

Proseminar :

Audio Kompression Algorithmen Mp3

Bearbeiter: Klein-Dasdamirov Makhir
Seminar Leiter: Dr. Gerhard Schrott

Inhalt:

1. Warum braucht man Audiokompression

2. Audiokompression Arten

3. MP3

3.1 Kodierung von Stereosignalen

3.2 Der MP3-Kodierer

3.3 Der MP3-Dekodierer

4. Filterbank

5. Format des Datenstroms

6. Datenreduktionsverfahren

1. Warum braucht man Audiokompression

Der Idee der Komprimierung von Audiodateien lag der Wunsch zugrunde, bessere **Sprachqualität** über Telefonleitungen aus Kupfer zu übertragen. Dies erübrigte sich bald durch die rapide Entwicklung des Glasfasernetzes und neuen Anschlussmöglichkeiten für den Privatbenutzer.

Technologien wie ISDN brachte die Sprachübertragung entscheidend voran. Durch diese Entwicklung erhielt man neue, visionäre Ansatzpunkte für Einsatzgebiete der Audiocodierung. Erlanger Wissenschaftler wollten nicht nur die Übertragung von Sprache, sondern auch von **Musik** über Telefonleitung oder anderer Kanäle, bei denen man besonders sparsam mit der verfügbaren Datenrate umgehen muss, ermöglichen denn: Speichern von Audiodateien in CD-Qualität benötigt sehr viel Platz. Um 1 Minute mit 44,1 KHz, 16 Bit und Stereo zu archivieren benötigen wir ca. 10 MB. Um das ganze dann noch in Echtzeit über z.B. das Netz zu übertragen benötigen wir eine Leitung die 2.7 Megabit pro Sekunde bringt. Damit können wir streaming für „normale“ Anwender nicht ermöglichen.

Um nun trotzdem Audiodaten bequem über das Netz verschicken zu können müssten wir ganz empfindlich weit mit der Qualität herunter gehen um riesengroße Wartezeiten zu vermeiden oder aber die Daten komprimieren. Am Fraunhofer Institut entwickelten daraufhin die Wissenschaftler drei Jahre weiter, bis sie 1989 ein Patent anmeldeten welches 1992 von der ISO anerkannt wurde. Und in die MPEG Spezifikation integriert wurde.

Der Schlüssel zum Erfolg liegt allerdings in der anfänglichen Freigabe, der Nutzung und Entwicklung der Technologie. Mit der Entwicklung der ersten MP3 Player 1997 (AMP Advanced Multimedia Products) von Tomislav Ucelak erlangte das Format seinen endgültigen Durchbruch.

2.Audiokompression Arten

1.Verlustlose Audiokompression (Datenkomprimierung).

Verglichen mit Bildkomprimierung werden verlustfreie Komprimierungsmethoden in der Audiokomprimierung nicht so häufig verwendet. Die Hauptbenutzer von verlustfreier Komprimierung sind Toningenieure und ihre Kunden, die den Qualitätsverlust der verlustbehafteten Kompressionsmethoden vermeiden wollen.

Der Grund für die geringe Verwendung ist die geringe Kompression, die hierbei erreicht werden kann.

Die große Mehrzahl der Tonaufnahmen sind natürliche Töne, aufgenommen aus der realen Welt; solche Daten können nicht gut komprimiert werden. Man kann das mit Photos vergleichen, die sich nicht so gut komprimieren lassen wie computergenerierte Bilder. Aber auch computergenerierte Tonabfolgen können sehr komplizierte Wellenformen enthalten, die sich mit vielen Kompressionsalgorithmen nur schlecht verkleinern lassen. Dies liegt an der Natur der Schallwellen, die sich im Allgemeinen schwer vereinfachen lassen ohne eine zwangsweise verlustbehaftete Konvertierung in Frequenzfolgen, wie sie im menschlichen Ohr stattfinden.

Außerdem ändern sich die Werte der Audiosamples sehr schnell und es gibt selten Folgen von gleichen Bytes, weswegen allgemeine Datenkompressionsalgorithmen nicht gut funktionieren.

Die verlustfreien Audiocodecs unterscheiden sich von allgemeinen Algorithmen zur Datenreduktion dadurch, dass sie speziell an die typische Datenstruktur von Audiodateien angepasst sind. So können

(44,1 kHz) als Voraussetzung für die zu komprimierenden Daten annehmen. Dies hat zur Folge, dass die Audiocodecs in fast allen Fällen Audiodateien besser komprimieren als z. B. die Zip- oder RAR-Algorithmen. Die Kompressionsrate von CD-Tracks liegt meist zwischen 40 und 80 %.

Beispiele

- Shorten
- Free Lossless Audio Codec (FLAC)
- Lossless Predictive Audio Compression (LPAC)
- Lossless Transform Audio Compression (LTAC)
- OptimFROG
- Monkey's Audio
- LA (Lossless Audio)
- Apple Lossless (auch: Apple Lossless Encoding oder Apple Lossless Audio Codec)

Bei verlustfreien Codecs gibt es keine Qualitätsunterschiede des Audiosignals, weswegen man die Auswahl auf folgende Eigenschaften legen kann:

- Geschwindigkeit der Kompression sowie der Dekompression
- Kompressionsrate
- Soft- und Hardwareunterstützung

2. Verlustbehaftete Audiokompression (Datenreduktion)

Die meisten verlustbehafteten Kompressionsalgorithmen basieren auf simplen Transformationen wie modifizierte Diskrete Kosinus Transformation (MDCT), welche die aufgenommene Wellenform in ihre Frequenzabfolgen umwandeln. Einige moderne Algorithmen benutzen Wavelets, aber es ist noch nicht sicher ob solche Algorithmen besser funktionieren als die auf MDCT basierenden. Die meisten Algorithmen versuchen nicht den mathematischen Fehler zu reduzieren, sondern die subjektive menschliche Wahrnehmung der Tonfolgen zu verbessern. Da das menschliche Ohr nicht alle Information eines ankommenden Tones analysieren kann, ist es möglich eine Sounddatei stark zu verändern, ohne dass die subjektive Wahrnehmung des Hörers beeinträchtigt wird. So kann ein Codecc zum Beispiel einen Teil der sehr hohen und sehr tiefen Frequenzen (welche für Menschen fast unhörbar sind) weglassen. Auf ähnliche Weise werden Frequenzen, die durch andere Frequenzen überlagert sind, mit geringerer Genauigkeit wiedergegeben. Eine andere Art der Überlagerung ist, dass ein leiser Ton nicht erkennbar ist, wenn er unmittelbar vor oder nach einem lauten Ton kommt. Ein solches Modell der Ohr-Gehirn Verbindung, welches für diese Effekte verantwortlich ist, wird häufig psychoakustisches Modell genannt (auch: "**Psychoacoustic Model**", "**Psycho-model**" oder "**Psy-model**").

Aufgrund der Natur der verlustbehafteten Algorithmen verschlechtert sich die Qualität wenn eine solche Datei dekomprimiert und anschließend wieder komprimiert wird (Generationsverluste). Das passiert vor allem, wenn eine Audio-CD aus verlustbehafteten Audiodateien gebrannt wird und diese später wieder komprimiert wird. Dies macht verlustbehaftete Dateien sehr ungeeignet für Anwendungen in professionellen Tonbearbeitungsbereichen. Allerdings sind solche Dateien sehr beliebt bei Endbenutzern, da ein Megabyte ungefähr für eine Minute Musik bei annehmbarer Qualität reicht, was einer Kompressionsrate von ca. 1:11 entspricht.

Bearbeiter: Klein-Dasdamirov Makhir
Seminar Leiter: Dr. Gerhard Schrott

Beispiele

Bei den Beispielen werden auch die Bitraten angegeben bei denen eine komprimierte Datei kaum mehr vom Original unterscheidbar ist. (bei konzentriertem Zuhören mit gutem Zubehör und einem ausgereiften Codec des jeweiligen Kompressionsschemas; abhängig von der Art der Musik). Die Bitrate von CDs ist 1378 kbps (Kilobit pro Sekunde).

- MP2 Layer 2 Audio Codec (MPEG-1, MPEG-2): 280-400 kbps
- MP3 Layer 3 Audio Codec (MPEG-1, MPEG-2, Lame): 180-250 kbps
- Musepack: 160-200 kbps
- Vorbis: 160-220 kbps
- AAC (MPEG-2, MPEG-4): 160-220 kbps
- WMA
- ATRAC (bei MiniDiscs)
- DTS
- AC-3

3.MP3

Das Kürzel MP3 steht eigentlich für MPEG/Audio Layer-3. Damit bezeichnet man ein Verfahren zur Kompression, also zur Verdichtung von Audio-Daten. Ein Ziel bei der Entwicklung des MP3-Standards war die Verringerung der Datenmenge von Audiosignalen unter Beibehaltung der ursprünglichen Klangqualität. Mit der Komprimierung von Multimedia-Daten beschäftigt sich ein Team der „International Standards Organisation“ (ISO), dem internationalen Pendant zur DIN, dem es um Qualitätssicherung und Standardisierung geht, und der IEC, der „International Electro-Technical Commission“. Der Name dieses Teams ist „Moving Picture Experts Group“, kurz MPEG. Hauptsächlich ist das Ziel der MPEG die Videokomprimierung, der Audio-Part wurde in diesem Fall einer anderen Kompetenz übergeben. Nach jahrelanger Entwicklungszeit stellte diese Gruppe nach anderen Audio-Komprimierungs-algorithmen auch den MP3-Standard vor. Dieser große Schritt gelang Ende des Jahres 1997 dem Fraunhofer Institut für Integrierte Schaltungen (IIS) in Erlangen in der Bundesrepublik Deutschland. Als Vater des Verfahrens gilt Ing. Dr. Karlheinz Brandenburg, Abteilungsleiter des Bereichs Audiotechnik und Multimedia beim Fraunhofer Institut. Bereits zehn Jahre vorher begannen die Arbeiten an Audio-Kompressionsverfahren im Rahmen eines Projekts der Europäischen Union, des EUREKA-Projekts für digitale Audiodatenübertragung. Durch eine Kooperation mit der Universität Erlangen gelang es dem IIS schließlich, einen mächtigen Algorithmus auszuarbeiten, der von der International Standards Organisation als ISO-MPEG Audio Layer-3 standardisiert wurde.

Layer 1 ---Frequenzmaskierung.

Layer 2 ---Komplexer(welche Informationen weggelassen)

Layer 3 ---Huffmann-Kodierung.

Sound Quality	Bandwidth	<u>Mode</u>	<i>Bitrate</i>	<u>Reduction Ratio</u>
Telephone sound	2.5 kHz	mono	8 kbps	96:1
Better than hertz	4.5 kHz	mono	16 kbps	48:1
Better AM radio	7.5 kHz	mono	32 kbps	24:1
Similar FM radio	11 kHz	stereo	56-64 kbps	26-24:1
Near-CD	15 kHz	stereo	96 kbps	16:1
CD	>15kHz	stereo	112-128 kbps	14-12:1

Bearbeiter: Klein-Dasdamirov Makhir
Seminar Leiter: Dr. Gerhard Schrott

3.1 Kodierung von Stereosignalen

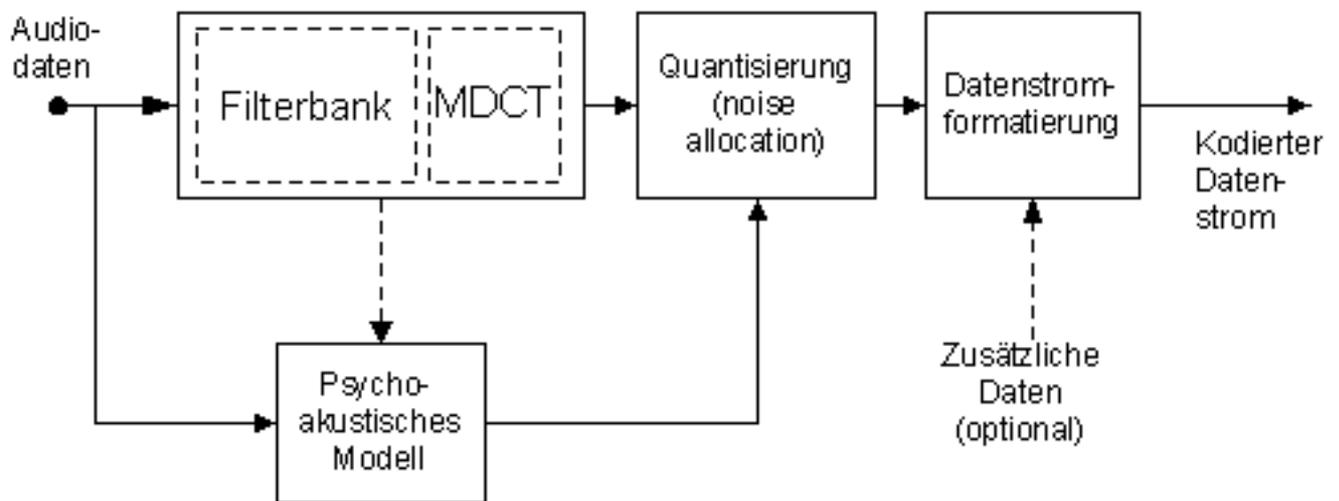
Bisher wurde nur die Kodierung eines einkanaligen Audiosignals berücksichtigt, doch Musikstücke liegen üblicherweise als Stereomaterial vor.

Prinzipiell ergibt sich der Stereoeindruck durch Phasen- und Pegeldifferenzen zwischen linkem und rechtem Kanal. Die Bestimmung der Richtung eines Schallereignisses geschieht je nach Frequenz unterschiedlich. Tiefe Töne sind räumlich gar nicht ortbar, mittlere Töne werden durch eine Kombination von Lautstärkeunterschied und Zeitpunkt des Eintreffens am linken und rechten Ohr geortet, hohe Frequenzen hingegen nur noch durch den Lautstärkeunterschied. Das Stereosignal (bestehend aus linkem und rechtem Kanal) wird vom MP3-Kodierer in die Mitte/Seite-Kodierung umgewandelt, das heißt in ein Mitten- (L+R) und ein Seitensignal (L-R) zerlegt. Das Seitensignal enthält sehr viel weniger Informationen als ein normales Monosignal und kann deshalb stärker komprimiert werden.

3.2 Der MP3-Kodierer

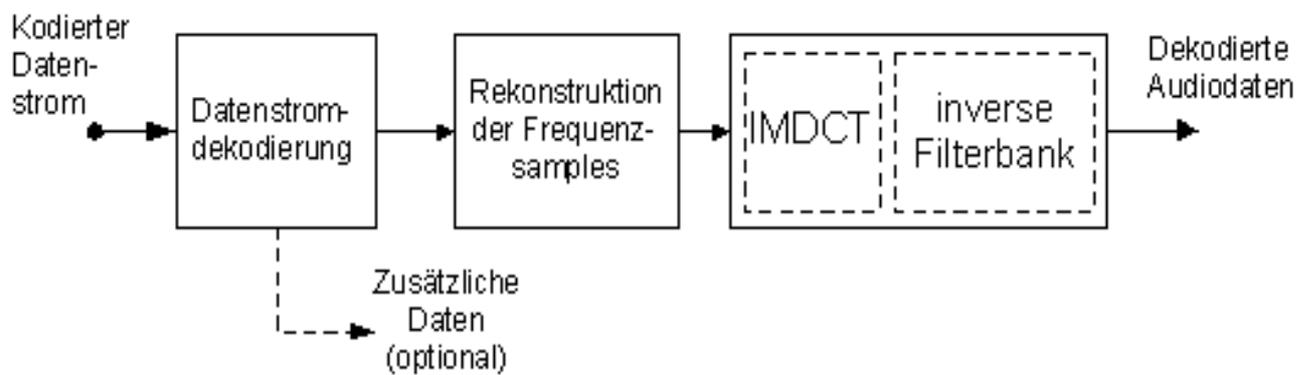
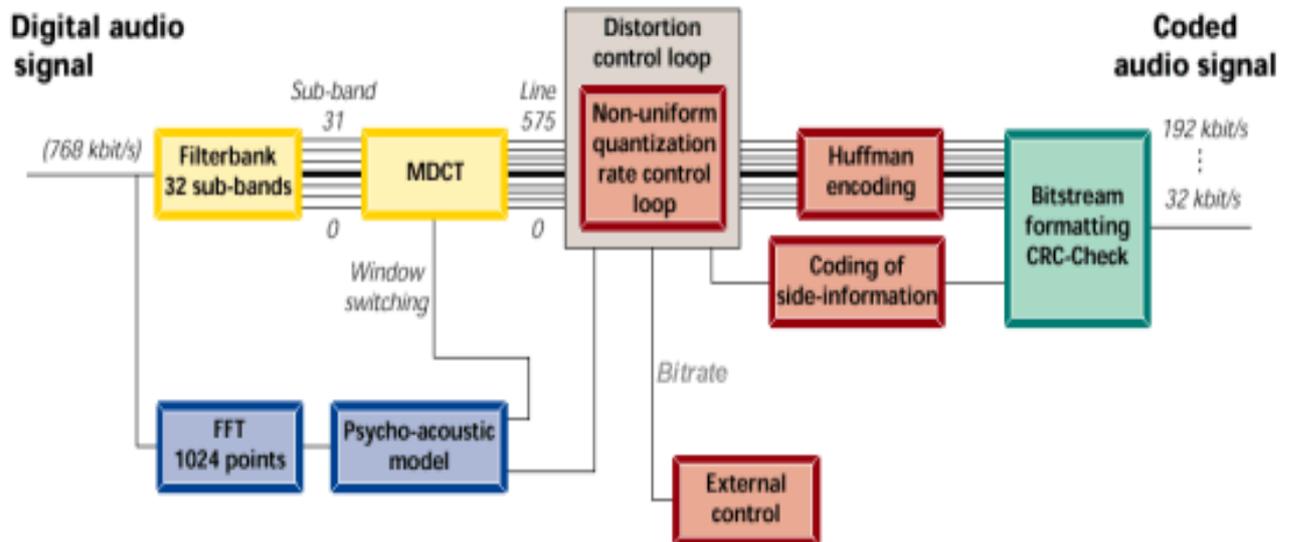
Der Komprimierungs- und Kodierungsprozeß läßt sich in mehrere Stufen unterteilen. Bild 5 zeigt den schematischen Aufbau eines MP3-Kodierers:

Bild : Blockdiagramm eines MP3-Kodierers



3.3 Der MP3-Dekodierer

Der Aufwand zur Dekodierung eines MP3-Datenstroms ist verglichen mit dem der Kodierung gering. Es müssen lediglich die kodierten Frequenzsamples rekonstruiert werden, die dann durch eine inverse modifizierte Cosinus-Transformation (IMDCT) und eine inverse Filterbank wieder in den Zeitbereich zurückgewandelt werden

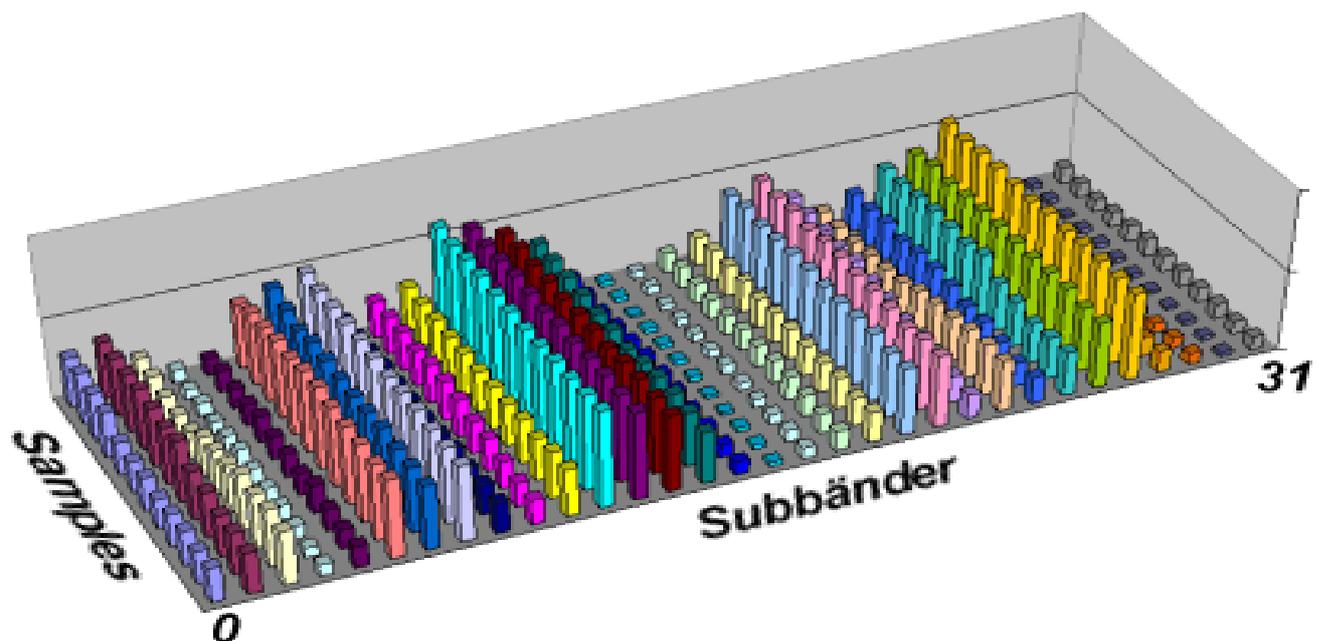


Bearbeiter: Klein-Dasdamirov Makhir
Seminar Leiter: Dr. Gerhard Schrott

4. Filterbank

Die Filterbank transformiert das zu kodierende Audiosignal vom Zeit- in den Frequenzbereich, wobei es in 32 Frequenzbänder (Subbänder) gleicher Breite unterteilt wird, welche die kritischen Bänder approximieren sollen. Für 32 eingelesene Samples wird pro Subband ein Sample ausgegeben. Eine unerwünschte Eigenschaft der Filterbank ist das sogenannte "Aliasing": Die 32 Subbänder überlappen sich, und so kann ein Ton einer festen Frequenz zwei Bänder beeinflussen.

Die Subbänder werden durch eine modifizierte diskrete Cosinus-Transformation (MDCT) jeweils nochmals in 18 Teilbereiche unterteilt. Dadurch ergibt sich eine höhere Spektralauflösung, außerdem ist es nun möglich, den Aliasingeffekt teilweise zu entfernen



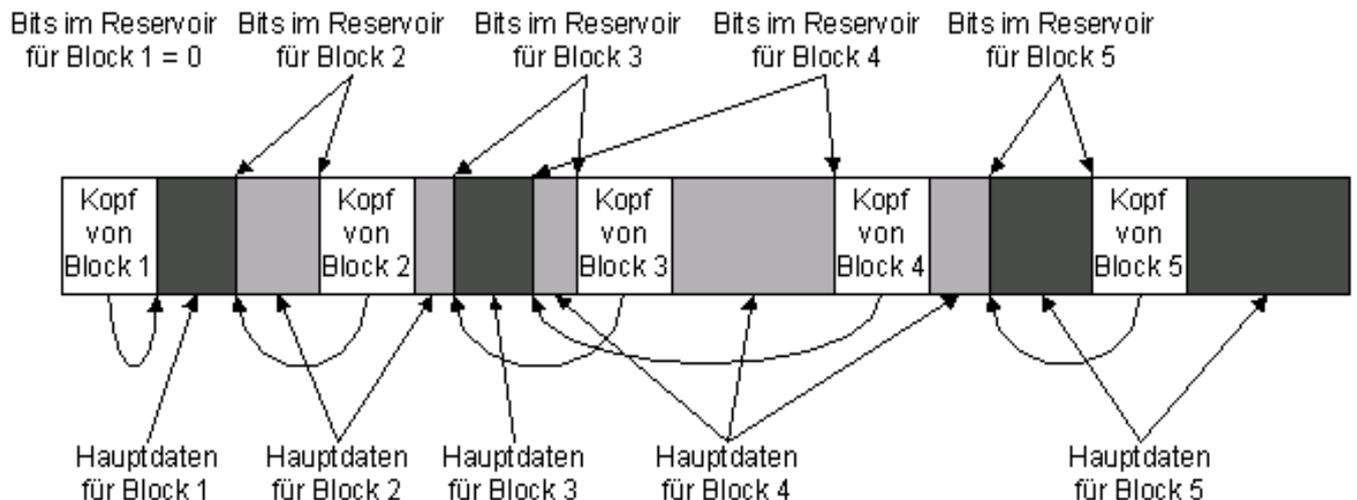
5.Format des Datenstroms

Ein MP3-Datenstrom beginnt immer mit einem Kopf, in dem z.B. Informationen über die benutzte Kodierung (Layer I bis III), die Datenrate und die Abtastfrequenz abgelegt sind. Danach folgen die einzelnen Datenblöcke mit je 1152 Samples, die folgendermaßen aufgebaut sind:

Kopf (32)	CRC (0, 16)	"Side Information"	Hauptdaten
--------------	----------------	--------------------	-------------------

Der CRC-Block (Cyclic Redundancy Check, siehe Vortrag 4) ist optional. Im Kopf des Datenstroms steht, ob er vorhanden ist.

Im "Side Information"-Block steht unter anderem, mit welchen Huffman-Bäumen die Hauptdaten kodiert sind und ein Zeiger auf den Beginn der tatsächlich zum Block gehörenden Daten. Die Hauptdaten sind Huffman-kodiert und enthalten die Skalierungsfaktoren und die quantisierten Samples. Es besteht die Möglichkeit, zusätzliche Daten in den Datenstrom einzubinden (z.B. zusätzliche Audiokanäle oder Liedtexte), die vom Dekodierer ignoriert werden, wenn er nichts mit ihnen anzufangen weiß. Wegen des Bit-Reservoirs gehören die Daten nicht zwangsläufig zum aktuellen Datenblock, wie Bild 8 verdeutlicht.



6. Datenreduktionsverfahren

Einführung

Wie werden diese tollen Kompressionsraten erreicht, ohne dass ein hörbarer Unterschied auftritt?

Maskierung

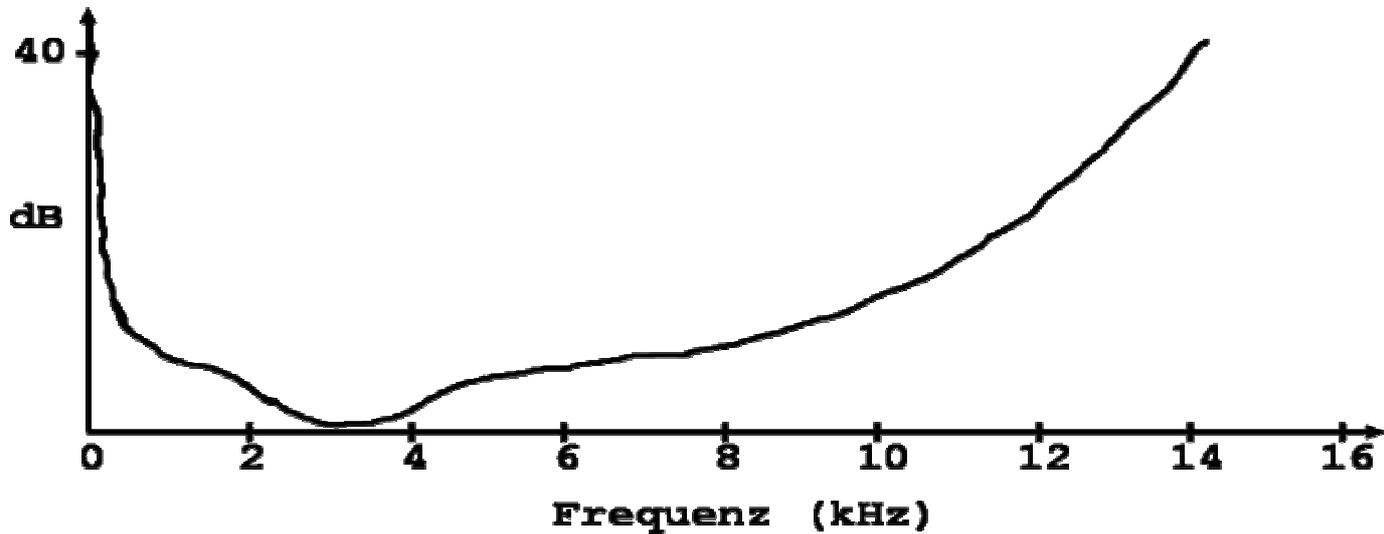
Beispiel: In der Disko haben wir 110 dB.
Gespräch - maskierung.

Frequenzmaskierung (Energieniveau)
Hörschwellenmaskierung

Encoder



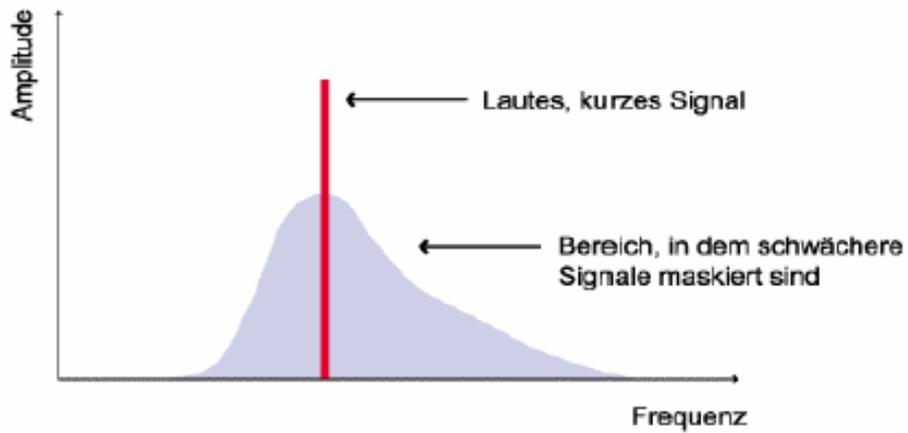
Hörschwellenmaskierung



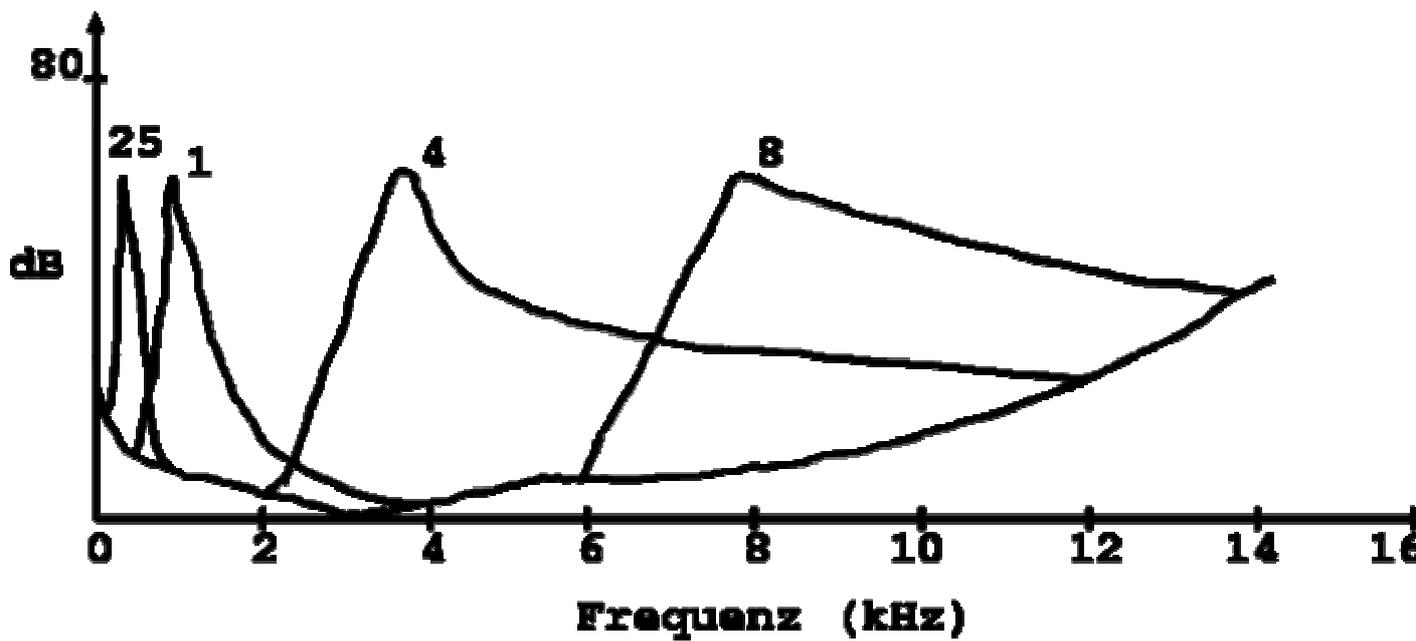
Zuerst wird in jedem Intervall jedes Paketes eine Hörschwellenmaskierung vorgenommen, d.H., dass die Signale aus dem Frequenzspektrum entfernt werden, die unter der Hörschwelle liegen (Abb. [1](#)) und somit vom Menschen gar nicht wahrgenommen werden können.

Wie in Abbildung [1](#) ersichtlich, würde z.B. ein Ton mit einer Frequenz von 16 kHz und einer Lautstärke von 40 dB entfernt werden, da das menschliche Gehör ihn nicht wahrnimmt

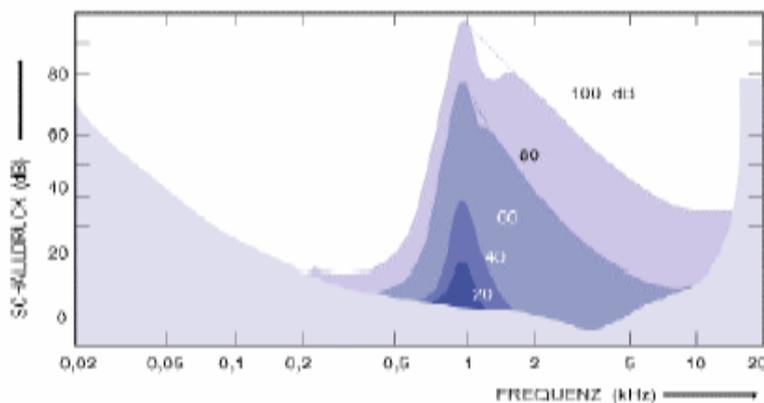
Frequenzmaskierung



Nach der Hörschwellenmaskierung folgt eine Frequenzmaskierung. Bei dieser Maskierung werden genau die Signale entfernt, die von Signalen mit einer anderen Frequenz übertönt werden.



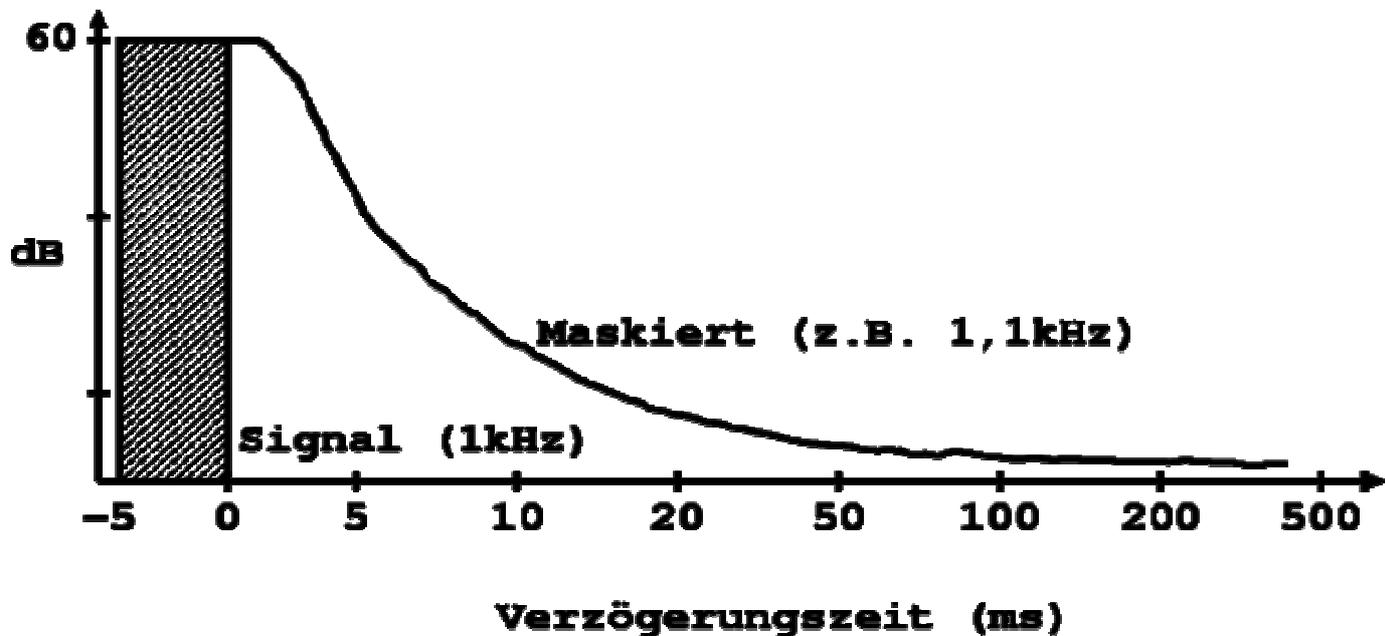
Wir haben einen Ton bei 1 kHz und einen bei 1,1 kHz. Dieser ist ca. 18 dB leiser. Er wird vom ersten Ton vollkommen verdeckt, da er unterhalb der Mithörschwelle liegt.



Also werden auch andere schwächere Töne in unmittelbarer Nähe des ersten Tones maskiert. Ein anderer Ton mit 2 kHz wäre jedoch hörbar, da er in einem anderen Frequenzband liegt.

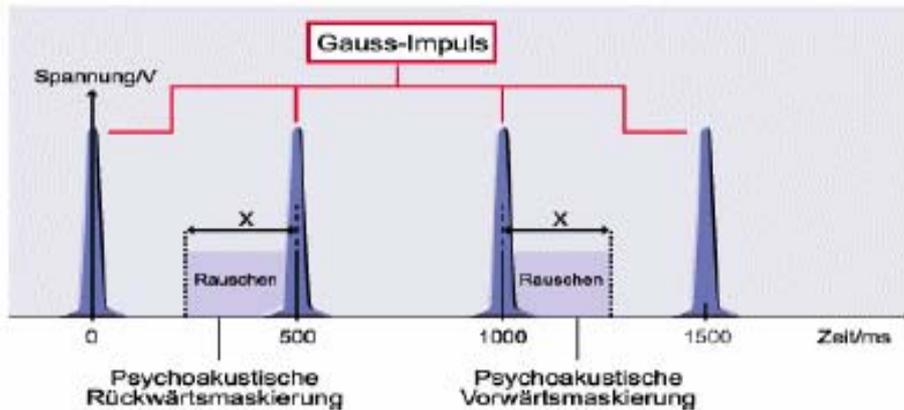
Die Grafik zeigt die Mithörschwellen bei Tönen, die von Schmalbandrauschen der Mittenfrequenz 1 kHz verdeckt werden, in Abhängigkeit von der Frequenz

Zeitmaskierung



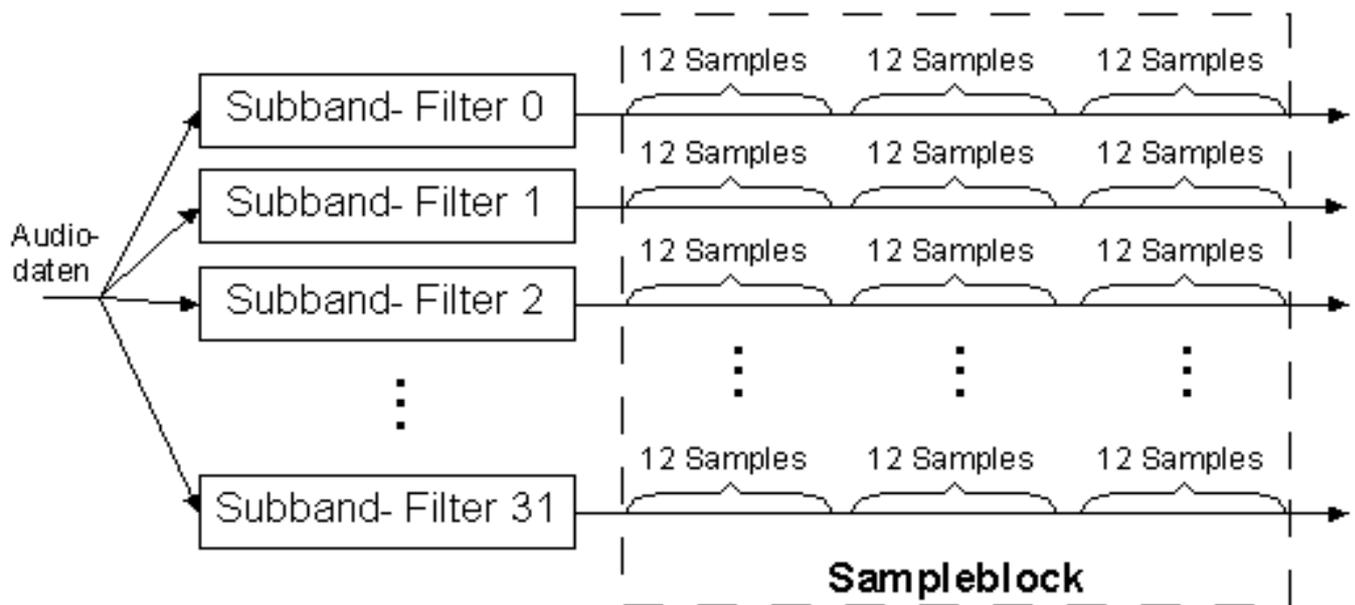
Wie bei der Frequenzmaskierung werden bei der Zeitmaskierung ebenfalls Signale entfernt, die von anderen Signalen übertönt werden.

Als weitere Komponente wird hierbei jedoch auch die zeitlich Abfolge der Signale beachtet. Wird z.B. ein Ton bei 1 kHz/60 dB abgeschaltet, maskiert dieser noch 5 Millisekunden nach dem Abschalten einen Ton mit 1,1 kHz/40 dB



Diese Art der Maskierung ist nicht nur nach dem Abschalten eines Signals zu beobachten, sondern auch in einer schwächeren Ausprägung vor dem Anschalten eines Signale.

Quantisierung



Die Frequenzsamples der Subbänder werden zu Blöcken mit je 1152 Samples zusammengefaßt (12 Samples x 32 Subbänder x 3), die gemeinsam quantisiert werden

Das prinzipielle Verfahren des Quantisierers ist folgendes: Wenn die Signalintensität eines Subbandes unter dessen Maskierung liegt, muß das Signal gar nicht kodiert werden, weil es unhörbar bleibt. Ansonsten werden die Frequenzsamples mit gerade so vielen Bits quantisiert, daß das dadurch eingeführte Quantisierungsrauschen durch die Maskierung gerade noch unhörbar bleibt.

Für jede Gruppe von 3x12 Samples eines Subbandes gibt es einen Wert, der angibt, mit wie vielen Bits Auflösung diese Samples quantisiert werden und bis zu drei Skalierungsfaktoren (höchstens einer für 12 Samples). Die Skalierungsfaktoren werden bei der Dekodierung mit den quantisierten Samples multipliziert, wodurch sich eine Verbesserung der Quantisierungsauflösung ergeben kann. Drei Skalierungsfaktoren pro Block werden nur verwendet, wenn es unbedingt nötig ist, um Verzerrungen des Signals zu vermeiden. Skalierungsfaktoren werden von zwei oder allen drei 12er-Gruppen gemeinsam genutzt wenn

(1) sich die Werte der Skalierungsfaktoren ähnlich genug sind, oder (2) wenn der Kodierer erkennt, daß die zeitliche Maskierung des Gehörs die eingeführten Fehler unhörbar macht. Es wird eine Iterationsschleife durchlaufen, die Quantisierungsparameter in geordneter Weise variiert, die Samples quantisiert und das dadurch erzeugte Quantisierungsrauschen tatsächlich ausrechnet, um zu prüfen, ob es im unhörbaren Bereich bleibt. Wenn dies nicht der Fall ist, wird für die entsprechenden Subbänder eine feinere Quantisierung gewählt ("noise allocation"). Diese Schleife benötigt den Hauptteil der Rechenzeit beim Quantisierungsprozeß. Außerdem werden die quantisierten Samples zusätzlich Huffman-kodiert, um eine weitere Reduzierung des Speicherbedarfs zu erzielen. Die verwendeten

Huffman-Bäume sind statisch. Für ein Audiosignal, das mit einer Abtastfrequenz von 44,1 kHz aufgezeichnet wurde und durch Kompression auf eine Datenrate von 128 kBit/s reduziert werden soll, ergibt sich pro Datenblock eine Größe von: $1152 \text{ (Samples/Block)} \times 128000 \text{ (Bits/s)} / 44100 \text{ (Samples/s)} = 3344 \text{ (Bits/Block)}$ Benötigt ein Block weniger als diese Anzahl an Bits, so werden die übrigen Bits an das sogenannte "Bit-Reservoir" übergeben. Läßt sich umgekehrt ein Block nicht ohne hörbaren Qualitätsverlust mit der vorgegebenen Blockgröße kodieren, so können Bits aus dem Bit-Reservoir entnommen werden, die zusätzlich zur Kodierung des Blocks verwendet werden. Die Blöcke, die nicht die ganze Blockgröße benötigen, werden mit Daten der nächsten Blöcke aufgefüllt; siehe Bild 8. Es dürfen jedoch zu keiner Zeit mehr Bits entnommen werden als im Bit-Reservoir vorhanden sind.

Literatur

Davis Yen Pan: A Tutorial on MPEG/Audio Compression, IEEE Multimedia Journal, Vol. 2, No. 2, S. 60 ff. (Als PostScript-Datei: <http://www.bok.net/~pan/>)

Davis Yen Pan: Digital Audio Compression, Digital Technical Journal, Vol. 5, No. 2, 1993 (Als PostScript-Datei:

<ftp://ftp.digital.com/pub/Digital/info/DTJ/v5n2/>

[Digital_Audio_Compression_01oct1993DTJA03P8.ps](http://www.digital.com/pub/Digital/info/DTJ/v5n2/Digital_Audio_Compression_01oct1993DTJA03P8.ps))

Matthias Carstens: Musik kompakt - Audio-Kompression mit MPEG Layer-3, c't-Magazin, Heft 21, 1998, S. 242 ff.

Björn Wesén: A DSP-based decompressor unit for high-fidelity MPEG-Audio over TCP/IP networks,

,<http://www.sparta.lu.se/~bjorn/whitney/compress.htm>

Homepage des Fraunhofer-Instituts für integrierte Schaltungen:

<http://www.iis.fraunhofer.de/amm/>

Begleitmaterial zu Kapitel 4.4 ("Audio Compression") des Kurses "Multimedia Systems" der Simon Fraser University:

<http://www.cs.sfu.ca/CourseCentral/365/li/material/notes/Chap4/Chap4.4/>

[Chap4.4_prev.html](http://www.cs.sfu.ca/CourseCentral/365/li/material/notes/Chap4/Chap4.4_prev.html)

MP3Tech: <http://mp3tech.cjb.net>

MPEG.ORG - MPEG Audio Resources and Software:

<http://www.mpeg.org/MPEG/audio.html>