VYSOKÁ ŠKOLA
CHEMICKO-TECHNOLOGICKÁ
V PRAZE

Signal processing, image noises, image denoising, compression

Jan Švihlík

Jan.Svihlik@vscht.cz

Institute of Chemical Technology in Prague

Faculty of Chemical Engineering

Department of Computing and Control Eng.

Content

- Convolution
  - Definition, convolution theorem, implementation
- Noise models
  - Additive, multiplicative, transform of noise models
- Image noise models
  - Gaussian, Heavy-tailed, salt & pepper, quantization, PCN
- Additive noise suppression
  - Convolution filter, mask in a spectral domain, median filtration
- Suppression of signal dependent noise
  - Homomorphic filter, Anscomb transform
- Compressions

Convolution (LSI systems)

Definition – 1D convolution

$$f(t)*g(t)=\int_{-\infty}^{+\infty} f(\tau)g(t-\tau)d\tau$$

Definition – 2D convolution

$$f(x,y)*g(x,y)=\int_{\mathbf{R}\times\mathbf{R}} f(\alpha,\beta)g(x-\alpha,y-\beta)d\alpha d\beta$$

Convolution properties

$$f*g=g*f$$ - Commutativity

$$c_1 f*c_2 g=c_1 c_2(f*g)$$ - Multiplication by constant

$$f*(g_1+g_2)=f*g_1+f*g_2$$ - Distributivity

$$f_1*[f_2*f_3]=[f_1*f_2]*f_3$$ - Associativity

Convolution

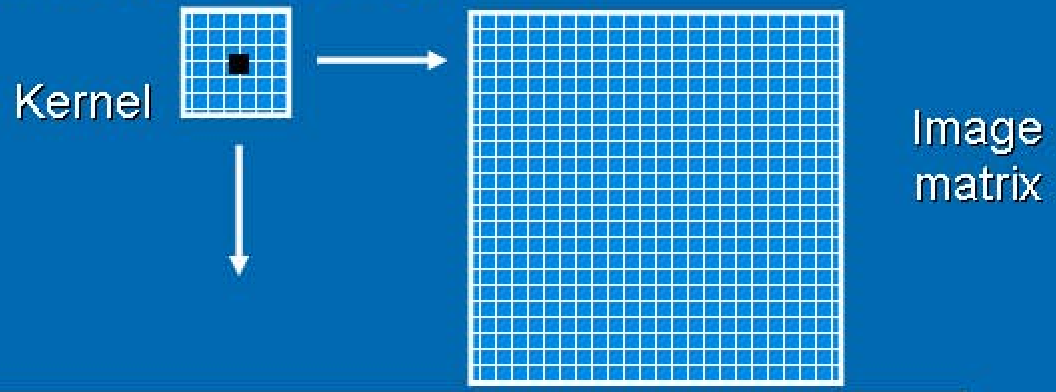
Discrete convolution

$$c(i,j)=\sum_{m=0}^{Mr}\sum_{n=0}^{Nr} r(m,n)\cdot h(i-m,j-n),$$
$$0\le i<Mr+Mh-1, 0\le j<Nr+Nh-1,$$

- r a m present convolved matrixes, Mr a Nr is height and width of r, Mh a Nh are sizes of h.

Implementation of discrete convolution

- A convolution kernel is moving across an image matrix. A sum of products between kernel coefficients and image pixels are computed. It is necessary to reserve correct boundary conditions (zero padding, mirror)



Basic methods for image noise suppression

Noise models

Additive noise

$$y = x + n$$

Multiplicative noise

$$y = x \cdot n$$

Models transform

$$e^y = e^{x+n} = e^x \cdot e^n$$

$$\log(y) = \log(x \cdot n) = \log(x) + \log(n)$$

Image noise models

Gaussian noise

- Probability density function (PDF) of Gaussian noise with mean value μ and variance σ^2 is given by

$$p_n(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}, \qquad x \in (-\infty, \infty)$$

- There also exist noises, which can be modeled as a Gaussian, but only in limiting cases, e.g., photon counting noise and film grain noise.

- Brada image contaminated by noise with $\sigma = 20$.

Image noise models

Heavy-tailed noise

- PDF of Heavy-tailed noise with μ and σ^2 is given by

$$p_n(x) = \frac{1}{2\sigma} e^{\frac{-|x-\mu|}{\sigma}}, \qquad x \in (-\infty, \infty)$$

- The conditions of the Central Limit Theorem cannot be well satisfied. Number of the terms of the sum is not so large, or small random variables are not so independent.
- A comparison of Gaussian and Heavy Tailed PDF. Heavy-Tailed PDF approaching zero more slowly .

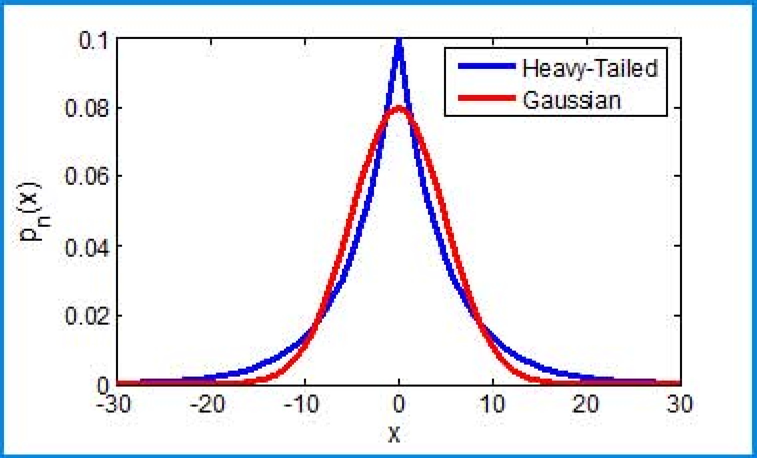


Image noise models

Salt & pepper

- Probability that a bit will be flipped (V->W) during the transmitting over a digital channel is given by $$P\left(|V-W|=2^{i}\right)=\varepsilon,\quad i=0,1,\ldots,B-1.$$
- A few pixels are set to the small value and other to the large value.

$$P(y=a)=1-\gamma,\qquad P(y=0)=\frac{\gamma}{2},\qquad P(y=255)=\frac{\gamma}{2},$$

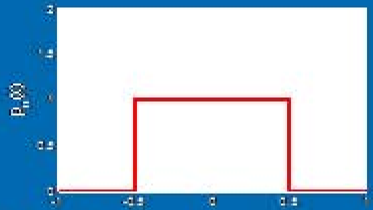
- Only the changed most significant bit (MSB) caused the origin of black or white pixel. The MSE introduced by MSB is equal to $$\varepsilon\left(2^{B-1}\right)^{2}=\varepsilon 4^{B-1}$$
- Brada image contaminated by S&P noise, γ = *0.05*.

Image noise models

Quantization noise

- Quantization noise is modeled as an uniform with q. step Δ and PDF

$$p_n(x)=\begin{cases}\frac{1}{\Delta}, & -\frac{\Delta}{2}\le n\le\frac{\Delta}{2},\\ 0, & jinde\end{cases}$$



- When the continuous random variable is transformed to the discrete random variable or when the discrete one is converted to another discrete one with fewer levels then the quantization noise is occurred.

Fixed threshold

Random threshold

Floyd-Steinberg dither

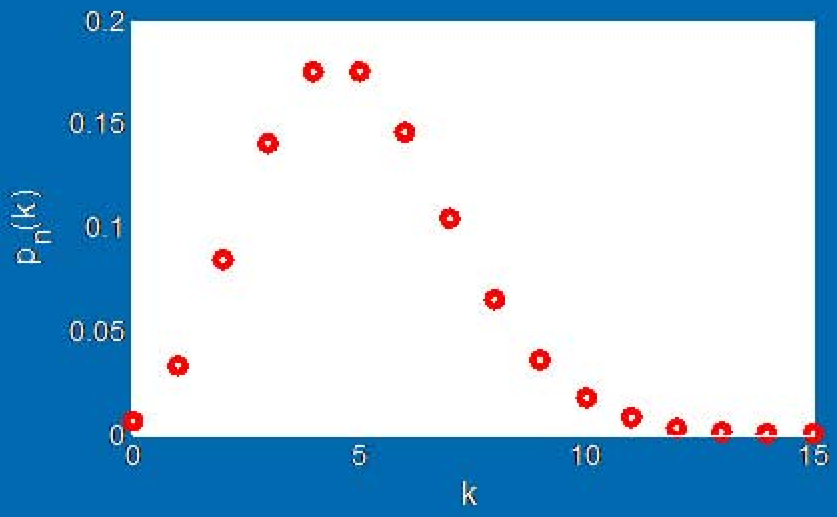
Image noise models

Photon counting noise (PCN)

- PDF is given by

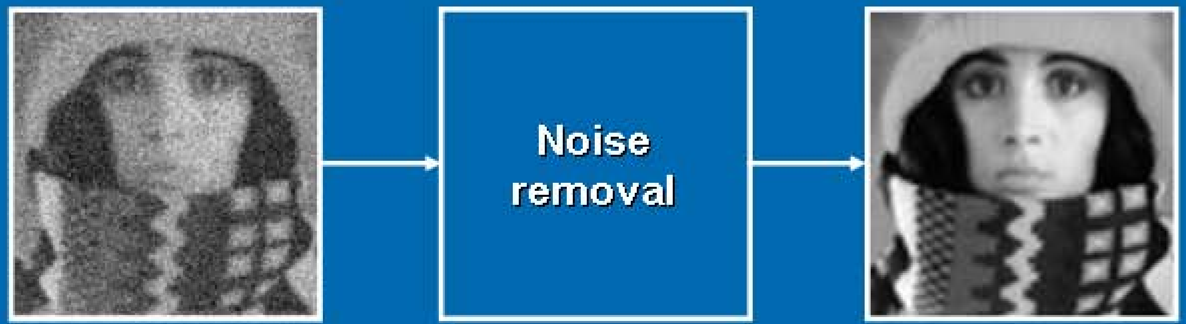
$$P(\Omega = k) = p_n(k;\lambda) = \frac{\lambda^k e^{-\lambda}}{k!}, \qquad k = 0, 1, 2, \ldots$$

- PCN is usually occurred in CCD sensors. Sensor is counting random number of photons λ in certain time interval.



Additive noise suppression

We have | We want



Convolution filtration

- It belongs to the basic noise removal methods. A convolution filter suppresses spectrum, which lies outside of useful signal. Space resolution is reduced!
- Kernel for simple averaging.

Gauss $\sigma = 20$

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

Filtered

Spectral domain

Mask in the spectral domain

- Method is based on mask application in the spectral domain. Masks can realize any filter.

- Knife-edge boundary mask causes unwanted artifacts.

$$C(u,v) = R(u,v) \cdot H(u,v) = R(u,v) \cdot \left| H(u,v) \right| \cdot \Phi(u,v),$$

Gauss $\sigma = 20$

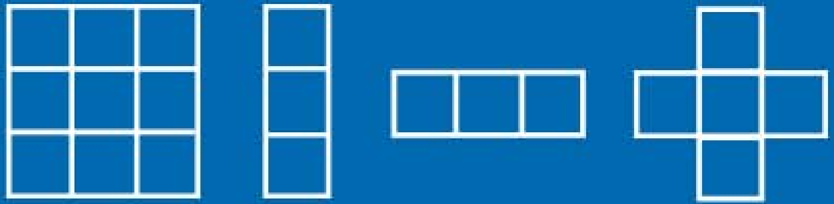
Filtered

Additive noise suppression

Median filtering

- MF is nonlinear. Pixel is substituted by median value computed in median field.
- Median *m* of random variable *X* is given

$$P(X \le m) \ge 0.5 \le P(X \ge m).$$



- Median of the series with odd term number can be explained as the series term value, where the half of the series term are greater than median and half of the series term are smaller than the median value .

Gauss $\sigma = 20$

Filtered

- better edge preservation than convolution filtering.
- best results are achieved for impulse noise .

Signal dependent noise removal

Multiplicative noise

Homomorphic filter

- Processed by taking logarithms from multiplicative model. After a denoising, exponential transfer function should be applied.

$$y = x \cdot n$$

$$\log(y) = \log(x) + \log(n)$$

Photon counting noise

Anscomb transform

- Anscomb transform stabilizes variance and transforms Poissonian data *I(λ)* into approx. Gaussian one with distribution *N(0,1)*.

$$AT\{I(\lambda)\} = 2\sqrt{I(\lambda) + \frac{3}{8}}.$$

- Time Averaging can be also used for PCN removal.

Image Compressions

Basic principles

Motivation

- Reduction of amount of data, which are to be archived or transferred.

Reduction of redundancy (lossless)

- Image usually contains correlated pixels. Pixels are statistically dependent. Thus we need to make an image decorrelation to decrease the number of bits.

Reduction of irrelevance (lossy)

- An observer cannot perceive all informations (e.g. high image frequencies) included in an image. Thus we decrease the information content of an image (we decrease the entropy), but an observer mustn't register an information loss.

Compression ratio

- Compression ratio is ratio between the non-compressed data and the compressed data.

Division of Compression Methods

Lossless (error-free) methods

- Basic
  - RLE (run length encoding)
- Statistical methods
  - VLC (variable length coding)
    - Aritmetic coding
    - Huffman coding
  - LZW (Lempel-Ziv-Welch)
- Lossless predictive methods

Lossy methods

- Transform domain
  - DCT (discrete cosine transform) - e.g. JPEG
  - DWT (discrete wavelet transform) - e.g. JPEG 2000
- Lossy predictive methods

Compressions

Transforms

- DCT (Discrete Cosine Transform)
- FFT (Fast Fourier Transform)
- DWT (Discrete Wavelet Transform)

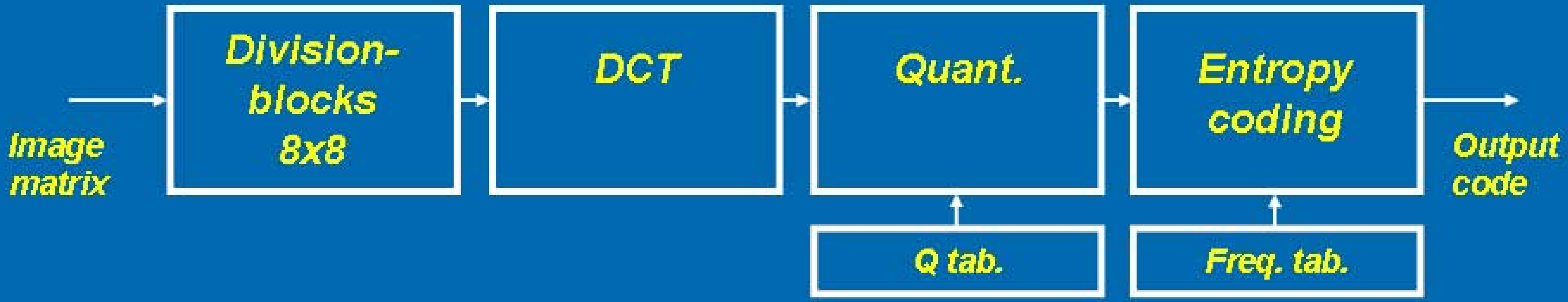
Non Transforms

- PCM (Pulse Code Modulation)
- DPCM (Differential PCM)
- LPC (Linear Predictive Coding)

Transform methods

JPEG (Joint Photographic Experts Groups)

- JPEG uses the discrete cosine transform (DCT) and it should be divided into these steps:
  - Image division into blocks
  - DCT evaluation in block of size 8x8
  - DCT coefficients quantization
  - Coding of DC component and “cik cak” reading
  - Entropy coding (Huffman or aritmetic)



Discrete Cosine Transform

Discrete cosine transform (DCT)

- DCT uses an orthogonal system of discrete cosine function.
- DCT should be seen as special case of DFT (discrete Fourier transform) with even extension of image function. Thus we obtain only cosine components = real components.
- DCT definition (size 8x8):

$$F(u,v)=\frac{1}{4}C(u)C(v)\sum_{x=0}^{7}\sum_{y=0}^{7}f(x,y)\cos\frac{(2x+1)\pi\cdot u}{16}\cos\frac{(2y+1)\pi\cdot u}{16},$$

$$C(u),C(v)=\frac{1}{\sqrt{2}}, \qquad pro\ u,v=0$$

$$C(u),C(v)=1, \qquad elsewhere$$

JPEG

Division into blocks

- Image is divided into block of size 8x8 pixels. Pixel value is transformed from the range $[0 \ldots 2^b - 1]$ into the range $[-2^{b-1} \ldots 2^{b-1} - 1]$, where b denotes the number of bits.

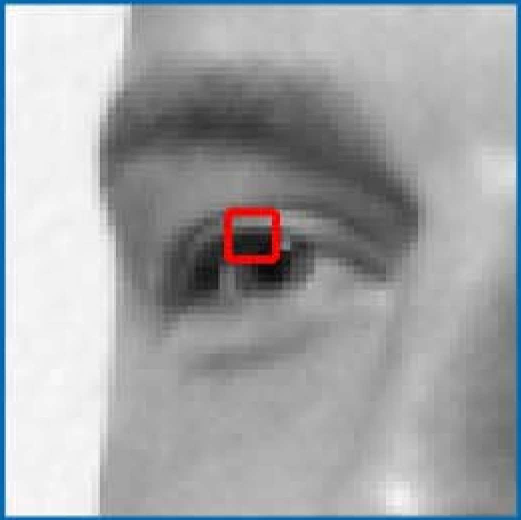
Discrete cosine transform

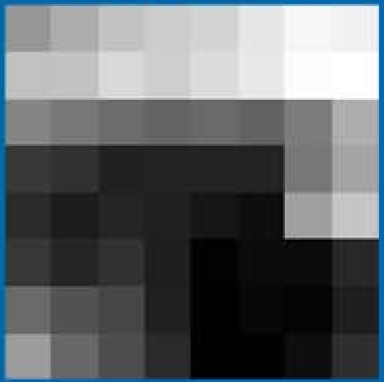
- Every block 8x8 is transformed using DCT. We obtain 64 frequency coefficients. Coefficient $F(0,0)$ stands for the DC component of an image. All other coefficients represent certain space frequencies.

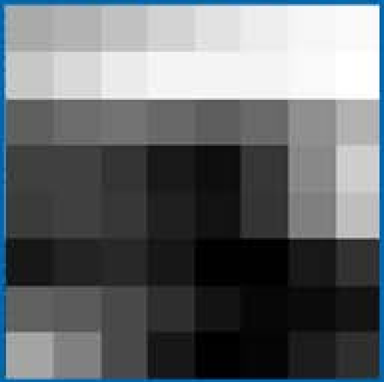
JPEG

Quantization

- Every coefficients *F(u,v)* is quantized using quantization table *Q(u,v)*.







$$Q = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

$$F_Q = round\left(\frac{F(u,v)}{Q(u,v)}\right)$$

$$e = \frac{1}{64}\sum_{x=1}^{8}\sum_{y=1}^{8}\left|f(x,y) - f_k(x,y)\right| = 5.16$$

JPEG

DC component coding and "cik-cak" reading

- The DC component is coded separately as a difference from the DC component of previous blocks. All quantized frequency coefficients are read "cik-cak".

30	0	0	0	0	0	0	0
-7	-8	1	1	0	0	0	0
-11	6	0	-1	0	0	0	0
-5	-3	0	0	0	0	0	0
-7	-3	2	0	0	0	0	0
-4	4	0	0	0	0	0	0
-1	0	-1	0	0	0	0	0
-3	1	0	0	0	0	0	0

JPEG

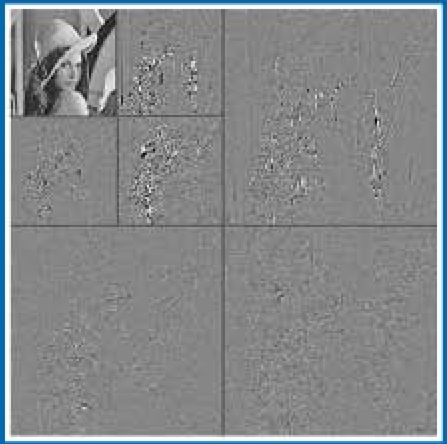
Entropy coding

- Entropy coding is realized using Huffman or arithmetic coder.

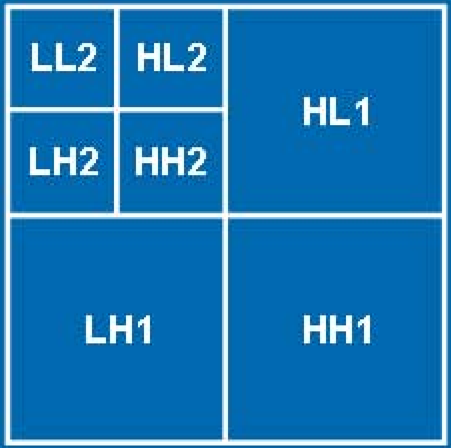
Compression using DWT

Image

Decomposition



Wavelet bands



- Based on wavelet coefficients thresholding.
- Coefficients are thresholded using threshold ε. It means that coefficients less than threshold are set to zero, the greater one are preserved.

- $\varepsilon > 0$ *Lossy compression*
- $\varepsilon = 0$ *Lossless compression*

Compression using DWT

Compression using Haar transform

Without c. $\varepsilon = 50$ *Lossy c.*

- We obtain sparse matrixes of wavelet bands (HL, LH, HH) after thresholding.
- Practically all wavelet coefficients are zeros -> simple coding.

References

[1] Boncelet, Ch.: *Image Noise Models. Handbook of Image and Video Processing*, Academic Press, 2000, pp. 325-335.

[2] Gonzales, C – Woods, R.: *Digital Image Processing*. Second edition. New Jersey, 2002. ISBN 0-201-18075-8.

Any questions

...?