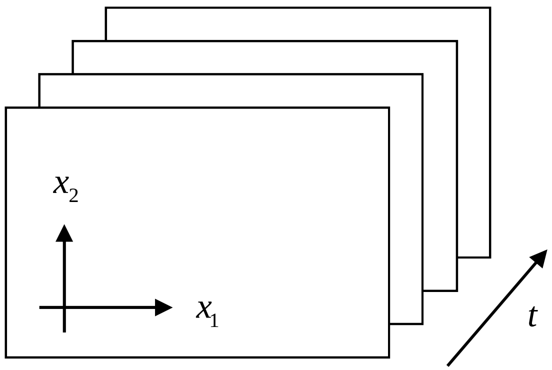
4. Methoden zur Bildcodierung – der JPEG-Standard

Das Ziel einer Transformationscodierung ist, die Korrelation zwischen den Pixeln zu reduzieren. Man transformiert und hat mit den transformierten Entwicklungs-koeffizienten (Fourierkoeff.) weitgehend unkorrelierte Verhältnisse. Man eliminiert damit die stoch. Abhängigkeit, so dass die transf. Bilder unkorreliert oder weiß sind (keine Möglichkeit mehr zur Prädiktion).

Man unterscheidet zwischen:
- **Einzelbildcodierung**: Dekorrelation im Ortsbereich **x** (still image coding)
- **Bildsequenzcodierung:** Dekorrelation im Zeitbereich *t* (image sequence coding)

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$



Stochastisches Modell für die Einzelbildcodierung

Ein Bild wird unterteilt in Teilbilder z.B. der Größe 8×8. Die Einzelbilder $\{\mathbf{X}_{ij}\}$ der Größe 8×8 bilden das Ensemble für die stochastische Betrachtung. Also hier z.B. für eine Bildmatrix der Dimension $N \times N = 32 \times 32$:

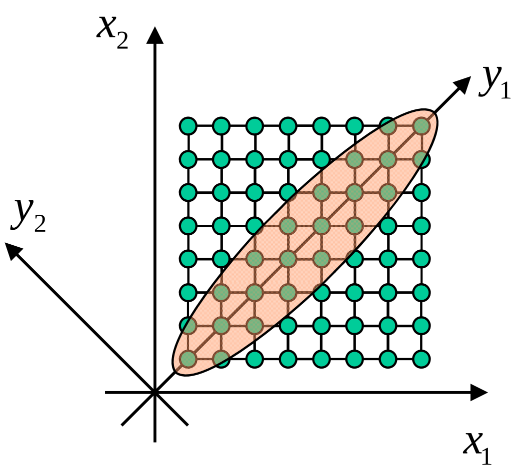
$$\underbrace{\begin{bmatrix} \mathbf{X}_{00} & \mathbf{X}_{01} & \mathbf{X}_{02} & \mathbf{X}_{03} \\ \mathbf{X}_{10} & \mathbf{X}_{11} & \mathbf{X}_{12} & \mathbf{X}_{13} \\ \mathbf{X}_{20} & \mathbf{X}_{21} & \mathbf{X}_{22} & \mathbf{X}_{23} \\ \mathbf{X}_{30} & \mathbf{X}_{31} & \mathbf{X}_{32} & \mathbf{X}_{33} \end{bmatrix}}_{\mathbf{X}}$$

Zur Illustration diene das folgende Beispiel

Wir nehmen eine 3-Bit-Codierung der Grauwerte an (8 Graustufen). Wir gruppieren zwei benachbarte Pixel der Größe 1×2 zu Teilbildvektoren:

$$\begin{bmatrix} [\bullet \quad \bullet] & [\bullet \quad \bullet] \\ [\bullet \quad \bullet] & [\bullet \quad \bullet] \end{bmatrix} \qquad \mathbf{X}_{ij} = \begin{bmatrix} x_i & x_j \end{bmatrix}$$

Falls man das Ensemble von Teilbildvektoren $\{\mathbf{X}_{\text{ij}}\}$ im zweidimensionalen Raum aufträgt, erhält man die folgenden 64 Vektoren:



Die Vektoren sind jedoch nicht alle gleichwahrscheinlich! Es ist unwahrscheinlich dass x_1 einen hohen Wert hat und x_2 einen niedrigen. Da benachbarte Pixel mit hoher Wahrscheinlichkeit ähnlich große Werte besitzen, gruppieren sich die Vektoren um die 1. Hauptdiagonale:

$x_1 \approx x_2\,!$

=> Dekorrelation mit KLT !!

Ziele des vollen JPEG-Standards

1. Einstellbare Qualitäts-/Kompressionswerte (Parameter)
2. Verlustfreie Codierung möglich
3. Hierarchische Codierung (Verschiedene Auflösungen möglich)

Baseline-Sequential-CODEC

(CODEC ist die Zusammenfassung von Coder und Decoder.)

1. Shift der Grauwerte von $[0,2^p-1]$ zu signed integers $[-2^{p-1},2^{p-1}-1]$
2. Anwendung der DCT (8×8): => 1 DC-Koeffizient, 63 AC-Koeff.
   (Separater Entwurf für 8-Bit (Video) und 12-Bit Bilder (Medizin))
3. Quantisierer (verlustbehaftet)
4. Separate Codierung für die DC-Komponente, da wertemäßig sehr hoch! Differential encoding (DPCM) wegen hoher Korrelation zu benachbarten Blöcken.
5. Zig-Zag-Sequenz der AC-Koeffizienten. Verbessert die anschließende Entropie-Kodierung (niederfrequente Anteile sind größer als hochfrequente Anteile), da in der Zig-Zag-Sequenz die Pixel stärker korreliert (benachbart) sind.
6. Entropie-Kodierung (verlustfrei)
   a) Huffman-Kodierung bei Baseline
   b) Arithmetische Kodierung (benötigt keine Kodier-Tabellen, adaptiert sich ans Bild)

   Die Huffman-Tabellen sind vordefiniert, eine für DC und eine für AC.
   Vor der Huffman-Kodierung wird zuerst eine Lauflängenkodierung (run-lenght-coding) durchgeführt.

Der Übergang vom RGB- auf den YUV-Farbraum

1. R,G,B – Farbaum (rot,grün,blau)
2. Y,U,V: Aufteilung in 1 Luminanzkanal Y und in 2 Chrominanz-Signale U,V

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{matrix} R+G+B \\ G+B \\ R+G \end{matrix} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} = \mathbf{T} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \mathbf{T}^{-1} \begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 1 & 1 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} Y \\ U \\ V \end{bmatrix}$$

3. C-M-Y-K, eine redundante Farbkodierung wird in der Druckindustrie verwendet (cyan, magenta, gelb, schwarz).

Nichtlinearität der menschlichen Wahrnehmung:

Das Auge ist wesentlich empfindlicher für die Luminanz (Helligkeit) als für die Chrominanz (Farbe). Deshalb gröbere Quantisierung der Chrominanz als bei der Luminanz und auch geringere örtliche Abtastung:

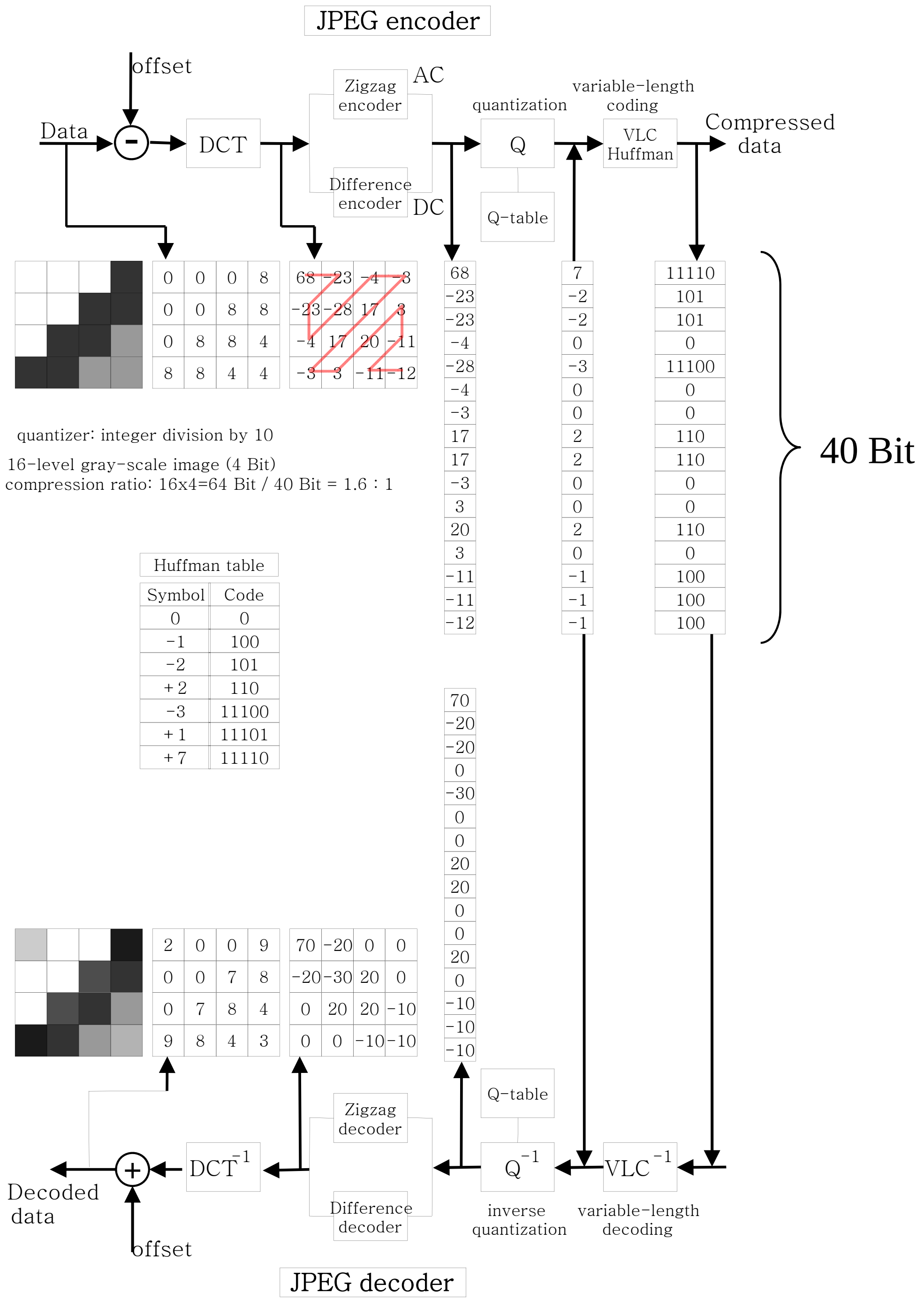
Qualität und Codierungsaufwand

Unkomprimiert: 8 Bit/Farbkanal => 24 Bit/Pixel

JPEG-typische Kompression: 8:1 bis 18:1, d.h. 24/8=3 Bit/Pixel bzw. 24/18=1,33 Bit/Pixel.

Qualität gemessen in Bits/Pixel: Gibt die Gesamtanzahl der Bits an, einschließlich der Chrominanzkomponenten, dividiert durch die Anzahl der samples in der Luminanz-komponente.

• 0,25-0,50 Bits/colour pixel	mäßige bis gute Qualität
• 0,50-0,75 Bits/colour pixel	gute bis sehr gute Qualität
• 0,75-1,50 Bits/colour pixel	exzellente Qualität
• 1,50-2,00 Bits/colour pixel	i.a. vom Original nicht unterscheidbar


JPEG encoder
offset
Data
DCT
Zigzag encoder
AC
Difference encoder
DC
quantization
Q
Q-table
variable-length coding
VLC Huffman
Compressed data
40 Bit
quantizer: integer division by 10
16-level gray-scale image (4 Bit)
compression ratio: 16x4=64 Bit / 40 Bit = 1.6 : 1
Huffman table
Symbol Code
0 0
-1 100
-2 101
+2 110
-3 11100
+1 11101
+7 11110
Q-table
Zigzag decoder
Difference decoder
DCT^{-1}
Q^{-1}
VLC^{-1}
inverse quantization
variable-length decoding
Decoded data
offset
JPEG decoder

Vordefinierte Standardquantisierungstabellen (Q-Tabellen) für die 64 DCT-Koeffizienten

Quantisierung d. DCT-Koeff.

(verlustbehaftet)

$$F^Q(u,v) = \left\lfloor \frac{F(u,v)}{Q(u,v)} + 0,5 \right\rfloor$$

Inverse Quantisierung

$$F^{Q'}(u,v) = F^Q(u,v) \cdot Q(u,v)$$

$Q(u,v)$

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

Luminanz Y (feiner)

17	18	18	24	21	24	47	26
26	47	99	66	56	66	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

Chrominanz U,V (gröber)

Quantisierungstabellen

- Die Q-Tabellen werden experimentell ermittelt und im JPEG-Standard zur Verfügung gestellt.
- Dabei wird ausgenutzt, dass die subjektive menschliche Wahrnehmung unempfindlicher ist gegenüber Quantisierungsrauschen bei höheren Frequenzen => gröbere Quantisierung von hochfrequenten DCT-Koeffizienten.
- In speziellen Anwendungsfällen wäre auch denkbar, die Quantisierungstabellen zusammen mit dem komprimierten Bild zu übertragen.

Die Entropie als Informationsmass

(Claude Shannon, 1948)

Gegeben: Auftretenswahrscheinlichkeiten für die zu codierenden statistisch unabhängigen q Symbole: $p_1, p_2, \ldots p_q$ (SUP). Der Informationsgehalt einer Nachricht wird definiert über die Entropie H

$$\text{Entropie:}\quad H = \underbrace{\sum_{i=1}^{q} \underbrace{p_i \cdot \underbrace{\operatorname{ld}\frac{1}{p_i}}_{\substack{\text{Informationsgehalt}\\ \text{eines Zeichens}}}}_{\substack{\text{multipliziert mit}\\ \text{seiner Häufigkeit}}}}_{\substack{\text{und summiert über}\\ \text{alle Zeichen}}} = -\sum_{i=1}^{q} p_i \cdot \operatorname{ld} p_i$$

Je unwahrscheinlicher eine Nachricht ist, desto höher ist ihr Informationsgehalt und umgekehrt. Die Entropie ist ein Maß für die Unordnung, die „Unsicherheit“, den „Überraschungsanteil“ der in einer Nachricht steckt.

Die Entropie als Informationsmass

Größte Entropie, falls alle Symbolwerte gleichwahrscheinlich sind. In diesem Fall besteht größtmögliche Unsicherheit über die zu erwartende Nachricht; folglich ist bei deren Eintreffen der Informationsgewinn am größten.

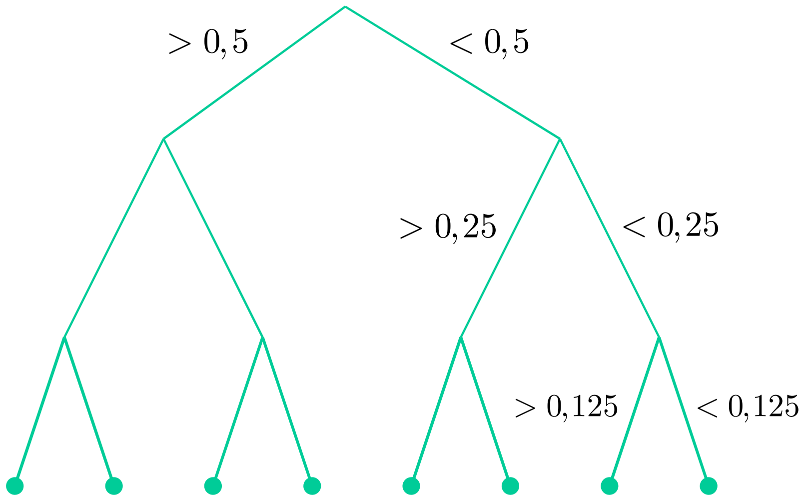
Für jede andere Verteilung ist die Entropie kleiner.

Die Entropie H liefert eine untere Grenze für die mittlere Codelänge:

$$L = \sum_{i=1}^{q} p_i \cdot l_i \geq H$$

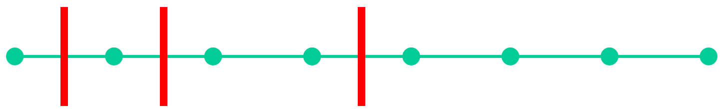
Gegeben ein Ensemble von Signalen $\{a_i\}$ mit einem Signalalphabet von 8 möglichen Amplituden zwischen 0 und 1. Alle Symbole haben die gleiche Wahrscheinlichkeit von 1/8.
Wie viel Entscheidungen werden benötigt um die richtige Amplitude zu bestimmen.
Der Informationsgehalt ist gegeben durch den Aufwand an binären Entscheidungen zur Bestimmung des richtigen Wertes eines ankommenden Datums.

Halbierungsstrategie



p_i: $\frac{1}{8}$ $\frac{1}{8}$ $\frac{1}{8}$ $\frac{1}{8}$ $\frac{1}{8}$ $\frac{1}{8}$ $\frac{1}{8}$ $\frac{1}{8}$

$\varepsilon = \frac{1}{7}$: $a_0 = 0$ $a_1 = \varepsilon$ $a_2 = 2\varepsilon$ $a_3 = 3\varepsilon$ $a_4 = 4\varepsilon$ $a_5 = 5\varepsilon$ $a_6 = 6\varepsilon$ $a_7 = 1$

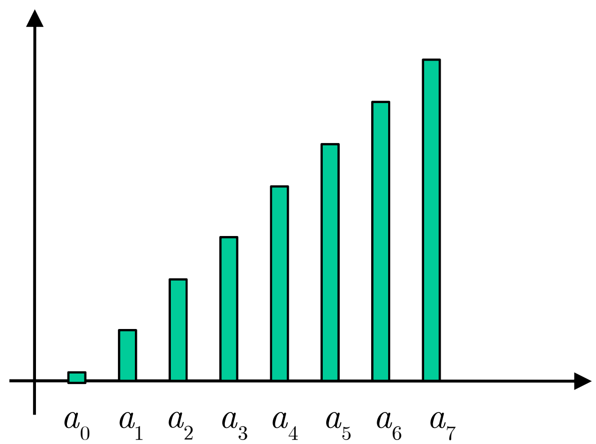


$$H = H_{\max} = \sum_{i=1}^{8} p_i \,\mathrm{ld}\frac{1}{p_i} = \sum_{i=1}^{8} \tfrac{1}{8}\,\mathrm{ld}\,8 = 8 \cdot \tfrac{1}{8} \cdot 3 = 3 \quad (\mathrm{ld}\, q)$$

Mit drei Entscheidungen (Zweige des Baums) ist man bei jedem Blatt des Baums. Balancierter Baum.

7	1
6	0,86
5	0,71
4	0,57
3	0,43
2	0,29
1	0,14
0	0

$\frac{1}{2}$ $\frac{1}{4}$ $\frac{1}{8}$



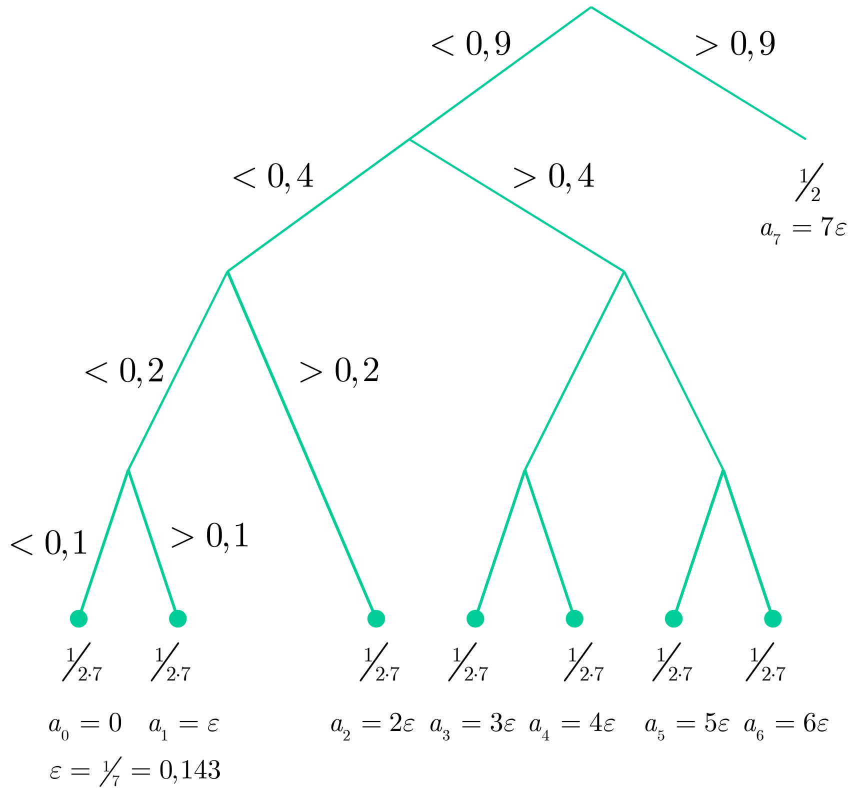
Ein Symbol a_7 habe die Wahrscheinlichkeit 1/2, alle anderen die gleiche W. von 1/(2·7).
Unbalancierter Baum:
Es wird nur eine Entscheidung für das Symbol a_7 =1 benötigt, welches am häufigsten vorkommt. Für alle anderen Symbole werden 4 und einmal 3 Entscheidungen benötigt.

$$H = \sum_{i=1}^{8} p_i \,\mathrm{ld}\frac{1}{p_i} = 0,5\cdot 1 + 7\cdot\tfrac{1}{2\cdot 7}\cdot \mathrm{ld}\,2\cdot 7 = 0,5(1+\mathrm{ld}\,14) = 2,4 \text{ bit codiert}$$

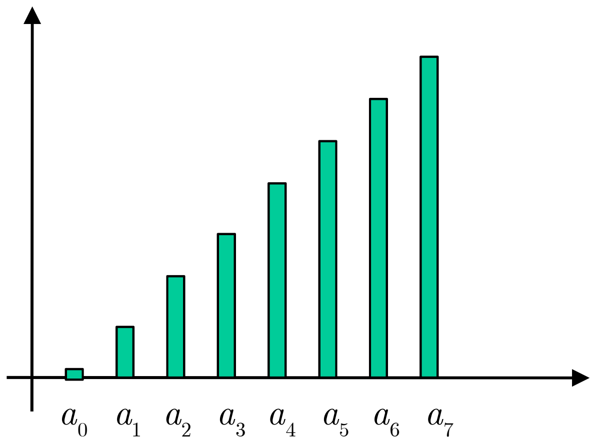
$$\mathrm{ld}\,8 = 3 \quad \text{bit uncodiert}$$

Die mittlere Baum- bzw. Entscheidungstiefe und damit auch Codierungslänge beträgt:

$$L = \sum_{i=1}^{q} p_i \cdot l_i = {(6\cdot 4 + 1\cdot 3 + 7\cdot 1)}\big/{14} = {34}/{14} = 2,428 > H$$



7	1	← 0,9
6	0,86	
5	0,71	
4	0,57	
3	0,43	← 0,4
2	0,29	← 0,2
1	0,14	← 0,1
0	0	



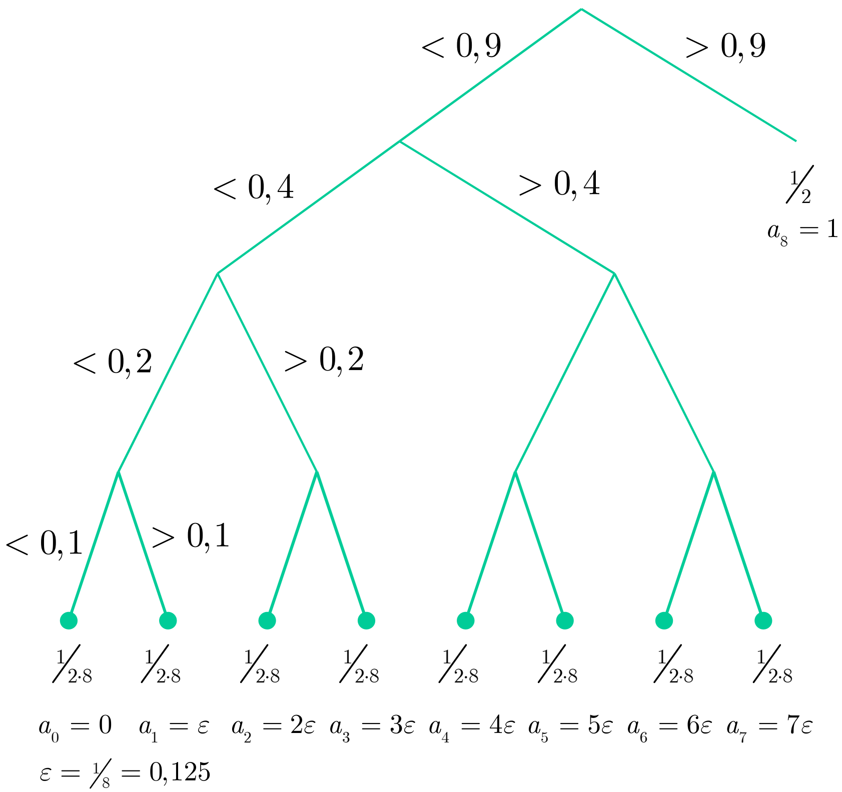
Ein Symbol a_8 habe die Wahrscheinlichkeit 1/2, alle anderen die gleiche W. von 1/2·8.
Es wird nur eine Entscheidung für das Symbol a_8 benötigt, welches am häufigsten vorkommt. Für alle anderen Symbole werden 4 Entscheidungen benötigt.

$$H = \sum_{i=1}^{9} p_i \operatorname{ld} \frac{1}{p_i} = 0,5 \cdot 1 + 8 \cdot \tfrac{1}{16} \cdot 4 = 0,5 + 2 = 2,5 \text{ bit codiert}$$

$$\operatorname{ld} 9 = 3,17 \text{ bit uncodiert}$$

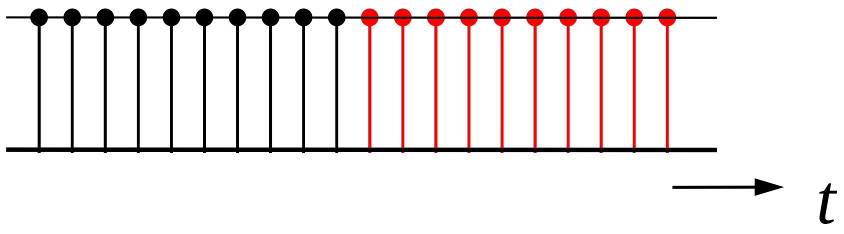
Die mittlere Baum- bzw. Entscheidungstiefe und damit auch Codierungslänge beträgt:

$$L = \sum_{i=1}^{q} p_i \cdot l_i = (8 \cdot 4 + 8 \cdot 1) / 16 = 2,5 = H$$



8	1	
7	0,875	← 0,9
6	0,75	
5	0,625	
4	0,5	
3	0,375	← 0,4
2	0,25	← 0,2
1	0,125	← 0,1
0	0	

Beispiel: SUP mit einem Alphabet von q=8 Amplituden

$\{a_i\}$ 

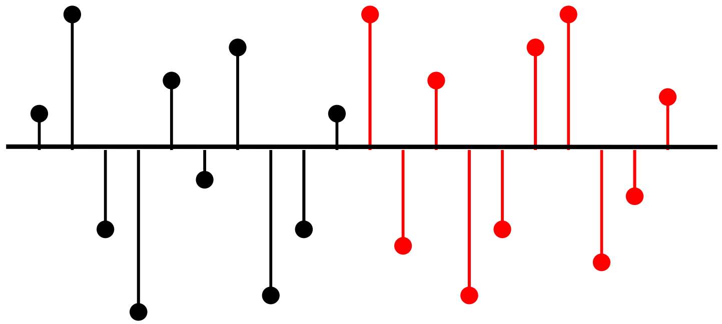
Es kommt nur eine Amplitude mit der Sicherheit 1 vor:

Entropie (SUP): $H = H_{\min} = \sum_{i=1}^{8} p_i \,\mathrm{ld}\, \frac{1}{p_i} = 1 \cdot \underbrace{\mathrm{ld}\, 1}_{=0} = 0$

keine statistische Unsicherheit; minimale Entropie.

Es werden 0 Bit zur Codierung benötigt.

Beispiel: SUP mit einem Alphabet von q=8 Amplituden



Entropie (SUP): $H = H_{\max} = \sum_{i=1}^{8} \frac{1}{8} \operatorname{ld} 8 = 8 \cdot \frac{1}{8} \cdot 3 = 3 \quad (\operatorname{ld} q)$

alle Amplituden des Alphabets der Größe q

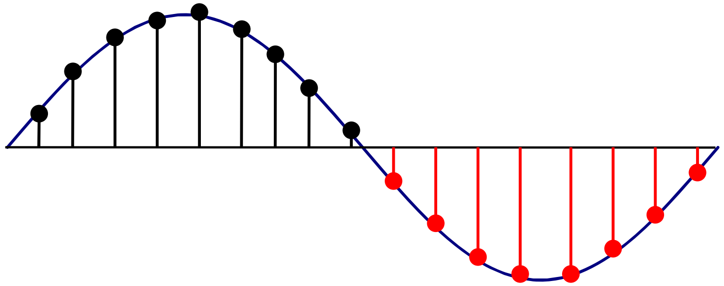
sind gleichwahrscheinlich $1/q$

Keine statistische Vorhersage möglich und damit auch

kein Codierungsgewinn realisierbar!

Es werden 3 Bit in einem Blockcode benötigt

Beispiel: SUP mit einem Alphabet von q=8 Amplituden



Auch hier sind alle Amplituden nahezu gleichverteilt:

Entropie (SUP): $H = H_{\max} = \mathrm{ld}\, q = 3$

Die Datenfolge ist deterministisch. Verwendet man als Modell bandbegrenzte Signale, so ergibt sich eine Entropie von nahezu Null!!

Oder: Wie groß ist die Entropie der Ziffernfolge von

π= 3,1415926535… ?

Antwort: Sie hat die Entropie 0! Es gibt keine Überraschung. Es gibt einen deterministischen Algorithmus zur Generierung aller Ziffern !!

Aufteilung einer Nachricht in einen Entropieanteil und eine Redundanz

$$\underbrace{\boxed{\text{Entropie H}}\boxed{\text{Redundanz R}}}_{\text{Nachricht}}$$

(uncodierter Blockcode)

Die durch die Entropie vorgegebene minimale mittlere Codelänge kann nicht ohne Informationsverlust weiter verkleinert werden.
Wenn bei einer entropiecodierten Nachricht Daten verloren gehen, dann bedeutet das zwingend einen Informationsverlust.
Fügt man Redundanz einer Nachricht hinzu, so können Übertragungsfehler detektiert und auch korrigiert werden.

In diesem einfachsten SUP-Modell von Shannon werden Verbundwahrscheinlichkeiten nicht berücksichtigt.

D.h. die folgenden Texte haben bei dem SUP-Modell den gleichen Informationsgehalt:

- Landau schonen
- lcsenan dunoa
- Claude Shannon

(Anagramme)

Deutsche Textnachrichten

• Uncodiert benötigt man:	ld 26=4,70 bit/Zeichen
• Die Entropiecodierung (SUP) benötigt nur:	4,0629 bit/Zeichen
• Redundanz:	0,637 bit/Zeichen (13,6%)

Die Analyse von Verbundwahrscheinlichkeiten führen zu wesentlich komplexeren Modellen (bedingte Entropie, Verbundentropie)

Die Entropie einer natürlichen Sprache läßt sich wesentlich genauer bestimmen, wenn auch die Auftrittswahrscheinlichkeiten von Buchstabenkombinationen (N-Gramme) berücksichtigt werden (Paare: Bigramme, Triple: Trigramme, ...).

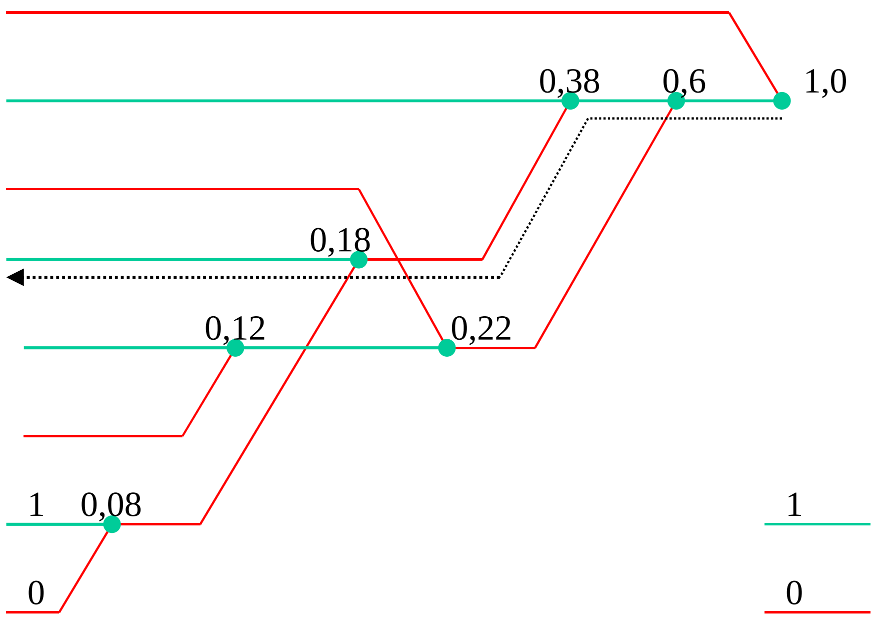
Claude Shannon bestimmte die Entropie der englischen Sprache 1950 auf 1,5 bit je Buchstabe
Karl Küpfmüller bestimmte die Entropie der deutschen Sprache 1954 auf 1,5 bit je Buchstabe
Damit liegt die Redundanz der deutschen/englischen Sprache bei etwa **3,2 bit je Buchstabe**.
Anders ausgedrückt:
etwa **68% der Zeichen** eines deutschen/englischen Textes sind – aus der Sicht des Informationsgehalts der Nachricht überflüssig.

Die Angaben von Shannon sind jedoch nicht konstruktiv; es handelt sich eigentlich nur um *Existenzsätze*. Es dauerte teilweise über 30 Jahre, bis konstruktiv optimale Codes für die Quell- und Kanalcodierung gefunden wurden.
Der vom Shannon-Fano-Algorithmus erzeugte Codierungs-Baum ist nicht immer optimal.

1. Huffman-Entropie-Codierung.
   David A. Huffman hat 1952 einen Algorithmus veröffentlicht, der beweisbar immer einen optimalen Baum für die gegebenen Symbolwahrscheinlichkeiten liefert.
2. BCH und Reed-Solomon-Codes mit Fehlererkennung und Fehlerkorrektur (1960). Große Bedeutung heute für CDs, DVDs, Festplatten, Satellitenbildcodierung.
3. Arithmetische Entropie-Codierung (Rissanen, IBM, 1980)

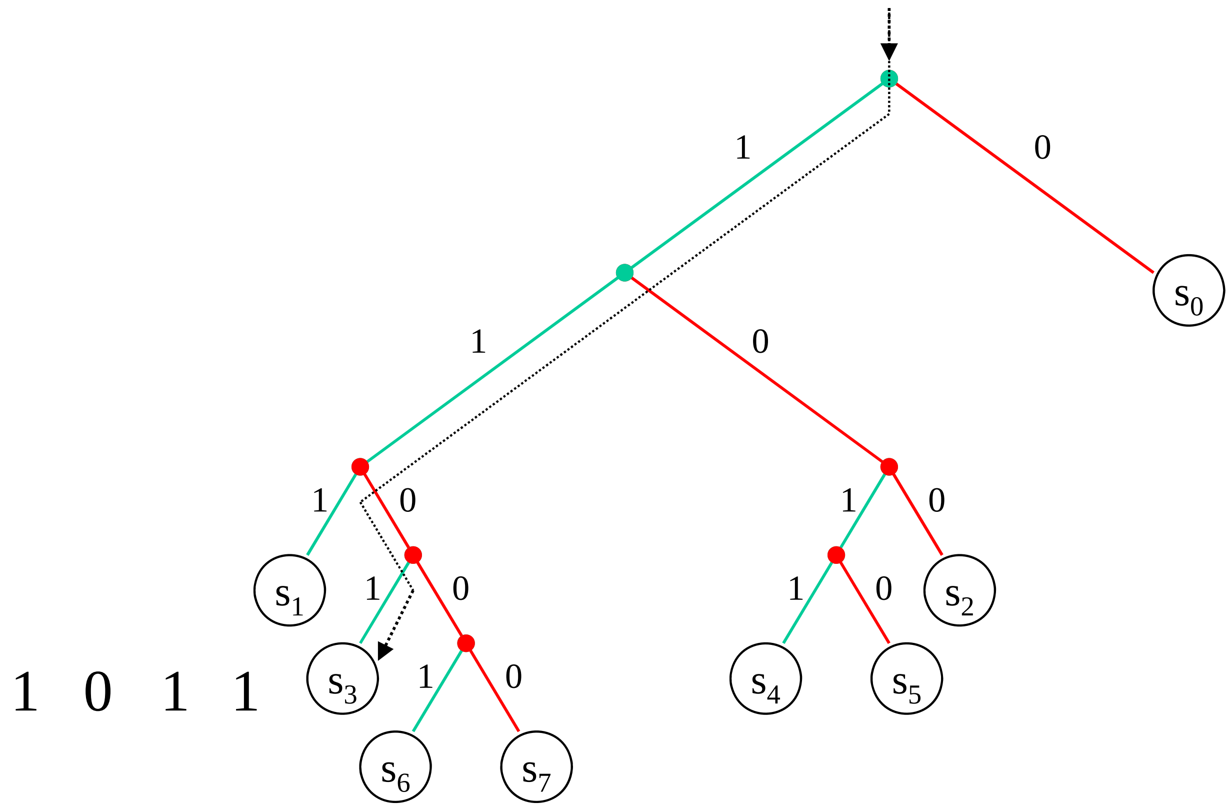
Huffman-Codierung

Code	Symbol	Auftritts-W. p_i
0	s_0	0,4
111	s_1	0,2
001	s_2	0,1
1011	s_3	0,1
1101	s_4	0,07
0101	s_5	0,05
10011	s_6	0,04
00011	s_7	0,04



Fasse in jeder Stufe die zwei kleinsten Wahrscheinlichkeiten zusammen und führe zur Unterscheidung eine Bitstelle ein

Decodierungsbaum



Der Huffman-Code minimiert die mittlere Codelänge:

$$L_m = \sum p_i l_i = 0,4 \times 1 + 0,2 \times 3 + 0,1 \times 3 + 0,1 \times 4 + 0,07 \times 4 + 0,05 \times 4 + 0,04 \times 5 + 0,04 \times 5 = 2,58$$

Die Entropie beträgt: $H = \sum p_i \cdot \lg \frac{1}{p_i} = 2,51 \leq L_m$

Ein Blockcode für 8 Zeichen hätte zum Vergleich eine Codelänge von 3.

Diese 3 Bit würden auch benötigt, wenn alle Symbole gleichwahrscheinlich wären.

VCDEMO
TU-Delft

Original

JPEG 2 Bit/pixel
MSE: 9,1

JPEG 1,5 Bit/pixel
MSE: 15,8

JPEG 0,7 Bit/pixel
MSE: 48,3

Original

JPEG 0,5 Bit/pixel
MSE: 74

JPEG 0,4 Bit/pixel
MSE: 101

Original

JPEG 0,3 Bit/pixel
MSE: 150

JPEG 0,2 Bit/pixel
MSE: 320

Original

JPEG 0,1 Bit/pixel
MSE: 496

Vergleich mit JPEG2000

JPEG 0,5 Bit/pixel
MSE: 74

Original

JPEG2000 0,5 Bit/pixel
MSE: 37,4

JPEG 0,3 Bit/pixel
MSE: 150

Original

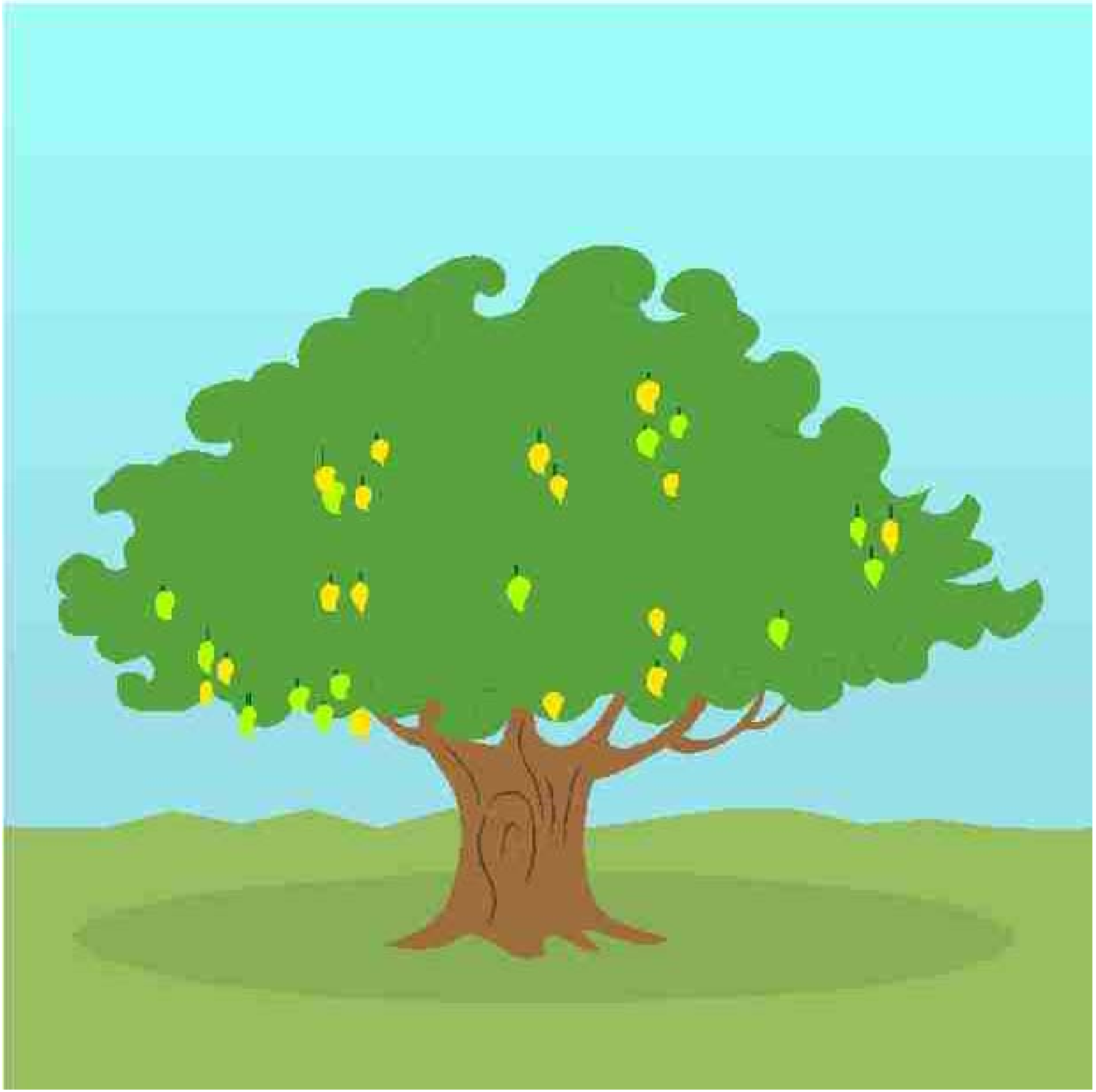
JPEG2000 0,3 Bit/pixel
MSE: 73

Welchen Gewinn erwarten wir durch die **semantische Codierung** ?

Größe des Originals: 1 Mio Pixel

Semantische Codierung:
Ein Auto vor einem Baum auf einer grünen Wiese (0,1 kB).

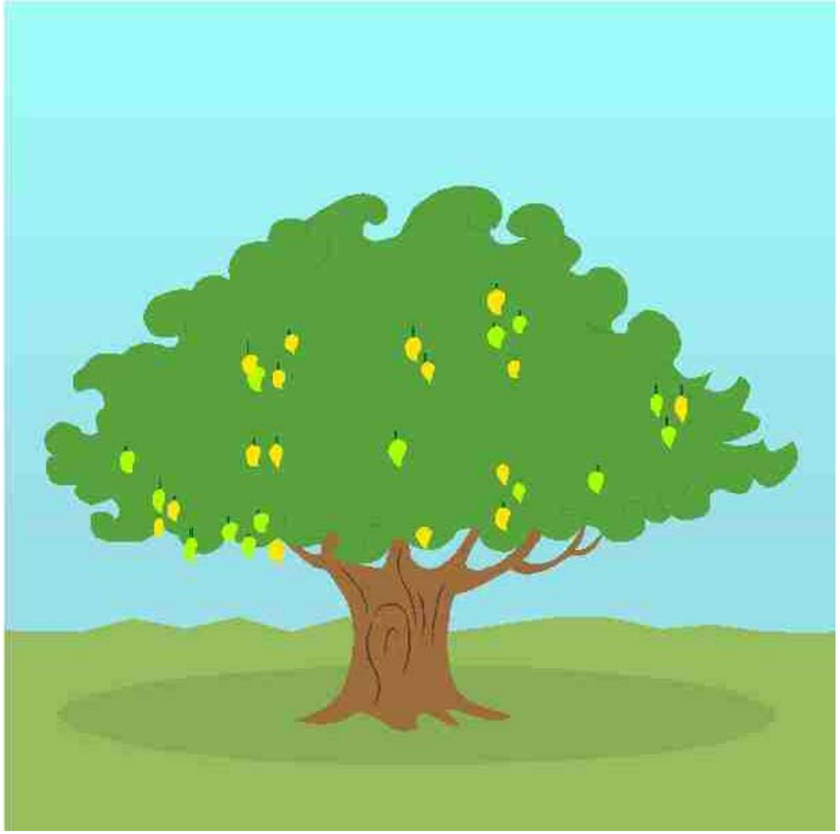
Damit ist bis auf einige Details fast alles über den Informationsgehalt des Bildes ausgesagt (verlustbehaftete Codierung)

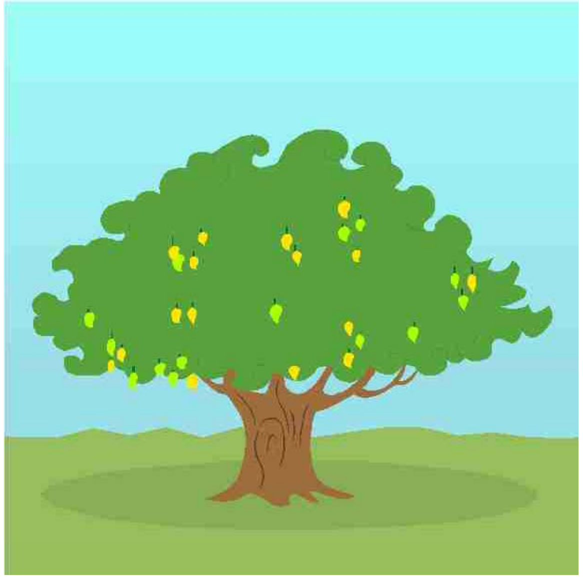


Beispiel Baum mit 1 Mio Pixel

Vergleich aller drei Codierungsarten:

1. Entropie (SUP)
2. JPEG
3. Semantische Codierung





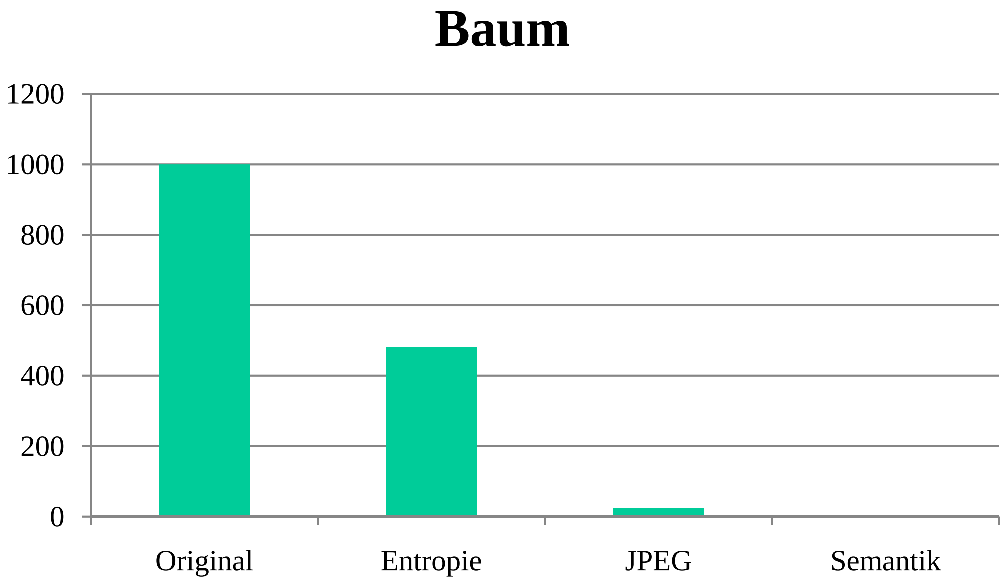
Original	Entropie (SUP)	JPEG	Se-mantik
1000 kB	480 kB	24 kB	1 kB
Einsparung:	52%	97,6%	99,9%

Pieter Brueghel: "Die holländischen Sprichwörter", 1559 1 Mio Pixel

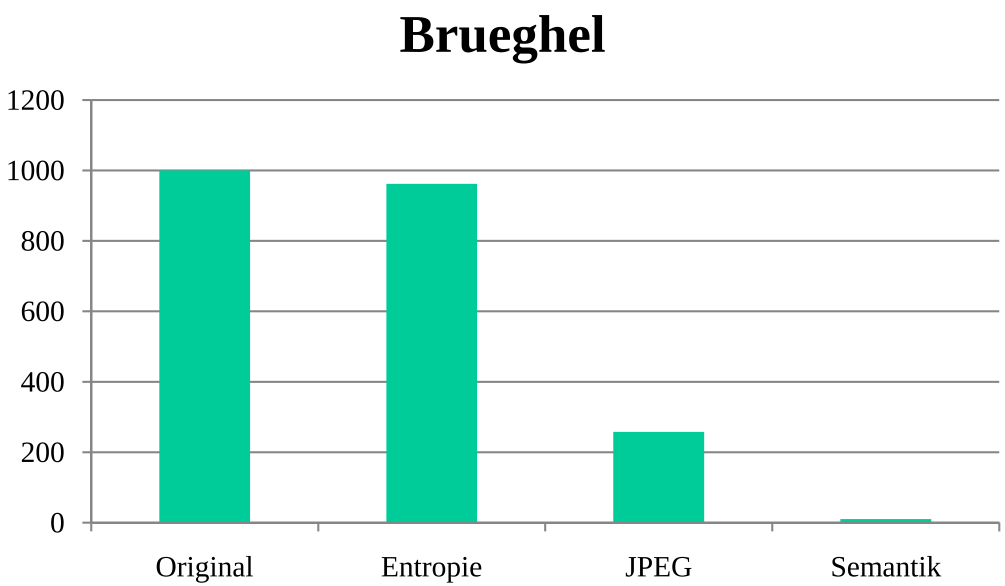
  

	Original	Entropie (SUP)	JPEG	SEM
	1000 kB	962 kB	250 kB	10 kB
Einsparung:		3,8%	75 %	99%

Pieter Brueghel: "Die holländischen Sprichwörter", 1559



Faktor 2 42 1000



Faktor 1,04 3,9 100

Video in Communication and Multimedia

Video & Image Coding Standards