Multimedia

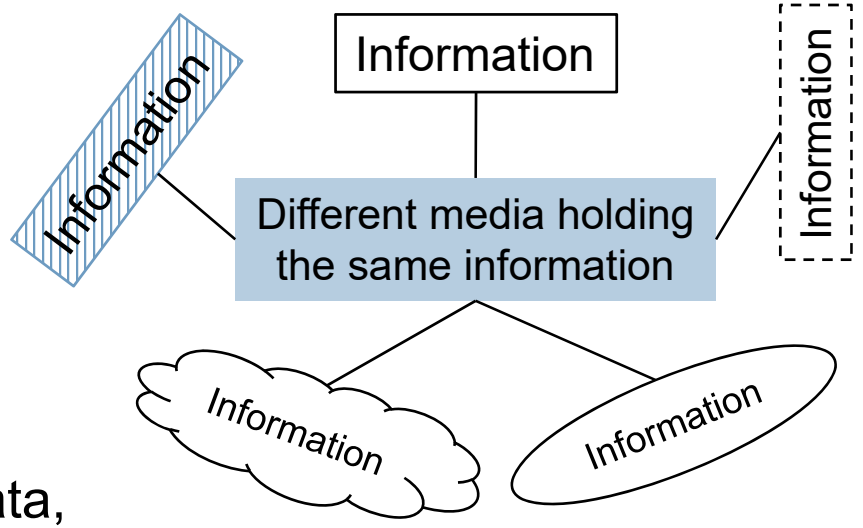
§3 Basics of data compression

Prof. Dr. Georg Umlauf

Content

Why data compression?

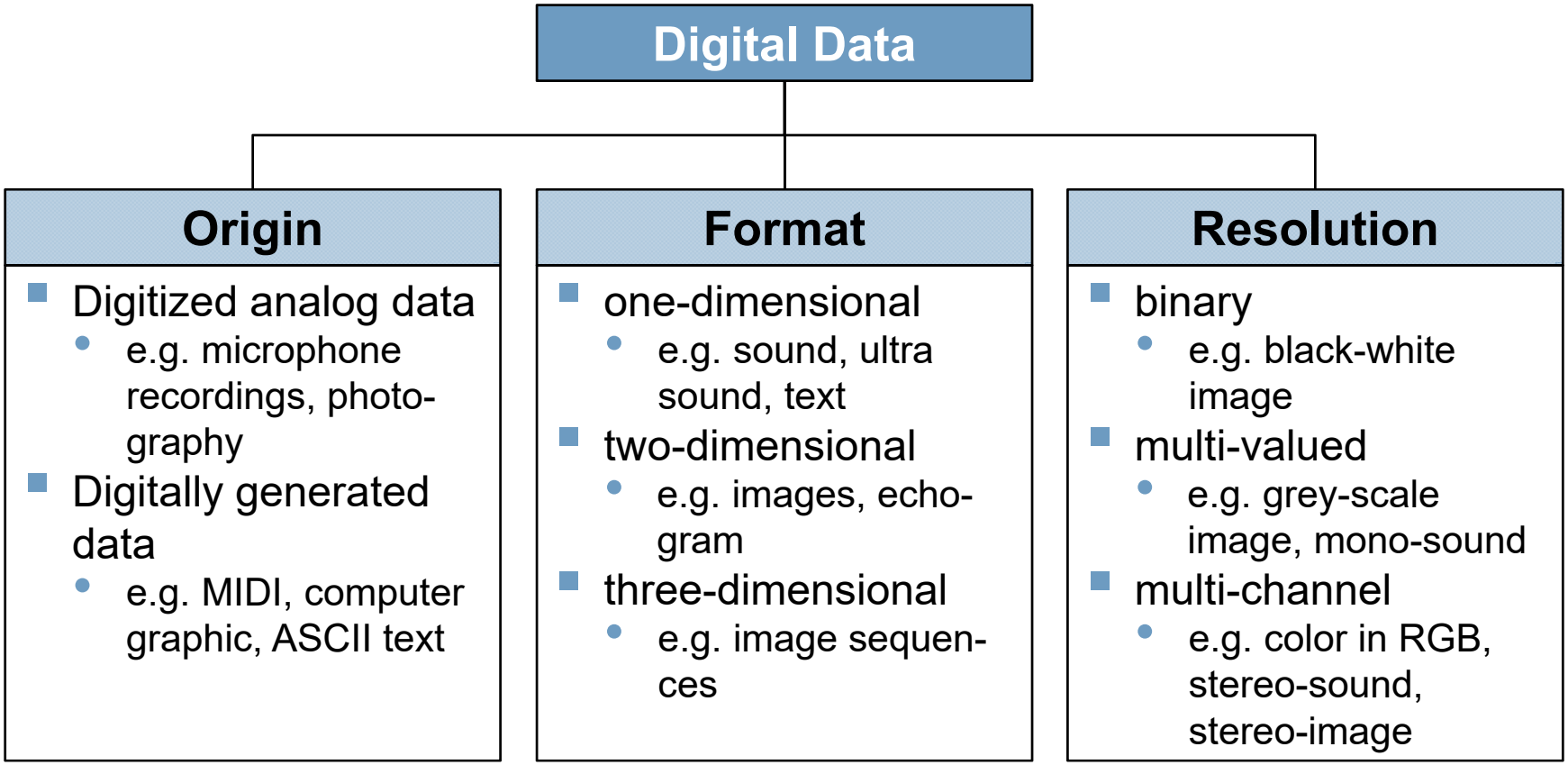
- Limited resources
  - Storage space
  - Transmission rate
- Data becomes increasingly complex
  - complex image-, video- and 3d-data,
  - possible due to powerful hardware.



➡ Storage and transmission of such data requires efficient and effective reduction of data.

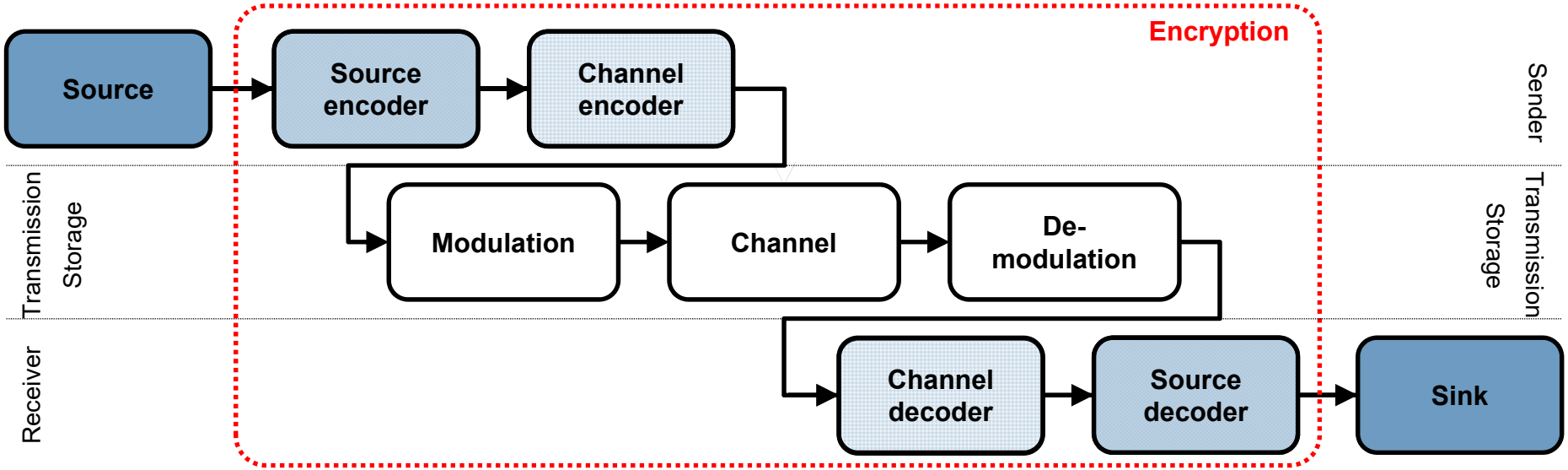
➡ Almost all multimedia data, e.g. audio-, image-, or video-data, is used today in compressed form.

Why data compression?



Encoding

- **Source coding:** Compression of data by sender.
- **Channel coding:** Protection of the data against random errors during transmission or storage.
- **Cryptography:** Protection of the data against interception or modification by unauthorized others.
- **Modulation:** Superimpose data onto the physical medium.



What is data compression?

- Reduce amount of (digital) data, necessary to represent a certain information.

- Compression methods
  - Many different approaches and techniques.
  - Developed for different applications.
  - Perform very different for different applications.

Content

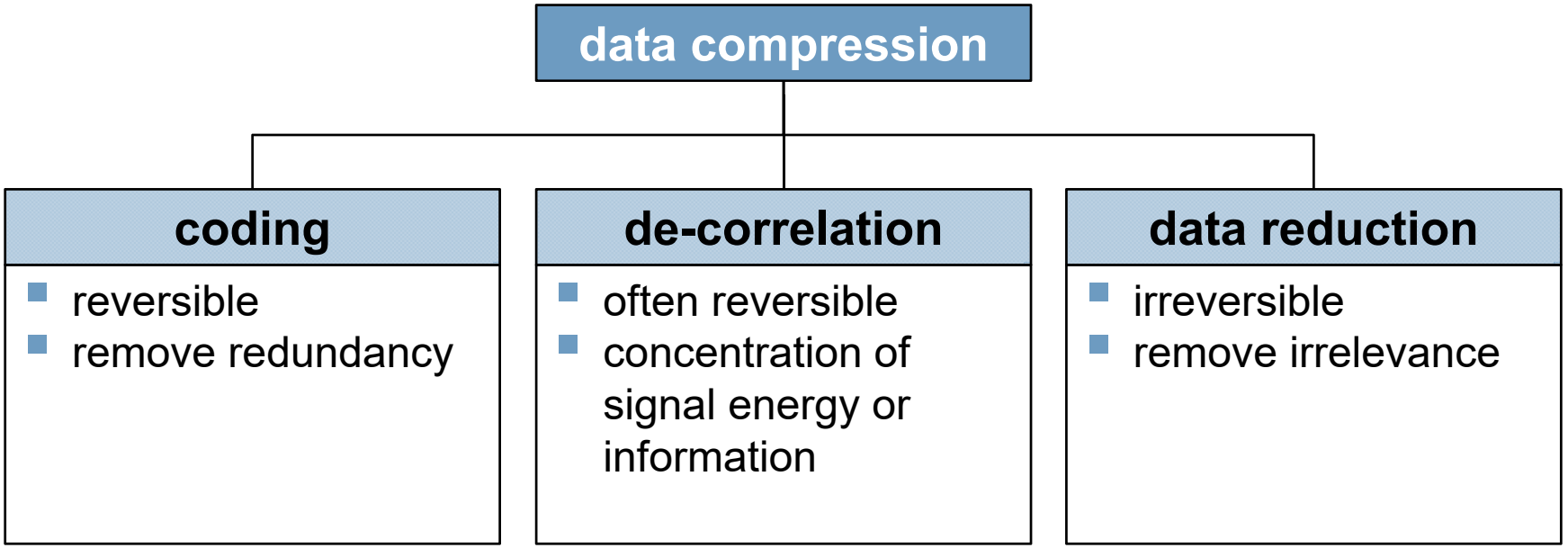
§3.2 Data compression

What is data compression based on?

- Content of files can be
  - redundant (repetitions, patterns, etc.) and/or
  - partially unimportant (too much detail, etc.).
- ➡ Use redundancy, regularity, repetitions, correlation, similarity
  - ➡ **Coding**
- ➡ Unimportant Information can be ignored.
  - ➡ **Data Reduction**
  - ➡ Depended on the perception of humans.

- ➡ Both aspects can be used to shorten the representation of the information, i.e. compress the representation of the information.

What is data compression based on?



Lossless compression

- Text-, program-, and data files must be compressed without loss of information.
- Only redundancy can be remove.
- ➡ Original can (in principle) be reconstructed perfectly without errors.
- E.g.: Huffman-Coding, run-length-coding, arithmetic Coding, LZW coding.

Lossy compression

- For image-, video-, and audio-files the compressed file can differ slightly, as long as the difference is unrecognizable (invisible or inaudible).
  - Used physiological properties of human eyes and ears.
  - ➡ Compress file to target size.
  - ➡ Compress data to target transfer rate.
- ➡ There is a difference between the original and the decoded object.
  - ➡ Data cannot be reconstructed perfectly without errors.
- Allows for larger compression rates.
- E.g.: JPEG, MPEG

Application: text files

- Order of characters in a file is usually not arbitrary.
- Letters of an alphabet occur with different frequencies.
- Encode frequent letters using „short“ symbols and infrequent letters using „long“ symbols.
  - ➡ Reduce the average length of symbol for the letters.
  - ➡ Encode analogously frequent and infrequent letter sequences.

Application: audio files

- A single tone (sinus) can be represented more compactly than a complex sound.

Application: image files

- Not all areas in an image have the same degree of complexity:
  - Large, homogenous areas (e.g. single-colored or regular patterns) can be represented more compactly.
  - Highly detailed areas need a more expensive representation.

Application: video files

- Similar to images:
  - Use coherence between succeeding frames.
  - Image areas that do not change, do not need to be encoded again.

Aspects of data compression (1)

- Compression and de-compression is realized in so-called **Codecs**.
- Complexity and run-time
  - Dependent on application, data volume and the data itself.
- Compression performance, compression rate
  - Increases usually with increasing recourses (computing power, storage space).
  - $\text{Compression rate} = \frac{\text{Un-compressed size of the data}}{\text{Compressed size of the data}}$ [in percent].
  - $\text{Saving} = 1 - \frac{1}{\text{Compression rate}}$ [in percent].

Aspects of data compression (2)

- Pre- or Post-Processing
  - (de)-coding of the data „online“ during the generation/transmission?
  - (de)-coding of the data „offline“ in a pre-/post-process?
- Robustness
  - What is the effect of an error in a compressed file?
- Usability
  - Is the compressed file self-extracting or does the receiver need a de-coder?
- Edit ability
  - Can individual blocks of the compressed file be modified without changing the rest of the file?

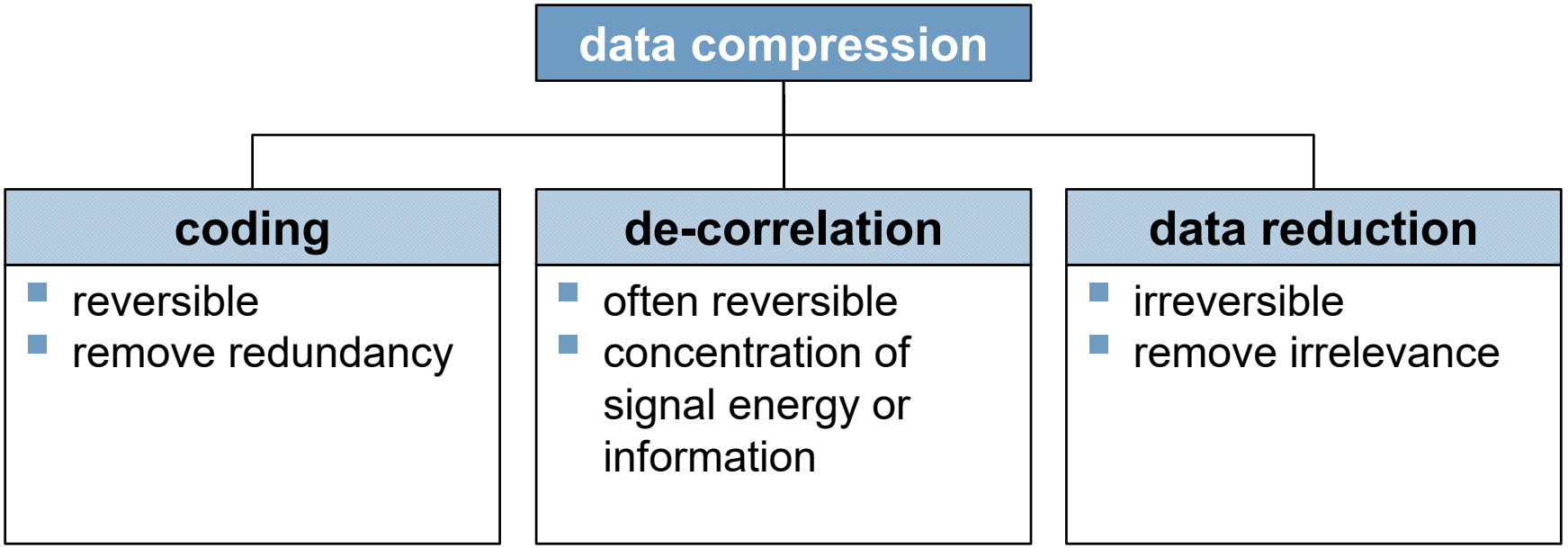
Examples for compressed file formats

- Text: ZIP, RAR
- Images: GIF, JPEG, PNG, TIFF
- Video: AVI, MPEG
- Audio: WAV, MP3, MPEG
- Hybrid: PDF
- Others: ZIP, GZIP, RAR

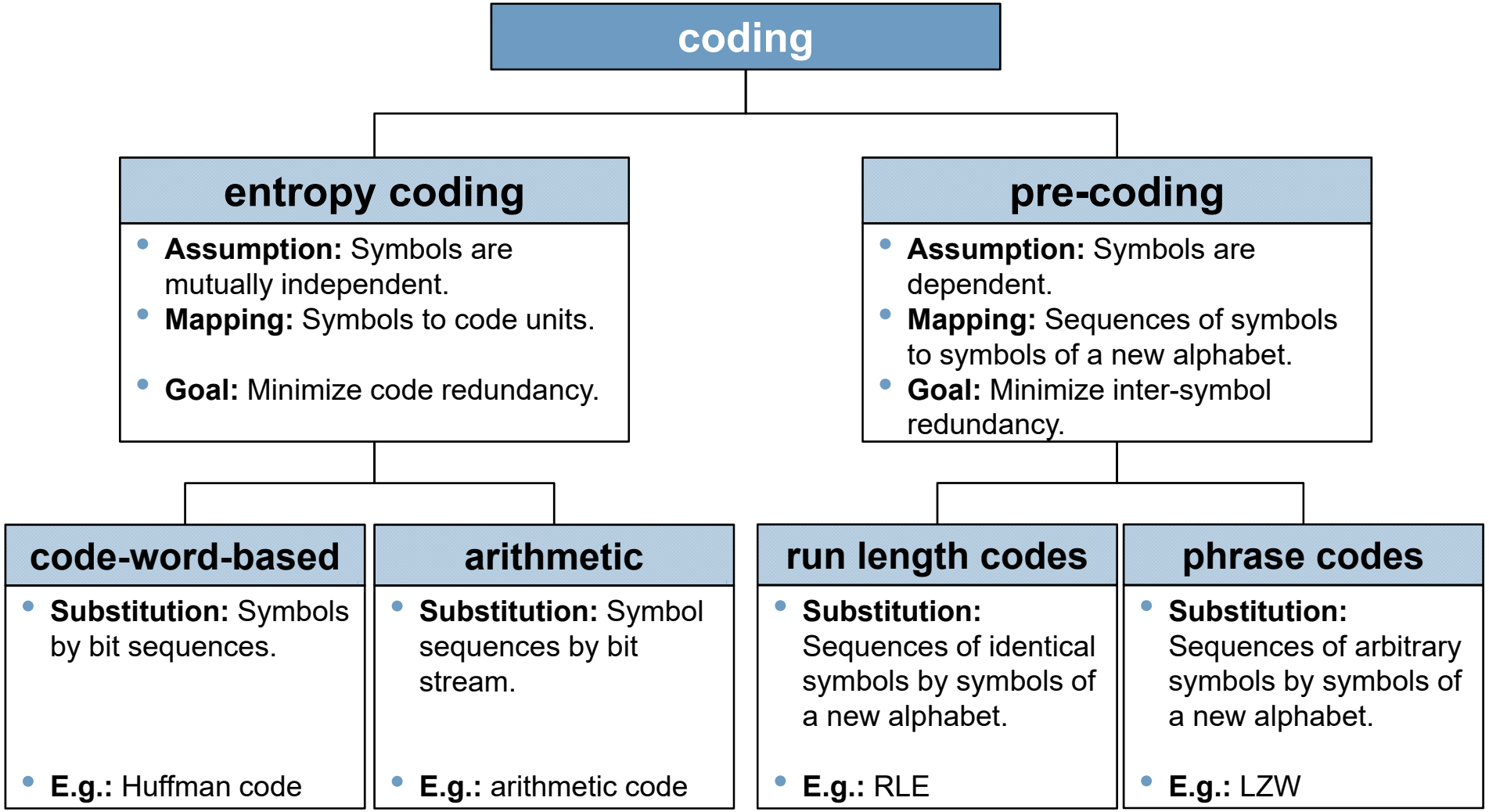
Content

§3.3 Classification of compression methods

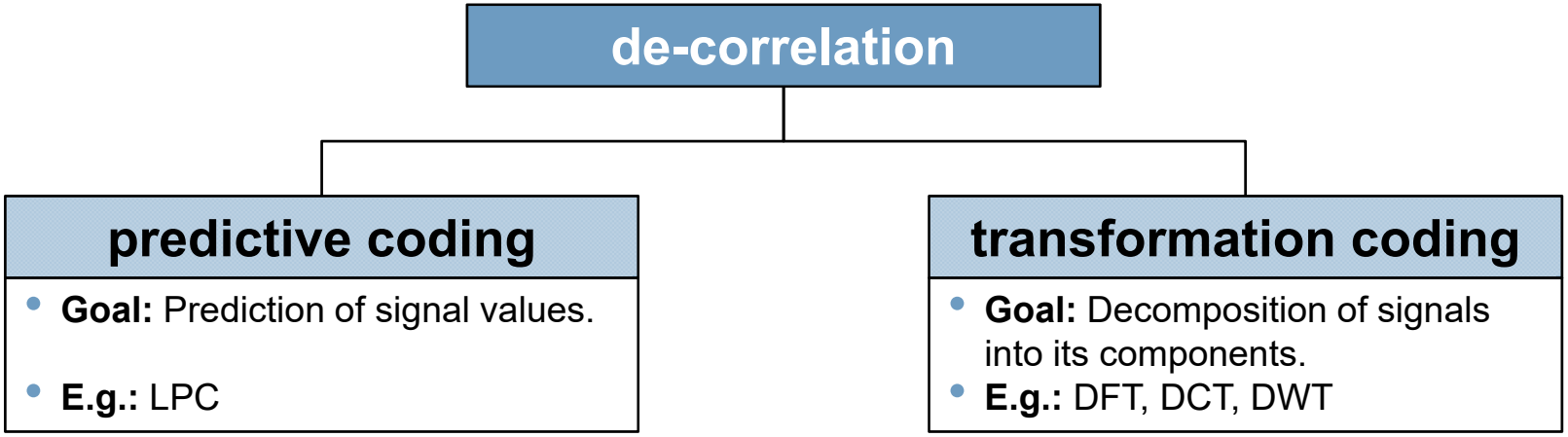
Overview



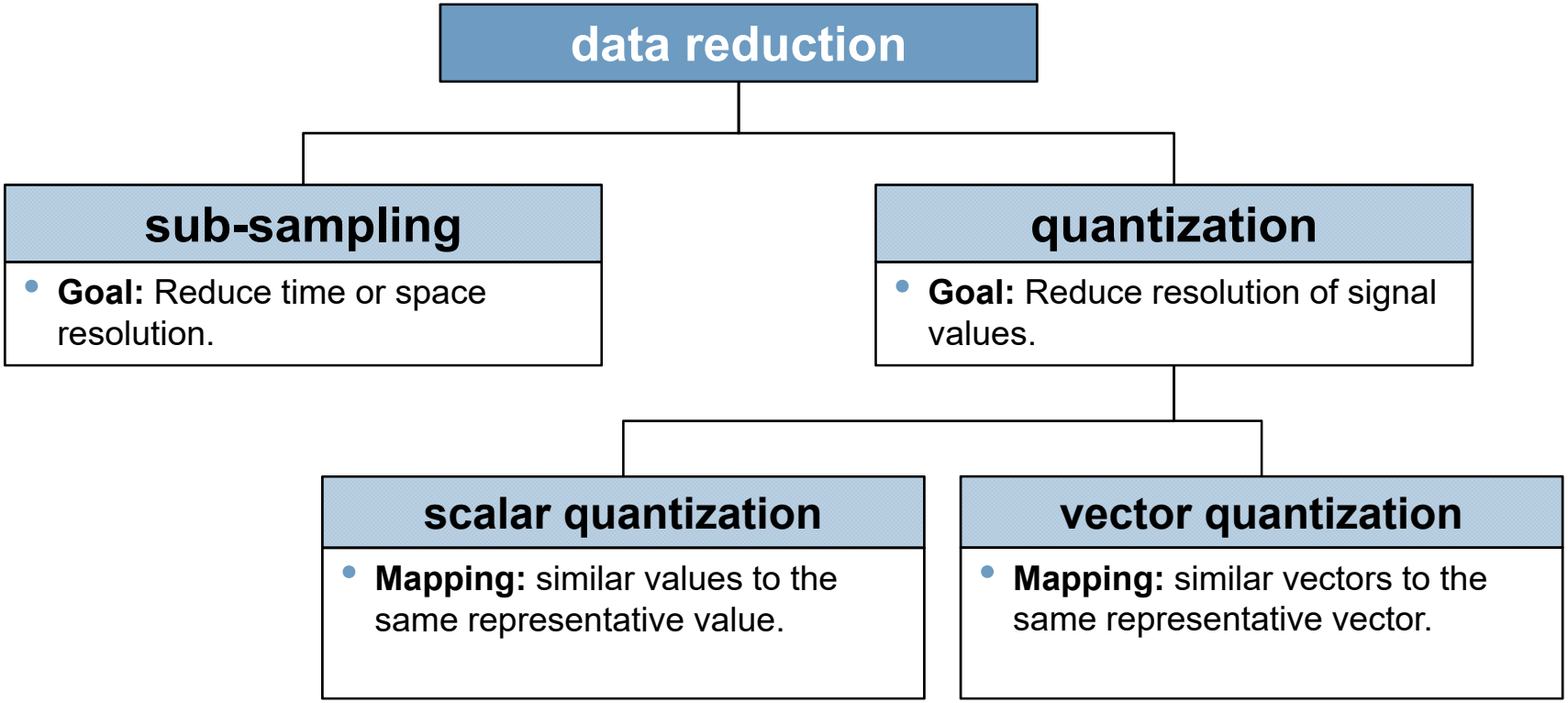
§3.3 Classification of compression methods



§3.3 Classification of compression methods

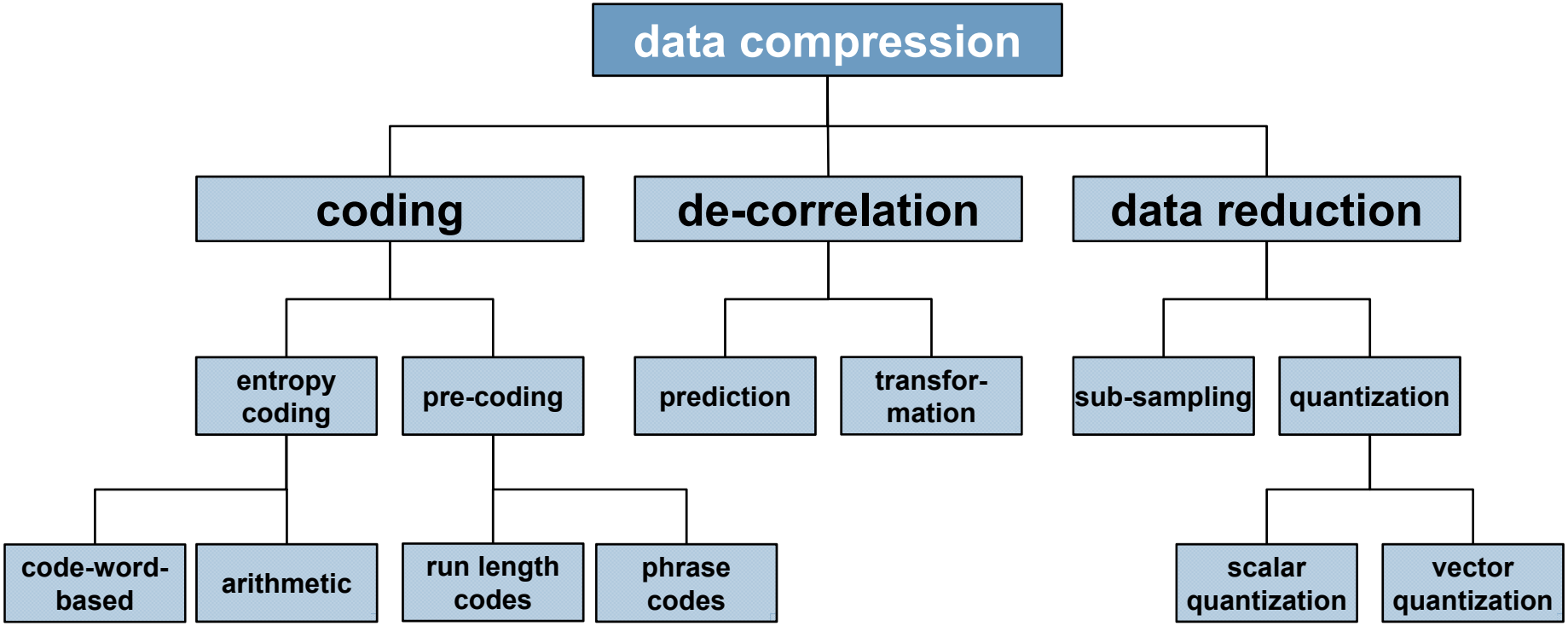


§3.3 Classification of compression methods



§3.3 Classification of compression methods

Overview



§3.3 Classification of compression methods

Overview of compression methods

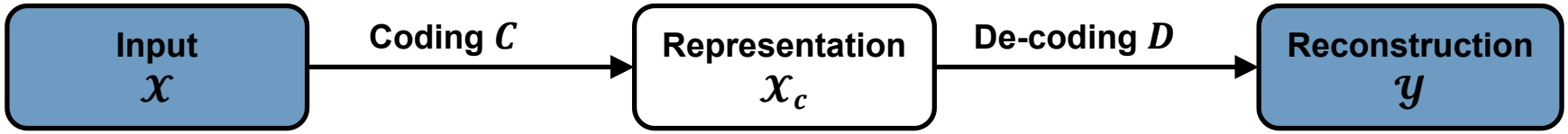
Type	Form	Method	Chapter
Lossless compression		• RLE (run-length coding)	§4.1
	Statistical coding	• LZW-coding • Huffman-coding • arithmetic coding	§4.4 §4.2 §4.3
Lossy compression	Layered coding	• Bit-position • Sub-sampling • Sub-band-coding	§5.1 §7.3.2
	Quantization	• Vector quantization	§5.2
	Transformation	• DFT/FFT (Fourier-Transformation) • DCT (Cosines-Transformation) • DWT (Wavelet-Transformation)	§6.1 §6.2 §6.3
	Prediction	• LPC (linear predictive coding)	§7.3.3
Hybrid compression		• JPEG • MPEG • H.264, H.265	§8.4.6 §9.2 §9.2

Content

§3.4 Measures for compression methods

How to evaluate the performance of a compression method?

1. Compare data size before and after the compression.
   - ➡ Compare x and x_c, e.g. compression rate.
2. Compare data quality before and after the compression.
   - ➡ Compare x and y, e.g. distortion measures.



- Lossless compression, if $x = y$.
- Lossy compression, if $x \neq y$.

Measures for the quality of a compression method

- Compression performance, compression rate
  - Compression rate $= \frac{\text{Un-compressed size of the data}}{\text{Compressed size of the data}}$ [in percent].
  - Saving $= 1 - \frac{1}{\text{Compression rate}}$ [in percent].
  - Compression factor: Ratio of compressed size of a file to the un-compressed size of the file.
    - Example: Compression factor 1:100 means, the compressed file size is a hundredth of the original file size.
- Speed for the compression.
- Speed for the de-compression.

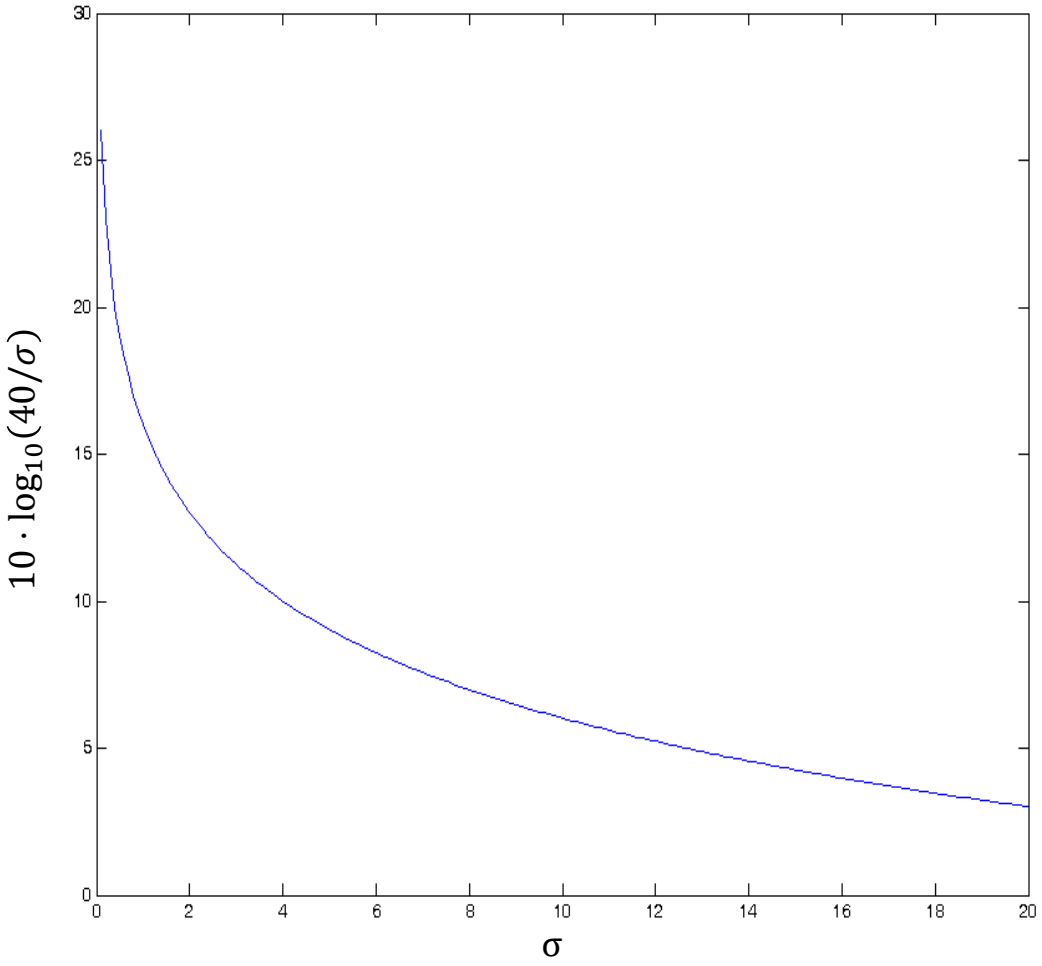
§3.4 Measures for compression methods

Distortion measures for $\mathcal{X} = (x_1, \ldots, x_n), \mathcal{Y} = (y_1, \ldots, y_n)$

- Distortion: $\mathcal{Y} - \mathcal{X}$.
- Absolute error: $|x_i - y_i|, \quad i = 1, \ldots, n.$
- Squared error: $(x_i - y_i)^2, i = 1, \ldots, n.$
- Average distance (AD): $d_1 := \frac{1}{n} \sum_{i=1}^{n} |x_i - y_i|$
- Mean squared error (MSE): $\sigma^2 := \frac{1}{n} \sum_{i=1}^{n} (x_i - y_i)^2$
- Signal-Noise-Ratio (SNR): $SNR = \sigma_{\mathcal{X}}^2 / \sigma^2$ with $\sigma_{\mathcal{X}}^2 := \frac{1}{n} \sum_{i=1}^{n} x_i^2$.
- Logarithmic SNR: $SNR(dB) = 10 \log_{10} \sigma_{\mathcal{X}}^2 / \sigma^2$.
- Peak-SNR (PSNR): $PSNR = \max_i x_i^2 / \sigma^2$.
- Logarithmic PSNR: $PSNR(dB) = 10 \log_{10} (\max_i x_i^2 / \sigma^2)$.

Trend of the PSNR

- A larger PSNR corresponds usually to a better reconstruction quality.



§3.4 Measures for compression methods

Example:

Uncompressed original

PSNR=45.53dB

PSNR=36.81dB

PSNR=31.45dB

Source: Wikipedia

§3.4 Measures for compression methods

- **Problem:** How do these measures correlate with the perceived distortion?

Original image Lena

Compressed image to 25% (jpg)

Goal

- What is source coding?
- What is the difference between lossless and lossy compression?
- When is lossless or lossy compression used?
- What is PSNR?