M I A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Lecture 2:
Degradations in Digital Images

Contents

M I A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Noise

- very common in digital images
- can have many reasons, e.g.
  - grainy photographic films that are digitised
  - specific acquisition methods:
    e.g. ultrasound imaging always creates ellipse-shaped speckle noise
  - sensor quality of a CCD chip
  - perturbed image transmission by interferring channels
  - atmospheric disturbance during wireless transmission
- our goal: classify and simulate noise
- Simulating noise is important for the evaluation of image processing methods, since one knows both the original and the noisy image.
- algorithms for denoising in later lectures
- Noise is modelled in a stochastic way.

M I A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Additive Noise

- most important type of noise
- Grey values and noise are assumed to be independent:

$$f_{i,j} \; = \; g_{i,j} + n_{i,j}$$

  where $g = (g_{i,j})$ is the original image, $n = (n_{i,j})$ the noise, and $f = (f_{i,j})$ the noisy image.

- noise n may have different distributions, e.g.
  - uniform distribution (quantisation noise)
  - Gaussian distribution (very common)

M I A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Uniform Noise

- simplest noise model
- has a constant density function within some interval $[-\frac{a}{2}, \frac{a}{2}]$:

$$p(x) = \begin{cases} \frac{1}{a} & \text{for } x \in [-\frac{a}{2}, \frac{a}{2}], \\ 0 & \text{else} \end{cases}$$

- appears e.g. as a byproduct of the quantisation process (cf. Lecture 1): If one rounds to the next integer number, one has $a = 1$.
- A random variable U with uniform distribution in $[-\frac{a}{2}, \frac{a}{2}]$ is easy to simulate. In C:

```
U = ((float)rand() / RAND_MAX) - 0.5) * a;
```

- Degrading some image f with uniform noise is done by replacing f by $f + U$ with some uniformly distributed random variable U.

Noise (4)

(a) Left: Original image, 256×256 pixels, grey value range: $[0, 255]$. **(b) Right:** After adding noise with uniform distribution in $[-70, 70]$. Resulting grey values outside $[0, 255]$ have been cropped. Author: J. Weickert (2007).

Noise (5)

MIA 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Gaussian Noise (White Noise)

- most important noise model: good approximation in many practical situations, e.g.
  - thermal sensor noise in CCD cameras
  - circuit noise caused by signal amplifications
- has density function

$$p(x) \;=\; \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)$$

  where μ is the mean and σ the standard deviation

- For a random variable X with Gaussian distribution, the following probabilities hold:

$$\begin{aligned} P(\mu-\sigma \le X \le \mu+\sigma) &\approx 68\% \\ P(\mu-2\sigma \le X \le \mu+2\sigma) &\approx 95.5\% \\ P(\mu-3\sigma \le X \le \mu+3\sigma) &\approx 99.7\% \end{aligned}$$

  Hence, Gaussian noise is almost completely in the interval $[\mu-3\sigma,\, \mu+3\sigma]$.

Noise (6)

How Can One Simulate Gaussian Noise ?

- **Box–Muller algorithm** for creating a random variable with *normal distribution* (Gaussian distribution with $\mu = 0$, $\sigma = 1$):
  - Create independent random variables U and V with uniform distribution in [0,1]. Implementation in C:

    ```
    U = (float)rand() / RAND_MAX;
    V = (float)rand() / RAND_MAX;
    ```

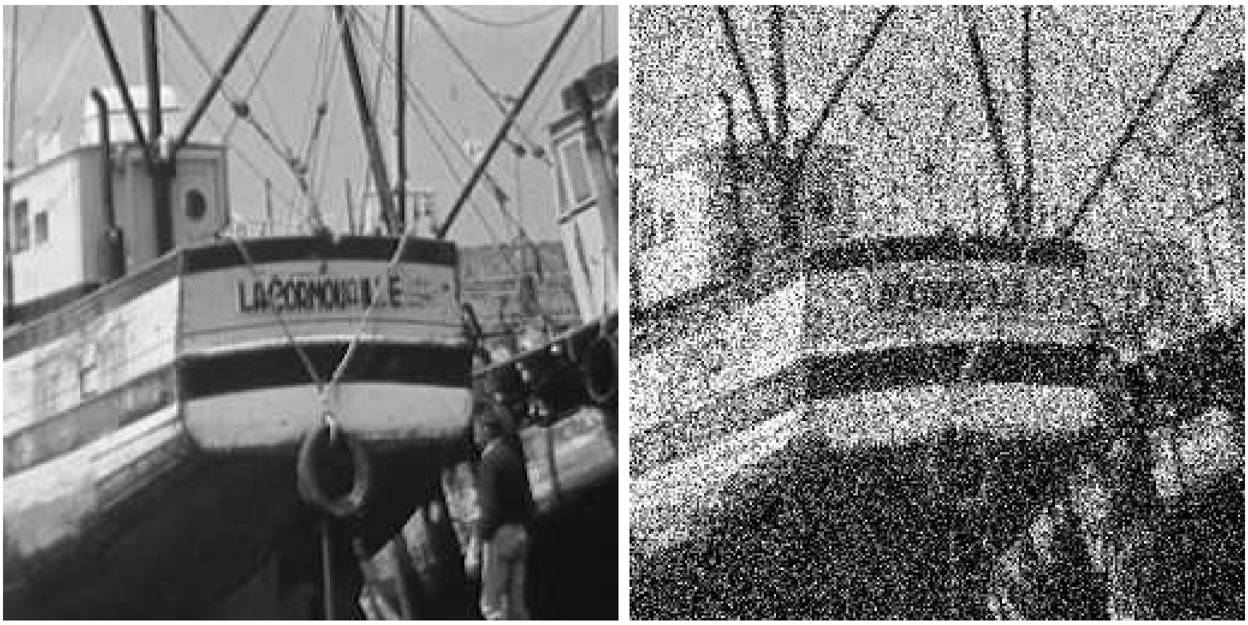
  - Compute

    $$\begin{aligned} N &= \sqrt{-2 \ln U}\, \cos(2\pi V), \\ M &= \sqrt{-2 \ln U}\, \sin(2\pi V). \end{aligned}$$

    Then N and M are independent random variables with normal distribution.

- How can one degrade a grey value f by additive Gaussian noise with mean 0 and standard deviation σ ?
  Take some random variable N with normal distribution and replace f by $f + \sigma N$.

Noise (7)



(a) Left: Original image, 256×256 pixels, grey value range: $[0, 255]$. **(b) Right:** After adding Gaussian noise with $\sigma = 64.48$. Grey values outside $[0, 255]$ have been cropped. Author: J. Weickert (2002).

Noise (8)

Be Careful !

- Images with byte-wise coding have grey values in $[0, 255]$.
- By adding Gaussian noise, one may leave this interval.
- Byte-wise coding either crops these values or misinterprets them.
- In both cases no real Gaussian noise is obtained.
- Remedy, if real Gaussian noise is required:
  Code image in floating point precision or add Gaussian noise during testing.
- Similar considerations apply for uniform noise.

Noise (9)

Multiplicative Noise

- Multiplicative noise is signal-dependent (unlike additive noise).
- Frequently it is proportional to the grey value:

$$\begin{aligned} f_{i,j} &= g_{i,j} + n_{i,j}\, g_{i,j} \\ &= (1 + n_{i,j})\, g_{i,j} \end{aligned}$$

- Example:
  Noise caused by the grains of a photographic emulsion.

Noise (10)

(a) Left: Original image, 256×256 pixels, grey value range: $[0, 255]$. **(b) Right:** After applying multiplicative noise where n has uniform distribution in $[-0.5, 0.5]$. Resulting grey values outside $[0, 255]$ have been cropped. Note that darker grey values are less affected by noise than brighter ones. Author: J. Weickert (2007).

Noise (11)

M I A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Impulse Noise

- degrades the image at singular (!) pixels where erroneous grey values are created (in contrast to additive or multiplicative noise that affects all pixels)
- Example: pixel defects in a CCD chip of a digital camera
- *Unipolar* impulse noise gives degradations with the same grey value, whereas *bipolar* noise attains two grey values.
- Bipolar noise with the highest and lowest grey values is called *salt-and-pepper* noise.

Noise (12)

(a) Left: Original image, 256×256 pixels. **(b) Right:** 20 % of all pixels have be degraded by salt-and-pepper noise, where bright and dark values have the same probability. Author: J. Weickert (2002).

Noise (13)

Signal–to–Noise Ratio (Signal–Rausch–Abstand)

- measures deterioration of image quality by noise
- Let $g = (g_{i,j})$ be the initial image with $M \times N$ pixels.
  Then its *mean (average grey value)* is given by

$$\mu(g) \ := \ \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} g_{i,j}$$

  and its *variance* is defined as

$$\sigma^2(g) \ := \ \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} (g_{i,j} - \mu)^2$$

  Its square root σ is called *standard deviation*.

M I A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Noise (14)

- Let $f = (f_{i,j})$ be a noisy version of g, degraded by additive noise with zero mean:

$$f_{i,j} \; = \; g_{i,j} + n_{i,j}.$$

- Then the *noise variance* of f is given by

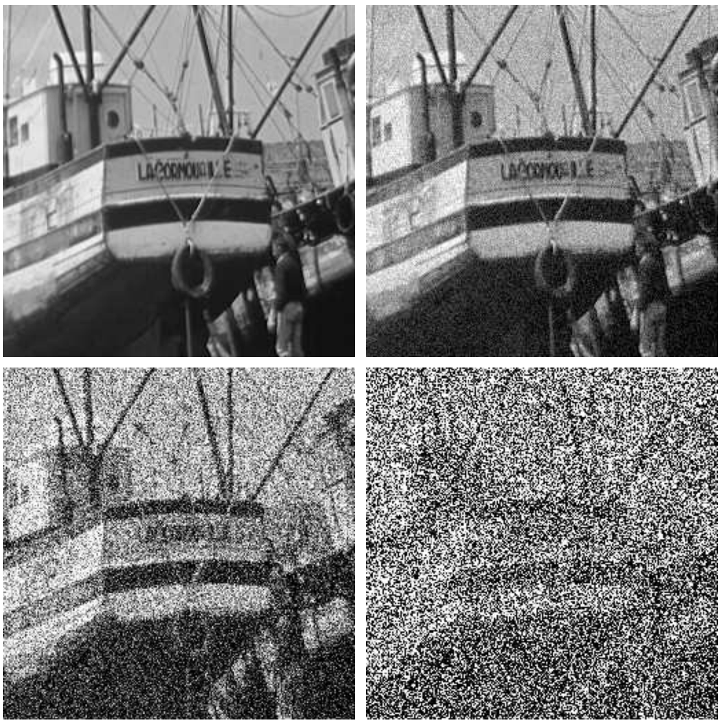
$$\sigma^2(n) \; = \; \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} n_{i,j}^2 \; = \; \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} (f_{i,j} - g_{i,j})^2.$$

- The *signal-to-noise ratio (SNR)* compares image variance and noise variance (different definitions exist):

$$\mathrm{SNR}\,(f,g) \; := \; 10 \, \log_{10} \frac{\sigma^2(g)}{\sigma^2(n)}$$

- Its unit is decibel (dB). The higher the better.
  Noise with SNR values ≥ 30 dB is basically invisible.

Noise (15)



(a) Top left: Original image, 256×256 pixels, standard deviation $\sigma(g) = 64.48$. **(b) Top right:** Adding Gaussian noise with $\sigma(n) = 16.12$ gives $\mathrm{SNR} = 12.04$ dB. **(c) Bottom left:** $\sigma(n) = 64.48$, $\mathrm{SNR} = 0$ dB. **(d) Bottom right:** $\sigma(n) = 257.92$, $\mathrm{SNR} = -12.04$ dB. Grey values outside $[0, 255]$ are cropped. Author: J. Weickert (2002).

M I A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Blur (Unschärfe)

- second source of image degradations (besides noise)
- caused e.g. by atmospheric disturbances, defocussing, imperfections of the optical system or motion during image acquisition
- For simplicity, let us assume that the blurring effect is identical at all locations.
- can be regarded as weighted averaging auf grey values within a certain neighbourhood
- weights for averaging and shape of neighbourhood depend on the source of degradation
- Weighted averaging can be mathematically described by a so-called convolution.

M I A

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Convolution (Faltung)

One-Dimensional Convolution

- convolution of two 1-D signals $g = (g_i)_{i\in\mathbb{Z}}$ and $w = (w_i)_{i\in\mathbb{Z}}$:

$$(g * w)_i \;:=\; \sum_{k\in\mathbb{Z}} g_{i-k}\, w_k$$

- The components of w can be regarded as (mirrored) weights for averaging the components of g.
- A continuous version of the convolution is given by

$$(g * w)(x) \;:=\; \int_{\mathbb{R}} g(x-x')\, w(x')\, dx'$$

Blur (3)

Example:

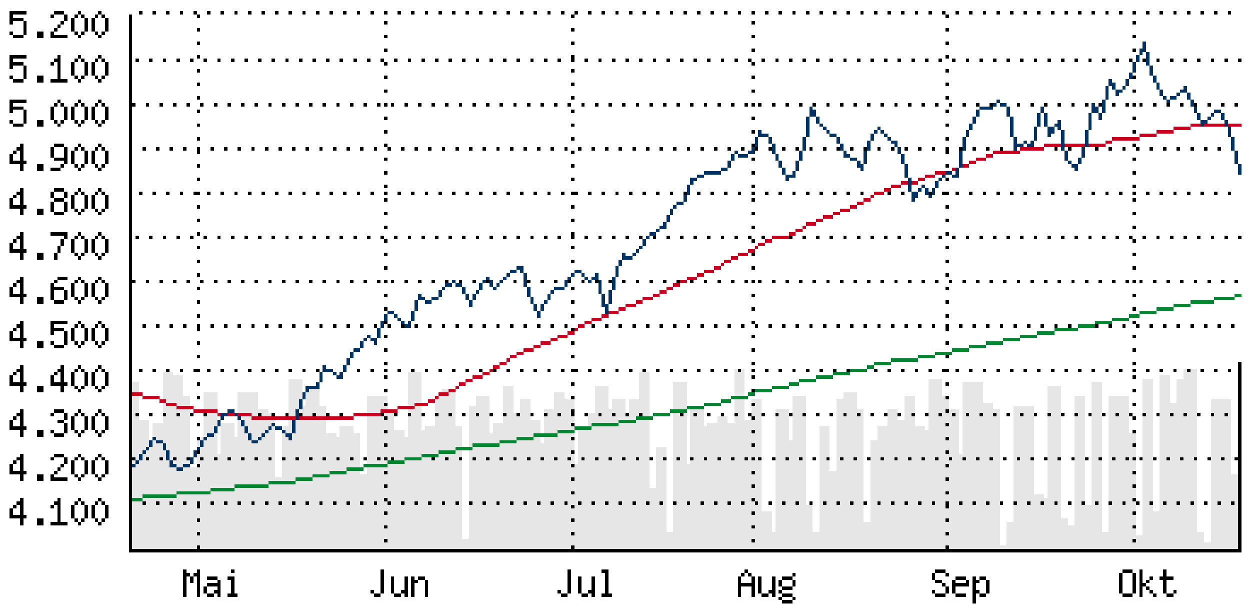
- Let g_i be a stock market price (Börsenkurs) at day i.
- Wanted: Average price f_i within the last 200 days.
- With

$$w_k := \begin{cases} \frac{1}{200} & \text{for } k = 0,...,199, \\ 0 & \text{else} \end{cases}$$

one obtains

$$f_i \;=\; \sum_k g_{i-k}\, w_k \;=\; \frac{1}{200} \sum_{k=0}^{199} g_{i-k}$$

Blur (4)



German stock market index (DAX) on October 20, 2005. **Blue:** Daily values. **Red:** Averaged over the last 38 days. **Green:** Averaged over the last 200 days. Source: `http://www.spiegel.de`.

M I A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Two-Dimensional Convolution

- Convolution of two discrete images $g = (g_{i,j})$ and $w = (w_{i,j})$:

$$(g * w)_{i,j} := \sum_{k \in \mathbb{Z}} \sum_{l \in \mathbb{Z}} g_{i-k,\, j-l}\, w_{k,l}$$

- The continuous 2-D convolution is defined by

$$(g * w)(x, y) := \int_{\mathbb{R}} \int_{\mathbb{R}} g(x-x', y-y')\, w(x', y')\, dx'\, dy'.$$

The double integral can be intergrated first with respect to x' and then with respect to y' (Fubini's theorem).

M I A 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Example

- Let the convolution kernel $w(x, y)$ be given by

$$w(x, y) := \begin{cases} \frac{1}{\pi r^2} & \text{for } x^2 + y^2 \leq r^2, \\ 0 & \text{else,} \end{cases}$$

- Then $g * w$ describes a smoothing of the image g by averaging all grey values within a disk-shaped neighbourhood of radius r.

- Remark:
  Computing this convolution by calculating the integral becomes time-consuming when r is large.
  We will soon study a more efficient alternative: the Fourier transform.

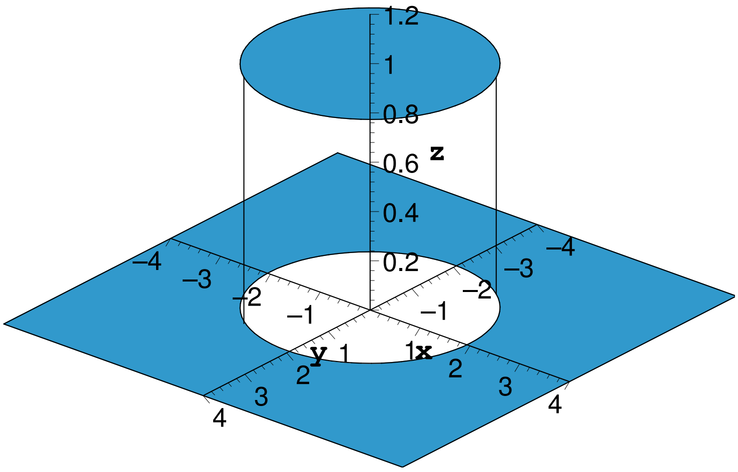
M I A 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Blur (7)

Modelling Blur by Convolutions

- **Defocussed Optical System:**

  gives cylinder–shaped convolution kernel



Cylinder-shaped convolution kernel. Author: B. Burgeth (2002).

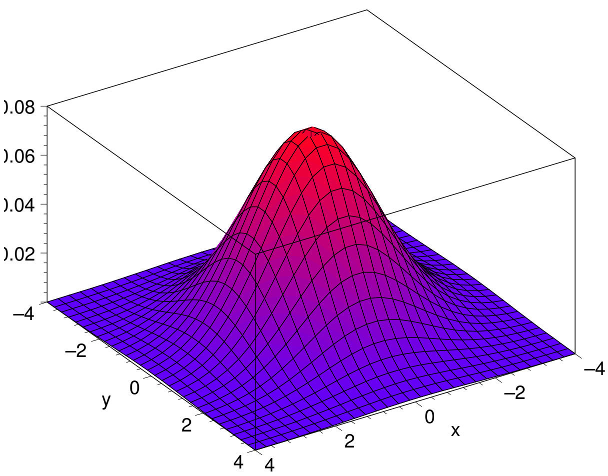
M I A 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Blur (8)

- **Atmospheric Disturbances (e.g. Telescopes):**

  can be approximated by a 2-D Gaussian (product of two 1-D Gaussians):

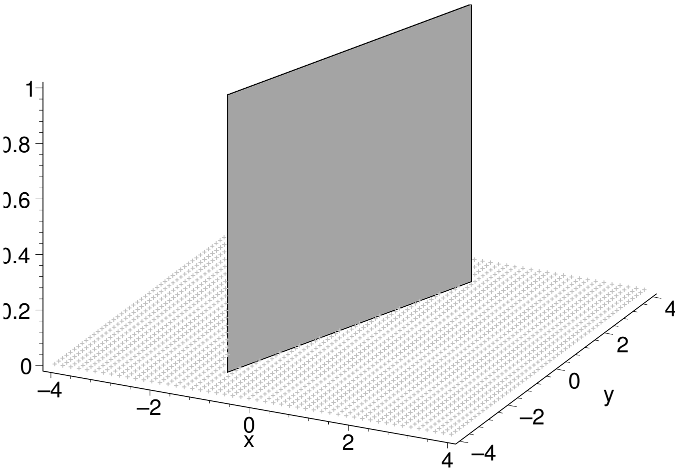
$$w(x,y) \;=\; \frac{1}{2\pi\sigma^2}\exp\left(\frac{-(x-\mu_1)^2-(y-\mu_2)^2}{2\sigma^2}\right)$$



2-D Gaussian. Author: B. Burgeth (2002).

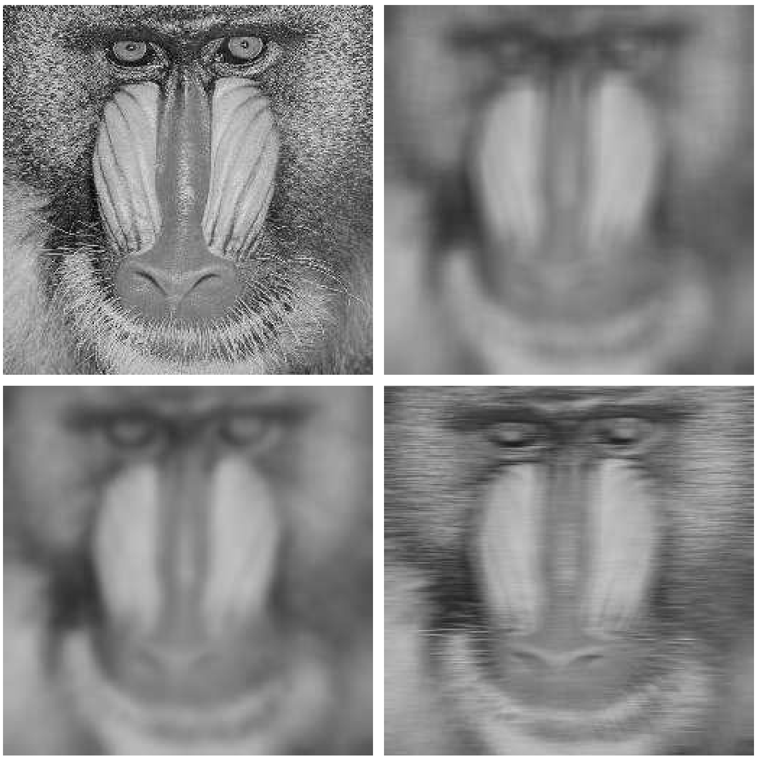
- **Motion Blur:**

  in the simplest case (uniform motion, all objects in equal distance to camera):
  1-D box function along the direction of motion



Kernel for a convolution with a 1-D box function. Author: B. Burgeth (2002).

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28



(a) Top left: Original image, 256×256 pixels. **(b) Top right:** Simulation of out-of-focus blur. **(c) Bottom left:** Simulation of Gaussian blur. **(d) Bottom right:** Simulation of motion blur in horizontal direction. Author: J. Weickert (2002).

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Summary

- Noise and blur are frequent degradations in digital images.
- Noise can be modelled and simulated in a stochastic way.
  Most important is additive Gaussian noise with zero mean.
  It can be simulated with the Box–Muller algorithm.
  Sometimes also multiplicative and impulse noise is present.
- Blur can be simulated by convolution with a suitable kernel (cylinder, Gaussian, 1-D box function).
  Often one assumes that this kernel does not depend on the location.
- A frequently used degradation model of some initial image g with a shift-invariant blurring kernel w and additive noise n is given by

$$f = g * w + n.$$

M I A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Literature

- R. C. Gonzalez, R. E. Woods: *Digital Image Processing*. Prentice Hall, Upper Saddle River, Second Edition, 2002.
  (Section 5.2 deals with noise models.)
- G. E. P. Box, M. E. Muller: A note on the generation of random normal deviates. *Annals of Mathematical Statistics*, Vol. 29, 610–611, 1958.
  (original paper on the Box–Muller method)
- C. Boncelet: Image noise models. In A. Bovik (Ed.): *Handbook of Image and Video Processing*. Academic Press, San Diego, pp. 325–335, 2000.
  (good description of noise models for images)