Komprimieren von Daten, wie funktioniert das?

| Dipl.-Ing. (FH) Thomas Burgard

Das Komprimieren von Daten (Datenkompression) ist ein Verfahren zur Reduzierung der Daten vor dem Speichern bzw. der Übertragung. Wie aber funktioniert Datenkompression und welche Verfahren gibt es? Dieser Artikel verschafft einen detaillierten Überblick.

Man könnte zu der Ansicht gelangen, dass in den heutigen Zeiten von Hochgeschwindigkeitsnetzen, extrem schneller Datenübertragung im Internet, DSL und immer schneller werdenden Computern eine Datenkompression nicht mehr notwendig sei. Es wird jedoch schnell klar, dass bspw. unkomprimierte Videodaten zu einem großen Problem werden.

Das MP3-Kompressionsverfahren kann die Videodaten sehr effektiv reduzieren, ohne die Qualität stark zu mindern. Mit dem DivX-Kompressionsverfahren kann der Video-Filmer mühelos die digital gespeicherten und komprimierten Daten in perfekter Qualität auf dem PC speichern oder man kann sich komplette Filme (mit DivX komprimiert) anschauen. Ohne Datenkompression wäre das mit der üblichen PC- und Speicherkartentechnik nicht möglich. Gut vergleichbar sind Autoparkplätze und Massenspeicher: Die Zahl der Parkplätze kann mit der immer stärker wachsenden Automenge nicht mithalten. So ähnlich verhält sich das Problem mit Massenspeichern, die ebenfalls der immer stetig wachsenden Datenmenge auf Dauer nicht gewachsen sind.

Gut vergleichbar sind Autoparkplätze und Massenspeicher: Die Zahl der Parkplätze kann mit der immer stärker wachsenden Automenge nicht mithalten. So ähnlich verhält sich das Problem mit Massenspeichern, die ebenfalls der immer stetig wachsenden Datenmenge auf Dauer nicht gewachsen sind.

Was muss eine Datenkompression bieten?

- Die Qualität der dekomprimierten Daten muss optimal sein.
- Die Kompression soll im besten Fall verlustfrei sein. Eine verlustbehaftete Datenkompression ist dann akzeptabel, wenn die Qualität dabei noch gut ist.

Was bedeutet Datenkompression nun genau?

Genau beschrieben ist die Datenkompression eine Technik zur systematischen Reduzierung der Datenmenge, die für die Wiedergabe eines gegebenen Inhaltes in einer von einem Computer lesbaren Form erforderlich ist. Prinzipiell wird eine Datenkompression mittels eines speziellen Softwareverfahrens durchgeführt. Hierbei werden die Daten von ihrer enthaltenen Redundanz befreit und in eine komprimierte Form gepackt (daher auch die Begriffe „packen" und „entpacken").

Möglich ist eine Datenkompression durch zwei fundamentale Prinzipien:

1. **Beseitigung von Redundanz** (Kompression) innerhalb der Daten: Hierbei bleiben die originalen Daten vollständig erhalten und sind nach einer Dekompression auch vollständig rekonstruierbar. In der Mathematik spricht man von einer „bijektiven Abbildung".

Ergebnis: **Verlustfreie Kompression**

Anwendung: Prinzipiell für alle Arten von Daten (alphanumerische, grafische und akustische) sehr gut geeignet.
Software: z.B. WinZip, 7Zip, WinRar

2. **Beseitigung von Irrelevanz** (Reduktion) innerhalb der Daten: Hierbei können die Daten nicht mehr fehlerfrei rekonstruiert werden. Von irrelevanten Daten spricht man, wenn die Daten vom Beobachter/Empfänger nicht wahrgenommen werden können.

Ergebnis: **(Anwendungsspezifische) verlustbehaftete Kompression**

Anwendung: Vorzugsweise im Multimediabereich, also Audio, Video, Bilder. Anwendungsspezifisch bedeutet, dass für die jeweiligen Datentypen entspre-

chende Verfahren zum Einsatz kommen.

Die Kompressionsrate gibt das Verhältnis der ursprünglichen Größe einer Datei zu der Größe ihrer komprimierten Datei an.

Die verlustfreien Kompressionsverfahren

Wie bereits beschrieben, basieren verlustfreie Verfahren auf Redundanzen. Die Informatik spricht hier auch von „unnötigen Informationen", die bei Entfernung keinerlei Qualitätsverluste bringen. Um die Redundanzen in den Daten zu erkennen, bedient man sich unterschiedlicher Methoden. Die Methoden werden in zwei Gruppen aufgeteilt:

- Die erste Methode ermittelt die Wiederholung von einzelnen Zeichen und ganzen Zeichenkombinationen in Dateien.
- Die zweite Methode ermittelt die Redundanzen über Häufigkeitsverteilung der Zeichen (Statistik basiert) in einer Datei.

Im Folgenden werden einige wichtige verlustfreie Kompressionsverfahren beschrieben:

Wiederholungsbasierte Verfahren

Word Coding: Dieses Verfahren ist ein einfaches Kompressionsverfahren und findet in der Textkompression Anwendung. Hierbei werden sogenannte „Verzeichnisse" aller im Text vorkommenden Wörter angelegt und über einen Verweis dann aufgerufen. Das Verfahren ist sehr schnell, da jedes Wort nur einmal im Verzeichnis abgelegt wird und durch den Verweis aufgerufen werden kann. Da das Verfahren im Verzeichnis Trennzeichen verwendet, kann es auch nur für Textdateien verwendet werden.

Lauflängen-Codierung (RunLengthEncoding = RLE): Dieses Verfahren basiert auf dem Suchen von Wiederholungen von Zeichen innerhalb der Daten. Hierbei werden die Anzahl der Wiederholungen und der entsprechende Zeichenwert als Wertepaar zusammen gespeichert. Dadurch ist es möglich, die Datenmenge sehr stark zu reduzieren. Das folgende Inhaltsbeispiel soll das Verfahren vereinfacht demonstrieren. Der Inhalt soll ein Ausschnitt einer Schwarz-Weiß-Skizze darstellen:

Ursprungsdatei: WWWWWWSSSSS WWWSSSSSSWWWWWWSSSWW

Komprimierte Datei: 6xW, 5xS, 3xW, 6xS, 7xW, 3xS, 2xW

Bei diesem kleinen Beispiel lässt sich die Datenreduktion bereits gut erkennen. Prinzipiell kann man sagen, dass das RLE-Verfahren sehr einfach ist und sich für einfache Bildkompressionen verwenden lässt. Sind die Bilddateien komplex und mit sehr feinen Farbabstufungen versehen, ist dieses Verfahren nicht gut geeignet.

LZ77-Verfahren (Lempel-Ziv-Verfahren): Im Gegensatz zum RLE-Verfahren werden beim LZ77-Verfahren zusätzlich auch ganze Sequenzen (Kombinationen von Zeichen) verarbeitet. Ein kleines Beispiel soll das verdeutlichen:

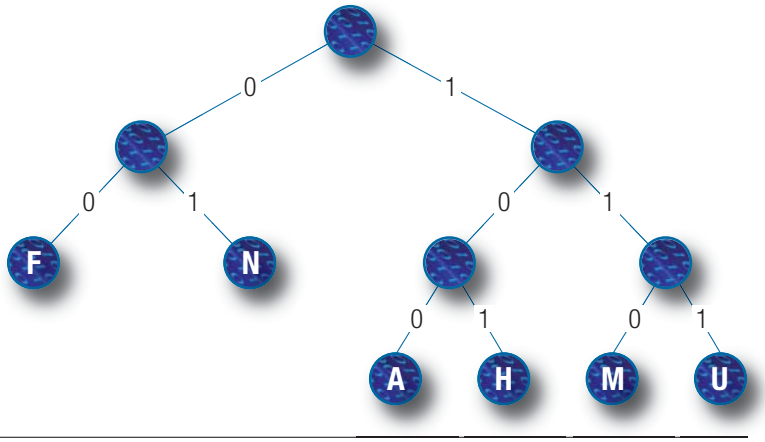
RLE-codiert: WSWSWSWSWS, 4xWS, 2xS

LZ77-codiert: 5xWS, 4xWS, 2xS

Es ist gut zu erkennen, dass das LZ77-Kompressionsverfahren nochmals eine verbesserte Verdichtung der Daten bringt. Es bietet im Allgemeinen eine gute Kompressionsleistung und wird in vielen bekannten Packprogrammen wie z.B. WinZip, PKZip, ... benutzt.
Eine Weiterentwicklung des LZ77-Verfahrens ist das LZ78-Kompressionsverfahren. Es verwendet ein eigenes „Wörterbuch", in welchem eine Folge von Zeichen unter einem Index abgelegt wird. Dieses Wörterbuch wird während der Erstellung der komprimierten Daten aufgenommen und in bestimmten Varianten bei sehr großen Datenmengen ständig angepasst. LZ78 wird bei dem verlustfreien TIFF-Dateiformat verwendet.

Häufigkeitsbasierte Verfahren

Diese verlustfreien Verfahren betrachten die Häufigkeit der Zeichen in einer Datei. In einer Textdatei wird für jedes Zeichen 8 Bit gespeichert (das Zeichen wird mit 8 Bit codiert). Wie häufig das Zeichen in der Textdatei vorkommt, spielt keine Rolle. Dies ist natürlich nicht gerade eine effektive Speichermethode. Besser wäre es doch, wenn die Häufigkeit eines vorkommenden Zeichens in der Textdatei mit berücksichtigt werden würde. Anders ausgedrückt: Ist die Häufigkeit eines Zeichens im Text groß, wird ein kurzer Code dafür verwendet. Tritt ein Zeichen selten auf, ist der Code länger, was ja o.k. ist. Häufig vorkommende Zeichen im Text werden mit zwei Zeichen codiert und selten vorkommende Zeichen werden mit mehreren Zeichen codiert. Genau diese Idee ist auch die Basis für den Morse-Code (verwendet drei Zustände: kurz, lang und Pause).
Wie funktioniert nun die Huffmann-Codierung genau? Zu Beginn wird die Häufigkeit aller vorkommenden Zeichen in einer Datei ermittelt. Auf Basis dieser Häufigkeitsermittlung wird dann ein sogenannter „Binärbaum" generiert. An diesen Baum werden immer die beiden Knoten mit den niedrigsten Zeichenhäufigkeiten an einen gemeinsamen „Elternknoten" drangehängt, der als Wert die Summe der Zeichenhäufigkeiten seiner Kinderknoten beinhaltet. Mit allen Knoten wird so verfahren. Der Binärbaum ist dann fertig generiert, wenn sich alle Knoten mit einem sogenannten „Wurzelknoten" (der erste Knoten im Baum, von dem alles ausgeht) befinden. Alle linken Äste im Baum bekommen den Wert „0" und



Generierter Binärbaum für eine Textdatei mit dem Wort „Huffmann".

alle rechten Äste den Wert „1". So können alle vorkommenden Zeichen in der Datei durch eine bestimmte Sequenz von 0 und 1 codiert werden. Die Abbildung zeigt den generierten Binärbaum der Huffmann-Codierung einer kleinen Textdatei mit dem Wort „Huffmann" (8 Buchstaben x 8 Bit = 64 Bit).

Codierungstabelle für die vorkommenden Zeichen:

F: 00
N: 01
A: 100
H: 101
M: 110
U: 111

Ergebnis-Code für das Wort Huffmann ergeben dann 20 Bit:
10111100001101000101

Die beiden Zeichen „F" und „N" treten am häufigsten auf und befinden sich deshalb in der niedrigsten Ebene mit nur zwei Bit codiert. Alle anderen Zeichen sind eine Ebene tiefer eingehängt und werden somit mit drei Bit codiert. Alle Zeichen mit der gleichen Häufigkeit befinden sich also immer in der gleichen Ebene. Das Ergebnis der Huffmann-Codierung sind also nur 20 Bit für die Speicherung im Gegensatz zur Ausgangssituation mit 64 Bit.

Die verlustbehafteten Kompressionsverfahren

Diese Kompressionsart wurde prinzipiell für ganz bestimmte Dateitypen wie Audio, Video und Bilder entwickelt. Das Ergebnis nach einer Datenkompression ist stets eine Reduktion der Daten, also immer mit einem Qualitätsverlust behaftet. Es ist aber so, dass nur die Daten aus der Datenmenge reduziert werden, die dem menschlichen Auge nicht auffallen.

JPEG-Kompression

Diese JPEG- (Joint Photographic Expert Group-)Kompressionsart ist wohl das bekannteste Verfahren für Bilddateien und ist sehr leistungsfähig. Das Verfahren verwendet gleich mehrere Kompressionsverfahren, die nacheinander zum Zuge kommen. Die JPEG-Kompression verfolgt folgende Ziele:

- Unabhängigkeit von der Bildbeschaffenheit
- Kompressions-Komplexität, die noch akzeptabel ist
- Die Bildqualität (durch die Kompression) soll vom Anwender beeinflussbar sein.

Was ist bei einer JPEG-Datei für den Anwender zu beachten? Wird eine Bilddatei mit einer Bildbearbeitungssoftware geöffnet und nach einer Bearbeitung wieder gespeichert, so verschlechtert sich die Qualität der Bilddatei (Datenreduktion). Es gilt: Bei jeder Speicherung (auch ohne Änderungen) verschlechtert sich die Qualität.

MP3-Kompression (MPEG-Audio-Layer 3)

In den 1990er-Jahren entwickelt, gelangte das MPEG- (Movie Picture Expert Group-)Verfahren zu einem nie vorher dagewesenen Bekanntheitsgrad. Man kann Audiodaten um den Faktor 10 komprimieren und trotzdem kaum Unterschiede zur CD-Qualität feststellen. Prinzipiell beruht das Verfahren auf den Schwächen des menschlichen Ohres, auch als „Maskierung" bezeichnet. Zum Beispiel können sehr tiefe Töne nicht geortet werden (Subwoofereffekt) und müssen folglich nicht in Stereo gespeichert werden. Es werden diejenigen Töne entfernt, die unterhalb der Hörschwelle liegen. Sehr laute Töne überdecken nach dem Auftreten andere Töne. MP3 entfernt die unhörbaren Töne in einem bestimmten Zeitbereich. Es können also sehr viele Tondaten vom MP3-Verfahren entfernt werden, ohne dass ein merklicher Qualitätsverlust auffällt.

Video-Kompression

Die aktuelle Entwicklung in der Videodaten-Kompression ist der DivX-Codec, auch als MPEG-4 bezeichnet. Hierbei werden die Bildinformationen und -unterschiede nicht pixelweise, sondern objektweise gespeichert. Das Kompressionsverfahren erkennt aufgrund von Farb- und Helligkeitswechseln die Kanten von Objekten und kann das Bild somit in mehrere Objekte aufteilen. Diese können dann aufgrund ihrer Eigenschaften mit unterschiedlicher Genauigkeit codiert werden.

autor.

Thomas Burgard entwickelt Applikationssoftware und professionelle Internetauftritte für Unternehmen.

kontakt.

Thomas Burgard Softwareentwicklung & Webdesign
Dipl.-Ing. (FH) Thomas Burgard
Bavariastr. 18b
80336 München
Tel.: 0 89/54 07 07-10
Fax: 0 89/54 07 07-11
E-Mail: info@burgardsoft.de
www.burgardsoft.de