# LU04b - Komprimierung

## Einleitung

In diesem Kapitel konzentrieren wir uns auf verlustfreie Verfahren zur Komprimierung von Daten. Bei diesen Verfahren können die ursprünglichen Daten wiederhergestellt werden. Zum Beispiel:

- Textdokumente
- Programmcode (als Sourcecode oder binärer Code)
- Geschäftsdaten

Verlustbehaftete Verfahren für Bilddateien finden Sie im Kapitel bildformate.

## Run length encoding (RLE)

Bei diesen Verfahren werden identische Zeichen oder Zeichenketten, die mehrmals hintereinander stehen, nur einmal abgespeichert. Je nach Art der Daten kann dieses Kompressionsverfahren massive Einsparungen bringen.

### Beispiel: Brief

Ein Brief besteht zu einem grossen Teil aus Leerzeichen (hier als _ dargestellt).

#### Text

```
Herr_____
___
Hans_Muster_____
___
Wegweg_0_____
___
_____
___
0000_Hier_____
___
_____
___
Sehr_geehrter_Herr_Muster_____
___
```

#### Codierung

Um Speicherplatz zu sparen, wird nicht jedes Leerzeichen einzeln gespeichert. Stattdessen wird angegeben, wie oft sich ein Zeichen wiederholt.

```
Herr\76_\
Hans_Muster\69_\
Wegweg_0\72_\
\80_\
0000_Hier\71_\
\80_\
Sehr_geehrter_Herr_Muster\65_\
```

Diese Codierung kann auf alle Zeichen angewandt werden, die mehrfach hintereinander folgen.

### Beispiel: Grafik

Eine Grafik besteht zum grössten Teil aus Pixeln mit der Hintergrundfarbe. Zum Beispiel hat diese Wiki-Seite, die Sie im Moment betrachten, wohl 80-90% weisse Pixel. Es wäre nun sehr ineffizient, jeden weissen Pixel einzeln zu übermitteln. Stattdessen wird einfach angegeben, wieviele weisse Pixel hintereinander folgen.

## Wörterbuch für Texte

Bei diesen Verfahren wird ein Wörterbuch mit allen im Text vorkommenden Wörtern erstellt. Im eigentlichen Text wird anstelle der Wörter nur noch die Position im Wörterbuch angegeben.

Zum Entpacken der Datei, muss das Wörterbuch in der komprimierten Daten mitgegeben werden.

#### Beispiel

```
Wenn Fliegen hinter Fliegen fliegen, fliegen Fliegen Fliegen hinterher.
```

##### Wörterbuch

| Index | Wort |
|---|---|
| 0 | Wenn |
| 1 | Fliegen |
| 2 | fliegen |
| 3 | hinterher |
| 4 | hinter |

##### Codierung

```
0 1 4 1 2 2 1 1 3
```

Je länger der Text ist, desto häufiger kommen die gleichen Wörter vor. Somit wird das Verfahren umso besser, je länger der Text ist.

#### Verfahren

- LZW-Algorithmus

# Entropiecodierung

Bei diesen Verfahren wird jedes Zeichen durch einen unterschiedlich langen Code ersetzt. Dabei wird den häufigsten Zeichen ein kurzer, den seltenen Zeichen ein längerer Code zugeordnet. Im Gegensatz zur ASCII-Codierung haben die Zeichencodes eine variable Länge. Dadurch können häufige Zeichen mit 2 oder 3 Bits codiert werden, was eine markante Einsparung bedeutet.

## Morse-Alphabet

Der im 19. Jahrhundert entwickelte Morsecode setzt eine Form der Entropiekodierung ein. Jedem Zeichen wird ein Code aus Punkten (kurzes Signal) und Strichen (langes Signal) zugeordnet. Dabei wird den häufigsten Buchstaben ein Code aus wenigen Signalen zugeordnet. Die selten genutzten Zeichen haben einen längeren Code.

Um die einzelnen Zeichen zu erkennen, wird nach jedem Zeichen eine längere Pause eingelegt. Beim Morsen mit einer Lichtquelle (z.B. Taschenlampe) gilt:

- Punkt = 1 Sekunde Licht.
- Strich = 3 Sekunden Licht.
- 1 Sekunde Pause zwischen zwei Signalen.
- 3 Sekunden Pause nach jedem Buchstaben.

#### Huffman

Viele aktuelle Kompressionsverfahren nutzen (nebst anderen Verfahren) die Entropiecodierung. Die Grundlagen dazu finden Sie im Kapitel Huffman.

---

m114-A0G

Marcel Suter

From:
https://wiki.bzz.ch/ - **BZZ - Modulwiki**

Permanent link:
**https://wiki.bzz.ch/de/modul/m114/learningunits/lu04/komprimierung?rev=1769631167**

Last update: **2026/01/28 21:12**