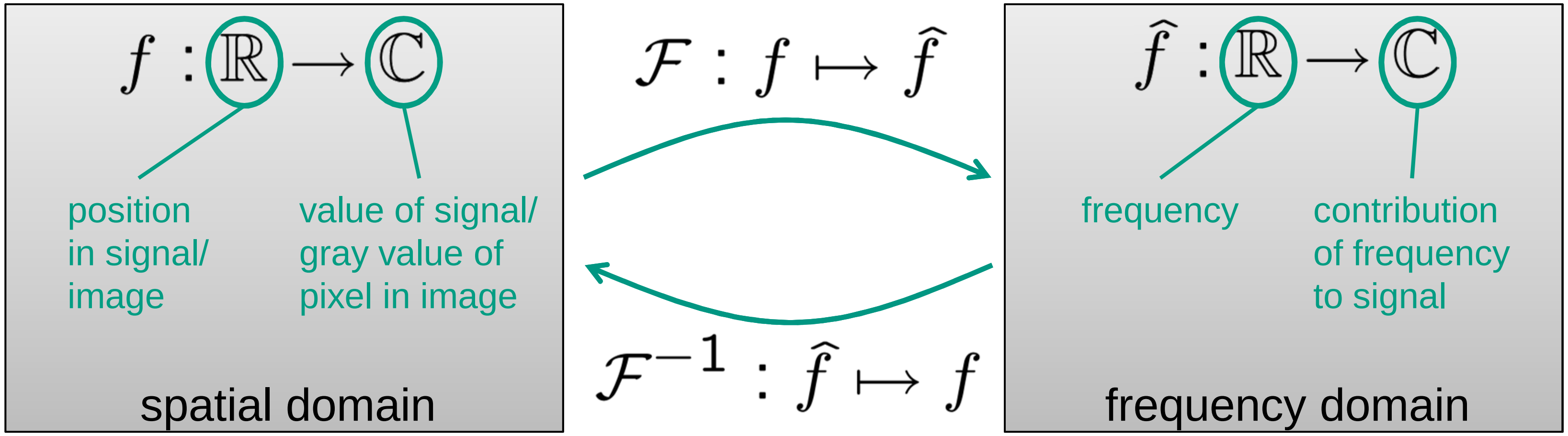
Nyquist-Shannon Sampling Theorem

If f is band bounded signal with cutoff frequency k_0 then it is completely determined by giving its ordinates at a series of points spaced at most $\frac{1}{2k_0}$,i.e. the sample frequency must be larger than $2k_0$

- Questions:
  - what is a band-bounded signal?
  - what is a cutoff frequency?

Fourier Transform

- Assume a periodic signal
  $f : \mathbb{R} \to \mathbb{C}$
- Then, we can define the Fourier transform of f



$$\widehat{f}(k) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i k x} dx$$

$$f(x) = \int_{-\infty}^{\infty} \widehat{f}(k) e^{2\pi i k x} dk$$

Fourier Transform

- Properties:
  - the Fourier transform is linear

$$\mathcal{F}\Big\{\alpha f(x) + \beta g(x)\Big\}(k) = \alpha \hat{f}(k) + \beta \hat{g}(k)$$

  - shifting a signal along the x-axis only changes the complex angles in frequency domain but not the amplitudes

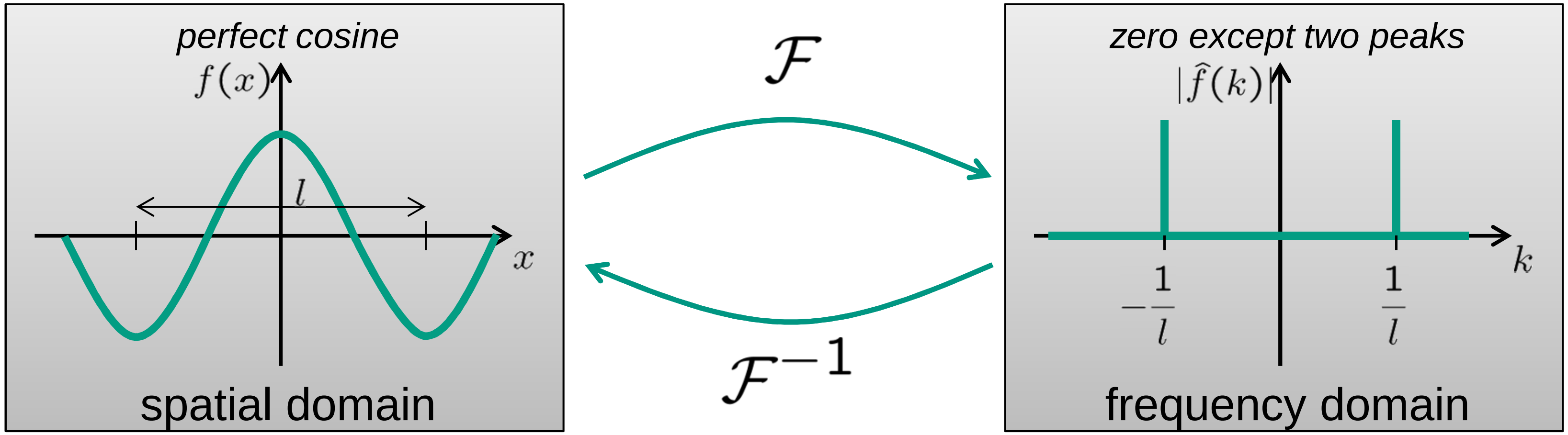
$$\mathcal{F}\Big\{f(x-\xi)\Big\}(k) = e^{-2\pi i \xi k} \hat{f}(k)$$

  - rescaling the x-axis in the spatial domain rescales the frequency axis in a reciprocal way

$$\mathcal{F}\Big\{f(\alpha x)\Big\}(k) = \frac{1}{|\alpha|} \hat{f}(\frac{k}{\alpha})$$

Fourier Transform

- Properties:
  - a cosine in spatial domain generates two peaks in frequency domain



  - the peaks are located at position reciprocal to the period length
  - if the signal in spatial domain is a linear combination of cosines, the Fourier transform will be a set of peaks in frequency domain
  - intuitive interpretation: the Fourier transform decomposes a periodic signal into a (potentially infinite) linear combination of cosines

Fourier Transform

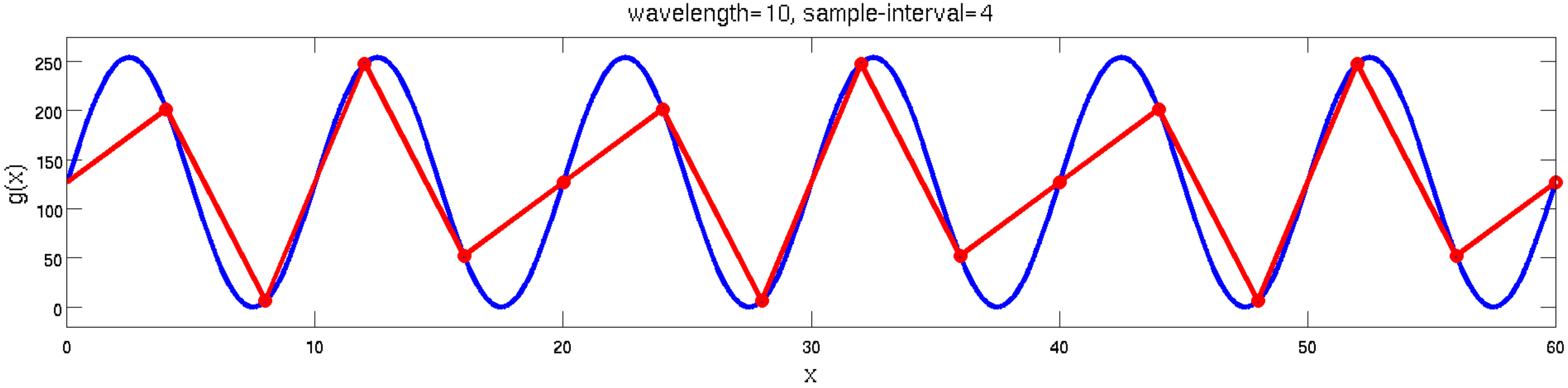
- Observation
  - smooth periodic functions with small slope can be composed out of cosines with large period
  - periodic functions with large slope require cosines with small period
  - periodic functions that are discontinuous or have discontinuous derivatives require cosines with unbounded frequencies

- Definition

  A signal f is band bounded with cutoff frequency k_0 if its Fourier transform is zero for all frequencies larger than the cutoff frequency, i.e.

$$\hat{f}(k) = 0 \text{ for all } k \text{ with } |k| \geq k_0$$

Nyquist-Shannon Sampling Theorem



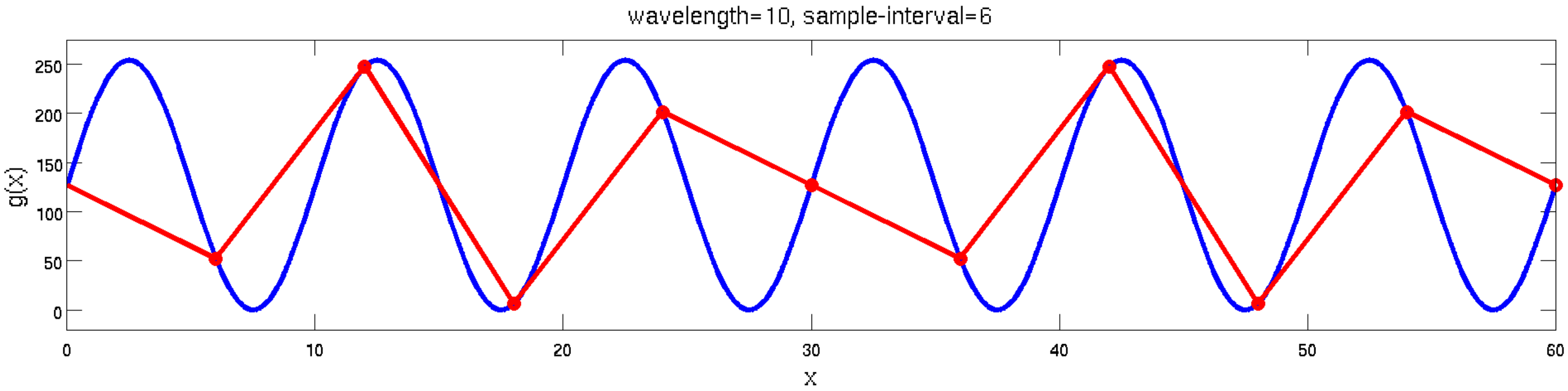
- signal is band bounded (sine function)
- sampling frequency high enough

$$f_{sample} = \frac{1}{4} > 2f_{signal} = \frac{2}{10}$$

- reconstruction of the signal possible

If f is band bounded signal with cutoff frequency k_0 then it is completely determined by giving its ordinates at a series of points spaced at most $\frac{1}{2k_0}$,i.e. the sample frequency must be larger than $2k_0$

Nyquist-Shannon Sampling Theorem



- signal is band bounded (sine function)
- but

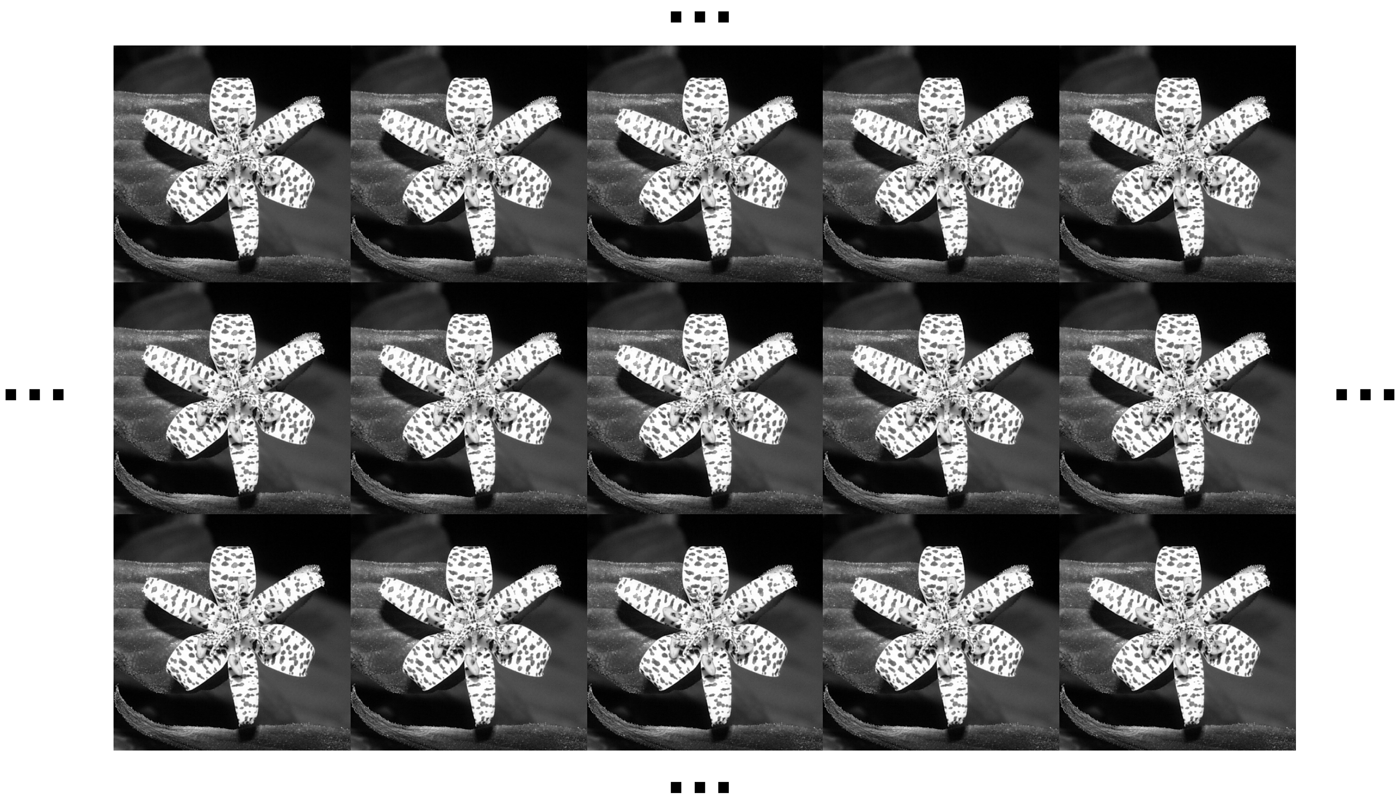
$$f_{sample} = \frac{1}{6} < 2f_{signal} = \frac{2}{10}$$

- reconstruction of the signal impossible

> If f is band bounded signal with cutoff frequency k_0 then it is completely determined by giving its ordinates at a series of points spaced at most $\frac{1}{2k_0}$,i.e. the sample frequency must be larger than $2k_0$

Sampling Theorem and Images

- Remarks:
  - analysis analogously possible for 2d signals
  - image is not periodic, but we can make it periodic by copying it repeatedly to the left, right, top, and bottom



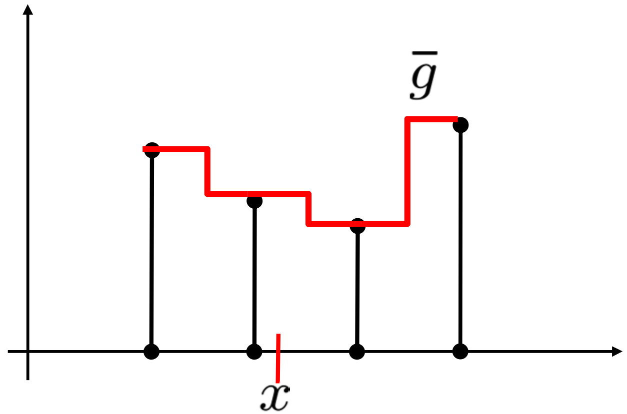
Sampling Theorem and Images

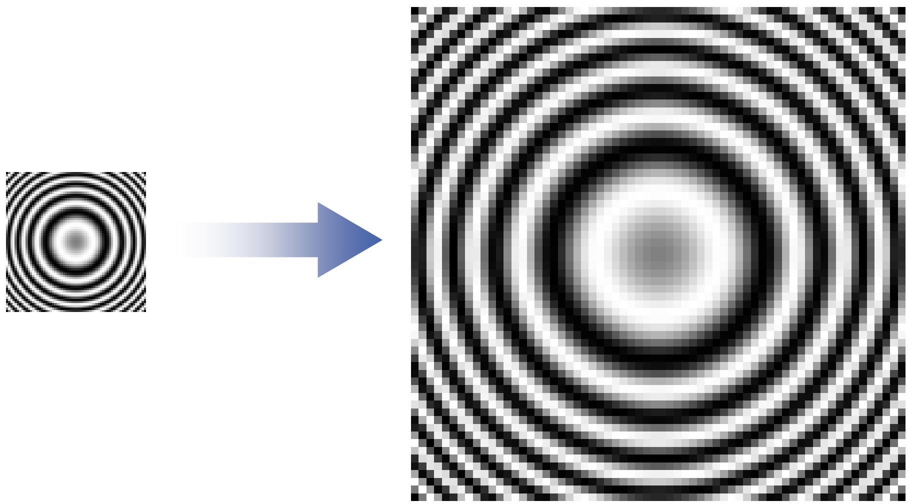
- Questions:
  - how can we determine the sampling frequency of a camera?
  - what can we do if we find that the sampling theorem is violated?

Image Scaling and Interpolation

- changing the image size
- scaling needs evaluation of the image at non-integer positions → interpolation

- nearest neighbor interpolation:
  - approximating the grey level function with a step function
  - take the grey value of the nearest integer position
  - problem: aliasing

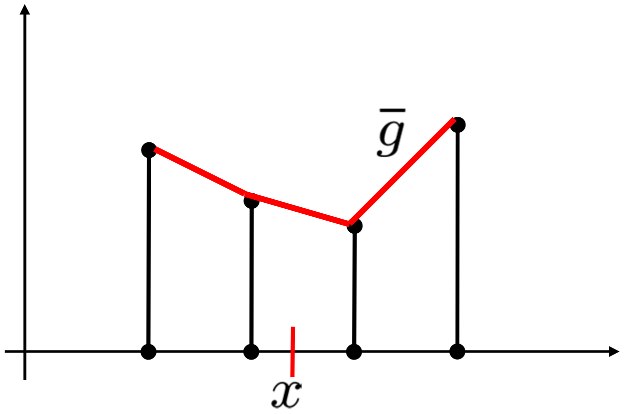




Interpolation cont.

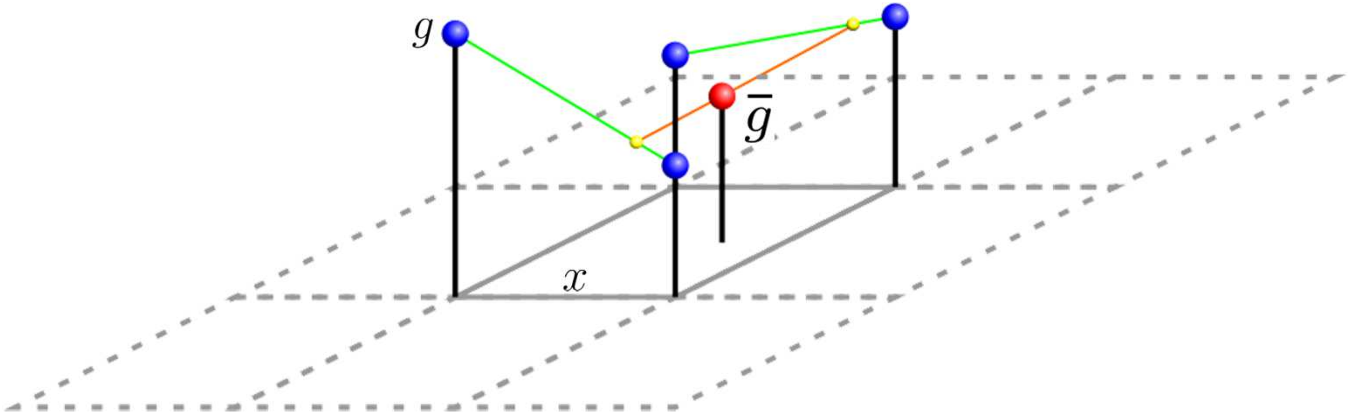
- linear interpolation in ID
  - fit linear function locally around x

$$\bar{g}(x) = g(\lfloor x \rfloor) + (x - \lfloor x \rfloor)(g(\lfloor x \rfloor + 1) - g(\lfloor x \rfloor))$$



Interpolation cont.

- extension of linear interpolation to 2D:



  – interpolate from 4 neighboring pixels

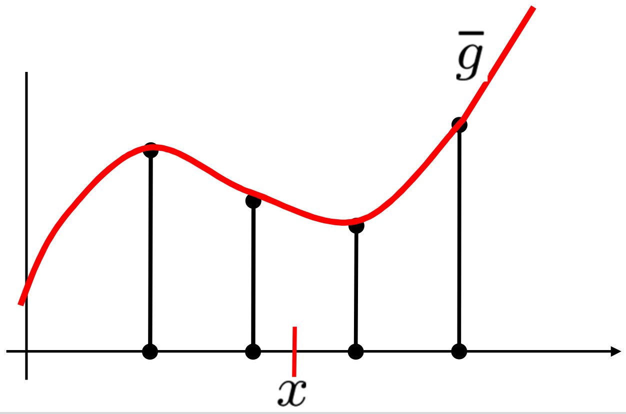
Interpolation cont.

- cubic interpolation
  - fit cubic polynomial to the grey level
  - solve

$$\bar{g}(x) = a \cdot (x - \lfloor x \rfloor)^3 + b \cdot (x - \lfloor x \rfloor)^2 + c \cdot (x - \lfloor x \rfloor) + d$$

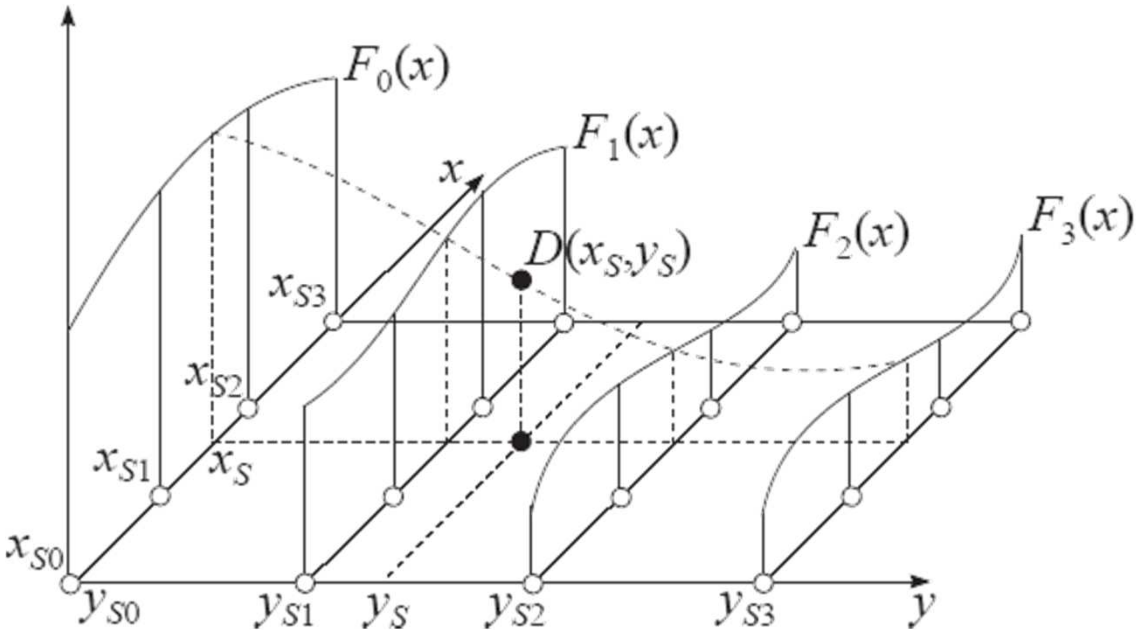
yields:

$$a = -\frac{1}{6} g(\lfloor x \rfloor - 1) + \frac{1}{2} g(\lfloor x \rfloor) - \frac{1}{2} g(\lfloor x \rfloor + 1) + \frac{1}{6} g(\lfloor x \rfloor + 2)$$
$$b = \frac{1}{2} g(\lfloor x \rfloor - 1) - g(\lfloor x \rfloor) + \frac{1}{2} g(\lfloor x \rfloor + 1)$$
$$c = -\frac{1}{3} g(\lfloor x \rfloor - 1) - \frac{1}{2} g(\lfloor x \rfloor) + g(\lfloor x \rfloor + 1) - \frac{1}{6} g(\lfloor x \rfloor + 2)$$
$$d = g(\lfloor x \rfloor)$$



Interpolation cont.

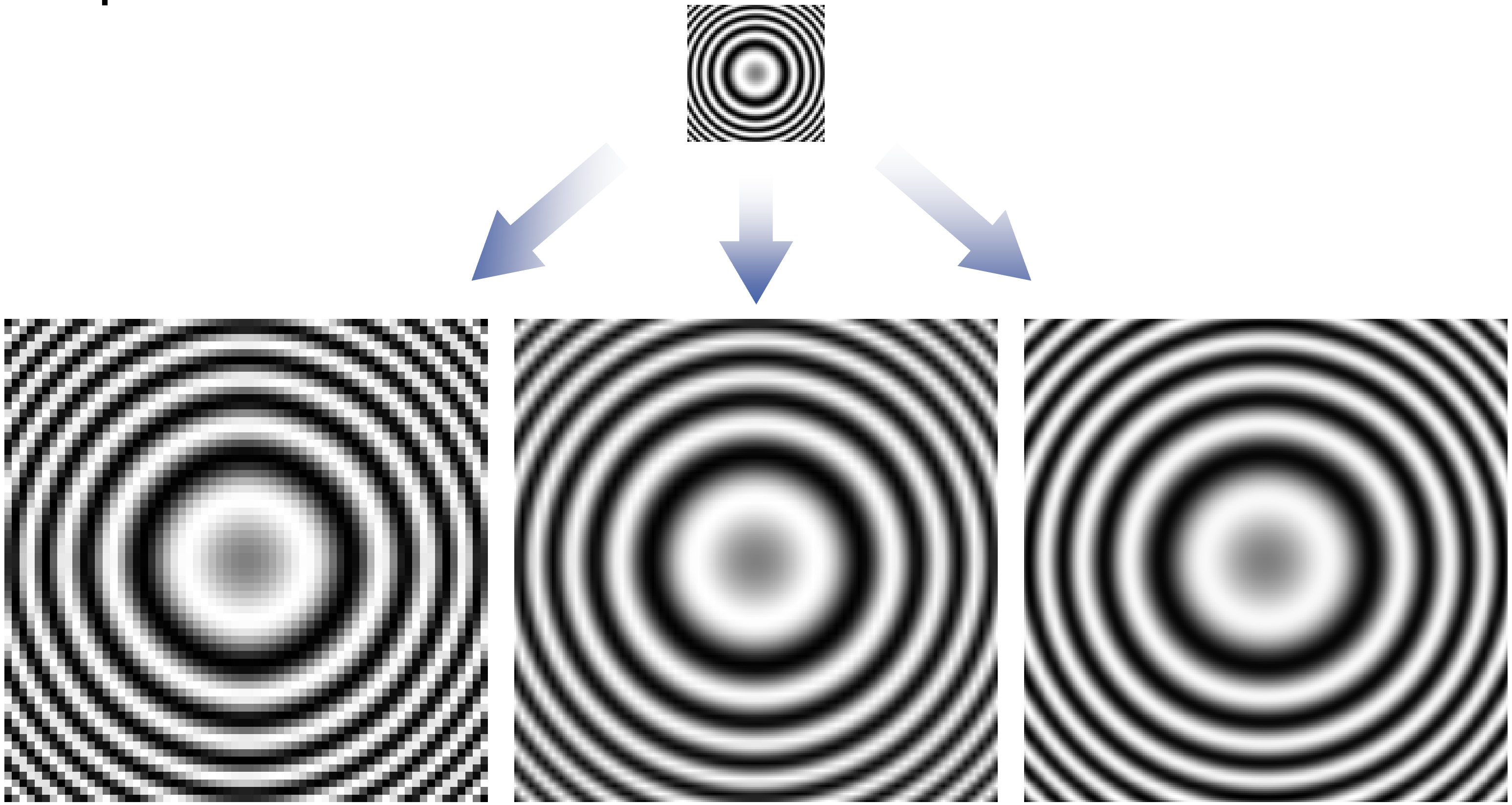
- extension of cubic interpolation to 2D:



  - interpolation from 16 neighboring pixels

Interpolation cont.

- Example:



Imager

- Process of image formation:
  - **sampling**
    evaluate light intensity on a regular grid of points
  - **quantization**
    map continuous signals to discrete values (natural numbers)
  - **blur and noise**
  - **color**
    will be discussed later. Here: only light intensity/grey level images

Quantization

- incident light:

$$I : \mathbb{R}^2 \rightarrow \mathbb{R}$$

- digital camera signals:

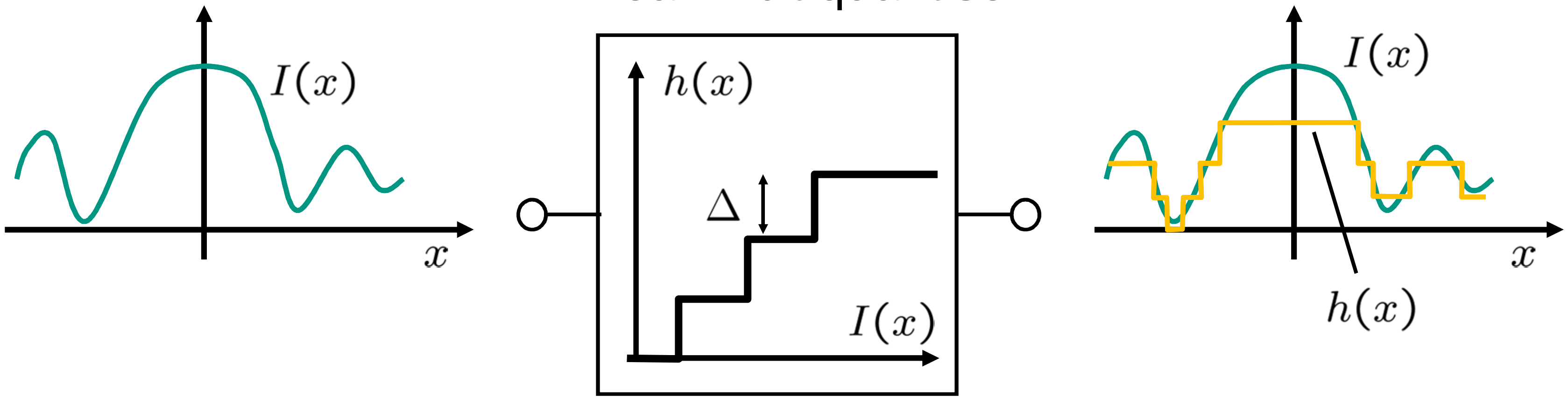
$$g : \{0, \ldots, w-1\} \times \{0, \ldots, h-1\} \rightarrow \{0, \ldots, g_{max}\}$$

$w, h:$ image width, height

- need transformation from real valued light intensity to discrete digital signals (analog-to-digital converter)

Quantization cont.

linear 2-bit quantiser



- characteristic with equidistant steps ("linear") of size Δ:

$$g(x) = \max\{0, \min\{g_{max}, \left\lfloor \frac{I(x)}{\Delta} + \frac{1}{2} \right\rfloor\}\}$$

$$h(x) = \Delta g(x)$$

- error of non-overdriven quantiser:

$$I(x) - h(x) \in [-\frac{\Delta}{2}, \frac{\Delta}{2}]$$

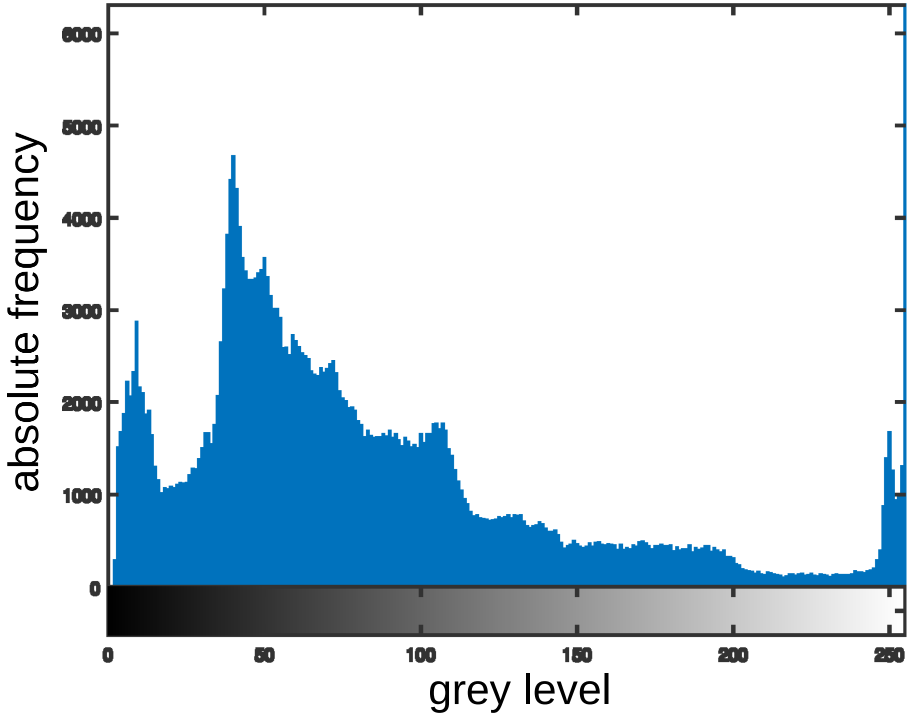
Quantization cont.

- characteristic of digital cameras:
  - linear
  - logarithmic
- grey level cutoff value
  - 1 (binary images, “bitmaps”) → 1 bit/pixel
  - 255 → 8 bit/pixel = 1 byte/pixel
  - 4095 → 12 bit/pixel = 1.5 byte/pixel
  - 65535 → 16 bit/pixel = 2 byte/pixel
- correction of grey level distribution
  - image too dark/too bright
  - low contrast
  - non-linear camera characteristic

linear characteristic

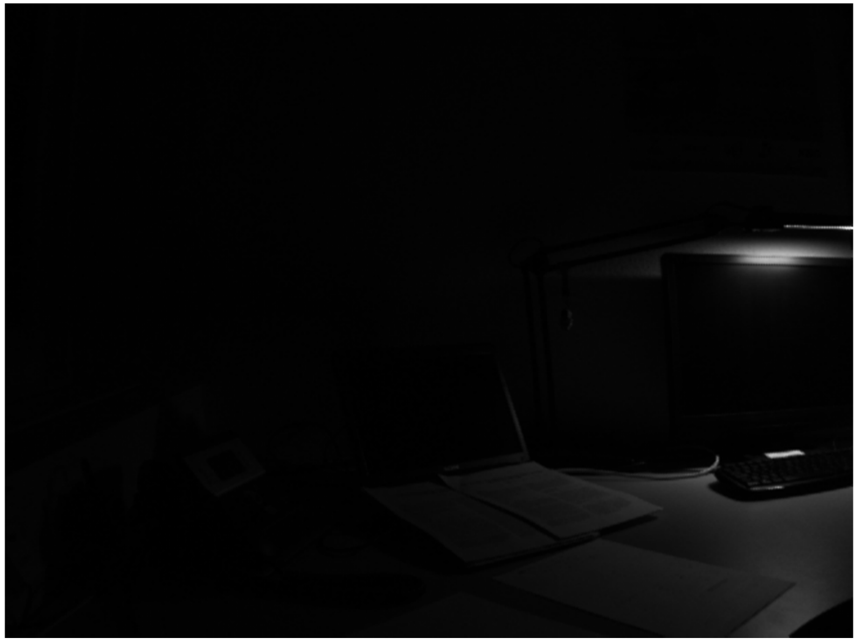
logarithmic characteristic

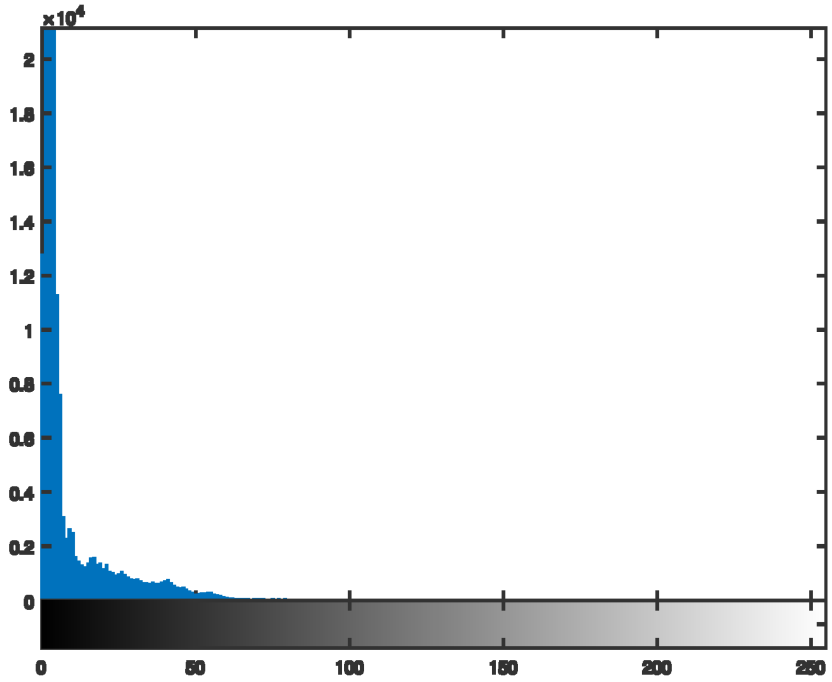
Grey Level Histogram



- grey level histograms display distribution of grey levels

Grey Level Histogram cont.



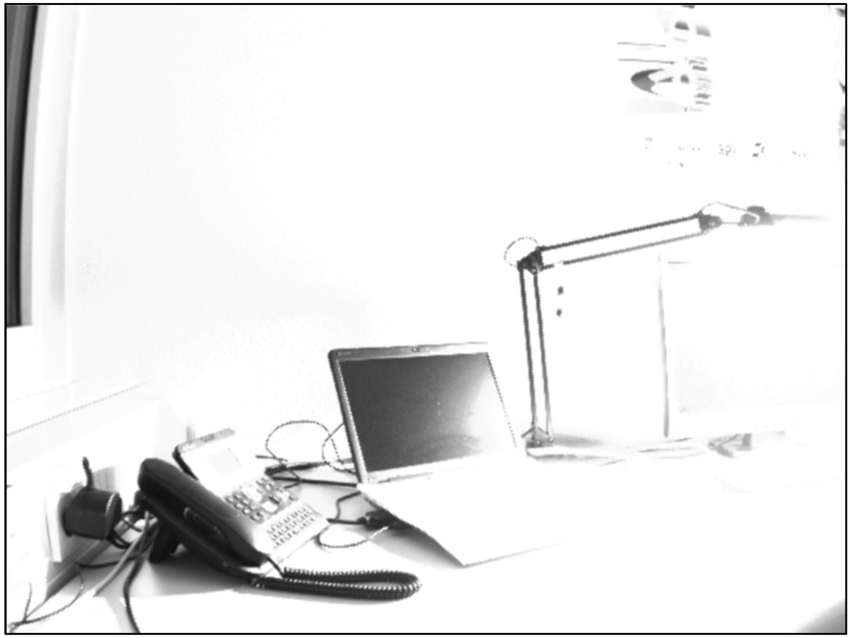


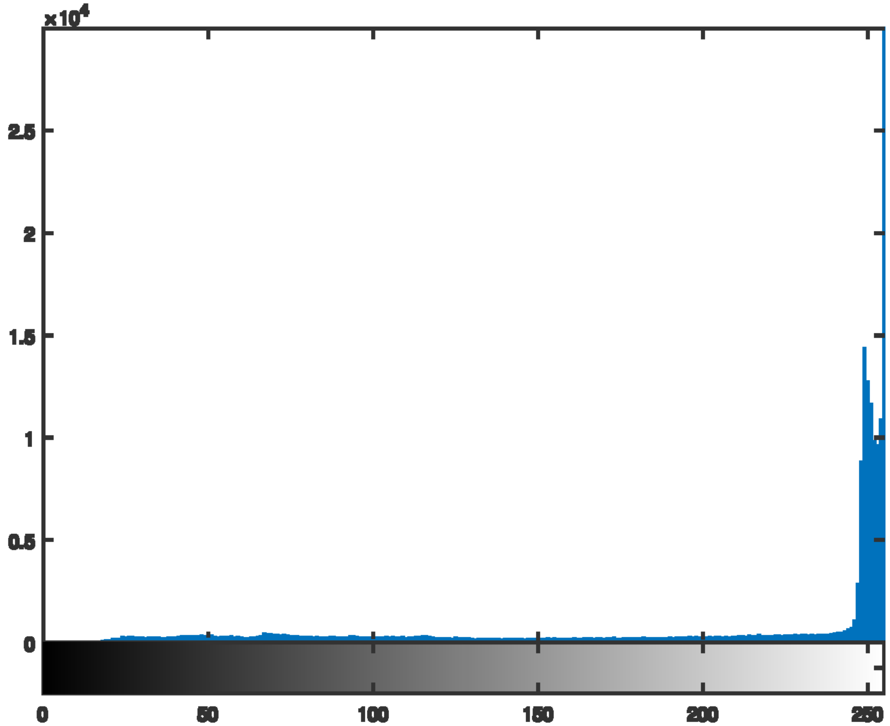
Underexposed images:

- open aperture of camera
- increase exposure time of camera
- increase gain
- add additional light sources

- multiply grey values by a constant
- auto-exposure implemented in many digital cameras

Grey Level Histogram cont.





Overexposed images:

- information loss due to cutoff value, no reconstruction possible
- close aperture of camera
- reduce exposure time of camera
- auto-exposure

Grey Level Histogram cont.

$\gamma = 1$

$\gamma = 0.5$

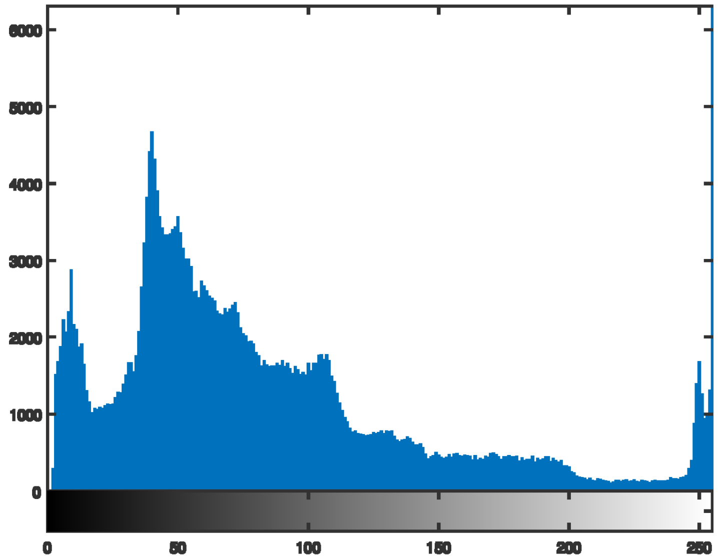
$\gamma = 2$

Gamma correction:

$$g_{out} = g_{max} \left(\frac{g_{in}}{g_{max}} \right)^{\gamma}$$

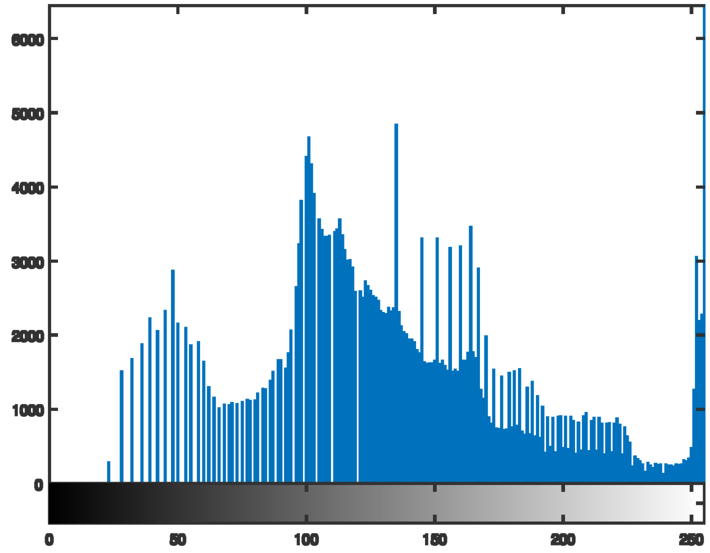
- keeps black and white
- nonlinear transformation

Grey Level Histogram cont.

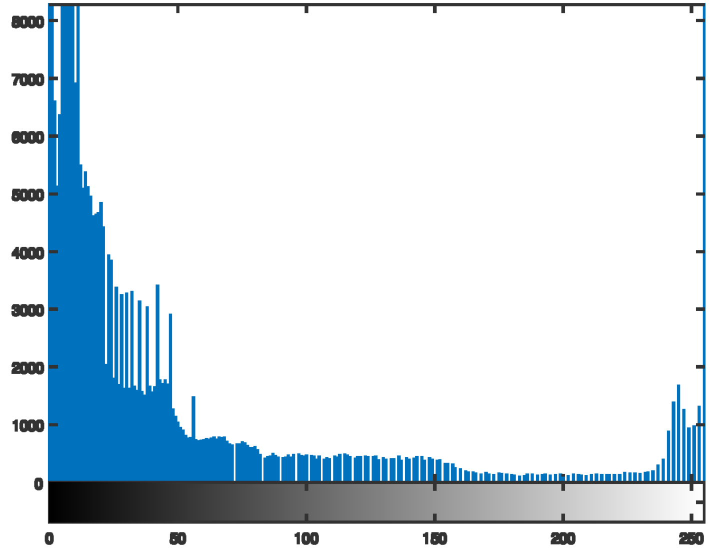


$\gamma = 1$

$\gamma = 0.5$



$\gamma = 2$

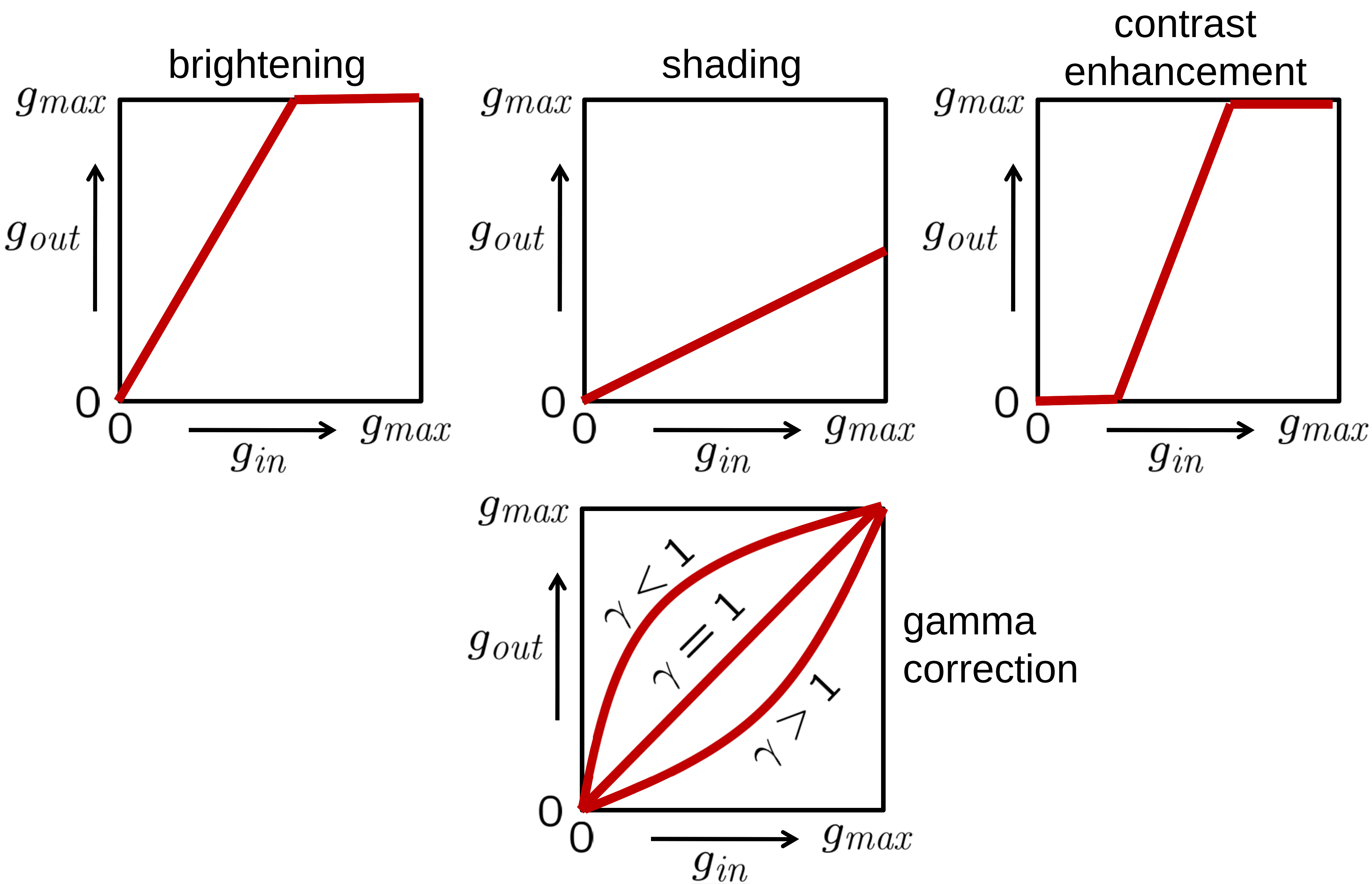


Gamma correction:

$$g_{out} = g_{max} \left(\frac{g_{in}}{g_{max}} \right)^{\gamma}$$

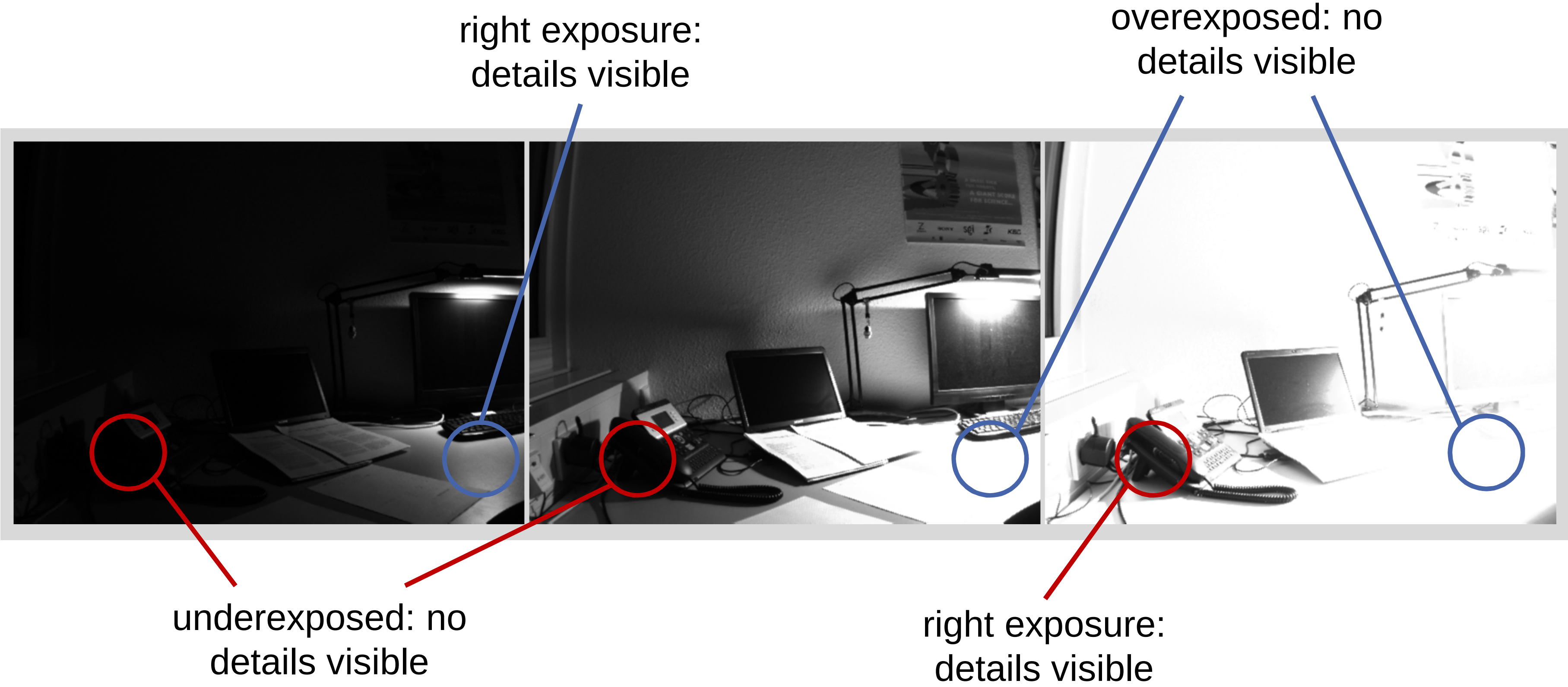
- keeps black and white
- nonlinear transformation

Grey Level Transformations

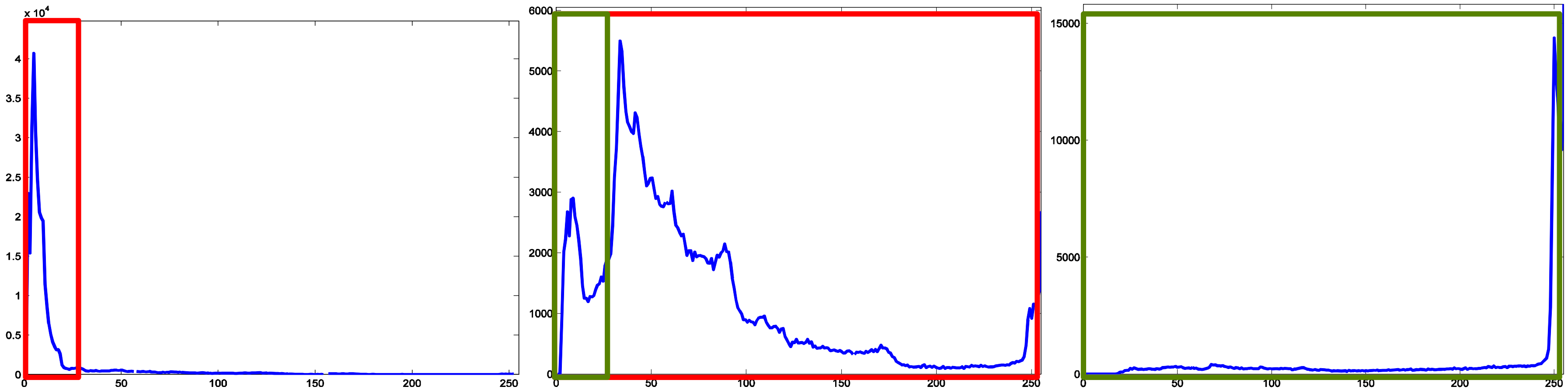


Exposure Series

– exposure bracketing, high dynamic range imaging (HDRI):
  increase the grey value resolution combining over- and underexposed images



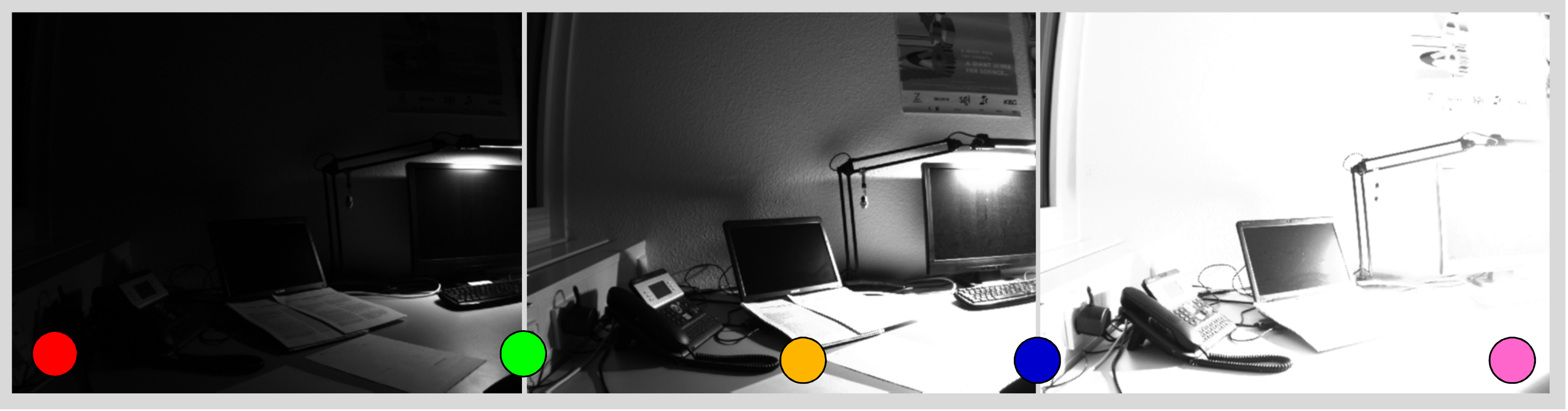
Exposure Series cont.



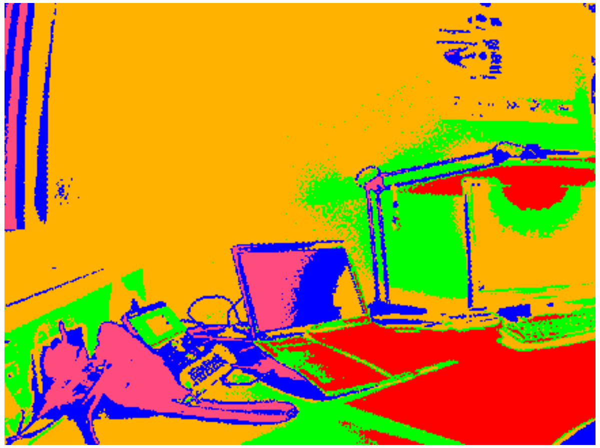
gray level histograms

gray scales differ by a constant factor

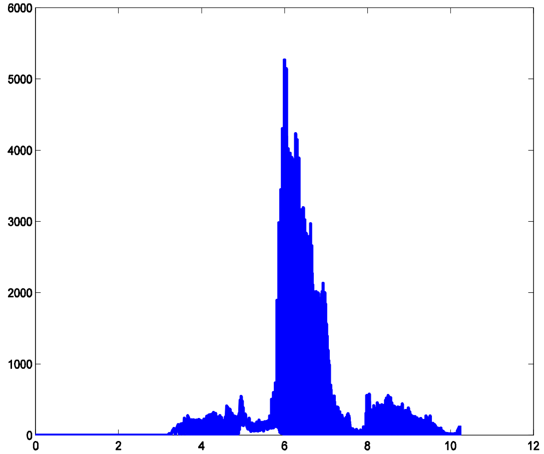
Exposure Series cont.



HDRI image (after log transform)



HDRI mixing



HDRI histogram
(after log transform)

Imager

- Process of image formation:
  - **sampling**
    evaluate light intensity on a regular grid of points
  - **quantization**
    map continuous signals to discrete values (natural numbers)
  - **blur and noise**
  - **color**
    will be discussed later. Here: only light intensity/grey level images

Convolution Operator

The convolution operator

- takes two functions f, g
- creates a new function $h = g * f$
- which is defined pointwise by

$$h(x) = \int_{-\infty}^{\infty} f(\tau) g(x - \tau) d\tau$$

- we interpret
  - g is a gray level image
  - f is a filter function
  - h is a filtered image

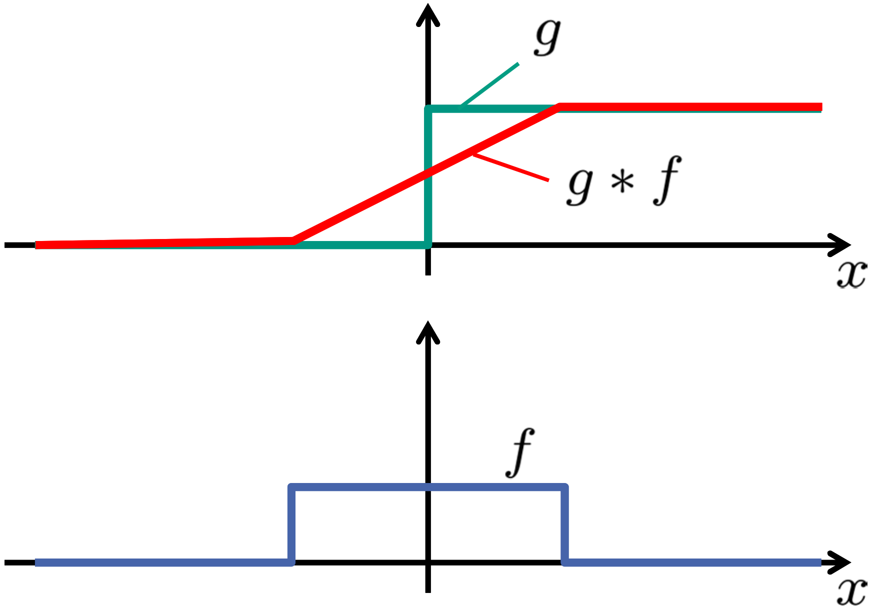
- convolution implements a linear filter

Convolution Operator

- Example

$$g(x) = \begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}$$

$$f(x) = \begin{cases} 0 & \text{if } x < -1 \\ \frac{1}{2} & \text{if } -1 \leq x \leq +1 \\ 0 & \text{if } x > +1 \end{cases}$$



$$(g * f)(x) = \int_{-\infty}^{\infty} f(\tau) g(x - \tau) d\tau = \int_{-\infty}^{x} f(\tau) \cdot 1 d\tau + \int_{x}^{\infty} f(\tau) \cdot 0 d\tau$$

$$= \int_{-\infty}^{x} f(\tau) d\tau = \begin{cases} 0 & \text{if } x < -1 \\ \frac{1}{2}(x+1) & \text{if } -1 \leq x \leq +1 \\ 1 & \text{if } x > +1 \end{cases}$$

Convolution Operator

- Properties of convolution
  - commutativity

$$f * g = g * f$$

  - associativity

$$(f * g) * h = f * (g * h)$$

  - linearity

$$f * (\alpha g + \beta h) = \alpha(f * g) + \beta(f * h)$$

  - relationship with Fourier transform

$$\mathcal{F}(f * g) = \mathcal{F}(f) \cdot \mathcal{F}(g)$$

$$\mathcal{F}(f \cdot g) = \mathcal{F}(f) * \mathcal{F}(g)$$

Convolution of Images

- Convolution can be extended
  - to the 2d case

$$f, g : \mathbb{R}^2 \rightarrow \mathbb{R}$$
$$(g * f)(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\tau, \rho) g(x - \tau, y - \rho) d\tau d\rho$$

  - to the case of function which we can evaluate only at integer positions

$$f, g : \mathbb{Z} \rightarrow \mathbb{R}$$
$$(g * f)(u) = \sum_{k=-\infty}^{\infty} f(k) g(u - k)$$

$$f, g : \mathbb{Z}^2 \rightarrow \mathbb{R}$$
$$(g * f)(u, v) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f(k, l) g(u - k, v - l)$$

Convolution of Images

filter

$$(g * f)(u, v) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f(k, l) g(u - k, v - l)$$

gray level image

pixel to evaluate

problem: sums run to infinity!

– in practice, filters and images have limited size.
We assume that all gray levels outside of filter size are 0

Convolution of Images

- Example

g :

3	4	2	2	2	2	4
5	2	1	1	1	0	2
6	3	2	2	0	1	3
4	4	8	5	4	7	3
6	6	7	8	9	9	7

f :

2	2	1
1	0	0
0	1	3

$g * f$:

	30	26	15	12	18	
	47	39	30	32	30	
	61	52	53	57	45	

$(g * f)(5,1) =$
$+ f(-1,-1)g(6,2) + f(0,-1)g(5,2) + f(1,-1)g(4,2)$
$+ f(-1,0)g(6,1) + f(0,0)g(5,1) + f(1,0)g(4,1)$
$+ f(-1,1)g(6,0) + f(0,1)g(5,0) + f(1,1)g(4,0)$

boundary pixels are typically left free since convolution requires evaluation of pixels outside of image *g*

$$(g * f)(u,v) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f(k,l)g(u-k,v-l)$$

Blur and Noise

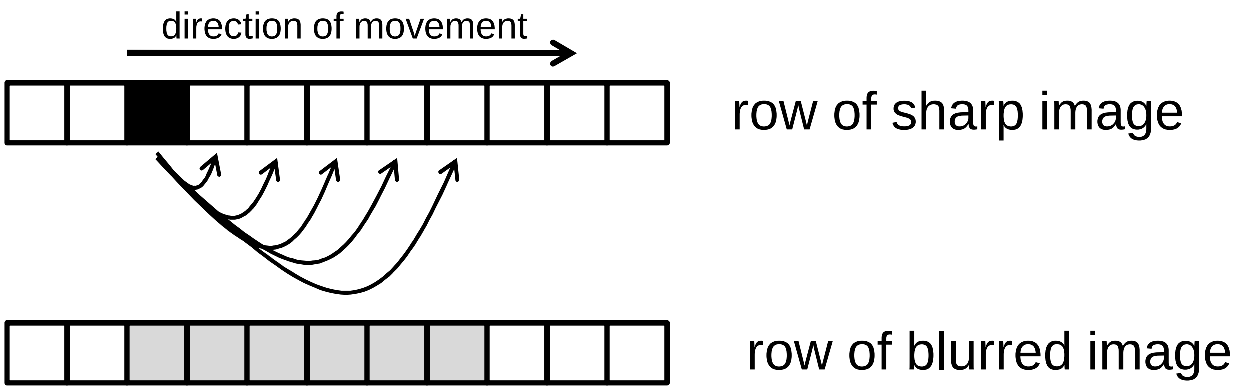
- types of blur and noise:
  - motion blur
  - defocus aberration
  - statistical noise of sensor cells and amplifiers
  - malfunctioning sensor cells



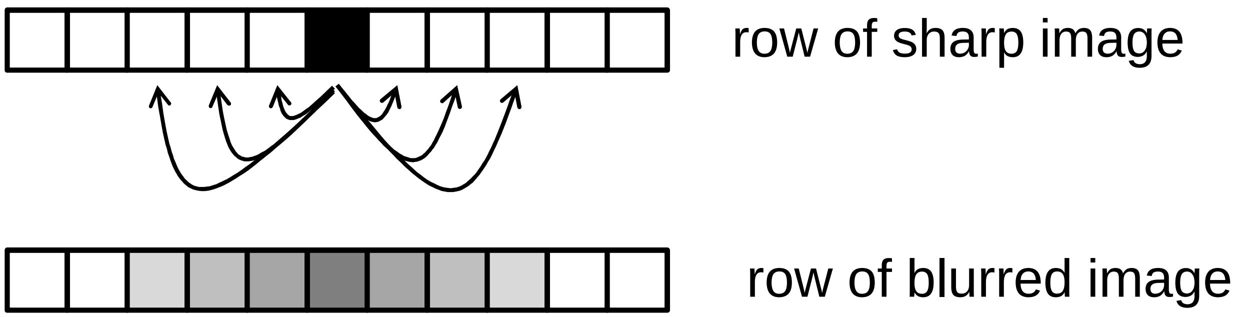
image source: wikipedia

Models of Blur

- Motion blur:



- Gaussian blur:

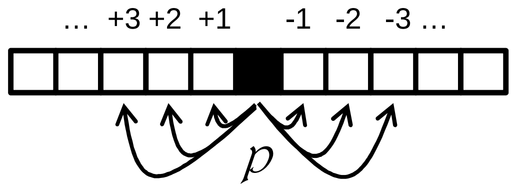


Models of Blur cont.

- blur can be modeled with convolution

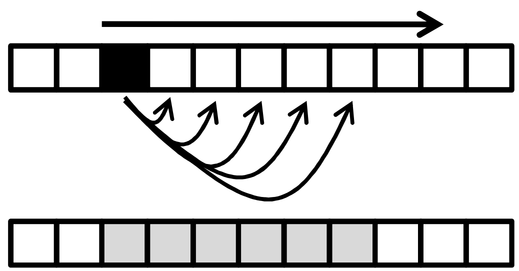
$$g_{blurred} = g_{sharp} * p$$

p : “point-spread-function” models blur



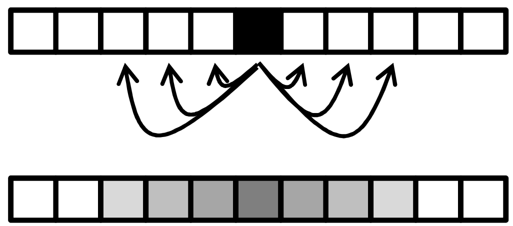
- – motion blur (along x-axis by n pixels):

$$p_{motion}(x) = \begin{cases} \frac{1}{n} & \text{if } -n < x \leq 0 \\ 0 & \text{otherwise} \end{cases}$$



- – Gaussian blur (with variance σ^2):

$$p_{Gauss}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\frac{x^2}{\sigma^2}}$$



Wiener Deconvolution

- techniques to obtain sharp image from blurred image based on Wiener filter

$$g_{blurred} = g_{sharp} * p + v$$

p : point-spread-function

v : pixel noise

assume g_{sharp} and v be independent

$$g_{restored} = f * g_{blurred}$$

find optimal f that minimizes:

$$e(k) = \mathbb{E}\left[|\hat{g}_{sharp}(k) - \hat{g}_{restored}(k)|^2\right]$$

($\hat{g}$ denotes Fourier transform of g)

($\mathbb{E}$ denotes expectation value)

Wiener Deconvolution cont.

$$
\begin{aligned}
e(k) =& \mathbb{E}\left[|\hat{g}_{sharp}(k) - \hat{g}_{restored}(k)|^2\right] \\
=& \mathbb{E}\left[|\hat{g}_{sharp}(k) - \hat{f}(k)\hat{g}_{blurred}(k)|^2\right] \\
=& \mathbb{E}\left[|\hat{g}_{sharp}(k) - \hat{f}(k)(\hat{p}(k)\hat{g}_{sharp}(k) + \hat{v}(k))|^2\right] \\
=& \mathbb{E}\left[|(1 - \hat{f}(k)\hat{p}(k))\hat{g}_{sharp}(k) - \hat{f}(k)\hat{v}(k)|^2\right] \\
=& (1 - \hat{f}(k)\hat{p}(k))(1 - \hat{f}(k)\hat{p}(k))^* \mathbb{E}\left[\hat{g}_{sharp}(k)\hat{g}^*_{sharp}(k)\right] \\
& - (1 - \hat{f}(k)\hat{p}(k))\hat{f}^*(k)\mathbb{E}\left[\hat{g}_{sharp}(k)\hat{v}^*(k)\right] \\
& - \hat{f}(k)(1 - \hat{f}(k)\hat{p}(k))^* \mathbb{E}\left[\hat{v}(k)\hat{g}^*_{sharp}(k)\right] \\
& + \hat{f}(k)\hat{f}^*(k)\mathbb{E}\left[\hat{v}(k)\hat{v}^*(k)\right]
\end{aligned}
$$

independence of signal and noise yields:

$$\mathbb{E}\left[\hat{g}_{sharp}(k)\hat{v}^*(k)\right] = \mathbb{E}\left[\hat{v}(k)\hat{g}^*_{sharp}(k)\right] = 0$$

denote:

$$S(k) = \mathbb{E}\left[\hat{g}_{sharp}(k)\hat{g}^*_{sharp}(k)\right], \quad N(k) = \mathbb{E}\left[\hat{v}(k)\hat{v}^*(k)\right]$$

$$e(k) = (1 - \hat{f}(k)\hat{p}(k))(1 - \hat{f}(k)\hat{p}(k))^* S(k) + \hat{f}(k)\hat{f}^*(k)N(k)$$

Wiener Deconvolution cont.

- zeroing the derivative of e to obtain the minimum yields:

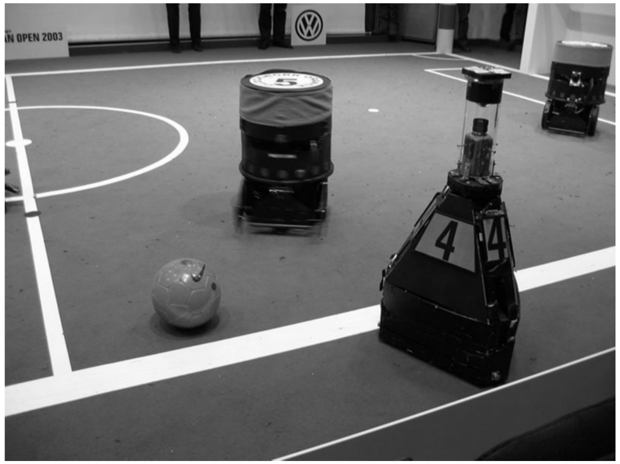
$$\hat{f}(k) = \frac{\hat{p}^*(k)S(k)}{\hat{p}(k)\hat{p}^*(k)S(k) + N(k)} = \frac{\hat{p}^*(k)}{|\hat{p}(k)|^2 + (\frac{S(k)}{N(k)})^{-1}}$$

  which defines the optimal linear filter (Wiener filter)
- $\frac{S(k)}{N(k)}$ is the signal-to-noise ratio
- in the noiseless case:

$$\hat{f}(k) = \frac{1}{\hat{p}(k)} \quad (\text{if } N(k) = 0)$$

- but: $\frac{S(k)}{N(k)}$ and $\hat{p}(k)$ must be known

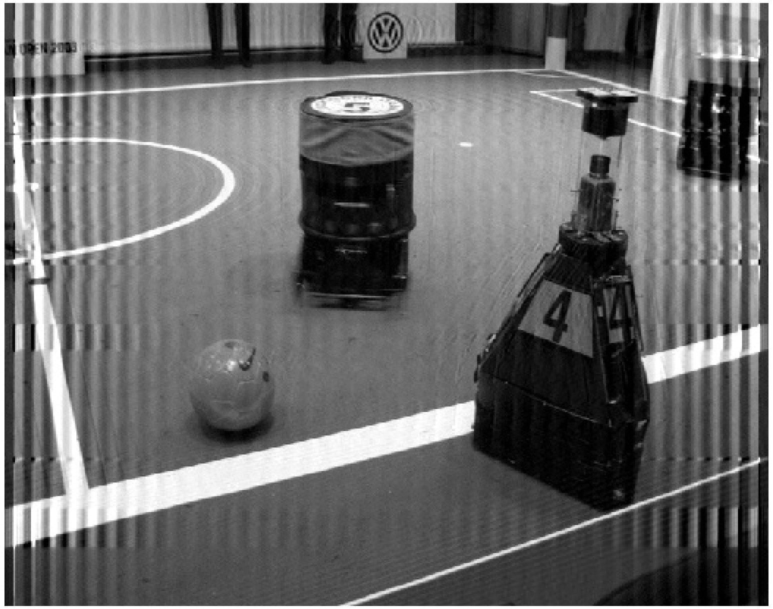
Wiener Deconvolution cont.



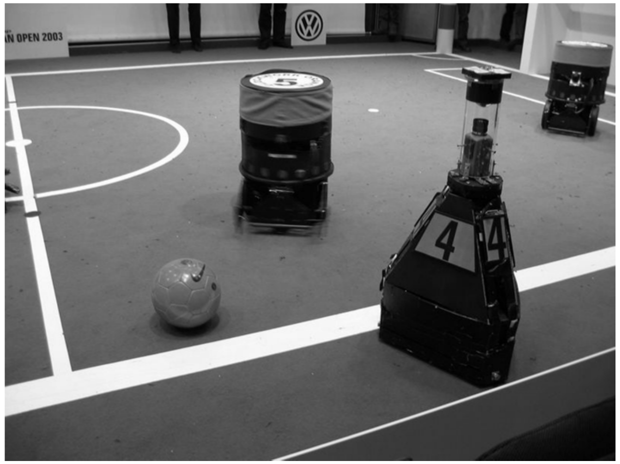
original image

motion blur

restored image



Wiener Deconvolution cont.



original image

Gaussian blur

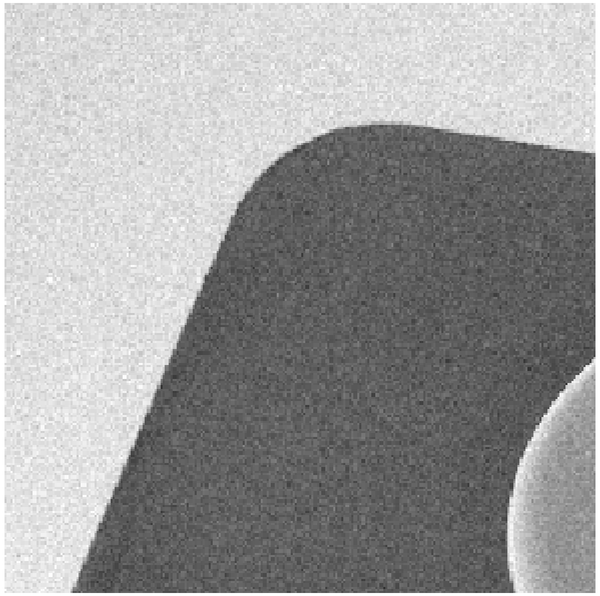
restored image

Models of Noise

- statistical noise:

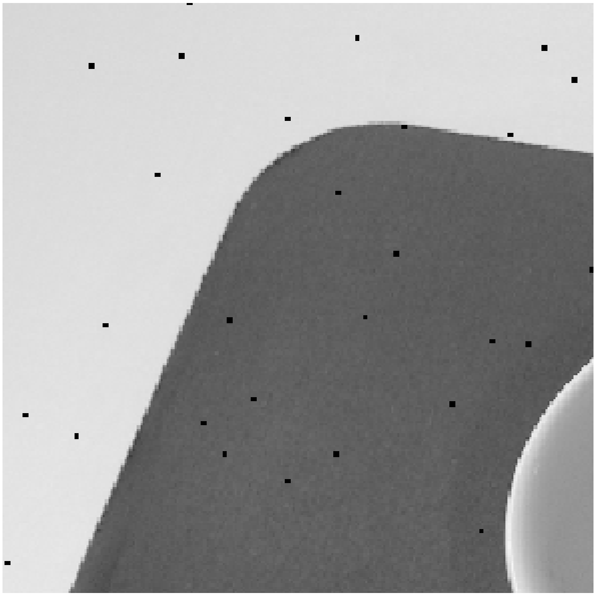
$$g_{noisy}(x,y) = g_{sharp}(x,y) + v(x,y)$$
$$v(x,y) \sim N(0,\sigma^2) \ i.i.d.$$

(i.i.d. = independent and identically distributed)



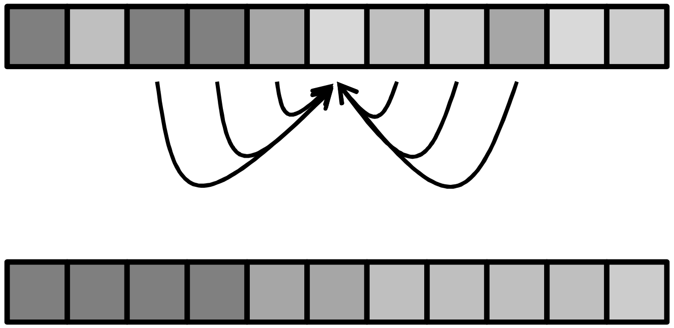
- malfunctioning sensors:

$$g_{noisy}(x,y) = \begin{cases} g_{sharp}(x,y) & \text{with probability } p \\ \text{arbitrary} & \text{otherwise} \end{cases}$$

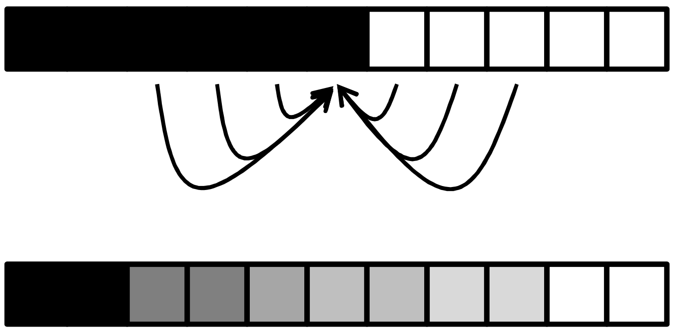


Statistical Noise

- basic idea: averaging (smoothing)



- works well in homogeneous areas, but fails at grey level edges



Smoothing Filters

- rectangular filter

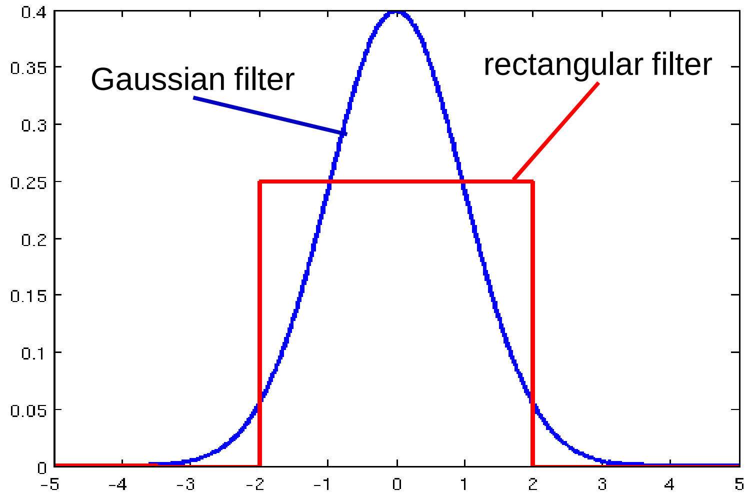
$$f(x) = \begin{cases} \frac{1}{a} & \text{if } |x| < \frac{a}{2} \\ 0 & \text{otherwise} \end{cases}$$

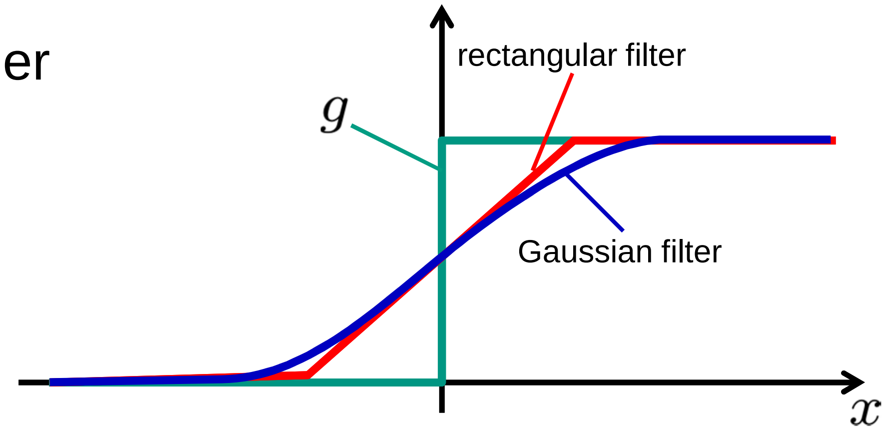
the larger parameter a , the stronge smoothing

- Gaussian filter

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\frac{x^2}{\sigma^2}}$$

the larger parameter σ, the stronger smoothing



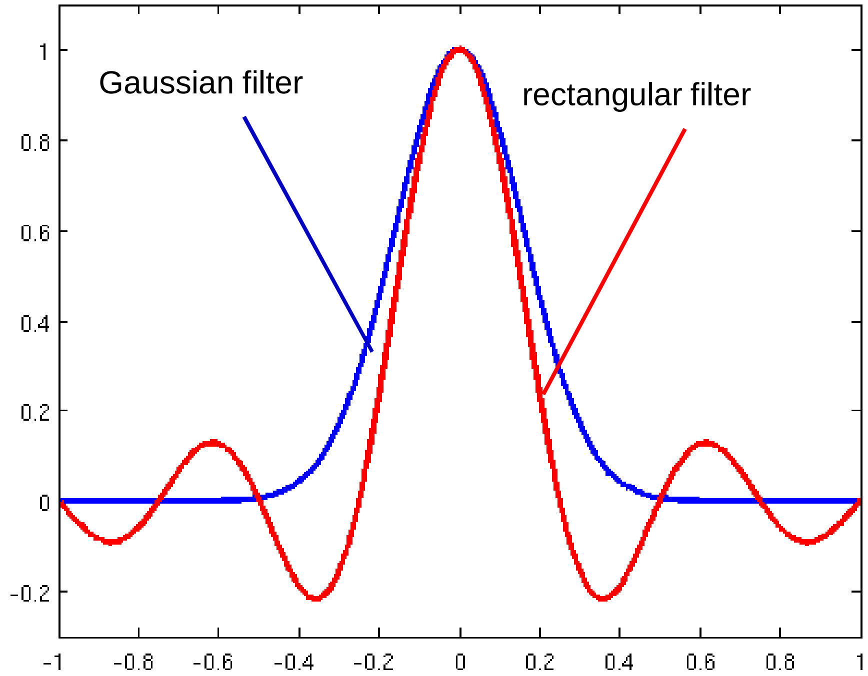


Smoothing Filters

- Fourier transform of smoothing filters

$$\mathcal{F}(rect.filter)(k) = sinc(ak) = \begin{cases} \frac{\sin(\pi x)}{\pi x} & \text{if } x \neq 0 \\ 1 & \text{if } x = 0 \end{cases}$$

$$\mathcal{F}(Gauss.filter)(k) = e^{-2\pi^2\sigma^2k^2}$$



→ Smoothing: low pass filtering

Smoothing Filters for Images

– rectangular filter masks:

$$\frac{1}{9}\cdot\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix}$$

3x3 square mask

$$\frac{1}{25}\cdot\begin{bmatrix}1&1&1&1&1\\1&1&1&1&1\\1&1&1&1&1\\1&1&1&1&1\\1&1&1&1&1\end{bmatrix}$$

5x5 square mask

$$\frac{1}{19.8}\cdot\begin{bmatrix}0.1&0.8&1&0.8&0.1\\0.8&1&1&1&0.8\\1&1&1&1&1\\0.8&1&1&1&0.8\\0.1&0.8&1&0.8&0.1\end{bmatrix}$$

5x5 disc mask

Discrete Convolution cont.

– Gaussian filter masks:

$$\frac{1}{39}\cdot\begin{bmatrix}1 & 4 & 1\\ 4 & 19 & 4\\ 1 & 4 & 1\end{bmatrix}$$

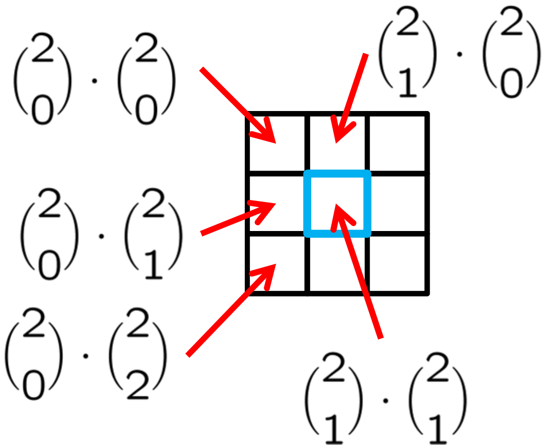
3x3 mask, σ^2=0.25

$$\frac{1}{273}\cdot\begin{bmatrix}1 & 4 & 7 & 4 & 1\\ 4 & 16 & 26 & 16 & 4\\ 7 & 26 & 41 & 26 & 7\\ 4 & 16 & 26 & 16 & 4\\ 1 & 4 & 7 & 4 & 1\end{bmatrix}$$

5x5 mask, σ^2=1

– binomial filter masks

approximations to Gaussian masks using binomial coefficients $\binom{n}{k}$



$$\frac{1}{16}\cdot\begin{bmatrix}1 & 2 & 1\\ 2 & 4 & 2\\ 1 & 2 & 1\end{bmatrix}$$

3x3 mask

$$\frac{1}{256}\cdot\begin{bmatrix}1 & 4 & 6 & 4 & 1\\ 4 & 16 & 24 & 16 & 4\\ 6 & 24 & 36 & 24 & 6\\ 4 & 16 & 24 & 16 & 4\\ 1 & 4 & 6 & 4 & 1\end{bmatrix}$$

5x5 mask

Smoothing Filters cont.

Gaussian, sigma = 7

Bilateral filter

Gaussian filter

- filter mask independent of image content

$$\underbrace{\tilde{g}(u,v)}_{\text{gray level after filtering}} \propto \sum_{i,j} \Big(\underbrace{e^{-\frac{1}{2}\frac{i^2+j^2}{\sigma^2}}}_{\text{distance dependent weight}} \cdot \underbrace{g(u+i, v+j)}_{\text{gray level}} \Big)$$

- smooth over edges and gross outliers

Bilateral filter

- filter mask dependent on image content

$$\underbrace{\tilde{g}(u,v)}_{\text{gray level after filtering}} \propto \sum_{i,j} \Big(\overbrace{e^{-\frac{1}{2}\frac{||g(u+i,v+j)-g(u,v)||^2}{\rho^2}}}^{\text{content dependent weight}} \cdot \underbrace{e^{-\frac{1}{2}\frac{i^2+j^2}{\sigma^2}}}_{\text{distance dependent weight}} \cdot \underbrace{g(u+i, v+j)}_{\text{gray level}} \Big)$$

- reduces smoothing at edges and gross outliers

Bilateral Filters cont.

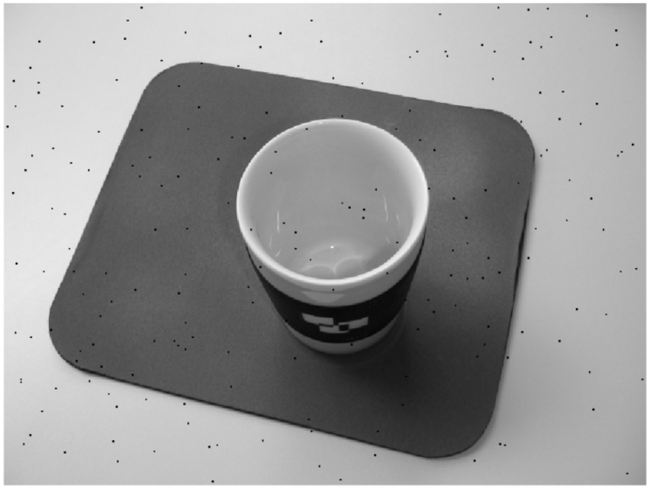
Gaussian filter

$\sigma = 7$

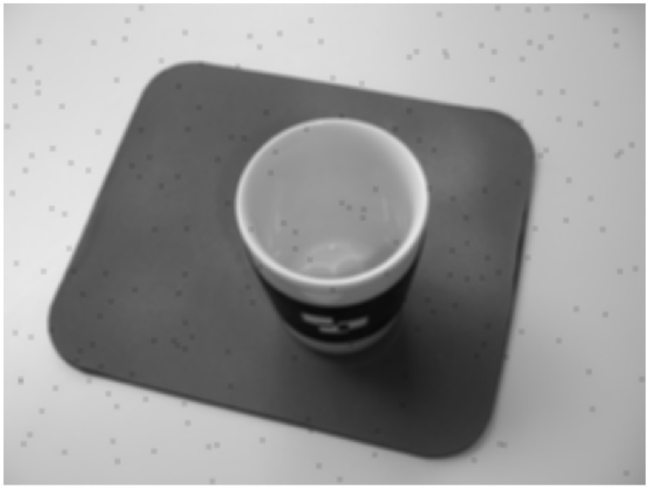
bilateral filter

$\sigma = 7, \ \rho = 20$

Salt-and-Pepper Noise

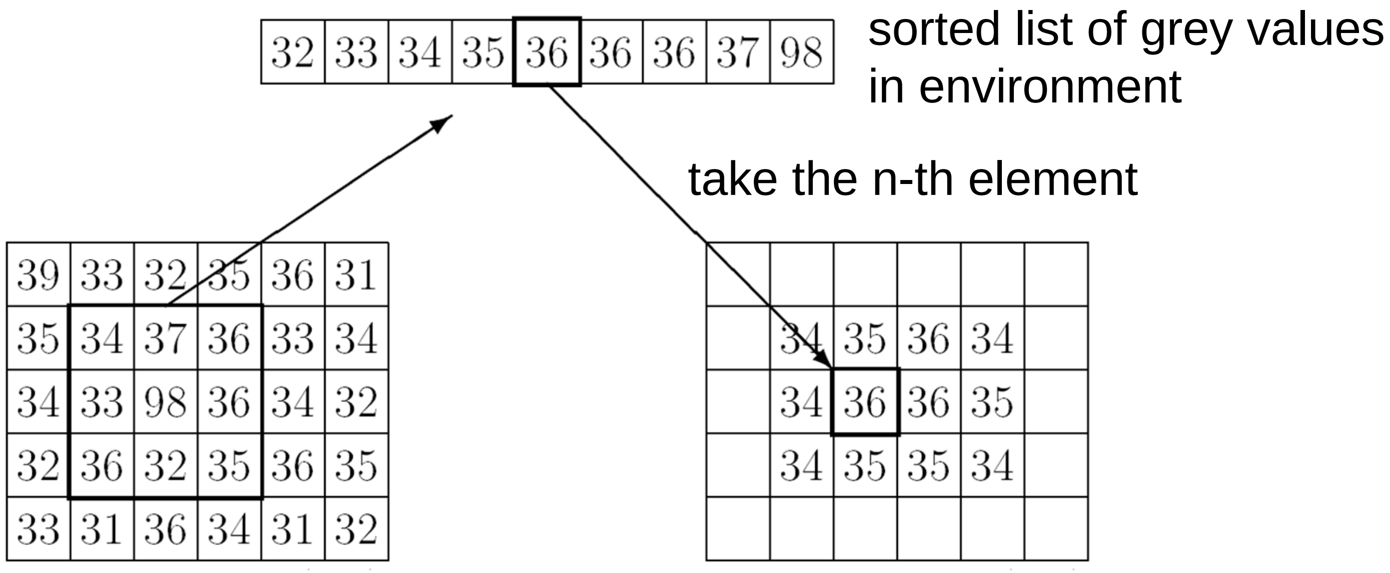


Gaussian filter



→ smoothing not appropriate for salt-and-pepper noise

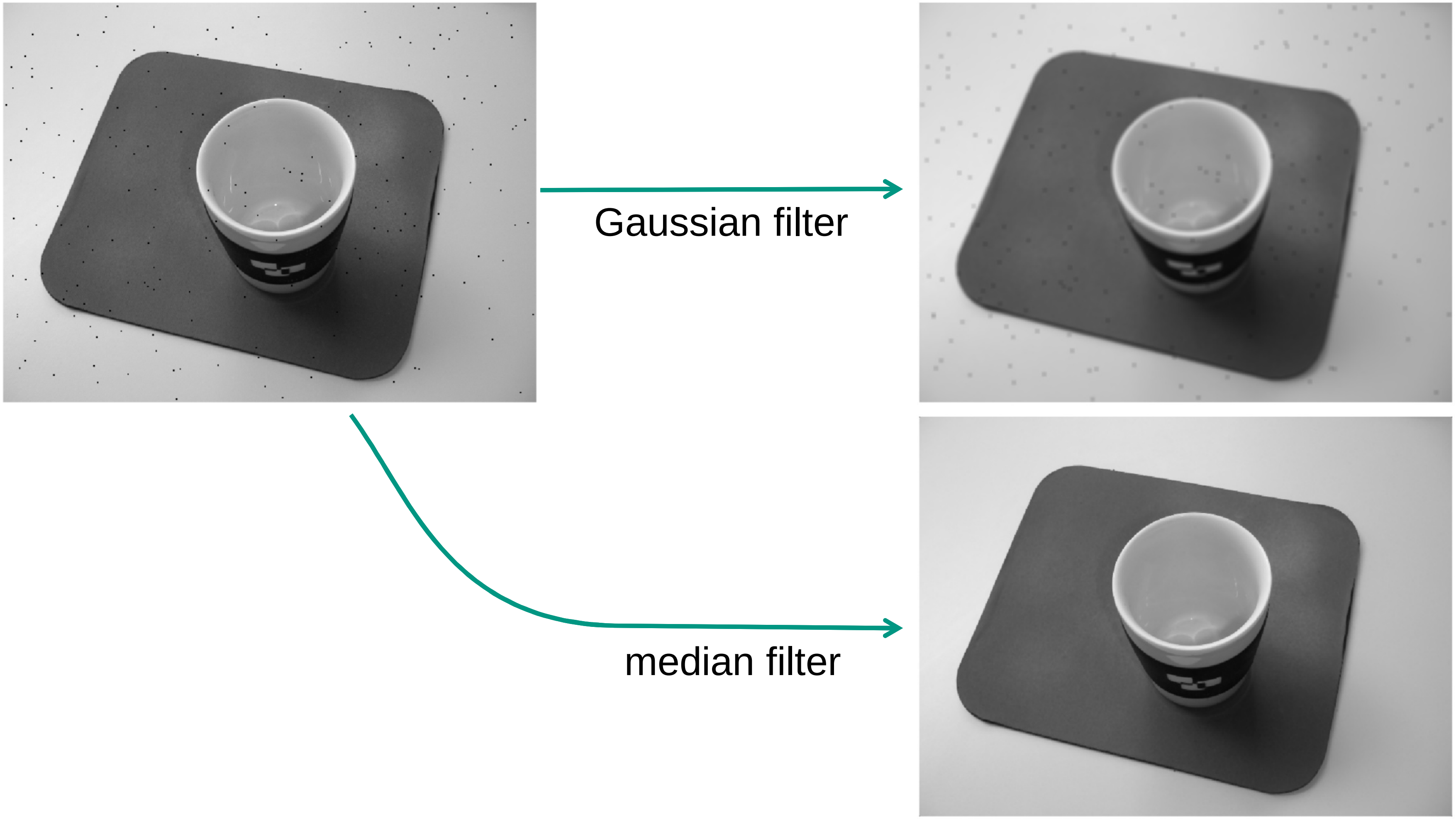
Median filter



median filter:

- sort grey values in environment around reference pixel
- take the grey value in the middle of the sorted list

Median filter



SUMMARY: IMAGE PREPROCESSING

Summary

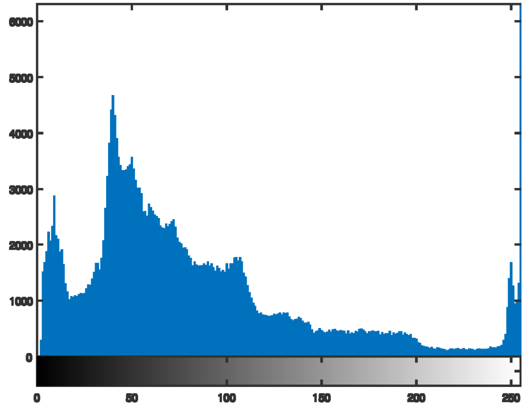
– **sampling**
  - Moiré patterns
  - sampling theorem
  - Fourier transform

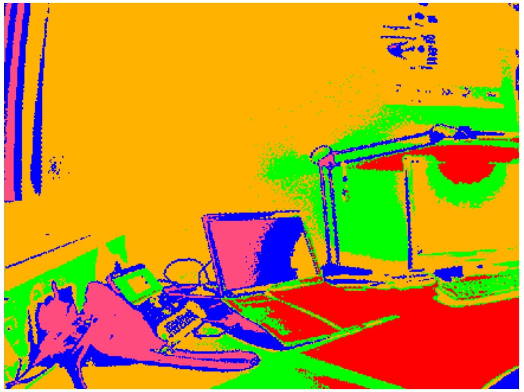
– **quantization**

– **blur and noise**

Summary cont.

- **sampling**
- **quantization**
  - discrete grey values
  - histogram transformation
  - high dynamic range imaging
- **blur and noise**





Summary cont.

- **sampling**
- **quantization**
- **blur and noise**
  - convolution
  - models of blur and noise
  - optimal image restoration (Wiener deconvolution)
  - smoothing filters
    - rectangular
    - Gaussian
    - bilateral
  - median filter