

Fachhochschule Augsburg (FHA)
Studiengang Informatik

DVA-Seminar Dokumentenmanagement

Sommersemester 2004
Dokumentation zur Präsentation #: 13

Datenkompression

Autor: Steve Moser
Datum der Präsentation: 2004 - 05 - 11
Dozent: Prof. Peter Fischer, Hochschule für Wirtschaft Luzern
Gastdozent an der FH Augsburg

Inhaltsverzeichnis

1	Ein paar Worte	3
2	Lempel Ziv Familie	5
2.1	LZ77	5
2.2	LZ78	5
2.3	LZW	6
2.4	Übersicht LZ-Familie	8
3	Huffman	9
4	Grafik	11
4.1	GIF	11
4.2	JPEG	12
4.3	JPEG2000	12
4.4	PNG	13
5	MPEG	14
5.1	Grundsätzliches zu MPEG	14
5.2	MP3	14
5.3	MPEG1	15
5.4	MPEG2	15
5.5	MPEG3	15
5.6	MPEG4	15
6	Welcher Kompression für welches Dokument	17
7	...	18

1 Ein paar Worte

In der heutigen Zeit, in der sich Information rasend über das Internet verbreitet und täglich sich mehr und mehr Daten ansammeln, die laufend weiterverbreitet, gehortet und archiviert werden, bemühen sich Entwickler und Wissenschaftler aus den naturwissenschaftlichen Richtungen Wege zu finden, Daten geeignet zu archivieren, zu komprimieren oder für das Internet brauchbar zu machen.

Eine große Rolle in dieser Entwicklung stellen Kompressions- und Dekompressionsalgorithmen dar.

Die Kompression und Dekompression sieht einem Urlaub sehr ähnlich. Vorher wird entschieden, was für Dinge einen auf die Reise begleiten sollen. Anschließend wird ein geeigneter Koffer ausgesucht. Dort wird alles fein säuberlich eingepackt und verschlossen. In vielen Fällen wird noch ein Schloss verriegelt, und ab geht es auf die Reise. Angekommen, wird das Schloss geöffnet und alles wieder ausgepackt. Bei einer Kompression geschieht dies in ähnlicher Weise. Man sucht sich seine Daten zusammen und komprimiert sie mit einem Algorithmus (Aussuchen eines geeigneten Koffers). Es entsteht ein Archiv (Koffer). Dieses Archiv kann nun über das Internet verschickt werden (Reiseweg) oder als Sicherung auf eine externe Quelle verlagert werden (Schrank). Das Archiv kann jederzeit wieder mit dem gleichen Algorithmus entpackt werden (vorgehensweise beim Auspacken des Koffers). Die Daten sind dann wieder in ihrem Ursprungszustand. Das Beispiel ist vielleicht etwas simpel gewählt, jedoch verdeutlicht es schön den Hintergedanken, welcher der Thematik zur Datenkompression zu Grunde liegt.

Es gibt die verschiedensten Kompressionsarten für verschiedenste Zwecke. Je nach Verwendungszweck der Daten werden verlustbehaftete und verlustfreie Kompressionsalgorithmen verwendet.

verlustfrei: Daten werden verlustfrei komprimiert und wieder dekomprimiert. Die Ursprungsdatei ist zu 100% gleich der dekomprimierten Datei.

verlustbehaftet: Daten können nicht in ihren Ursprungszustand dekomprimiert werden, da Datenfragmente bei der Komprimierung aus algorithmischer Sicht überflüssig waren.

Zum Beispiel ist es nicht sinnvoll wichtige Textdokumente mit einem verlustbehafteten Algorithmus zu komprimieren, da nach der Dekomprimierung der Urzustand nicht mehr sicher gestellt ist. Auf der anderen Seite ist es wiederum sinnvoll, ein Musikstück verlustbehaftet zu komprimieren. Hier können Informationen fallen gelassen werden, die das menschliche Gehör nicht mehr wahrnehmen kann.

Nun kann sich manch einer fragen: "Wozu das alles? Man hatte es doch früher auch nicht benötigt."

Hierzu ein paar Schlagworte:

- Sicherung von Daten
- Platz sparen
- Information schnell präsentieren
- **Internet**

Sowohl für Privatanwender als auch für Großbetriebe ist es wichtig Daten zu sichern. Wer viele Daten täglich, wöchentlich, monatlich sichert, braucht viel Speicherplatz und dieser ist nicht billig. In dieser Globalisierungszeit werden alle Arten der Information über das Internet ausgetauscht und präsentiert. Genau dort ist das Haupteinsatzgebiet der Kompressionsalgorithmen. Man findet sie auf jeder Website, ohne das man es merkt. Denn je schneller eine Website geladen ist, umso schneller kann der Anwender die Information verarbeiten. Ein Webauftritt einer Firma wäre nicht rentabel, sollte sich die Website nicht nach ein paar Sekunden öffnen. Wer wartet schon gerne drei Minuten auf eine Website bis sie endlich angezeigt wird? Wer würde sich ein Werbevideo ansehen, das erst nach 30 Minuten vollständig runtergeladen wird? Mit anderen Worten ist das Geld verdienen oder Geld einsparen, ein nicht zu unterschätzender Faktor, der die Entwicklung von Kompressionsalgorithmen beeinflusst oder gar beschleunigt hat.

Da es sehr viele und abgewandelte Verfahren der Komprimierung gibt, wird in diesem Dokument nur auf die wichtigsten und bekanntesten Algorithmen der Kompression eingegangen. Es soll eine Einleitung dessen sein was hinter den Bildern, Texten, Musik und Filmen stehen kann, mit denen wir täglich in Kontakt kommen.

2 Lempel Ziv Familie

2.1 LZ77

Das LZ77 Verfahren wurde 1977 von Abraham Lempel und Jakob Ziv entwickelt. Die Kompression erfolgt durch eine Kodierung des Textes mittels der Angabe der Distanz des aktuellen Zeichens zum Anfang des Textes und der Anzahl der von dort zu kopierenden Zeichen. Erscheint ein neues Zeichen, so wird dieses hinzugefügt. Es hat dann die Form: $(2, 1) B$

Die $(2, \dots)$ bedeutet, dass um zwei Stellen zurückgegangen wird und dieses Zeichen einmal $(\dots, 1)$ kopiert wird. Das darauffolgende B ist das neue Zeichen.

LZ77 Beispiel

Die zu komprimierende Zeichenfolge:

Stelle	1	2	3	4	5	6	7	8	9
Zeichen	a	a	b	c	b	b	a	b	c

Zur Kodierung:

Stelle	Hit	Zeichen	Code
1	-	a	(0,0)a
2	a	b	(1,1)b
3	-	c	(0,0)c
4	b	b	(2,1)b
5	ab	c	(5,2)c

¹

[Kow04]

Vorteil: Hohe Kompression bei wiederkehrenden, langen und gleichen Zeichenketten. Schnelle Dekompression durch einfache Adressberechnung.

Nachteil: Aufwendige Kompression durch die vielen Vergleiche, die dabei entstehen.

Bemerkung: Das "Suchfenster", die Anzahl der Buchstaben, um welche zurück gegangen wird, bewegt sich im Rahmen von 4000 - 68000 Zeichen.

2.2 LZ78

Eine Weiterentwicklung des LZ77 Verfahrens ist das LZ78 Verfahren. Dieses Verfahren verwendet ein eigenes "Wörterbuch". In diesem wird eine Folge von Zeichen unter einem Index abgelegt. Dieses Wörterbuch wird bei der Kompression erstellt und in einigen Varianten laufend angepasst, wenn es sich z.B. um große Datenmengen handelt.

¹<http://einstein.informatik.uni-oldenburg.de/rechnernetze/seite63.htm> (Stand 10.04.2004)

Wird ein langes Wort im Wörterbuch gefunden, dann wird der Index dieses Wortes im Wörterbuch sowie das folgende Zeichen ausgegeben. Das daraus sich neu ergebende Wort wird in das Wörterbuch aufgenommen.

LZ78 Beispiel

Die zu komprimierende Zeichenfolge:

Stelle	1	2	3	4	5	6	7	8	9
Zeichen	a	b	b	a	b	a	b	a	c

Zur Kodierung:

Stelle	Wörterbuch	Code
1	1:a	(0,a)
2	2:b	(0,b)
3	3:bc	(2,c)
4	4:bca	(3,a)
5	5:ba	(2,a)

²

[Kow04]

Vorteil: Schnellere Kompression durch weniger Vergleiche.

Nachteil: Zusätzlicher Speicher für das Wörterbuch bei der Kompression und Dekompression sollte berücksichtigt werden.

2.3 LZW

Das LZW verfahren ist eine Weiterentwicklung des LZ78 Verfahren und wurde 1984 von Terry Welch patentiert. Das Verfahren beruht darauf, dass es bei der Ausgabe nur noch Indizes des Wörterbuchs ausgibt, und keine Zeichen mehr enthalten sind.

Zu Beginn werden alle möglichen Zeichen der länge 1 im Wörterbuch explizit untergebracht. Bei der Kompression wird eine möglichst lange Zeichenkette im Wörterbuch gesucht und als Index ausgegeben. Diese Zeichenkette wird mit dem nächsten nicht passenden Zeichen als neue Zeichenkette in das Wörterbuch übernommen.

LZW Beispiel

Die zu komprimierende Zeichenfolge:

Stelle	1	2	3	4	5	6	7	8	9
Zeichen	a	b	b	a	b	a	b	a	c

²<http://einstein.informatik.uni-oldenburg.de/rechnernetze/seite7.htm> (Stand 11.04.2004)

Im Wörterbuch bereits enthalten: 1:a, 2:b, 3:c

Zur Kodierung:

Stelle	Wörterbuch	Code
1	4:ab	(1)
2	5:bb	(2)
3	6:ba	(2)
4	7:aba	(4)
6	8:abac	(7)
9	-	(3)

Zur Dekompression:

Dekompression (im Wörterbuch steht bereits 1:a, 2:b, 3:c)

Stelle	Eingabe	Wörterbuch	Ausgabe
1	(1)	1:a	a
2	(2)	4:ab	b
3	(2)	5:bb	b
4	(4)	6:ba	ab
5	(7)	7:aba	ab+a
6	(3)	8:abac	c

³

[Kow04]

Anwendungsgebiete: Unix Programm "compress", GIF (*Graphic Image Format*)

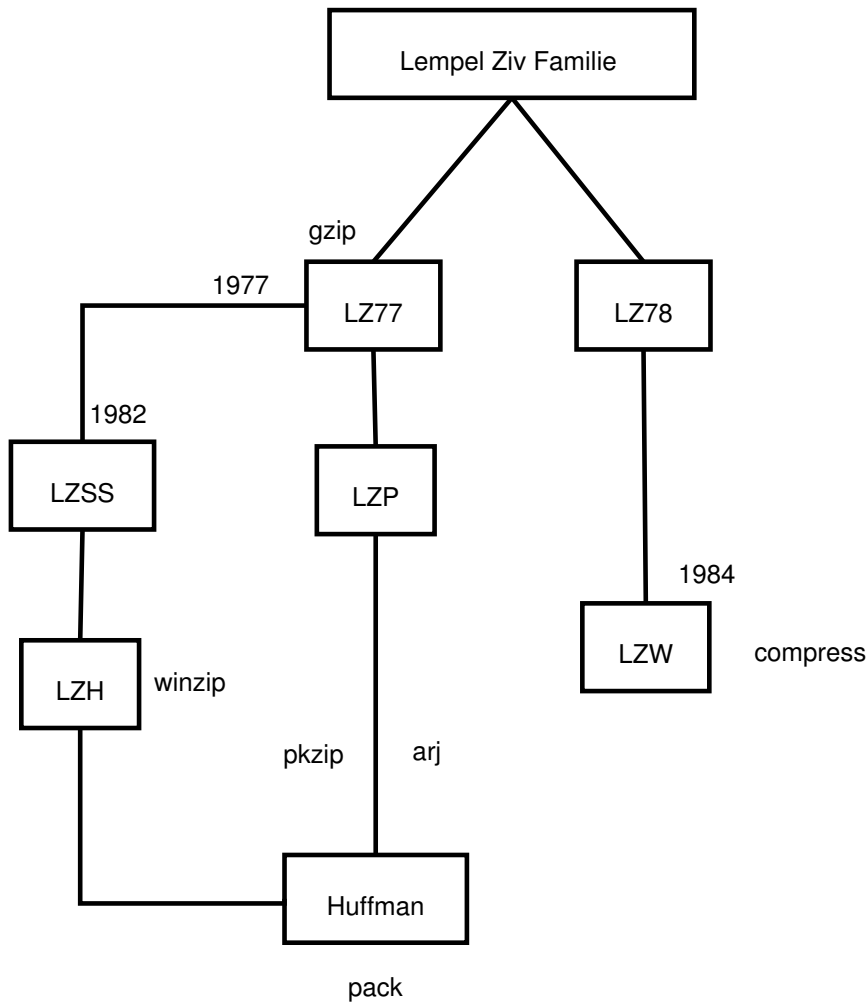
Die LZ-Verfahren sind bei der verlustfreien Kompression anzusiedeln.

³<http://einstein.informatik.uni-oldenburg.de/rechnernetze/seite8.htm> (Stand 12.04.2004)

2.4 Übersicht LZ-Familie

Aus den verschiedenen Ableitungen des Lempel-Ziv Verfahren, entstanden u.a. Winzip, gzip, pkzip, arj, compress und weitere. Die folgende Übersicht stellt die Entstehungshistorie der Familie der Lempel-Ziv Algorithmen dar.

Eine gute Zusammenfassung der einzelnen zip Algorithmen lässt sich unter <http://goethe.ira.uka.de/seminare/rftk/zip/> [Wan04] einsehen.



3 Huffman

Huffman-Kodierung geht auf das Morsealphabet zurück und nutzt die unterschiedlichen Wahrscheinlichkeiten aus, mit der einzelne Zeichen auftreten. Ein Zeichen, das sehr oft vorkommt, wird mit einer kurzen Kodierungssequenz ausgestattet wie z.B. 1 oder 01, und ein Zeichen, das sehr selten auftritt, mit einer längeren Kodierungsfolge wie z.B. 00001. Die Länge der Kodierung ergibt sich zum einen aus dem Vorrat der Zeichenmenge und zum anderen aus dem daraus resultierenden Ableitungsbaum. Typisches Einsatzgebiet der Huffman-Kodierung sind Texte. Das Huffmanverfahren ist ein verlustfreier Algorithmus. Das folgende Beispiel soll die Einfachheit dieser Art der Komprimierung darstellen.

Beispiel für Huffmankodierung

Gegeben sei eine beliebige Textfolge, wie z.B.

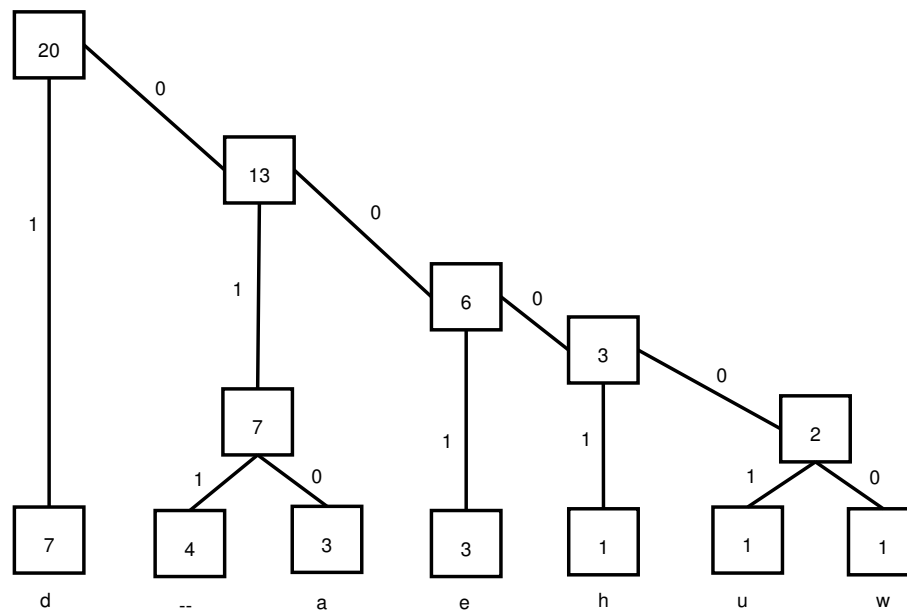
”wadde hadde du de da”

Normalerweise wird nun 1 Zeichen mit 8 Bit kodiert. Dies entspricht hier 20 Zeichen x 8 Bit = 160 Bit.

Nun wird die Häufigkeit der einzelnen Zeichen ermittelt.

d	-	a	e	h	u	w
7	4	3	3	1	1	1

Daraus wird ein Ableitungsbaum erstellt. Die zwei jeweils kleinsten Knoten werden zu einem neuen Knoten addiert. Die Reihenfolge beginnt von unten nach oben, und wird solange fortgesetzt bis nur noch der Wurzelknoten vorhanden ist. Anschließend werden an den Kanten die entsprechenden 0 und 1 angeschrieben. Linke Kanten 1, rechte kanten 0.



Ableitungsbaum Huffman-Codierung "wadde hadde du de da"

So ergibt sich eine Tabelle mit den Werten:

d	-	a	e	h	u	w
1	011	010	001	00011	00010	00000

Kodiert sieht es dann so aus (natürlich ohne Trenner):

00000 | 010 | 1 | 1 | 1 | 001 | 011 | 00011 | 010 | 1 | 1 | 1 | 001 | 011 | 1 | 00010 | 011 | 1 | 001 | 011 | 1 | 010

Die kodierten Zeichen werden mit der Entschlüsselungstabelle als neue Datei abgespeichert. Die neue Datei hat hier eine Länge von 51 Bit (ohne die Kodierungstabelle). Ein Ersparnis von ca. 69%.

Es können jedoch folgende Gründe für eine Verschlechterung sorgen. Variiert die Wahrscheinlichkeit der einzelnen Zeichen kaum, so kommen sie alle sehr selten vor. Der "worst-case" ist eine Gleichverteilung der zu kodierenden Zeichen. Ist der Text sehr kurz "bläht" die dazugehörige Kodierungstabelle die Datei unnötig auf.

4 Grafik

4.1 GIF

Das Graphic Interchange Format (GIF) ist einer der älteren Komprimierungsalgorithmen und wurde 1987 von der Firma CompuServe entwickelt. GIF wurde als plattformunabhängiges Format entwickelt und sollte den damaligen beschränkten geringen Übertragungsgeschwindigkeiten genüge tun. Mit anderen Worten, die Komprimierung musste groß sein und gleichzeitig sollte das komprimierte Bild noch gut bis sehr gut erkennbar sein. Das Ergebnis war ein auf dem LZW Verfahren basierender Algorithmus, der ein Rasterbild entwarf, das bis zu 256 Farben (8 Bit Farbtiefe) darstellen konnte. Dies reichte völlig aus, da Rechner zu dieser Zeit nicht mehr Farben unterstützten. Da der Algorithmus sich LZW zu nutze machte, ist das GIF Format bis heute patentiert und muss bei Grafikprogrammen, die dieses Format unterstützen, lizenziert werden. Vielleicht ist dies mit ein Grund warum nachfolgende Formate aus der Open Source Bewegung kommen und somit frei sind.

Durch die geringe Anzahl von Farben, eignet sich GIF für Bilder mit großflächigen Farben, wie z.B. Logos oder Zeichnungen. Verwendet man Realbilder, so muss man mit Einschränkungen rechnen.

GIF erstellt zu Anfang eine Farbtabelle und füllt diese mit den RGB-Werten (Rot, Grün, Blau) auf. Dies nennt man Farbindizierung. Farben, die nicht in der Tabelle enthalten sind, werden mit vorhandenen Farbpunkten interpoliert (annähernd berechnet). Diese Vorgehensweise nennt man Dither. Jedoch muss mit sichtbaren Rasterpunkten gerechnet werden.[Ser04] ⁴

Eine Weiterentwicklung erfuhr GIF zwei Jahre später, 1989 und trägt daher den offiziellen Namen GIF89a. Dieser Version wurden einige zusätzliche Optionen spendiert, wie z.B. transparente Hintergründe oder das Interlacing. Interlacing bedeutet, dass das Bild, obwohl es noch nicht vollständig übertragen wurde, zeilenweise am Zielrechner aufgebaut wird. Wenn die Übertragung vollständig ist, ist das Bild komplett sichtbar. Diesen Modus muss jedoch der Browser unterstützen.

Dies gilt auch für animierte GIFs. Ein animiertes GIF wird ähnlich einem Daumenkino aufgebaut. Mehrere Bilder werden aufeinander gelegt. Das jeweils obere Bild wird gegen sein darunterliegendes ausgetauscht. Dies geschieht nach eigenem Ermessen und kann über Optionen des Erstellungsprogrammes festgelegt werden. Beispielsweise stellen GIF Motion oder Image Ready von Adobe diese Option zur Verfügung.

Das GIF Patent lief in den USA am 20.06.2003 ab und wird am 18.06.2004 in Europa enden.

”We were able to search the patent databases of the USA, Canada, Japan, and the European Union. The Unisys patent expired on 20 June 2003 in the USA, but it does not expire in most of Europe until 18 June 2004, in Japan until 20 June 2004 and in Canada until 7 July 2004.” ⁵ — [Fou04]

⁴<http://www.tutorial.artems.de/help/dateiformate/gif.html> (Stand 20.04.2004)

⁵<http://www.gnu.org/philosophy/gif.html> (Stand 26.04.2004)

4.2 JPEG

Generell lässt sich sagen, dass Bilder meist viel Speicherplatz benötigen. So wird zum Beispiel bei einem Bitmap (bmp) jeder einzelne Farbpunkt abgespeichert, mitsamt all seinen Attributen wie Farbe, Position und weiteres. Die Gif Kompression lässt nur eine Kompression bis zu 8 Bit pro Farbpunkt und maximal 256 Farben zu. Bilder die in ihrer Beschaffenheit wesentlich komplexer sind, bedürfen einer anderen Art der Komprimierung. Hier hat sich das JPEG (Joint Photographic Experts Group) Format durchgesetzt. "JPEG bezeichnet kein Dateiformat, sondern eine ganze Familie von Algorithmen zur Kompression digitalisierter Standbilder in Echtfarbkqualität. Diese Sammlung unterschiedlichster Verfahren wurde 1993 unter der Bezeichnung ISO 10918 als Standard festgeschrieben." (siehe [Sch04])⁶ Die JPEG Komprimierung zählt zu den verlustbehafteten Komprimierungsalgorithmen. Jedoch besteht hier die Möglichkeit die Höhe der Kompression über einen Faktor anzugeben. Desweiteren bietet JPEG eine Farbtiefe bis zu 24 Bit an. Ein hoher Faktor bedeutet eine verluststarke Kompression, ein niedriger Faktor eine verlustarme Kompression. Je nach Vorlage lassen sich Farbbilder bis zu einem Faktor von 20 reduzieren ohne sichtliche Verluste. Eine Reduzierung bis zum Faktor 50 liefert meist eine ausreichende Bildqualität.

Das Verwenden von JPEG komprimierten Bildern ist im Internet neben PNG (*Portable Network Graphic*) zum Standard herangewachsen. Der Komprimierungsalgorithmus beruht auf folgender Idee.

Das zu komprimierende Bild wird auf Helligkeitsschwankungen und Farbraster untersucht. Weichen nun mehrere Pixel nur geringfügig z.B in ihrer Farbe oder Helligkeit ab, so werden diese quasi zusammengefasst. Der Faktor der Komprimierung gibt dem Algorithmus im wesentlichen vor, welche Abweichungen erlaubt sind. Die ausführliche Vorgehensweise kann unter http://www.mathematik.de/spurendermathematik/spudema_beitraege/beitraege/rooch/nkap04.html [SW04] nachgelesen werden.

4.3 JPEG2000

Das JPEG 2000 Verfahren wurde 2001 erstmals verabschiedet und von der Joint Photographic Experts Group entwickelt. Dieses Verfahren zählt zu den verlustfreien Komprimierungstypen. Daher ist eine Komprimierung nur in geringerem Faktor möglich. Im Gegensatz zu JPEG können in diesem Verfahren eine Vielzahl von Optionen eingestellt werden. So kann z.B bei einem Portrait das Gesicht sehr scharf erhalten bleiben, wobei der Hintergrund eine untergeordnete Rolle spielt und somit höher komprimiert werden kann. Das heisst, dass bestimmte Bereiche innerhalb eines Bildes unterschiedlich kodiert werden können. Mit dieser Art der Komprimierung ist es nun auch möglich weitere Informationen, wie z.B. das Datum der Erstellung, ein digitales Wasserzeichen oder Ähnliches, im Bild trotz der Komprimierung unterzubringen. Dies ist mit JPEG nicht möglich.

⁶http://i31www.ira.uka.de/docs/semin94/02_JPEG/ (Stand 21.04.2004)

4.4 PNG

PNG steht für Portable Network Graphics. Dieses Komprimierungsverfahren kommt aus den USA und wurde 1985 patentiert. Dieses Verfahren ist verlustfrei, bietet Transparenz und eine Farbauflösung von 281.474.976.710.656 (=48 Bit) Farben, sowie Information über Autor und ähnliches. PNG vereint die Möglichkeiten von JPEG2000, GIF, JPEG und anderen Bildkomprimierungsformaten.⁷ [dAi04]

Ursprünglich sollte dieses Format das GIF Format ablösen, zudem hat es u.a. eine breite Akzeptanz unter Internetbenutzern gefunden. Die genaue Vorgehensweise wird in einem Vortrag von Max Völkel beschrieben und kann unter der URL <http://goethe.ira.uka.de/seminare/redundanz/vortrag12/> [Völ04] eingesehen werden.

⁷<http://www.drweb.de/grafikgrundlagen/png.shtml> (Stand 14.04.2004)

5 MPEG

MPEG Verfahren wird hauptsächlich für die Komprimierung von Videodaten eingesetzt. Die Komprimierung kommt sowohl den bewegten Bildern zugute, als auch dem Audiopart. MPEG ist ein Komprimierungsverfahren, welches in verschiedenen Dateiformaten eingesetzt werden kann. Die 1988 ins Leben gerufene **Motion Picture Expert Group** Institution hatte die Aufgabe, Kompressionsalgorithmen für Audio und Videomaterial zu entwickeln. Zugleich bestand die Forderung, diese Entwicklung zum Standard zu machen, mit der Auflage, dass die Bildübertragung über lokale Netze, wie z.B. ISDN, oder auf allen Datenträger mit unterschiedlichen Datenraten funktionieren sollte.

MPEG ist als asymmetrisches Kodierungsverfahren konzipiert. Das heißt, die Komprimierung ist um ein Vielfaches aufwendiger als die Dekomprimierung.

”Das Entwicklungsziel war ein “generischer” Coder, der vom Prinzip her alle Optionen bietet, aber erst durch Weglassen von unnötigen Optionen und Parameterfestlegungen auf eine spezielle Anwendung getrimmt wird. MPEG schreibt nicht die einzelnen Kompressionsalgorithmen fest, sondern lediglich ein Dateiformat, das einen erlaubten Datenstrom beschreibt.”⁸ — [Doe04]

5.1 Grundsätzliches zu MPEG

Folgende Basis liegt allen bereits entwickelten MPEG-Video Verfahren zugrunde. Hat man zwei Bilder, so werden nicht beide vollständig kodiert, sondern nur die Details, die sich innerhalb der zwei Bilder unterscheiden. Man hat nun anstatt zwei Bilder ein kodiertes Bild mit weiterer Information über Details, die den Unterschied beschreiben. Diese werden über Vektoren dargestellt. So wird z.B. ein fahrender LKW auf einer Straße als Detail behandelt, das sich von Bild zu Bild verändert, der Hintergrund bleibt jedoch gleich. Nun wird die Strecke, auf der sich ein LKW von einem Bild zum nächsten bewegt, als Vektor in das zu kodierende Bild mit integriert. Findet nun ein kompletter Szenen- bzw. Bildwechsel statt, wird wieder ein neues Startbild generiert, von dem aus neue Veränderungen ausgehen. Jedoch werden nicht alle Objekte als Ganzes erkannt, sondern man beschränkt sich auf 16x16 Pixel sog. Makroblöcke. Weiterhin werden sog. Schwingungsüberlagerungen, die für das Auge nicht sichtbar sind, entnommen.

5.2 MP3

Die Audiokompression macht sich die neben grundsätzlichen Komprimierungsalgorithmen die Gehörschwäche des Menschen zu Nutze. So werden alle Töne, die das menschliche Ohr nicht hören kann, die aber von Instrumenten erzeugt werden, heraus gefiltert. Diese Kombination aus Minimierung der Information auf das Wesentliche und die anschließende Komprimierung machen es möglich aus einem Musikstück der Größe 60 MB (dies entspricht einem Lied auf einer

⁸http://www.rcs.ei.tum.de/courses/seminar/realzeit_bv/mpeg/node3.html (Stand 25.03.2004)

handelsüblichen CD) ohne hörbaren Unterschied auf 6 MB (bei 128 KBit) zu kürzen. Dieses Format ist unter dem Namen MP3 bekannt (MPEG1 Audio Layer3) und kann Audiodateien bis auf ein Zwölftel komprimieren⁹ [Wik04]. Das Fraunhofer Institut für Integrierte Schaltungen (IIS), das den MP3 Algorithmus patentiert hat, hat diesen zum AAC (MPEG-2 ACC) weiterentwickelt. Mit diesem neuen Kodierungsverfahren ist eine Reduzierung der Ursprungsaudiodatei auf ein Sechszehntel ihrer Ursprungsgröße ohne hörbare Verluste möglich.

5.3 MPEG1

Ursprünglich war MPEG1 ein hardwareabhängiges Komprimierungsverfahren für Video. In der heutigen Zeit sind Rechner aufgrund ihrer Schnelligkeit in der Lage dies auf Softwarebasis zu bewerkstelligen. Es bot eine Auflösung von 352x288 (SIF Auflösung). SIF bedeutet "simple input format" und steht für das gängige Europäische und Amerikanische Fernsehformat PAL und NTSC. MPEG1 kann nicht mit dem in MPEG2 bereits integriertem Interlace Verfahren umgehen (zeilenartiger Bildaufbau, wie im Fernsehen). MPEG1 stellt 25 Frames zur Verfügung, dies entspricht 25 Bilder/Sekunde, die nötig sind, damit Bilder laufen lernen; auch bekannt unter dem Namen VCD (Video Compact Disc).

5.4 MPEG2

Diese verbesserte Art der Kompression ist u.a. konzipiert für DVD (Digital Versatile Disc) und DVB (Digital Video Broadcasting). Im Unterschied zu MPEG1 bietet MPEG2 eine wesentlich effizientere Kompression. Der Qualitätszuwachs bietet u.a. Auflösungen bis zu 1920x1152 (High Definition TV). Erstmals war es möglich Filmmaterial in 1:1 Studioqualität zu verarbeiten. Die sehr gute Kompression machte sich auch der private Anwender zu nutze. Neben DivX ist es das beliebteste Kodierverfahren, um DVDs auf CD Größe zu komprimieren; auch bekannt geworden unter dem Namen SVCD (Super Video Compact Disc).

5.5 MPEG3

"Hätte der Standard für HDTV werden sollen. Es genügte aber eine Erweiterung von MPEG-2. MPEG-3 kam daher nie zustande."¹⁰ [Wik04]

5.6 MPEG4

MPEG4 ist seit 1996 in der Entwicklung und baut auf dem MPEG2 Verfahren auf. Die Kompression an sich bleibt fast gleich, jedoch erweitert MPEG4 MPEG2 um die Option, 3D Potenziale auszuschöpfen und künstliche Intelligenzen (AI's) oder Avatare (synthetische Repräsentationen

⁹<http://de.wikipedia.org/wiki/MP3> (Stand 25.03.2004)

¹⁰<http://de.wikipedia.org/wiki/MPEG> (Stand 25.03.2004)

realer Menschen) zu visualisieren bspw. eine virtuelle Konferenz mit 3D Avataren (s. Meldung vom 14.02.2001 "Virtuelle Konferenzen mit 3D Avataren"¹¹) [Neu04].

¹¹<http://www.glossar.de/glossar/index.htm> (Stand 25.03.2004)

6 Welcher Kompression für welches Dokument

Die folgende Tabelle listet als Zusammenfassung nochmal die wichtigsten Komprimierungsverfahren und ihre Verwendungsgebiete auf.

Format	Komprimierungsart	Komprimierungsfaktor	Bemerkung
Text			
Reiner Text	Huffman	1,5-3	
PDF	keines	-	lohnt nicht
Binärdaten			
CAD-Daten	Huffman	2-4	höhere Faktoren möglich
Grafik			
Schwarzweiß, Halbton	Lempel-Ziv	1,5-3	verlustfrei, TIFF
	JPEG	2-20	
	JPEG2000	2-100	feinstufig kontrollierbar
	PNG	2-20	
Farbe, Halbton 8 bit	Lempel-Ziv	1,5-3	verlustfrei, TIFF
	PNG	1,5-4	
Farbe, Halbton 24 Bit	Lempel-Ziv	1,2-2	verlustfrei, TIFF
	PNG	3-10	
	JPEG	2-30	bei geringen Qualitätsverlusten
	JPEG2000	2-100	feinstufig kontrollierbar
Audio			
	MPEG2	5-12	
	MP3/MP3 Pro	5-80	ohne hörbare Verluste
	OggVorbis	5-16	bessere Qualität als MP3
Video			
	MPEG2	5-50	Standard für DVD Video
	MPEG4/DivX	20-100	sehr weit verbreitet

12

— [JG02]

¹²Jürgen Gulbins, Markus Seyfried, Hans Strack-Zimmermann: "Dokumenten Management" 3. Auflage Seite 322, Springer Verlag ISBN 3-540-43577-8

7 ...

Es gibt noch sehr viel mehr Komprimierungsarten, die sich mit der Text-, Video- und Audiokompression beschäftigen und hier behandelt werden sollten. Nicht zu vergessen gegensätzliche oder besser gesagt parallele Entwicklungen der proprietärer Schiene und der Open Source Community. Bereits genannte Algorithmen und solche wie OggVorbis (Pendant zu MP3), Project Mayo und andere sind aus dem einfachen Grunde hier zu kurz gekommen, da sie die Seminararbeit in ihrem Umfang sprengen würden. Jedem, der Interesse oder weitere Information über genannte Algorithmen erfahren möchte, sei das Quellenverzeichnis eine Hilfe.

Bei meinen Recherchen ist mir noch ein Komprimierungsalgorithmus besonders aufgefallen. Dieser nennt sich Lzip und verspricht, dass er jegliche Dateien bis auf 0% ihrer Originalgröße komprimieren kann. Es wird allerdings hervorgehoben, dass dieser zu den verlustbehafteten Algorithmen gehört. Man sieht, auch in der Informatikwelt hat man Humor. :-)

Die offizielle Projektseite zu diesem genialen Komprimierungsalgorithmus ist unter <http://lzip.sourceforge.net/> zu finden.

Literatur

- [dAi04] drweb.de AG iGr. *Was ist PNG ?* World Wide Web, <http://www.drweb.de/>, 2004.
- [Doe04] Eric Doenges. *MPEG*. World Wide Web, <http://www.rcs.ei.tum.de/>, 2004.
- [Fou04] FSF Free Software Foundation. *GNU is not Unix*. World Wide Web, <http://www.gnu.org>, 2004.
- [JG02] Hans Strack-Zimmermann Jürgen Gulbins, Markus Seyfried. *Dokumenten Management*. Springer Verlag ISBN 3-540-43577-8, berlin, 2002.
- [Kow04] Prof. Dr. Wolfgang P. Kowalk. *Rechnernetzte, Kompression*. World Wide Web, <http://einstein.informatik.uni-oldenburg.de>, 2004.
- [Neu04] Glossar.de Neustadt. *Glossar*. World Wide Web, <http://www.glossar.de/glossar/index.htm>, 2004.
- [Sch04] Prof. A. Schmitt. *Dialogsysteme und Grafische Datenverarbeitung*. World Wide Web, <http://i31www.ira.uka.de/>, 2004.
- [Ser04] Artem Sergienko. *Tutorials, Screendesign, Webdesign-Köln*. World Wide Web, <http://www.tutorial.artems.de>, 2004.
- [SW04] Johannes Groß Sebastian Wickenburg, Aeneas Roch. *Die JPEG Kompression*. World Wide Web, <http://www.mathematik.de/>, 2004.
- [Völ04] Max Völkel. *Portable Network Graphics*. World Wide Web, <http://goethe.ira.uka.de/>, 2004.
- [Wan04] Xin Wang. *Zip Algorithmen*. World Wide Web, <http://goethe.ira.uka.de/>, 2004.
- [Wik04] Wikipedia. *MP3*. World Wide Web, <http://de.wikipedia.org>, 2004.