# Lecture 2:
# Foundations II: Degradations in Digital Images

## Contents

# Recently on IPCV ...

- Digital images are discretised in two ways:
  in the domain (sampling) and the co-domain (quantisation).

$256 \times 256$ pixels

$256$ greyscales

$32 \times 32$ pixels

$256$ greyscales

$256 \times 256$ pixels

$2$ greyscales

- generalisation of the domain:
  $m$-dimensional images, image sequences
- generalisation of the co-domain:
  vector-valued images, matrix-valued images

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 29 30 31 32 33

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 29 30 31 32 33

# Lecture 2:
# Foundations II: Degradations in Digital Images

**Contents**

# Noise

- very common in digital images
- can have many reasons, e.g.
  - image sensor of a digital camera, in particular in low light situations
  - grainy photographic films that are digitised
  - specific acquisition methods:
    e.g. ultrasound imaging always creates ellipse-shaped speckle noise
  - atmospheric perturbances during wireless transmission
- **our goal today**: classify and simulate noise
- Simulating noise is important for the evaluation of image denoising methods, since one knows both the original and the noisy image.
- Algorithms for denoising will be discussed in later lectures.
- Noise is modelled in a stochastic way.

## Additive Noise

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

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

$\boldsymbol{g} = (g_{i,j})$: original discrete image
$\boldsymbol{n} = (n_{i,j})$: noise
$\boldsymbol{f} = (f_{i,j})$: observed noisy image

- noise $\boldsymbol{n}$ may have different distributions, e.g.
  - uniform distribution (very simple)
  - Gaussian distribution (very common)

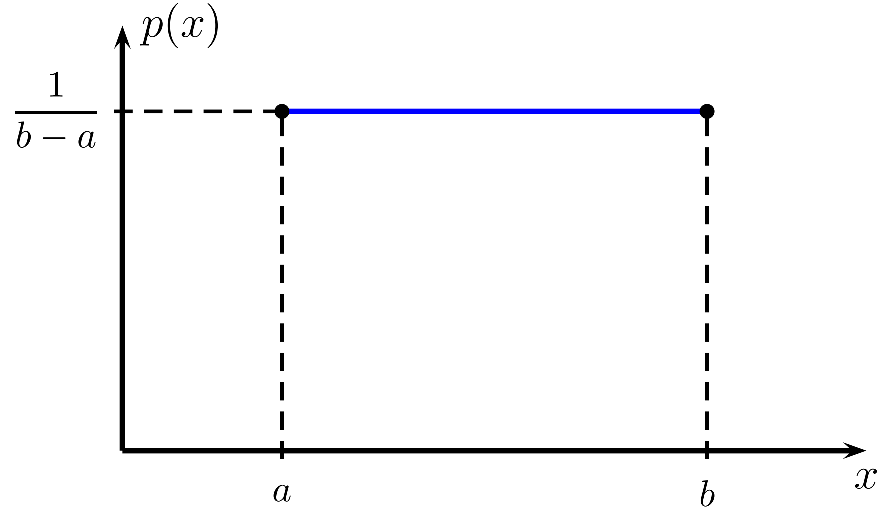
## Uniform Noise

- often not the most realistic noise model, but easy to simulate
- has a constant density function within some interval $[a, b]$:

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

Often one chooses a symmetric interval w.r.t. $0$, i.e. $[-b, b]$ with $b > 0$.

- can be used in connection with quantisation (cf. Assignment H1, Problem 5): adding uniform noise makes coarse quantisation levels visually more pleasant




Density function for uniform noise. Author: M. Mainberger.

## How Can One Simulate Uniform Noise ?

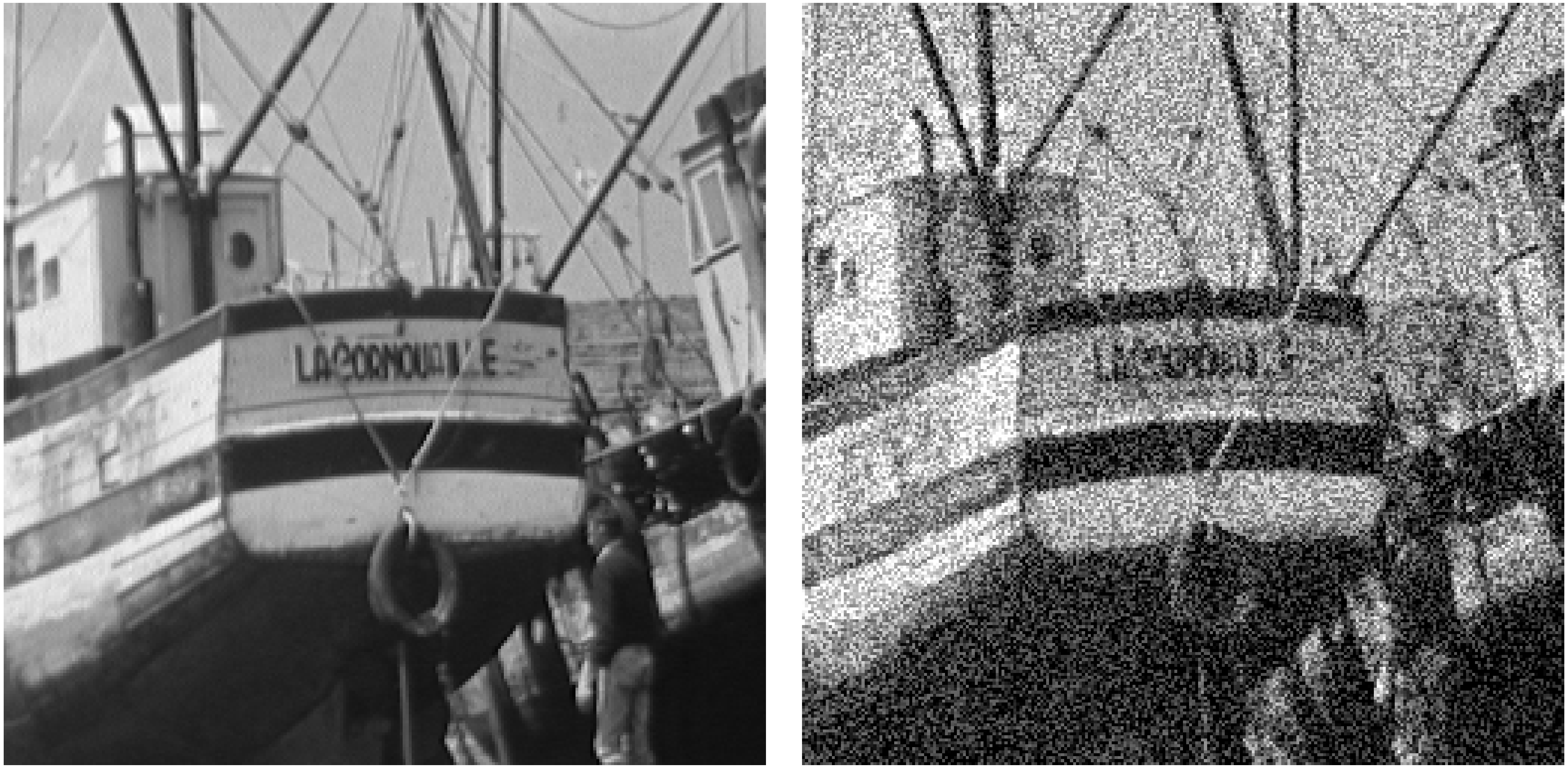
- a random variable $U$ with uniform distribution in $[a, b]$ simulated in C:

```
U = a + (double)rand() / RAND_MAX * (b - a);
```

  `(double)rand() / RAND_MAX` $\rightarrow$ uniformly distributed random number in $[0, 1]$

- Degrading some image $g$ with uniform noise is easy:
  Replace $g$ by $g + U$ with some uniformly distributed random variable $U$.

# Noise (5)



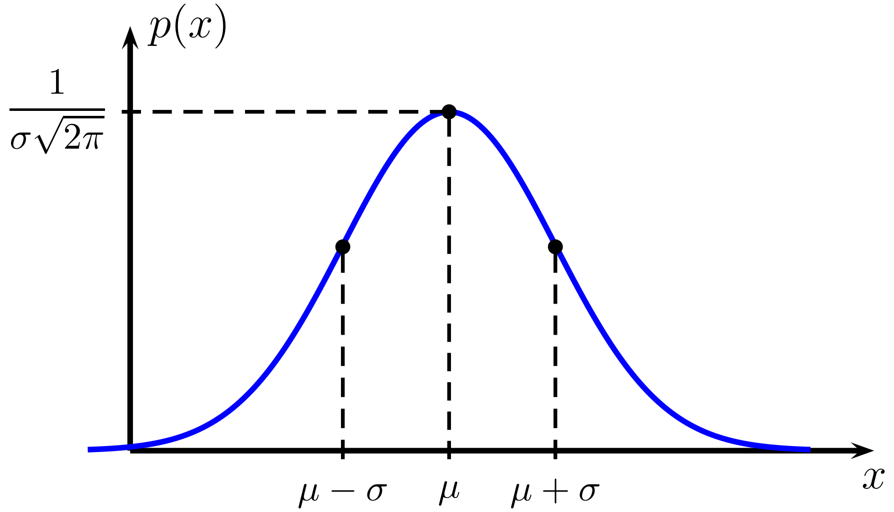
**Left:** Original image, $256 \times 256$ pixels, grey value range $[0, 255]$. **Right:** After adding noise with uniform distribution in $[-70, 70]$. Resulting grey values outside $[0, 255]$ have been cropped. Author: J. Weickert.

## Gaussian Noise (White Noise)

- most important noise model: good approximation in many practical situations, e.g.
  - thermal noise created by the imaging sensor
  - 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 $\mu$ is the mean and $\sigma$ the standard deviation of the Gaussian




Density function for Gaussian noise. Author: M. Mainberger.

- $p(x)$ has inflection points at $x = \mu - \sigma$ and $x = \mu + \sigma$.
- For a Gaussian-distributed random variable $X$ 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 lives almost completely in the interval $[\mu-3\sigma,\, \mu+3\sigma]$.

## How Can One Simulate Gaussian Noise ?

**Box–Muller algorithm** for creating two random variables with *normal distribution* (Gaussian distribution with $\mu = 0$, $\sigma = 1$) from two uniformly distributed variables:

- It works in 2D and uses polar coordinates.
  This setting is mathematically more pleasant to handle than the 1D case, since there is no analytical solution for the 1D Gaussian distribution

$$\Phi(x) = \int_{-\infty}^{x} p(s)\, ds\,.$$

- Create independent random variables $U$ and $V$ with uniform distribution in [0,1]:

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

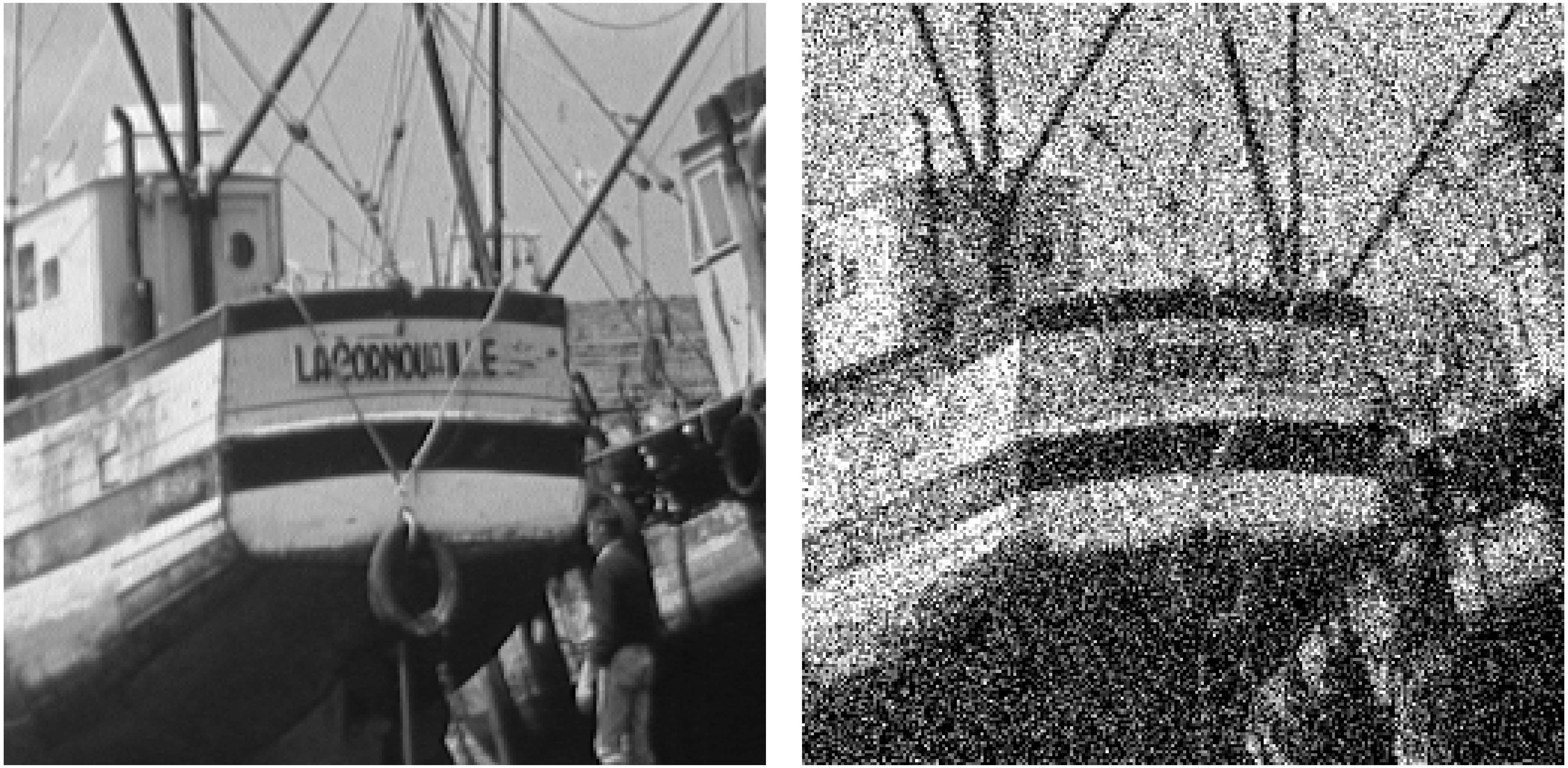
- If $U > 0$ compute

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

## Box–Muller algorithm (continued)

- In the Preparatory Assignment P1, Problem 2 the following is proven:
  $N$ 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 $\sigma$ ?
  Take some random variable $N$ with normal distribution and replace $f$ by $f + \sigma N$.



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

## 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.
- Two remedies, if real Gaussian noise is required:
  - Do not store the noisy image in a byte-wise manner.
    Use real numbers (`float` or `double`) instead.
  - Alternatively, avoid storing the noisy image at all.
    Simply create Gaussian noise within the program for testing.
- Similar considerations apply for uniform noise.

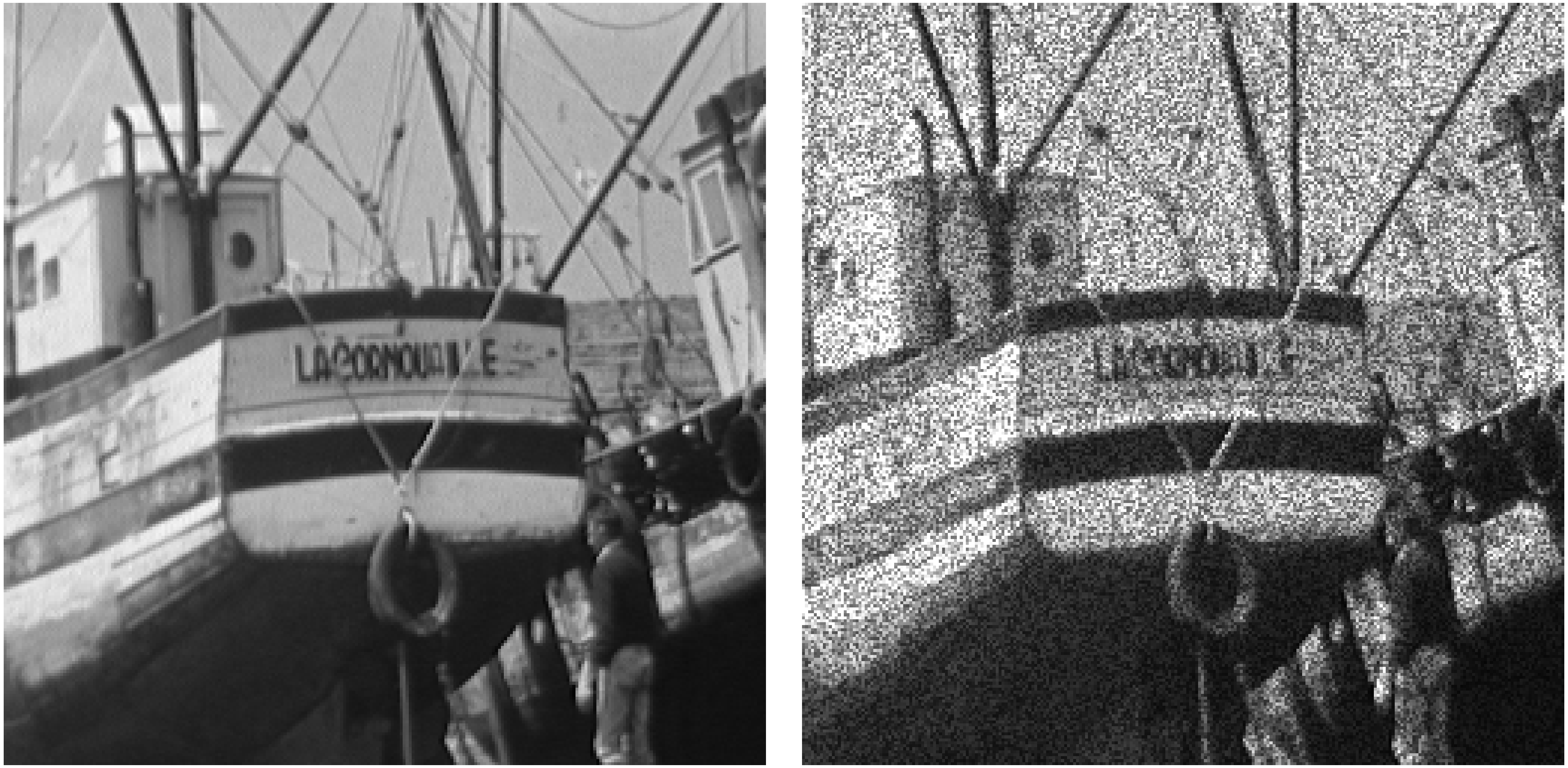
## Multiplicative Noise

- Multiplicative noise is signal-dependent (unlike additive noise).
- Usually one assumes that the perturbation 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 in analog photos.

# Noise (12)

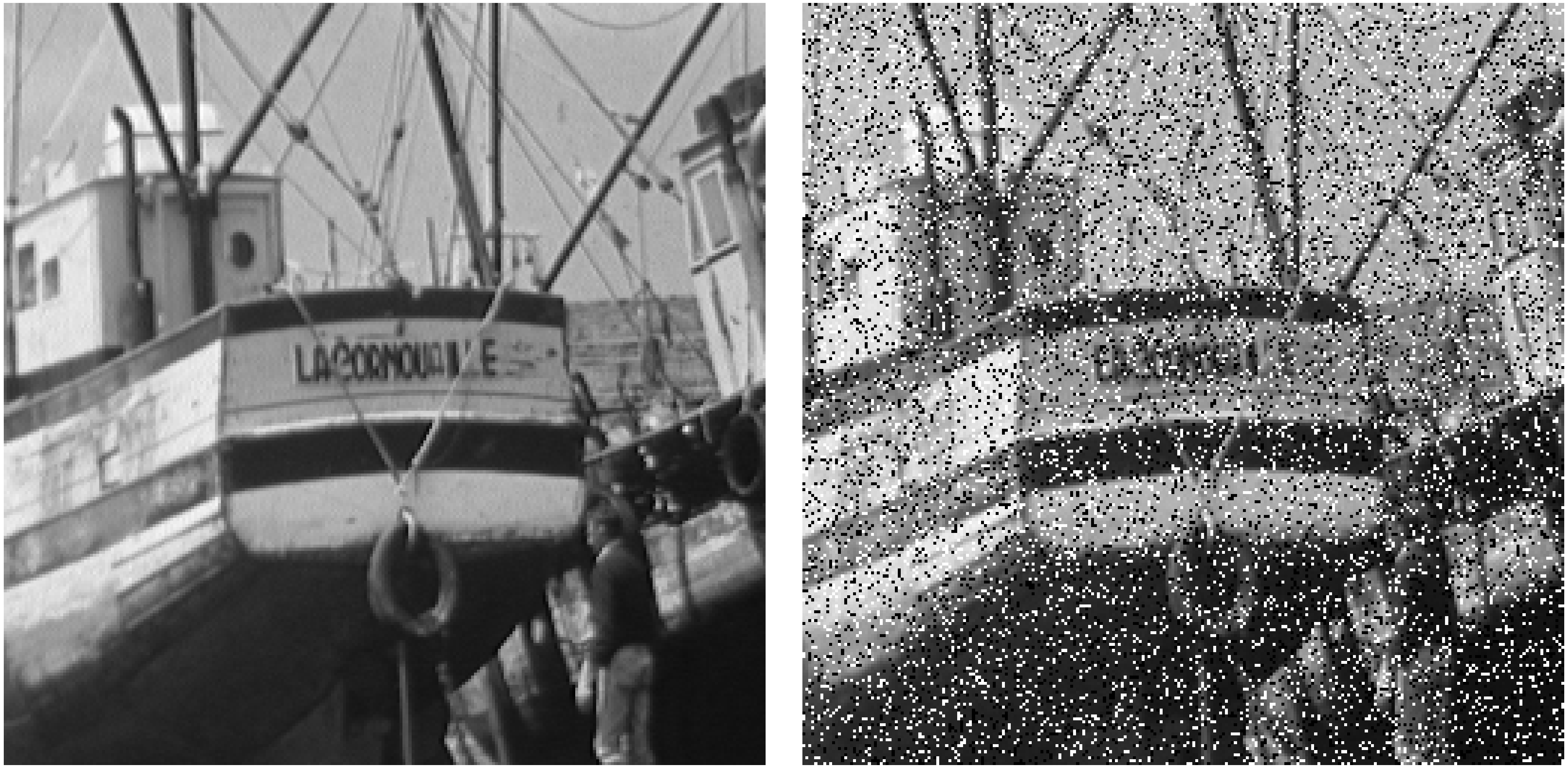


**Left:** Original image, $256 \times 256$ pixels, grey value range: $[0, 255]$. **Right:** After applying multiplicative noise where $n$ has uniform distribution in $[-0.5, 0.5]$. Resulting grey values outside $[0, 255]$ have been cropped. Note that darker grey values are less affected by noise than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

## Impulse Noise

- degrades the image in *some (!)* pixels where erroneous grey values are created (in contrast to additive or multiplicative noise that affects *all* pixels)
- Example: pixel defects in the sensor chip of a digital camera
- *Unipolar impulse noise* gives degradations having only one grey value.
  *Bipolar impulse noise* attains two grey values.
- *Salt-and-pepper noise* is bipolar noise attaining the highest and lowest grey values.

# Noise (14)



**Left:** Original image, $256 \times 256$ pixels. **Right:** 20 % of all pixels have been degraded by salt-and-pepper noise, where bright and dark values have equal probability. Author: J. Weickert.

## Measuring Noise

- Let $\boldsymbol{g} = (g_{i,j})$ be the undegraded image (without noise) with $M \times N$ pixels.
- Its *mean (average grey value, Mittelwert)* is given by

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

- The *variance (Varianz)* measures the average quadratic deviation from the mean:

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

- Its square root $\sigma(\boldsymbol{g})$ is called *standard deviation (Standardabweichung, Streuung).*

- Let $\boldsymbol{f} = (f_{i,j})$ be a noisy version of $\boldsymbol{g}$.
  The *mean squared error (MSE, mittlerer quadratischer Fehler)* of $\boldsymbol{f}$ w.r.t. $\boldsymbol{g}$ is

$$\mathrm{MSE}\,(\boldsymbol{f}, \boldsymbol{g}) \;:=\; \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} (f_{i,j} - g_{i,j})^2.$$

  **The smaller, the better.**

- In image compression one often uses the *peak-signal-to-noise ratio (PSNR, Spitzen-Signal-Rausch-Abstand).*
  It relates the noise to the grey value range in a logarithmic way.
  For a grey value range $[0, 255]$, the PSNR is defined as

$$\mathrm{PSNR}\,(\boldsymbol{f}, \boldsymbol{g}) \;:=\; 10\,\log_{10}\left(\frac{255^2}{\mathrm{MSE}\,(\boldsymbol{f}, \boldsymbol{g})}\right)$$

  Note that $255^2$ is the maximal MSE.
  The unit of the PSNR is *decibel (dB).* **The higher the better.**
  Noise with PSNR values $\geq 30$ dB is hardly visible.

# Noise (17)



**Top left:** Original image, $256 \times 256$ pixels. **Top right:** Adding Gaussian noise with $\sigma = 15$ gives $\text{MSE} = 226.06$ and $\text{PSNR} = 24.59$ dB. **Bottom left:** $\sigma = 30$ yields $\text{MSE} = 904.24$ and $\text{PSNR} = 18.57$ dB. **Bottom right:** $\sigma = 60$ yields $\text{MSE} = 3616.95$ and $\text{PSNR} = 12.55$ dB. Grey values outside $[0, 255]$ are cropped in the visualisation. Author: J. Weickert.

# Lecture 2:
# Foundations II: Degradations in Digital Images

**Contents**

# Blur (Unschärfe)

- second source of image degradations (besides noise)
- caused e.g. by defocussing, imperfections of the optical system, atmospheric perturbances, or motion during image acquisition.
- For simplicity, let us assume that the blurring effect is identical at all locations *(shift-invariant blur model).*
- This leads to weighted averaging of grey values within a certain neighbourhood.
- The shape of neighbourhood and the weights depend on the source of degradation.
- The mathematical tool to describe such a weighted averaging is called convolution.

A real-world example for motion blur: Zoom into a photo of a crab (grapsus adscenionis, Rote Felsenkrabbe) that has been degraded by camera motion. Image size: $768 \times 512$ pixels. Can you see the direction of the camera motion? Photo: J. Weickert.

# Convolution (Faltung)

## One-Dimensional Convolution

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

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

- Components of $\boldsymbol{w}$ are mirrored weights for averaging components of $\boldsymbol{g}$.
- *continuous convolution* of two 1-D signals $g, w : \mathbb{R} \to \mathbb{R}$:

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

## Example

- Let $g_i$ be the stock market price (Börsenkurs) at day $i$.
- We want to compute the average price $f_i$ within the last 200 days:

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

- This is as a discrete convolution between $g$ and a suitable "blur kernel" $w$:

$$f_i = \sum_{k \in \mathbb{Z}} g_{i-k} \, w_k$$

with

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

# Blur (5)


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`.

## Properties of the Convolution

For the continuous convolution, the following properties are not difficult to show (cf. Assignment H1, Problem 4), and they make life easier:

- Commutativity: $f * g \;=\; g * f$.
- Associativity: $(f * g) * h \;=\; f * (g * h)$.
- Distributivity: $(f + g) * h \;=\; f * h \;+\; g * h$,
  $f * (g + h) \;=\; f * g \;+\; f * h$.
- Differentiation: $(f * g)' \;=\; f' * g \;=\; f * g'$.
- Differentiability: If $f \in C^0(\mathbb{R})$ and $g \in C^n(\mathbb{R})$, then $(f * g) \in C^n(\mathbb{R})$.
  ($C^n(\mathbb{R})$: $n$ times continuously differentiable functions on $\mathbb{R}$)
- Linearity: $(\alpha f + \beta g) * h \;=\; \alpha(f * h) + \beta(g * h)$ with $\alpha, \beta \in \mathbb{R}$.
- Shift Invariance: $(T_b f) * g \;=\; T_b(f * g)$ for all translations $T_b$ with
  $(T_b f)(x) \;:=\; f(x - b)$.

These properties also hold for discrete convolution
(apart from the purely continuous properties "Differentiation" and "Differentiability").

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 29 30 31 32 33

## Two-Dimensional Convolution

- *discrete convolution* of two images $\boldsymbol{g} = (g_{i,j})_{i,j\in\mathbb{Z}}$ and $\boldsymbol{w} = (w_{i,j})_{i,j\in\mathbb{Z}}$:

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

- *continuous convolution* of two images $g, w : \mathbb{R}^2 \to \mathbb{R}$:

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

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

## Example

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

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

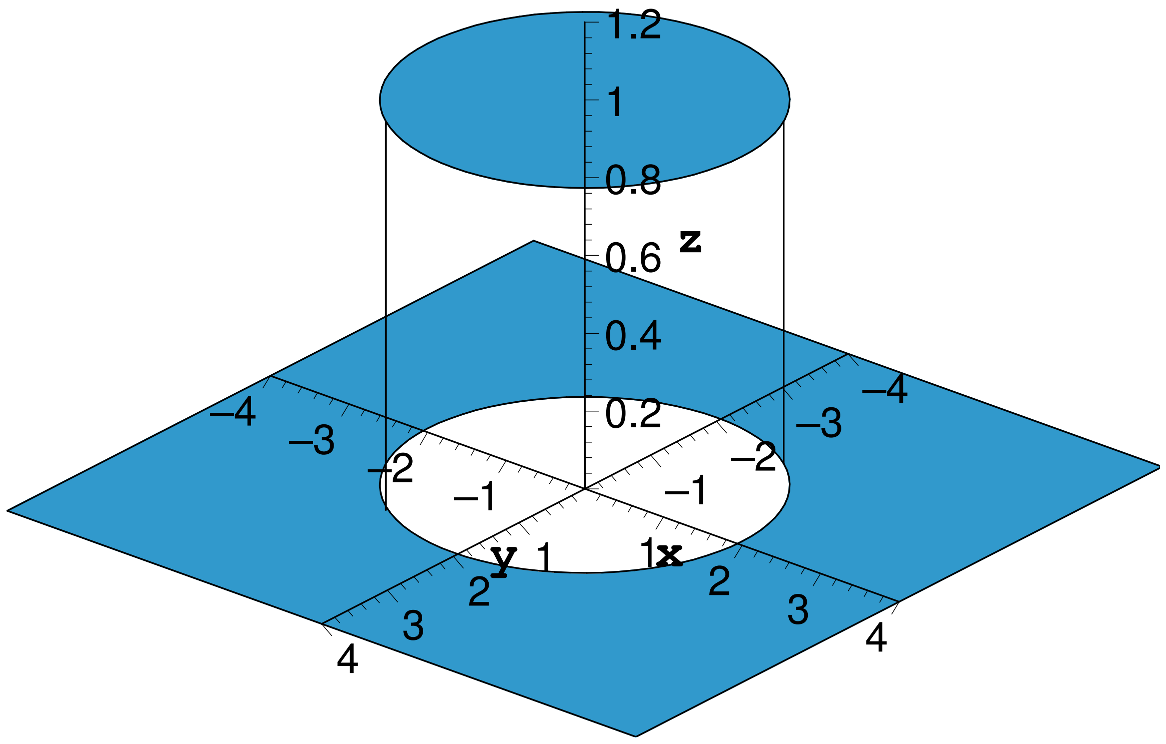
- Then $g * w$ describes a smoothing of the image $g$ by averaging all grey values within a disk-shaped neighbourhood of radius $r$.

- **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**.

## Modelling Blur by Convolutions

- **Defocussed Optical System:**

  usually approximated by a cylinder–shaped convolution kernel ("pillbox kernel")

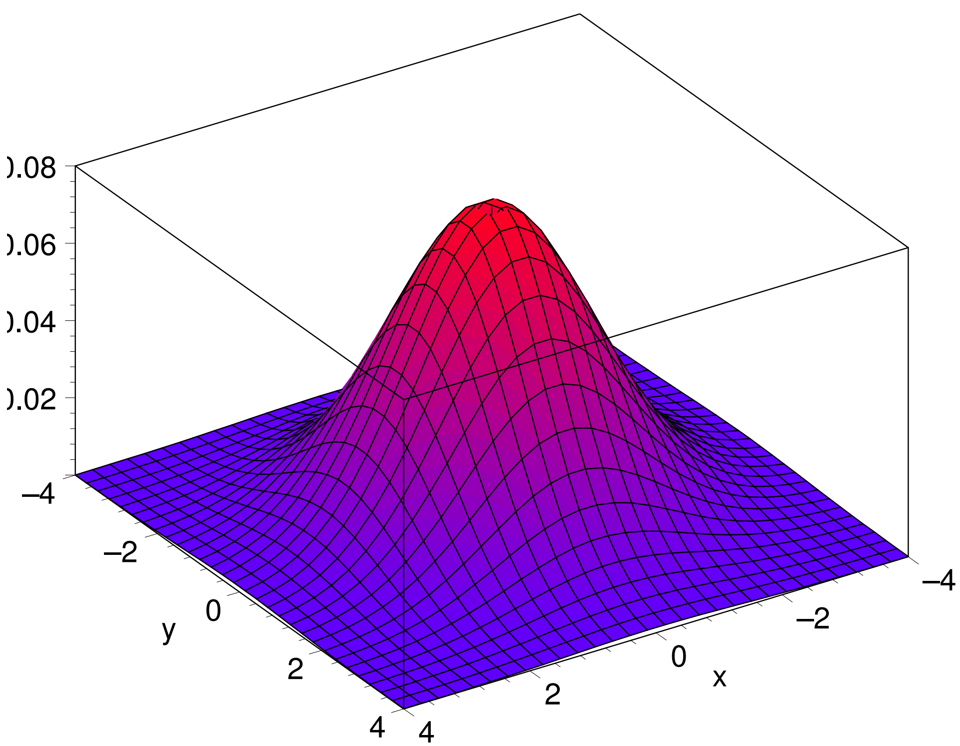



Cylinder-shaped convolution kernel. Author: B. Burgeth.

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

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

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

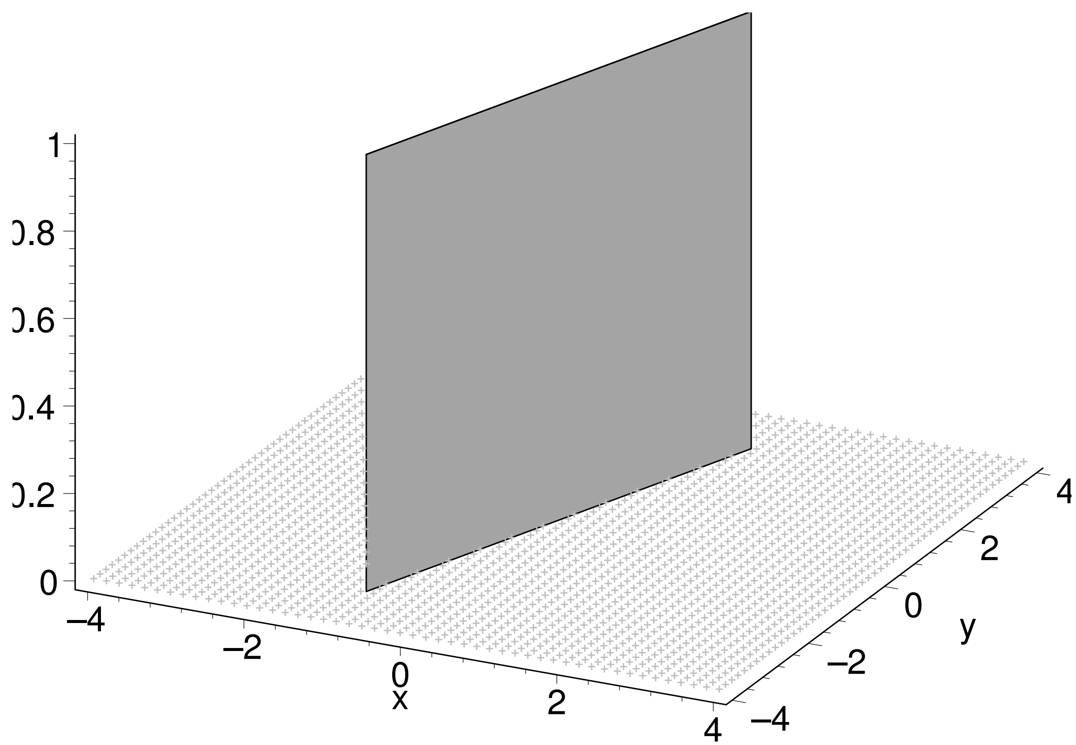



2-D Gaussian. Author: B. Burgeth.

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 29 30 31 32 33

- **Motion Blur:**

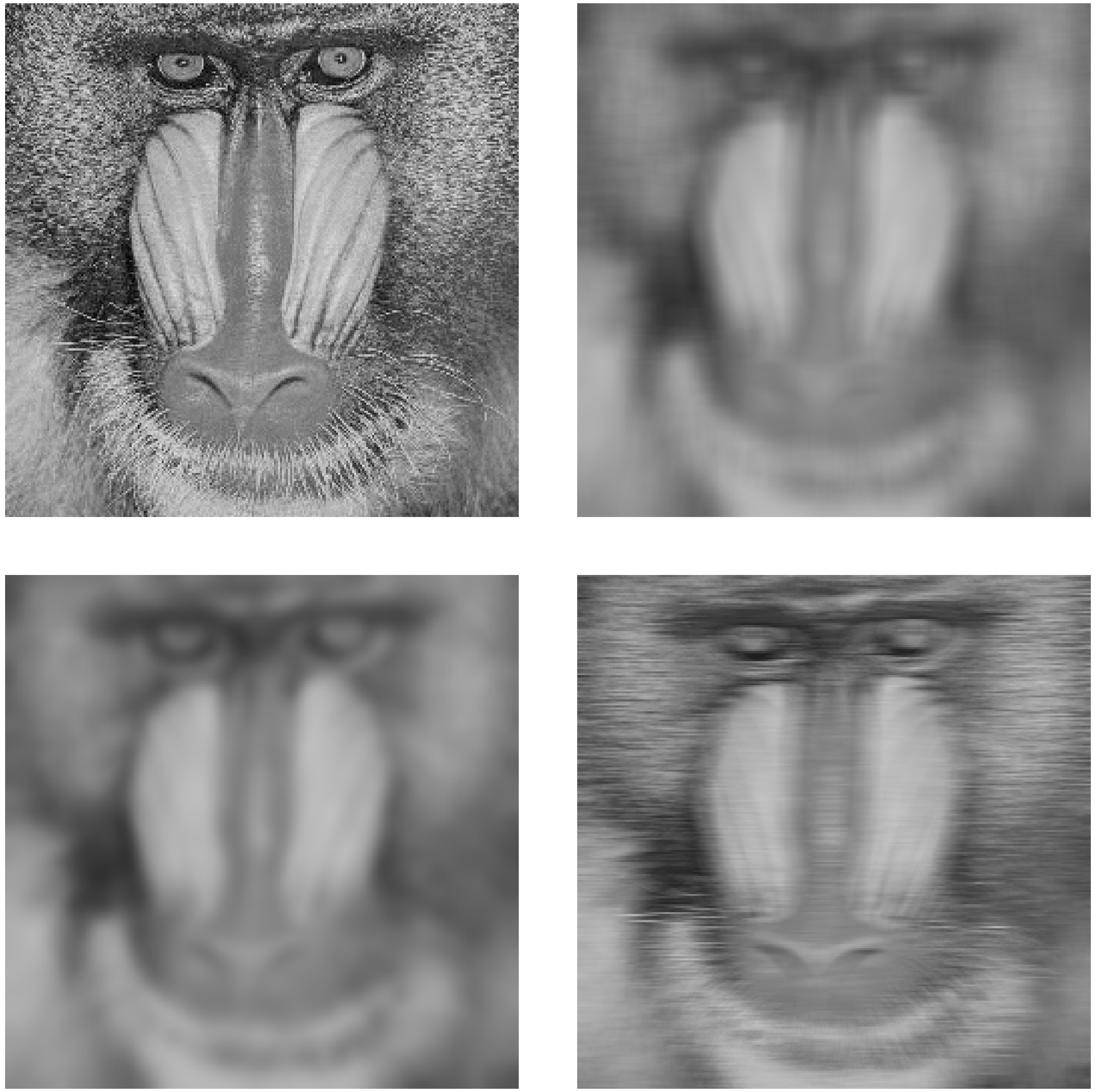
  in the simplest case (uniform motion, all objects in equal distance to camera): 1-D box function oriented along the motion direction




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

# Blur (12)

original image | out-of-focus blur



Gaussian blur | horizontal motion blur

Simulation of different types of blur.

# Lecture 2:
# Foundations II: Degradations in Digital Images

**Contents**

# Combined Blur and Noise

- Often digital images suffer from both blur and noise.
- A typical degradation model combines shift invariant blur with additive noise:

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

**Left:** Original image, $256 \times 256$ pixels. **Middle:** Blurred with a pillbox kernel of radius 5 pixels. **Right:** Further degradation by additive Gaussian noise with $\sigma = 30$. Values outside $[0, 255]$ are cropped. The result simulates a digital camera that is out of focus and suffers from sensor noise. Author: J. Weickert.

# 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.
- Shift-invariant blur can be simulated by convolution with a suitable kernel (e.g. pillbox kernel, Gaussian, 1-D box function).
- A frequently used degradation model of some initial image $g$ with a shift-invariant blur kernel $w$ and additive noise $n$ is given by

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

# References


- R. C. Gonzalez, R. E. Woods: *Digital Image Processing*. Prentice Hall, Upper Saddle River, International Edition, 2017.
  *(Chapter 5 deals with noise and blur 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)*
- H. Tijms: *Understanding Probability*. Cambridge University Press, Third Edition, 2012.
  *(elementary introduction to the probabilistic concepts used in this lecture)*