Image and Video Coding I: Fundamentals

Thomas Wiegand

Technische Universität Berlin
Organization

Vorlesung: Donnerstag 10:15-11:45
Raum EN-368

Material: http://www.ic.tu-berlin.de/menue/studium_und_lehre/

Literatur:
Introduction
Motivation for Source Coding

Source Coding / Data Compression
- Efficient transmission or storage of data
- Use less throughput for given data
- Transmit more data for given throughput

Source Coding: Enabling Technology
- Enables new applications
- Makes applications economically feasible

Examples
- Distribution of digital images
- Distribution of digital audio (first mobile audio players)
- Digital television (DVB-T, DVB-T2, DVB-S, DVB-S2)
- Internet video streaming (YouTube, Netflix, Amazon, ...)

T. Wiegand (TU Berlin) — Image and Video Coding: Introduction
Motivation

Practical Source Coding Examples

**File Compression**
- Compress large document archives
  - GZIP, WinZIP, WinRar, ...

**Audio Compression**
- Compress audio for storage on mobile device
  - FLAC, MP3, AAC, ...

**Image Compression**
- Compress images for storage and distribution
  - Raw camera formats, JPEG, JPEG-2000, JPEG-XR, ...

**Video Compression**
- Compress video for streaming, broadcast, or storage
  - MPEG-2, H.264 | AVC, VP9, H.265 | HEVC, ...
Types of Compression

Lossless Compression
- Invertible / reversible form of data compression
- Original input data can be completely recovered
- Required for document compression
- Examples: Lempel-Ziv coding (GZIP), FLAC, JPEG-LS

Lossy Compression
- Not invertible
- Only approximation of original input data can be recovered
- Achieves much higher compression ratios
- Dominant form of compression for media data (audio, images, video)
- Examples: MP3, AAC for audio
  - JPEG, JPEG-2000 for images
  - MPEG-2, H.264 | AVC, H.265 | HEVC for video
Source Data

Analog Signals
- Continuous-time/space and continuous-amplitude signals
- Typically resulting from physical measurements
- Most signals in reality (audio and visual signals)

Digital Signals
- Discrete-time/space and discrete-amplitude signals
- Often generated from analog signal
- Sometimes directly measured

Input Data for Source Coding
- Require digital signals
- Need to be stored and processed with a computer
- Compression methods are computer programs
Analog-to-Digital Conversion

**Sampling**
- Maps continuous-time/space signal into discrete-time/space signal

**Quantization**
- Maps continuous-amplitude signal into discrete-amplitude signal
- Simplest case: Rounding
Impact of Sampling (Spatial Resolution)

400 × 300 samples

200 × 150 samples

100 × 75 samples

50 × 38 samples
Impact of Quantization (Bit Depth)

- **8 bits per component**
- **4 bits per component**
- **3 bits per component**
- **2 bits per component**
Source Data Sampling and Quantization

Raw Images and Videos

Single-Component Image
- Matrix of integer samples
  \[ s[x, y] \]
- Characterized by
  - Number of samples in horizontal and vertical direction \( W \times H \)
  - Sample bit depth \( B \)

Color Images
- Three color components (typically RGB or YCbCr)
- Additionally characterized by color sampling format

Videos
- Sequence of images
- Additionally characterized by frame rate
Source Data Sampling and Quantization

**Color Sampling Formats**

- **RGB**
- **YCbCr 4:4:4**
- **YCbCr 4:2:2**
- **YCbCr 4:2:0**

Most common format
**Raw Data Rate**

- Bit rate $R$ of raw data format

$$R = \text{(samples per time unit)} \cdot \text{(bit depth per sample)}$$  \hspace{1cm} (1)

**Example: Full High Definition (HD) Video**

- $1920 \times 1080$ luma samples, 50 frames per second (Europe)
- 4:2:0 chroma format (chroma components with 1/4 luma resolution)
- 8 bits per sample

$$\Rightarrow \text{raw data rate} = 50\, \text{Hz} \cdot 1920 \cdot 1080 \cdot (1 + 2/4) \cdot 8\, \text{bits} \approx 1.24\, \text{Gbit/s}$$

**Example: Ultra High Definition (UHD) Video**

- $3840 \times 2160$ luma samples, 60 frames per second (USA, Japan)
- 4:2:0 chroma format, 10 bits per sample

$$\Rightarrow \text{raw data rate} = 60\, \text{Hz} \cdot 3840 \cdot 2160 \cdot (1 + 2/4) \cdot 10\, \text{bits} \approx 7.5\, \text{Gbit/s}$$
**Typical Video Communication Scenario**

- **Capture**: Video samples
- **Pre-processing**: Raw input
- **Video encoder**: Encoded bitstream
- **Transmission channel (can be replaced by storage)**
  - **Channel encoder**
  - **Modulator**
  - **Channel**
  - **Demodulator**
  - **Decoder**
- **Received bitstream**: Video samples
- **Video decoder**: Raw output
- **Post-processing**: Raw output
- **Display and perception**: Displayed video
Typical Video Communication Scenario

- Capture
- Pre-processing
- Raw input video samples
- Video encoder
- Encoded bitstream
- Channel encoder
- Transmission channel (can be replaced by storage)
- Channel decoder
- Demodulator
- No transmission errors
- Received bitstream
- Video decoder
- Post-processing
- Raw output video samples
- Display and perception
Typical Video Communication Scenario

1. Capture
2. Pre-processing
3. Video encoder
4. Transmission channel
5. Channel encoder
6. Modulator
7. Channel
8. Demodulator
9. Channel decoder
10. Video decoder
11. Post-processing
12. Raw output video samples
13. Bitstream
14. Display and perception

The Source Coding Problem
## Application Examples

<table>
<thead>
<tr>
<th></th>
<th>HD movie on Blu-ray Disc</th>
<th>UHD broadcast over DVS-S2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>raw video format</strong></td>
<td>1920×1080 luma samples</td>
<td>3840×2160 luma samples</td>
</tr>
<tr>
<td></td>
<td>4:2:0 chroma format</td>
<td>4:2:0 chroma format</td>
</tr>
<tr>
<td></td>
<td>8 bits per sample</td>
<td>10 bits per sample</td>
</tr>
<tr>
<td></td>
<td>24 frames per second</td>
<td>60 frames per second</td>
</tr>
<tr>
<td><strong>raw data rate</strong></td>
<td>ca. 600 Mbit/s</td>
<td>ca. 7.5 Gbit/s</td>
</tr>
<tr>
<td><strong>channel bit rate</strong></td>
<td>36 Mbit/s (read speed)</td>
<td>58 Mbit/s (8PSK 2/3)</td>
</tr>
<tr>
<td><strong>video bit rate</strong></td>
<td>ca. 20 Mbit/s</td>
<td>ca. 25 Mbit/s</td>
</tr>
<tr>
<td><strong>required compression</strong></td>
<td><strong>ca. 30:1</strong></td>
<td><strong>ca. 300:1</strong></td>
</tr>
</tbody>
</table>
The Basic Communication / Source Coding Problem

Basic Source Coding Problem

Two equivalent formulations:

1. Representing source data with highest fidelity possible within an available bit rate
2. Representing source data using lowest bit rate possible while maintaining a specified reproduction quality

Source Codec

- **Source codec**: System of encoder and decoder
### Characteristics of Source Codecs / Overall Communication

- **Bit rate**: Throughput of the communication channel
- **Quality**: Fidelity of the reconstructed signal
- **Delay**: Start-up latency, end-to-end delay
- **Complexity**: Computation, memory, memory access

### Practical Source Coding Problem

Given a maximum allowed complexity and a maximum delay, achieve an optimal trade-off between bit rate and reconstruction quality for the transmission problem in the targeted application.

In this course:

- Will concentrate on source codec
- Ignore aspects of transmission channel (e.g., transmission errors)
Image Compression Example

Original Image (1024 × 678 image points, 945 KB)
Image Compression Example

Lossless Compressed: GZIP

67.03 %

100 %
Image Compression Example

Lossless Compressed: PNG

46.50 %

100 %
Image Compression Example

Lossy Compressed: JPEG (Quality 95)

100 %

12.03 %

100 %
Image Compression Example

Lossy Compressed: JPEG (Quality 75) 3.18 %

100 %
Image Compression Example

Lossy Compressed: JPEG (Quality 50)
Image Compression Example

Lossy Compressed: JPEG (Quality 25)
Image Compression Example

Lossy Compressed: JPEG (Quality 1)
Trade-Off between Quality and Compression Ratio

Coding Efficiency

- Ability to trade-off bit rate and reconstruction quality
- Want **best reconstruction quality for a given bitrate** (or vice versa)

![Graph showing trade-off between quality and bit rate for codecs A and B. Codec B is better in quality for a given bit rate.](image-url)
Coding Efficiency Example

1:100 Compression: JPEG
Coding Efficiency Example

1:100 Compression: H.265 | HEVC
Coding Efficiency

Measuring Coding Efficiency

**Average Bit Rate**

\[ R = \frac{\text{number of bits in bitstream for a video sequence}}{\text{nominal duration of the video sequence}} \]  \hspace{1cm} (2)

**Quality / Distortion**

- Typically: Mean square error (MSE) and peak-signal-to-noise ratio (PSNR)
- For a \( W \times H \) array \( s[x, y] \) of original samples, the array \( s'[x, y] \) of reconstructed samples, and a sample bit depth \( B \)

\[
\text{MSE} = \frac{1}{W \cdot H} \sum_{x,y} (s[x, y] - s'[x, y])^2 \hspace{1cm} (3)
\]

\[
\text{PSNR} = 10 \cdot \log_{10} \left( \frac{(2^B - 1)^2}{\text{MSE}} \right) \hspace{1cm} (4)
\]

- For videos: Average PSNR over entire video sequence
Outline of Course

Image and Video Coding I: Fundamentals
- Probability, Random Variables, and Random Processes
- Lossless Coding
- Rate-Distortion Theory
- Quantization
- Predictive Coding
- Transform Coding

Image and Video Coding II: Algorithms and Applications
- Acquisition, Representation, Display, and Perception of Images
- Video Coding Overview
- Video Encoder Control
- Intra-Picture Coding
- Inter-Picture Coding
- Video Coding Standards