

Diplomarbeit

„MPEG-4 und seine Derivate“

Ausgeführt zum Zweck der Erlangung des akademischen Grades eines
Dipl.-Ing. (FH) für Telekommunikation und Medien
am Fachhochschul-Diplomstudiengang Telekommunikation und Medien St. Pölten

unter der Leitung von

Prof. Dr. Jakob Wassermann

Zweitbegutachter: FH-Prof. Dipl.-Ing. Georg Barta

ausgeführt von

Georg Waser
tm021123

St. Pölten, am

Unterschrift:

Ehrenwörtliche Erklärung

Ich versichere, dass

- ich diese Diplomarbeit selbständig verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und mich auch sonst keiner unerlaubten Hilfe bedient habe.
- ich dieses Diplomarbeitsthema bisher weder im Inland noch im Ausland einem Begutachter/einer Begutachterin zur Beurteilung in irgendeiner Form als Prüfungsarbeit vorgelegt habe.

Diese Arbeit stimmt mit der von den Begutachtern beurteilten Arbeit überein.

.....

Ort, Datum

.....

Unterschrift

Mein Dank gilt meiner Freundin Lisi, die mir während des Studiums und im Besonderen während des Verfassens der Diplomarbeit stets zur Seite gestanden ist.

Bedanken möchte ich mich auch bei meinen Eltern, die das Studium erst ermöglicht haben.

Mein Dank gilt auch meinem Betreuer, Prof. Dr. Jakob Wassermann, der mich mit Anregungen und Verbesserungsvorschlägen bei meiner Diplomarbeit unterstützt hat.

Zusammenfassung

Seit es digitale, audiovisuelle Daten gibt, existiert das Bestreben nach Komprimierung. Ständig wird daran gearbeitet, Daten bei möglichst geringem Verlust an Qualität zu verkleinern.

In der objektorientierten Videokomprimierung kommt man nicht an MPEG-4 vorbei, dem dritten Standard der MPEG-Reihe. Aufbauend auf MPEG-2 gilt als Besonderheit das große Ziel der Filterung und einzelnen Codierung von Objekten. Als Objekte werden sowohl visuelle (Personen, Hintergrund), als auch auditive Daten angesehen, die voneinander unabhängig im Datenstrom übertragen werden sollen. Abgesehen davon wurde natürlich auch der Komprimierungsvorgang weiterentwickelt und verbessert, der vor allem durch die Einführung von MPEG-4 Part 10 eine weitere Steigerung erfuhr.

Unter den Abwandlungen von MPEG-4, den so genannten Derivaten, haben sich einige herauskristallisiert, die den hohen Ansprüchen entsprechen. In dieser Arbeit werden die Derivate DivX, XviD, HDX4, 3ivx und Nero Digital, die den Kern MPEG-4 Part 2 gemeinsam haben, miteinander verglichen, um den in verschiedenen Aspekten leistungsfähigsten auszumachen.

Mit speziellen objektiven Algorithmen, eingebunden in Anwender-Software, wird ein mathematischer Ansatz zur Ermittlung der Qualität von komprimiertem Material erstellt. Auch offensichtliche Merkmale fließen in die Beurteilung der Qualität ein.

Die besten Werte der objektiven Berechnung der Güte erzielt der Codec Nero Digital, dessen Codiergeschwindigkeit allerdings die geringste ist. Auch DivX und HDX4 erreichen gute Werte, während XviD und vor allem 3ivx zurückliegen. Ähnliche Ergebnisse bringt der Vergleich einzelner Bilder.

Abstract

MPEG-4 plays a major role in the field of video and audio compression. Based on MPEG-2 the format was standardised in 1999 and involved as the most important progression the framework of subdividing a video scene into various objects. These elements can be video, audio and other data objects. The experts group MPEG aims for arbitrary editing of objects, independent of the original scene. But of course also the act of compression has been enhanced and improved.

There are many modified formats of MPEG-4, so-called derivatives. Well-known codecs are DivX, XviD, HDX4, 3ivx and Nero Digital. They all base upon MPEG-4 but changed the process in terms of technical refinements.

To detect differences between the derivatives there are objective algorithms in software-packets available. In this thesis, discrepancies between compressed and the original footage are revealed with mathematical approaches.

The best results provides the codec Nero Digital but it was at the same time the slowest derivative. Also DivX and HDX4 made good results regarding quality. XviD and particularly 3ivx were not able to compare with the best.

Inhaltsverzeichnis

1. Einführung.....	1
2. Überblick	2
2.1. Einsatzgebiete	3
2.2. Entwicklung von MPEG	3
3. Kompression und MPEG-2-Grundlagen.....	5
3.1. Redundanz und Irrelevanz.....	5
3.2. VLC und RLC	6
3.3. DCT	8
3.4. Quantisierung	9
3.5. DPCM	10
3.6. Hybride DCT.....	12
3.6.1. Group of Pictures	13
3.7. MPEG-2.....	14
3.7.1. Profiles und Levels.....	14
3.8. MPEG-Systems	16
3.8.1. Layer	18
4. MPEG-4	20
4.1. Wavelet-Transformation	20
4.2. Bildaufbau.....	21
4.3. MPEG-4 Part 10 (AVC).....	22
4.4. Profiles und Levels	25
4.5. Skalierbarkeit.....	27
4.6. Audio	29
4.7. MPEG-4-Systems	30
4.7.1. TransMux	31
4.7.2. FlexMux	32
4.7.3. Sync-Layer.....	32
4.7.4. Präsentation	33
4.8. Objekt-Framework	33
4.8.1. Media Objects	34

4.8.2. Strukturierung	35
4.8.3. Object Description	36
4.8.4. Scene Description	39
4.8.5. Bewegungsschätzung und -kompensation.....	42
4.9. Vergleich zu MPEG-2	42
5. Derivate von MPEG-4	45
5.1. DivX	45
5.2. XviD	46
5.3. HDX4	46
5.4. 3ivx	46
5.5. Nero Digital	47
6. Empirische Studie	48
6.1. Vorgehensweise	48
6.2. Hard- und Software	49
6.3. Codec-Implementierung	50
6.4. Qualitätsbestimmung	52
7. Ergebnisse der Studie	56
7.1. Darstellung und Interpretation	56
7.1.1. PSNR	57
7.1.2. SSIM	63
7.1.3. Direktvergleich	69
8. Schlussbetrachtung.....	77
9. Literaturverzeichnis	78

Abbildungsverzeichnis

Abbildung 1: Beispiel für eine variable Längencodierung nach Huffman	6
Abbildung 2: Beispiel für eine Lauflängencodierung	8
Abbildung 3: Transformation von Grauwerten in quantisierte DCT-Koeffizienten ...	9
Abbildung 4: Bild- und Übertragungsreihenfolge bei MPEG	13
Abbildung 5: Aufbau des PES-Pakets	17
Abbildung 6: Daten- und Transportmultiplex	18
Abbildung 7: Grundprinzip der Wavelet-Transformation	21
Abbildung 8: Möglichkeit der Makroblock-Unterteilung an Objektgrenzen	23
Abbildung 9: Übertragungsmöglichkeiten von Slice-Gruppen	24
Abbildung 10: Bezüge zwischen Base und Enhancement Layers.....	29
Abbildung 11: MPEG-4-Systems Architektur.....	31
Abbildung 12: BIFS-Knotenpunkt referenziert mittels OD auf Audio-Bitstrom.....	36
Abbildung 13: Beispiel für Szene aus mehreren Objekten	39
Abbildung 14: Zeit relativ zur internen Zeit eines BIFS-Befehls gesehen.....	41
Abbildung 15: Bilderfolge aus dem 1. Ausschnitt	52
Abbildung 16: Bilderfolge aus dem 2. Ausschnitt	53
Abbildung 17: PSNR des 1. Ausschnittes bei DivX.....	57
Abbildung 18: PSNR des 2. Ausschnittes bei DivX.....	58
Abbildung 19: PSNR des 1. Ausschnittes bei XviD	59
Abbildung 20: PSNR des 2. Ausschnittes bei XviD	59
Abbildung 21: PSNR des 1. Ausschnittes bei HDX4	60
Abbildung 22: PSNR des 2. Ausschnittes bei HDX4	60
Abbildung 23: PSNR des 1. Ausschnittes bei 3ivx	61
Abbildung 24: PSNR des 2. Ausschnittes bei 3ivx	61
Abbildung 25: PSNR des 1. Ausschnittes bei Nero Digital	62
Abbildung 26: PSNR des 2. Ausschnittes bei Nero Digital	62
Abbildung 27: SSIM des 1. Ausschnittes bei DivX	63
Abbildung 28: SSIM des 2. Ausschnittes bei DivX	64
Abbildung 29: SSIM des 1. Ausschnittes bei XviD	65

Abbildung 30: SSIM des 2. Ausschnittes bei XviD	65
Abbildung 31: SSIM des 1. Ausschnittes bei HDX4	66
Abbildung 32: SSIM des 2. Ausschnittes bei HDX4	66
Abbildung 33: SSIM des 1. Ausschnittes bei 3ivx	67
Abbildung 34: SSIM des 2. Ausschnittes bei 3ivx	67
Abbildung 35: SSIM des 1. Ausschnittes bei Nero Digital	68
Abbildung 36: SSIM des 2. Ausschnittes bei Nero Digital	68
Abbildung 37: PSNR-Vergleich aller Codecs des 1. Ausschnittes	69
Abbildung 38: PSNR-Vergleich aller Codecs des 2. Ausschnittes	70
Abbildung 39: SSIM-Vergleich aller Codecs des 1. Ausschnittes.....	70
Abbildung 40: SSIM-Vergleich aller Codecs des 2. Ausschnittes.....	71
Abbildung 41: Bildvergleich bei Frame 12125	73
Abbildung 42: Bildvergleich bei Frame 13490	74
Abbildung 43: Bildvergleich bei Frame 119982	75
Abbildung 44: Bildvergleich bei Frame 120870	76

Tabellenverzeichnis

Tabelle 1: Levels und Profiles bei MPEG-2.....	15
Tabelle 2: Auswahl von MPEG-4 Video-Objekttypen	25
Tabelle 3: Auswahl von MPEG-4 Video-Profiles	26
Tabelle 4: Auswahl von Profiles mit dazugehörigen Levels	27
Tabelle 5: Chip Online Archiv: Standard-Vergleich MPEG2 versus MPEG4.....	44
Tabelle 6: Eckpfeiler des Films	48
Tabelle 7: Grundangaben zur Codierung	48
Tabelle 8: MOS in Beziehung zu PSNR.....	54
Tabelle 9: Codier-Geschwindigkeiten der Codecs	56
Tabelle 10: Dateigrößen der Codecs	56
Tabelle 11: Direktvergleich der Derivate	72

1. Einführung

MPEG-4 ist ein Standard, der in seiner wichtigsten Rolle Verfahren zur Video- und Audiokompression beschreibt. Vor allem durch das digitale Fernsehen und die Kommunikation per Video werden MPEG-4 und seine Nachfolge-Versionen immer präsenter. Basierend auf MPEG-2 stellt das Framework der Unterteilung einer Videoszene in Objekte den größten Entwicklungssprung dar. Einher mit dem Erfolg gehen zahlreiche Abwandlungen, so genannte Derivate, des Standards. Bekannte Codecs sind DivX, XviD, HDX4, 3ivx und Nero Digital.

Die Derivate bauen auf demselben Kern auf, offenbaren aber in technischen Spezifikationen ausschlaggebende Unterschiede. Daraus ergibt sich folgende Forschungsfrage:

Welcher dieser Codecs ist, verschiedene Aspekte betrachtet und zusammengefasst, am leistungsfähigsten?

Dies bedeutet grundsätzlich die Aufgabe, bei möglichst hoher Komprimierung der Daten möglichst wenig an Qualität einzubüßen. Die Güte der Endergebnisse ist nur schwer zu bewerten und obliegt zumeist einer Beurteilung des menschlichen Auges.

Mit speziellen mathematischen Algorithmen, die teilweise mit subjektiven Erkenntnissen verbunden sind und in Software-Paketen zur Verfügung stehen, wird in dieser Arbeit mit Beobachtungen verschiedener Parameter eine Annäherung an die Qualitätsfrage der Codecs erreicht. Ausgehend von sachlichen Erkenntnissen werden Stärken und Schwächen der Derivate über die allgemeine Leistungsfähigkeit entscheiden.

2. Überblick

Bei MPEG-4 handelt es sich um einen ISO/IEC Standard (International Organization for Standardization/International Electrotechnical Commission) im Bereich der Video- und Audiokompression.

Entwickelt wurde dieses Verfahren von MPEG – der Moving Pictures Experts Group. Nach dem Erfolg der beiden Vorgänger MPEG-1 und MPEG-2 kam es im Oktober 1998 zur Fertigstellung von MPEG-4. Anfang 1999 erfuhr dieses Verfahren seine Standardisierung.¹

Die jüngste Weiterentwicklung beruht nach dem Zusammenschluss mit der Spezialistengruppe ITU bei ISO-MPEG auf der Bezeichnung MPEG-4/AVC (Advanced Video Coding). Es handelt sich bei der 2003 standardisierten Version um den zehnten MPEG-4-Teil (MPEG-4 Part 10), der wie Part 2 für den Videobereich zuständig ist. Die verschiedenen „Parts“ repräsentieren die einzelnen Teilbereiche, z.B. deckt Part 3 das gesamte Audiogebiet ab.

ITU gab dem Verfahren den Namen H.264. Das Vorgänger-Format H.262 entspricht MPEG-2, während H.263 dem Advanced Simple Profile von MPEG-4 Part 2 gleichkommt.²

Die Expertengruppe zielt mit MPEG-4 auf folgende Bereiche ab: Das Format soll vermehrt beim digitalen Fernsehen, das derzeit noch von MPEG-2 dominiert wird, zum Einsatz kommen. Eingebunden wird MPEG-4 bereits bei der digitalen Übertragung per DVB-H (Handheld) z.B. für Handys und DVB-S2, der Weiterentwicklung von DVB-S (Satellitenübertragung).³

Das große Ziel ist die Erkennung und Bereitstellung von Objekten, die z.B. in Video- und Audio-Objekte unterteilt werden können. Durch das in der Entwicklungsphase befindliche Vorhaben soll es möglich werden, multimediale Objekte einer weitreichenden Palette an Anwendungen und Netzwerken zur Verfügung zu stellen.

¹ vgl. <http://www.mpegif.com/mpeg4/>

² vgl. http://de.wikipedia.org/wiki/MPEG-4_Part_10 und http://en.wikipedia.org/wiki/MPEG-4_Part_2

³ vgl. <http://www.alm.de/index.php?id=244&L=> und <http://de.wikipedia.org/wiki/DVB-S2>

Außerdem soll ein gewisser Grad an Interaktivität durchführbar sein, die bei MPEG-4 durch die Szenenbeschreibung im Entstehen ist. Der User soll diese ohne die Notwendigkeit eines Rückkanals nutzen können.

MPEG strebt einen Zugang für jedermann an. Das bedeutet, dass die Technologie nicht nur Benutzern mit hoher Bandbreite vorbehalten wird, sondern auch jene mit niedriger Bandbreite davon profitieren sollen.

Weiters ist es wichtig, dass MPEG-4 in verschiedenen und möglichst vielen Medientypen bzw. Dateitypen integriert wird, wobei der Code an jeden Typ optimal angeglichen wird. ⁴

2.1. Einsatzgebiete

MPEG-4 erschließt mit seiner „Revolutionierung“ im Kompressionsbereich neue Gebiete, auf denen dieses Verfahren eingesetzt werden kann und teilweise auch schon wird: ⁵

- Streaming über das Internet
- Mobile Kommunikation
- Echtzeit-Kommunikation
- Digitale Fernsehübertragung
- Video-Postproduktion
- Virtuelle Meetings
- Lagerung und Archivierung von Multimedia-Inhalten

2.2. Entwicklung von MPEG

Die Moving Picture Experts Group (MPEG) wurde im Jänner 1988 gegründet und setzte sich zum Ziel, einen herstellerunabhängigen Standard für die Komprimierung von Video- und Audiosignalen zu entwickeln. Ein neues

⁴ vgl. Pereira/Ebrahimi 2002, S.38f

⁵ vgl. Pereira/Ebrahimi 2002, S.56

Datenformat sollte also barrierefrei in Multimedia-Systemen zur Anwendung kommen und austauschbar sein können. Anfangs konzentrierte sich die Arbeitsgruppe nur auf Videoverfahren, später fügte man allerdings auch den Audiobereich zu.⁶

MPEG-1 wurde im Jahr 1992 eingeführt und stach, als die Videokompression noch in der Anfangsphase war, vor allem dadurch hervor, dass es möglich war, Videosequenzen auf einer CD unterzubringen.

Zwei Jahre später, 1994, brachte die Moving Pictures Experts Group mit MPEG-2 den zweiten Standard heraus, der sich über viele Jahre hinweg im audiovisuellen Sektor etablierte. Anwendungen wie DVD-Videos und Super VideoCDs bauen heutzutage darauf auf. Auch bei der Übertragung per Digital Video Broadcasting (DVB) findet MPEG-2 seine Verwendung.⁷

Als nächste Erweiterung sollte MPEG-3 als Standard-Name fungieren, doch die Entwicklung wurde eingestellt, nachdem MPEG-2 bereits die meisten neuen Funktionen beherrschte. Also übersprang die Expertengruppe diese Bezeichnung und veröffentlichte MPEG-4, mit dessen Entwicklung schon 1993 begonnen wurde.⁸

⁶ vgl. Heyna/Briede/Schmidt 2003, S.69

⁷ vgl. Heyna/Briede/Schmidt 2003, S.72f

⁸ vgl. Heyna/Briede/Schmidt 2003, S.78

3. Kompression und MPEG-2-Grundlagen

Um auf die Feinheiten von MPEG-4 eingehen zu können, muss bei den Grundsätzen von MPEG begonnen werden. Da MPEG-4 nicht von Grund auf neu entwickelt wurde und essentielle Verfahren wie zum Beispiel die Bewegungsschätzung auch schon bei MPEG-2 existieren, wird im folgenden Teil die Basis des MPEG-2-Standards erläutert.

Die Kompression setzt im allgemeinen bei Ähnlichkeiten zwischen einzelnen Bildern, aber auch innerhalb eines Bildes an. Des Öfteren besteht ein Frame aus größeren Flächen mit ähnlichen Farb- und Grauwerten. Hier liegt bezüglich der Datenmenge Einsparungspotenzial vor, es muss nicht die ganze Information jedes einzelnen Pixels übertragen werden, sondern z.B. nur die Veränderung zwischen Nachbarpixel. Weiters kommt es überwiegend vor, dass sich zwischen benachbarten Bildern in Szenen eines Films nur wenig ändert. Darum erscheint es logisch, nur die Änderungen zwischen den Bildern zu übermitteln, anstatt jedes Bild einzeln zu übertragen.⁹

Im Folgenden werden die einzelnen Aspekte der Datenreduzierung näher erläutert.

3.1. Redundanz und Irrelevanz

Bei der Redundanz handelt es sich um eine umfangreiche Beschreibung einer Information, die nicht immer notwendig ist und daher komprimiert dargestellt werden kann. Zum Beispiel muss die Information eines Standbildes nicht 25-mal in der Sekunde übertragen werden, es genügt einmal. Folglich bedeutet dies eine Redundanz im Umfang von 24 Bildern. Die Beseitigung von Redundanz stellt einen gänzlich reversiblen Prozess dar und verliert damit keinerlei Information.¹⁰

⁹ vgl. Heyna/Briede/Schmidt 2003, S.53

¹⁰ vgl. Heyna/Briede/Schmidt 2003, S.53

Im Gegensatz dazu gehen bei der Irrelevanzreduktion Informationen abhanden und können nicht wiederhergestellt werden. Welche Daten dabei verloren gehen, hängt von der jeweiligen Anwendung ab.¹¹

Die beiden Verfahren können kombiniert werden.

3.2. VLC und RLC

Die variable Längencodierung (Variable Length Coding; VLC) und die Lauflängencodierung werden zur Reduktion von Redundanz eingesetzt. Letztere wird mit RLC (Run Length Coding) oder RLE (Run Length Encoding) abgekürzt. Information geht dabei nicht verloren und kann wiederhergestellt werden.

Bei der VLC werden die einzelnen Zeichen einer ankommenden Information mit einem bestimmten Muster umformatiert. In Abbildung 1 wird der weit verbreitete Huffman-Code angewandt. Jedes Element einer Zeichenkette wird nach Auftrittshäufigkeit sortiert. Der Buchstabe „E“ kommt 4-mal vor, gefolgt von „R“ (2-mal) und „D“ bzw. „B“ (1-mal).

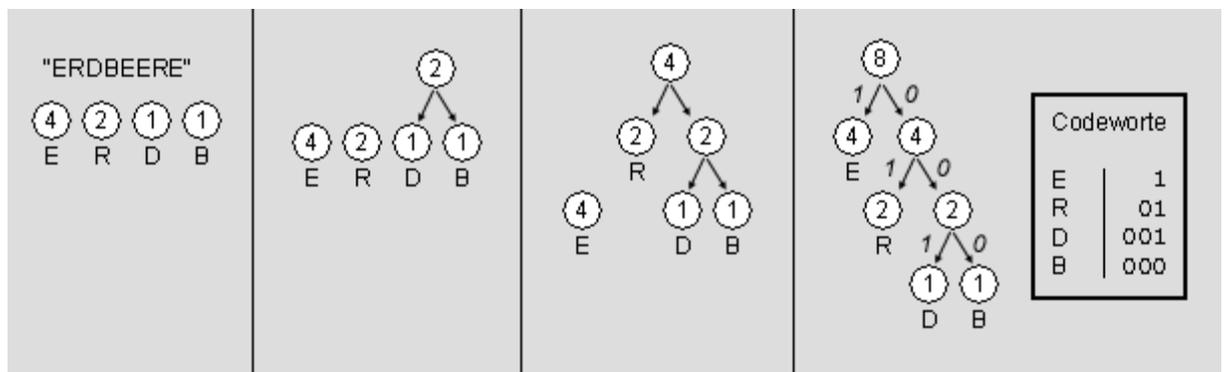


Abbildung 1: Beispiel für eine variable Längencodierung nach Huffman

Quelle: eigene Darstellung

Bei den seltensten Zeichen (hier D und B) werden die Häufigkeiten addiert und ein neuer Knoten gebildet. Wieder werden die kleinsten Ziffern – abgesehen von den bereits verarbeiteten – addiert und Knoten erzeugt, so lange, bis alle vorkommenden Zeichen erfasst sind. Das Ergebnis präsentiert sich in Form einer

¹¹ vgl. Heyna/Briede/Schmidt 2003, S.54

Baumstruktur, in der alle Verzweigungen anschließend mit den binären Zahlen versehen werden. Dieser Struktur folgend erhält man die in der Abbildung aufgelisteten jeweiligen Codeworte. Je häufiger demzufolge ein Zeichen vorkommt, desto kürzer ist dessen Codewort.

Schon beim Morsealphabet wurde die Tatsache berücksichtigt, dass gewisse Buchstaben zahlreicher vorkommen als andere – und daher eine kürzere Codierung erhalten.¹²

Wenn das Wort „ERDBEERE“ per ASCII-Code gespeichert würde, wären 8 Bytes notwendig – 1 Byte für jedes Zeichen, da die Datenverarbeitung in der Regel mit 1 Byte als kleinste Einheit zur Datenspeicherung arbeitet. Der Huffman-Code ersetzt diese Zeichenkette mit den resultierenden Codeworten – „10100100011011“. Die Zahlenfolge besteht aus 14 binären Werten, d.h. es wären lediglich 2 Bytes (14 werden zu 16 Bits aufgefüllt) notwendig. Dieser Vergleich entspricht einer Einsparung von 75 Prozent.

Um den entstandenen Code auf Empfängerseite wieder umwandeln zu können, liegt dort eine festgelegte Huffman-Tabelle bereit oder diese wird mitgeschickt.

Die Lauflängencodierung bewirkt eine zusätzliche Reduktion. Viele Daten setzen sich aus identischen, hintereinander auftretenden Bytes zusammen. Je größer deren Anzahl ist, desto mehr kann eingespart werden. Zur verkürzten Darstellung sich wiederholender Zeichen wird ein bestimmter Code verwendet, der sich aus dem Zeichen selbst, einer speziellen Markierung und der Häufigkeit zusammensetzt. Die Markierung sollte im besten Fall nicht Bestandteil der Zeichenfolge sein, da sie sonst nicht als solche erkennbar ist. Im folgenden Beispiel wird das Rufzeichen („!“) gewählt.¹³

Abbildung 2 stellt die zwei Fälle, die auftreten können, dar. Der Buchstabe „C“ kommt in der originalen Datenfolge zweimal vor und erfordert somit 2 Bytes an Speicherplatz. Gemäß der Lauflängencodierung werden die beiden Zeichen mit der Folge „C!2“ ersetzt, die allerdings 3 Bytes erfordert. Folglich müssen sich die

¹² vgl. Meyer 1999, S.254

¹³ vgl. Steinmetz 1999, S.121

Zeichen in den Originaldaten mehr als drei Mal wiederholen, um eine Komprimierung zu erreichen.

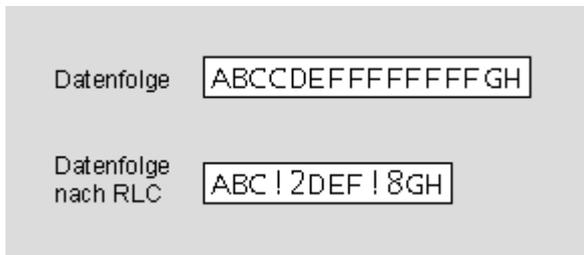


Abbildung 2: Beispiel für eine Laufängencodierung

Quelle: eigene Darstellung

Genau dies ist beim Buchstaben „F“ der Fall, dessen Folge statt 8 Bytes nur 3 Bytes („F!8“) beansprucht.

3.3. DCT

Bevor das Videosignal die Redundanzreduktion erfährt, wird es unter anderem mittels der DCT (Diskrete Cosinus Transformation) aufbereitet. DCT bearbeitet die Signale im Frequenzbereich und beruht auf der Fourier-Reihen-Entwicklung. Die Fourier-Transformation summiert unterschiedlich gewichtete Sinus- und Cosinus-Schwingungen auf und nähert sich so dem Verlauf periodischer Signale an. Im aperiodischen Bereich kommen die kontinuierliche bzw. die Diskrete Fourier-Transformation (DFT) vor. Auf Basis der DFT wurde z.B. die Fast Fourier Transformation (FFT) und eben die DCT entwickelt.

Die DCT teilt das gesamte Bild in Blöcke – meist zu 8 mal 8 Pixeln – auf. Anschließend werden die Pixelwerte in eine gleich große Matrix mit DCT-Koeffizienten transformiert, die als geordnete Darstellung zu verstehen ist und Aufschluss über die Stärke der Block-Frequenzen gibt. Während sich links oben der mittlere Grauwert des Blocks befindet, stellt die Matrix, nach rechts unten gehend, immer höhere werdende Frequenzen dar. Diese werden als AC-

Koeffizienten bezeichnet (Alternating Current), während es sich beim Mittelwert um den DC-Koeffizienten (Direct Current) handelt.¹⁴

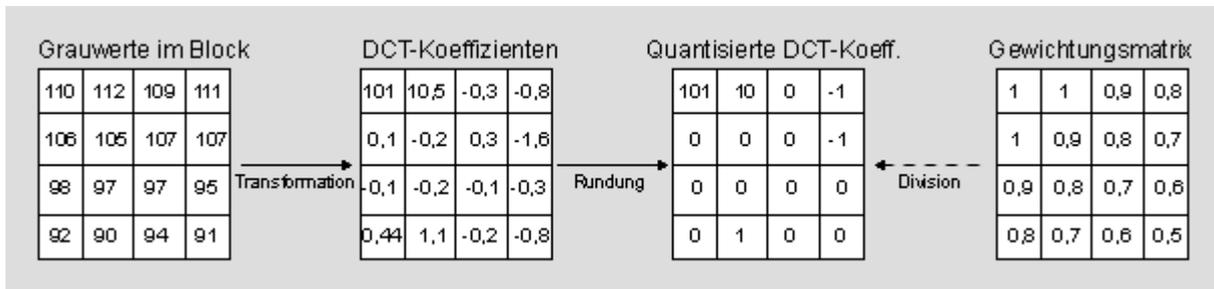


Abbildung 3: Transformation von Grauwerten in quantisierte DCT-Koeffizienten

Quelle: Heyna/Briede/Schmidt 2003, S.59

Die Berechnung der DCT-Koeffizienten geschieht mit folgender Formel:

$$F_{x,y} = \frac{2 * C(x) * C(y)}{N} * \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} f_{i,j} * \cos\left(\frac{(2i+1) * x * \pi}{2 * N}\right) * \cos\left(\frac{(2j+1) * y * \pi}{2 * N}\right)$$

N steht dabei für die Breite bzw. Länge des Blocks in Pixel, z.B. bedeutet N=8 ein Block von 8 * 8 Pixel Größe. $F_{x,y}$ ist ein N * N DCT-Koeffizient, wobei die Variablen x und y von 0 bis N-1 reichen. $f_{i,j}$ repräsentiert die N * N (i,j) Helligkeitswerte des Eingangsblocks und C(x) bzw. C(y) stehen für Konstanten:

$$C(n) = \begin{cases} \frac{1}{\sqrt{2}}, & n=0 \\ 1, & n \neq 0 \end{cases}$$

3.4. Quantisierung

Zur Quantisierung werden vorrangig die höheren Ortsfrequenzen herangezogen, die sich neben der zeitlichen Frequenz bei der Abfolge von Frames auf die Koordinaten eines einzelnen Bildes beziehen. Hohe Frequenzen machen sich durch Kanten im Bild bemerkbar und werden vom menschlichen Auge weniger

¹⁴ vgl. Heyna/Briede/Schmidt 2003, S.58ff

deutlich registriert. Aus diesem Grund können die AC-Koeffizienten grob quantisiert werden.

Im Gegensatz dazu taucht insbesondere bei zu grober Quantisierung der niedrigen Ortsfrequenzen der Nachteil dieses verlustbehafteten Verfahrens auf, der Blocking-Effekt. Bei diesem Effekt werden dadurch, dass diese Frequenzen großflächige Helligkeitsbereiche repräsentieren, die Grenzen der einzelnen Blöcke sichtbar.

Wie aus Abbildung 3 ersichtlich, werden die einzelnen Werte bei der Quantisierung gewichtet und gerundet, um eine effizient funktionierende Lauflängencodierung zu ermöglichen. Die Gewichtungsmatrix ist so angelegt, dass hier weniger Augenmerk auf die höheren Frequenzen gelegt wird.¹⁵

3.5. DPCM

Die DPCM bzw. Differenzielle Puls Code Modulation setzt im einzelnen Frame an (Intraframe-DPCM). Das Verfahren macht sich den Umstand zunutze, dass sich über ein Bild oft weite Regelmäßigkeiten erstrecken, also geringe Unterschiede zwischen Nachbarpixeln.¹⁶ Um diese Differenzen zu erfassen, wird zuerst aus einem oder mehreren vorangegangenen, abgetasteten Werten (z.B. aus einer Zeile) ein Vorhersagewert, der so genannte Prädiktionwert, erzeugt. Dieser wird anschließend vom aktuellen Abtastwert abgezogen, das Ergebnis ist der Prädiktionsfehler. Im Idealfall beträgt der Prädiktionsfehler Null, d.h. es liegt der gleiche Wert vor.

Empfangsseitig wird derselbe Prädiktionwert ermittelt. Mit Hilfe des übertragenen Prädiktionsfehlers kann nun mittels einfacher Addition der aktuelle Wert wiederhergestellt werden.

DPCM arbeitet bis hier verlustfrei. Da allerdings die Kompression noch zu gering ausfällt, werden die Prädiktionsfehler quantisiert (verlustbehaftet). Diese Quantisierung hat zur Folge, dass sich beim Prädiktor im Decoder ein kleiner

¹⁵ vgl. Heyna/Briede/Schmidt 2003, S.60

¹⁶ vgl. Heyna/Briede/Schmidt 2003, S.55f

Quantisierungsfehler einschleicht, wodurch das System nicht mehr akkurat arbeitet. Das Problem wird behoben, indem bereits dem Encoder-Prädiktor die quantisierten Werte zugeführt werden.

Die Ähnlichkeiten zwischen zeitlich aufeinander folgenden Frames sind noch stärker ausgeprägt, hier fallen die Differenzen im Schnitt noch geringer aus. (Interframe-DPCM). Der Prädiktor erhält seine Werte in diesem Fall aus mehreren, hintereinander auftretenden Bildern. Dieses Verfahren arbeitet gut, so lange sich zwischen den Frames wenig ändert. Tritt das Gegenteil ein, herrscht also viel Bewegung, wird der Prädiktionsfehler zu groß und die Effizienz der Kompression nimmt ab. Die deshalb erforderliche Bewegungsschätzung wird im folgenden Abschnitt erläutert.¹⁷

In einer Bilderfolge ändert sich, wie bereits erwähnt, nur wenig. Oft beschränken sich Bewegungen im Bild auf nur wenige Objekte, während der Rest des Bildes größtenteils gleich bleibt. Gelingt es nun, anstatt ganzer Frames nur diese Bewegungen zu erfassen und zu übertragen, wäre ein hoher Kompressionsfaktor möglich.

Bei MPEG wird für diese Bewegungsschätzung das Blockmatching-Verfahren angewandt. Hier werden die Frames in Blöcke (auch Makroblöcke genannt) zu $16 * 16$ Pixeln aufgeteilt, wobei angenommen wird, dass sich jeder Block in X- und Y-Richtung des der Bildgröße entsprechenden Koordinatensystems verschieben kann. Nun werden alle Blöcke des zeitlich vorangegangenen Bildes jeweils mit den aktuellen Blöcken verglichen. Dies geht so vonstatten, dass bei jedem Blockvergleich die Differenz der Pixelwerte der entsprechenden Blöcke gebildet wird. Anschließend werden diese Differenzen aufsummiert. Der niedrigste Betrag, der also den Blöcken mit den wenigsten Unterschieden entspricht, wird als der gesuchte Bereich im Vorgängerbild angesehen. Wie stark sich alle Blöcke in X- und Y-Richtung verschoben haben, wird in Form von Vektoren gespeichert.

Das aktuelle, aus dem Vorgänger-Bild erzeugte, wird als bewegungskompensiertes Bild bezeichnet. Bei der Interframe-DPCM werden die

¹⁷ vgl. Riempp/Schlotterbeck 1995, S.53ff

Unterschiede der beiden Bilder in Form eines Differenzbildes dargestellt. Wenn die Bewegungskompensation ihren Zweck erfüllt, müssen im Differenzbild idealerweise nur wenige Abweichungen gespeichert werden. Das bedeutet, dass nur wenige Daten übertragen werden, wodurch das Endziel einer hohen Kompression erreicht wird.

Der Decoder stößt bei der Rekonstruktion des vorhergesagten Bildes auf keinerlei Probleme. Mit dem alten Bild als Bezug kann hier mit Hilfe der übertragenen Vektoren und des Differenzbildes das neue Bild erzeugt werden.

Diese Abläufe bilden das Fundament der im nachfolgenden Kapitel behandelten Frametypen.¹⁸

3.6. Hybride DCT

Die Kombination aus den vorgestellten Prozeduren DPCM und DCT stellt die Basis des MPEG-Standards dar – die hybride DCT. Bei diesem Vorgang wird die Differenz zweier aufeinander folgender Frames gebildet und das Differenzbild im Anschluss transformationscodiert. Damit sind, je nach Qualitätsanspruch, Kompressionsfaktoren von über 500 möglich. Für eine akzeptable Beschaffenheit sollte allerdings ein Faktor im Bereich von 100 gewählt werden.¹⁹

Bei den Frames wird zwischen verschiedenen Typen unterschieden. Benachbarte Bilder sind die Grundlage der Interframe-Codierung. P-Frames repräsentieren unidirektionale Prädiktion, suchen also in einer Richtung nach Ähnlichkeiten im angrenzenden Bild.

Ohne Prädiktion arbeiten die I-Frames, sie werden rein intraframe-codiert. Hier erfolgt also kein Blockvergleich mit Nachbarbildern. I-Frames sind unerlässlich, da sie auf kein anderes Bild referenzieren und somit eventuell auftretende Fehler der Prädiktion ausmerzen können. Diese Frames kommen also in regelmäßigen

¹⁸ vgl. Riempp/Schlotterbeck 1995, S.57f

¹⁹ vgl. Heyna/Briede/Schmidt 2003, S.61

Abständen vor. Außerdem kann im Bereich der Videoproduktion der Schnitt nur an diesen prädiktionsfreien Frames vorgenommen werden.

Zur weiteren Optimierung der Kompression kommt die bidirektionale Prädiktion (B-Frames) zur Anwendung. Hier wird das aktuelle Bild nicht nur aus dem vorhergehenden, sondern zusätzlich aus dem zukünftigen geformt. Um dies zu ermöglichen, wurde auf Coder-Seite eine zeitliche Verzögerung eingebaut. Im Decoder wird die richtige Reihenfolge der Bilder wiederhergestellt.²⁰

3.6.1. Group of Pictures

Nachdem eine Bildfolge den Coder und Decoder durchlaufen hat, muss als letzter Schritt eine Bildumsortierung vorgenommen werden, da die Frames nicht in der richtigen Reihenfolge ankommen. Die entsprechenden, bereits erwähnten I-, P- und B-Frames werden dabei in einer gewissen Struktur – der Group of Pictures (GOP) – angeordnet. Wie in Abbildung 4 veranschaulicht, findet häufig GOP 12 seine Anwendung. Ein I-, drei P- und viermal zwei B-Frames ergeben dabei 12 Bilder in Folge.

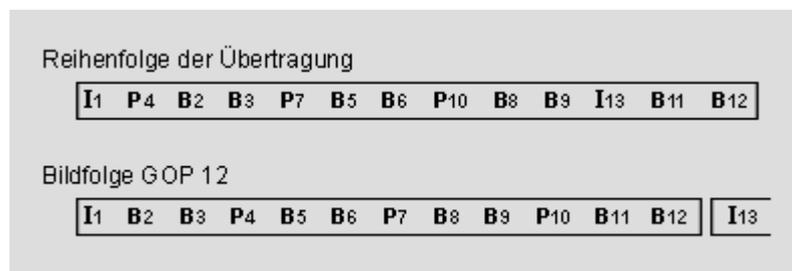


Abbildung 4: Bild- und Übertragungsreihenfolge bei MPEG

Quelle: Heyna/Briede/Schmidt 2003, S.72

Jede Group of Pictures startet mit einem I-Frame. Die bidirektionalen Bilder B_2 und B_3 erhalten ihre Informationen zur Prädiktion von I_1 und P_4 . Aus diesem Grund wird zuerst P_4 übertragen, um also den B-Frames das benötigte Referenzbild bereitzustellen.²¹

²⁰ vgl. Heyna/Briede/Schmidt 2003, S.58

²¹ vgl. Schmidt 2003, S.145

3.7. MPEG-2

Die hybride DCT stellt das Fundament der Standards MPEG-1 und MPEG-2 dar. Der neuere Standard bietet gegenüber seinem Vorgänger mehrere Bildauflösungen bis hin zu HDTV (1920 * 1152 Pixel) an. Eine andere Weiterentwicklung ist die MPEG-2-Fähigkeit, zwischen dem progressiven und Interlaced-Modus umschalten zu können, wogegen MPEG-1 nur im Vollbildmodus (progressiv) arbeiten kann.²²

Eine wesentliche Erweiterung ist die Skalierung. Dabei wird zwischen der räumlichen, der Amplituden-Skalierung und der Skalierung der Rate unterschieden.

Erstere ermöglicht eine Veränderung der horizontalen und vertikalen Auflösung (z.B. 352 * 288 Pixel) in demselben Datenstrom. Bei den Pixeln handelt es sich um die Luminanzwerte, während die Chrominanzanteile jeweils um den Faktor 2 unterabgetastet werden.

Die Skalierung der Rate bedeutet die Wiedergabe einer unterschiedlichen Anzahl von Frames in der Sekunde. Dies geschieht mit Hilfe von I-Frames, wobei ein regelmäßiges Auftreten über die ganze Sequenz vorausgesetzt wird.

Die Amplituden-Skalierung bewirkt eine unterschiedliche Bittiefe bzw. Auflösung der Quantisierung der DCT-Koeffizienten.²³

3.7.1. Profiles und Levels

MPEG-2 bietet ein so reichhaltiges Repertoire an verschiedenen Einstellungen und Funktionen an, dass die wichtigsten zusammengefasst werden. Es stehen fünf Profile bei vier Levels zur Verfügung.

Low Level mit einer Auflösung von 352 * 288 Pixel dient der Abwärtskompatibilität zu MPEG-1, während Main Level (720 * 576) der gegenwärtigen

²² vgl. <http://en.wikipedia.org/wiki/MPEG-2>

²³ vgl. Steinmetz 1999, S.161

Standardauflösung entspricht. Die beiden High Level wurden für HDTV-Anwendungen definiert.

Simple Profile ist am wenigsten aufwändig und arbeitet auch nicht mit bidirektionalen Frames. Am häufigsten kommt das Main Profile in Verbindung mit dem Main Level (MP@ML) zum Einsatz. Hier wird nicht das volle Potenzial von MPEG-2 ausgeschöpft, jedoch eignet es sich für viele Anwendungen. MP@ML ist vergleichbar mit der Qualität eines Standard-PAL-Signals und benötigt eine Datenrate von etwa 5 Mbps. Die maximal zulässige Datenrate liegt in diesem Fall bei 15 Mbps.

Profiles Levels	Simple Profile 4:2:0	Main Profile 4:2:0	SNR Scalable Profile 4:2:0	Spatially Scalable Profile 4:2:0	High Profile 4:2:0 oder 4:2:2
High Level < 60 fps		1920 * 1152 < 80 Mbps			1920 * 1152 < 100 Mbps
High 1440 < 60 fps		1440 * 1152 < 60 Mbps		1440 * 1152 < 60 Mbps	1440 * 1152 < 80 Mbps
Main Level < 30 fps	720 * 576 < 15 Mbps	720 * 576 < 15 Mbps	720 * 576 < 15 Mbps		720 * 576 < 20 Mbps
Low Level < 30 fps		352 * 288 < 4 Mbps	352 * 288 < 4 Mbps		

Tabelle 1: Levels und Profiles bei MPEG-2

Quelle: Heyna/Briede/Schmidt 2003, S.74

Die folgenden, skalierbaren Profile SNR Scalable, das sich auf den Signal-Rausch-Abstand (SNR) bezieht, und Spatially Scalable sind für den Broadcasting-Einsatz gedacht. SNR Scalable spielt bei der terrestrischen, digitalen Ausstrahlung eine Rolle, es soll den so genannten Brickwall-Effekt unterbinden. Bei diesem Effekt reißt der Datenstrom für das Fernsehbild aufgrund von Signalfehlern innerhalb kurzer Zeit ab. Das Profil soll ein Verhalten wie bei Analogsystemen, also nur sukzessive Qualitätsabnahme, garantieren. Spatially Scalable passt sich an die Gegebenheiten des Empfängers an und arbeitet mit verschiedenen Bildauflösungen, z.B. HDTV oder EDTV (Enhanced Definition).²⁴

²⁴ vgl. Heyna/Briede/Schmidt 2003, S.73ff

3.8. MPEG-Systems

Neben dem Video- und Audio-Part ist der Systems-Bereich der dritte Hauptteil bei MPEG-Videostandards. Systems garantiert den echtzeitfähigen Transport der Daten und hat Folgendes zur Aufgabe:

- Aufsplittung der Elementarströme in kleine Pakete
- Einbringen von Synchronisationsinformationen, um eine gleichlaufende Wiedergabe zu gewährleisten
- Multiplexbildung: Einzelne Elementarströme zu einem Gesamtdatenstrom vereinigen

Der erste Schritt ist eine Aufteilung der ankommenden Datenströme in Pakete, die Grunddatenströme (Packet Elementary Streams) darstellen. Diese Streams werden dabei hauptsächlich mit Daten aus der Video- und Audio-Abteilung gespeist. Ein Paket besitzt das in Abbildung 5 veranschaulichte Aussehen.

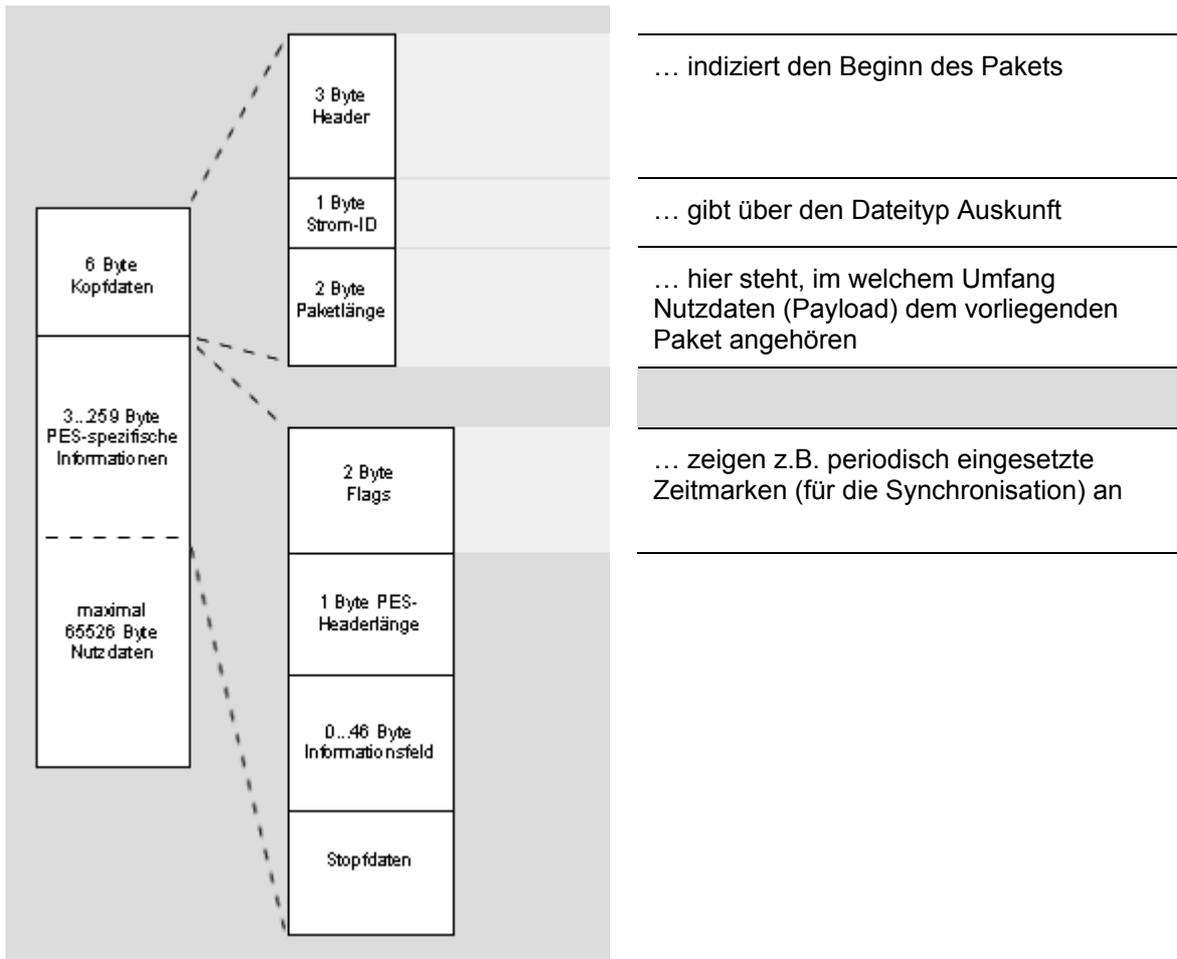
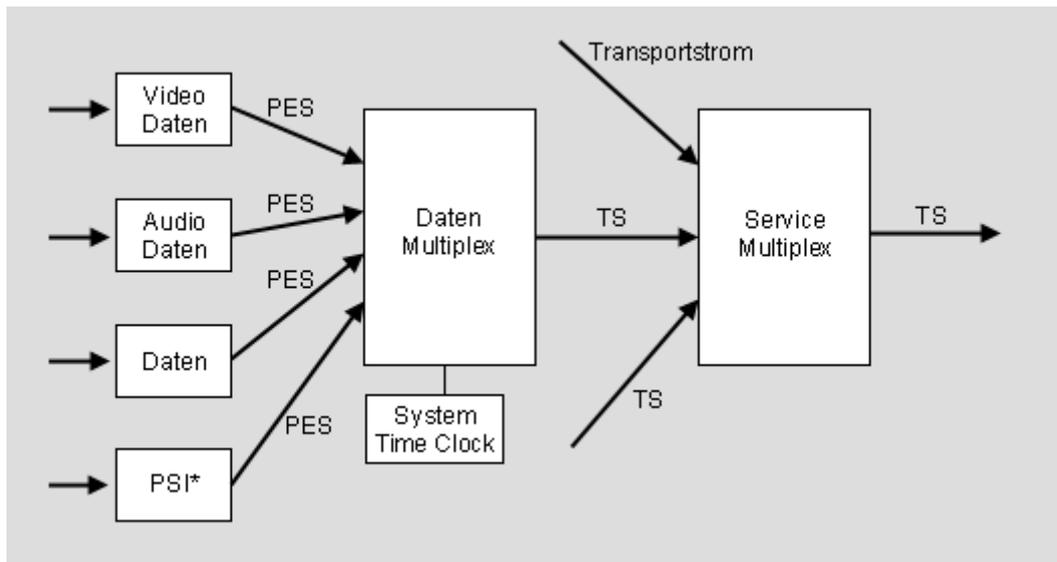


Abbildung 5: Aufbau des PES-Pakets

Quelle: Heyna/Briede/Schmidt 2003, S.125

Die Pakete aus dem Elementarstrom werden nicht direkt übertragen, sondern nochmals mit einer gewissen Kennzeichnung verpackt. Diese Kennzeichnung entscheidet darüber, ob es sich um einen Programm- oder Transportstrom handelt. Ersterer beinhaltet eine längere Folge von PES-Paketen, die allesamt zu einem einzigen Programm gehören. Der Programmstrom ist daher und aufgrund der Tatsache, dass Datenfehler nur schwer ausgeglichen werden können, für störresistente Anwendungen wie CD oder DVD geeignet.

Für das Gegenteil existiert der Transportstrom, dieser findet bei fehleranfälligen Übertragungen wie im Broadcast-Bereich seine Anwendung. Die Paketlänge ist hier fix und es können mehrere zeitliche Referenzen (Zeitbasen) vorhanden sein. Wie die Multiplex-Bildung, also das Vereinen der Datenströme, abläuft, wird in Abbildung 6 dargestellt.



* Programmspezifische Informationsdaten

Abbildung 6: Daten- und Transportmultiplex

Quelle: Heyna/Briede/Schmidt 2003, S.126

Eine wichtige Rolle in diesem Prozess spielt die Systemuhr (System Time Clock), die für die Synchronisation verantwortlich ist. Bei MPEG-1 arbeitet diese mit 90 kHz, bei MPEG-2 mit 27 MHz. Eine von der Systemuhr abgeleitete Zeitmarke (Program Clock Reference) wird regelmäßig im Datenstrom gesetzt, um zwischen Encoder und Empfänger eine Gleichmäßigkeit herstellen zu können. Diese Marken tauchen im Datenstrom mindestens alle 100 ms auf, bei DVB (Digital Video Broadcasting) sogar alle 40 ms. Außer der System Time Clock gibt es zwei relative Zeitmarken – Decoding Time Stamp und Presentation Time Stamp. Zwischen dem Decoding-Vorgang und dem Präsentationsstart muss deshalb unterschieden werden, da diese Zeitpunkte nicht zuletzt wegen der bidirektionalen Prädiktion ungleich sein können.²⁵

3.8.1. Layer

Der Videodatenstrom wird bei MPEG-2 in Schichten (Layer) aufgeteilt, wobei jede Schicht einen eigenen Header besitzt. Unter der obersten Schicht, dem Sequenz-Layer, befindet sich ein Layer, in dem die Group of Pictures definiert wird. Im

²⁵ vgl. Schmidt 2003, S.158ff

Header dieser Schicht werden Information zu Videoparametern (z.B. Pixelanzahl der Breite und Höhe) und Bitstream-Parametern (z.B. Bitrate oder Quantisierungstabelle) gespeichert.

Im Picture-Layer werden alle Einzelbilder abgelegt. Der beigeordnete Header enthält Angaben über den Timecode (Bildnummerierung) und GOP-Einflussgrößen. Die unterste Ebene wird als Block-Layer bezeichnet und birgt sowohl die eigentlichen codierten Daten als auch Quantisierungsparameter.²⁶

²⁶ vgl. Heyna/Briede/Schmidt 2003, S.75f

4. MPEG-4

MPEG-4 besticht gegenüber seinen Vorgängern mit einer Erhöhung der Kompressions- und Codiereffizienz, die in den folgenden Unterkapiteln und besonders beim 2003 standardisierten MPEG-4 Part 10 deutlich wird.

Die erste Version von MPEG-4 wurde Anfang 1999, wenige Monate nach dessen Fertigstellung, zu einem Standard. Ende dieses Jahres folgte MPEG-4 Version 2 als weiterentwickeltes und vollständig abwärtskompatibles Format nach.²⁷

Die bisher letzte verabschiedete, nochmals verbesserte Version stellt zum Beispiel mit Advanced Coding Efficiency (ACE) eine Anwendung zur Verfügung, die das Blockmatching-Verfahren mit einer Viertel-Pixel Genauigkeit betreibt und so um 30% bis 50% effizienter gegenüber MPEG-4 Version 1 arbeitet.²⁸

4.1. Wavelet-Transformation

Eine Weiterentwicklung stellt die Wavelet-Transformation dar, die beim neuesten Standard MPEG-4 eingesetzt wird.²⁹ Bei MPEG-4 Part 10 wird die Wavelet-Transformation allerdings nicht mehr angewandt. Ein Manko der DCT ist unter anderem das fallweise Auftreten von Blockartefakten. Dies wird bei der diskreten Wavelet-Transformation (DWT) unterbunden, indem das Bild als Ganzes betrachtet wird und keine Blöcke gebildet werden. Das Grundprinzip besteht aus einer Zerlegung in Teilbänder (Subbands), die Träger unterschiedlicher Frequenzen sind und eindimensionalen, digitalen Filtern unterworfen werden. Die Zeilen und Spalten eines Bildes erfahren eine Tiefpass- und Hochpassfilterung (TP bzw. HP), den Pixeln werden dabei Wavelet-Koeffizienten zugeordnet. In der Folge werden stets die rein Tiefpass-gefilterten Teilbilder von neuem einer

²⁷ vgl. <http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm>

²⁸ vgl. http://iphome.hhi.de/wiegand/assets/pdfs/video_coding_standards_03.pdf

²⁹ vgl. Watkinson 2004, S.16

Aussonderung unterzogen. Um eine Vervielfachung der Bildfläche zu verhindern, wird davor jeder zweite Koeffizient unterdrückt, der aber rekonstruierbar ist.³⁰

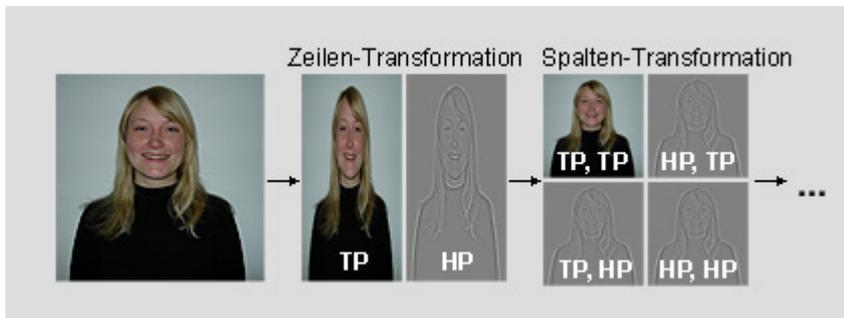


Abbildung 7: Grundprinzip der Wavelet-Transformation

Quelle: Heyna/Briede/Schmidt 2003, S.62

Die hybride DCT bleibt trotz der diskreten Wavelet-Transformation Bestandteil von MPEG-4, wird also nicht durch die DWT ersetzt. Es kommt immer die entsprechend günstigere Transformation zum Einsatz, z.B. wird bei Inhalten wie statischen Texturen im Sinne von austauschbaren Grafik-Mustern die DWT angewandt.

4.2. Bildaufbau

Die Wavelet-Koeffizienten werden vor der Übertragung einer Quantisierung unterzogen und anschließend nach abnehmender Signifikanz übermittelt. Den Vorzug erhalten jene Koeffizienten, die das Bild am wenigsten fehlerhaft erscheinen lassen. Somit ist es auch möglich, wichtige Bildteile – so genannte Regions of Interest (ROI) – zuerst aufzubauen und den Hintergrund folgen zu lassen. Weiters kann der Bildaufbau vom Benutzer jederzeit unterbrochen werden, um einen unnötigen Verbrauch von Bandbreite zu verhindern.³¹

³⁰ vgl. Riempp/Schlotterbeck 1995, S.63f

³¹ vgl. Dicks/Götting 2001, S.363ff

4.3. MPEG-4 Part 10 (AVC)

Im Jahr 2001 schloss sich die ITU (International Telecommunications Union) mit der MPEG zum Joint Video Team (JVT) zusammen. Das Ergebnis dieser Vereinigung war AVC (Advanced Video Coding) bzw. H.264. AVC ist die MPEG-Bezeichnung und repräsentiert den zehnten Teil von MPEG-4 (MPEG-4 Part 10), ITU nennt den Standard H.264.³² AVC weist folgende Veränderungen und Erweiterungen gegenüber seinen Vorgängern auf:

Die Diskrete Cosinus-Transformation (DCT; 8 * 8 Pixel) wird von der daraus abgeleiteten Integertransformation (4 * 4 Pixel) abgelöst. Ein Vorteil schlägt sich in der Reduzierung von Ringing-Artefakten nieder. Diese Artefakte treten auf, wenn ein hochfrequentes Signal zu stark quantisiert wird. Die Integertransformation kommt mit ganzen Zahlen aus, die lediglich addiert, subtrahiert oder binär verschoben werden.³³ Die DCT dagegen gebraucht irrationale Zahlen, was eventuelle geringe Rundungsfehler nach sich zieht.³⁴ Weiters existieren bei der räumlichen Prädiktion von 4 * 4 Pixel-Blöcken neun verschiedene Prädiktionsmodi. Die horizontale Prädiktion etwa füllt die Blöcke in horizontalen Reihen mit Nachbarpixel auf.

Bei der Entropiecodierung besitzt AVC neben dem Huffman-Code auch eine besser arbeitende arithmetische Codierung (Context Adaptive Binary Arithmetic Coding).³⁵ Abgekürzt CABAC, passt sich diese Anwendung dynamisch an die vorliegenden Gegebenheiten an, d.h. je nach Art der Daten wird das optimale Modell eingesetzt. Jedes Symbol wird zunächst in ein binäres Codewort umgewandelt. Anschließend erhält das Symbol eine Markierung, ob es sich um ein Syntax- oder Datenelement handelt. Darüber hinaus wird jedes Bit mit einer Zusatzinformation für das bestmögliche Modell versehen. Also unterzieht CABAC auch die Header- und Meta-Daten einer Komprimierung.

³² vgl. Watkinson 2004, S.5

³³ vgl. <http://de.wikipedia.org/wiki/H.264>

³⁴ vgl. <http://research.microsoft.com/~malvar/papers/MalvarCSVTJuly03.pdf>

³⁵ vgl. <http://de.wikipedia.org/wiki/H.264> und Watkinson 2004, S.332f

Die Bewegungsschätzung kann variiert werden. Blöcke können eine Größe von $16 * 16$ Pixeln aufweisen oder auch in $4 * 4$ -Blöcke unterteilt werden. Dies garantiert eine präzisere Erfassung von Bewegungen.

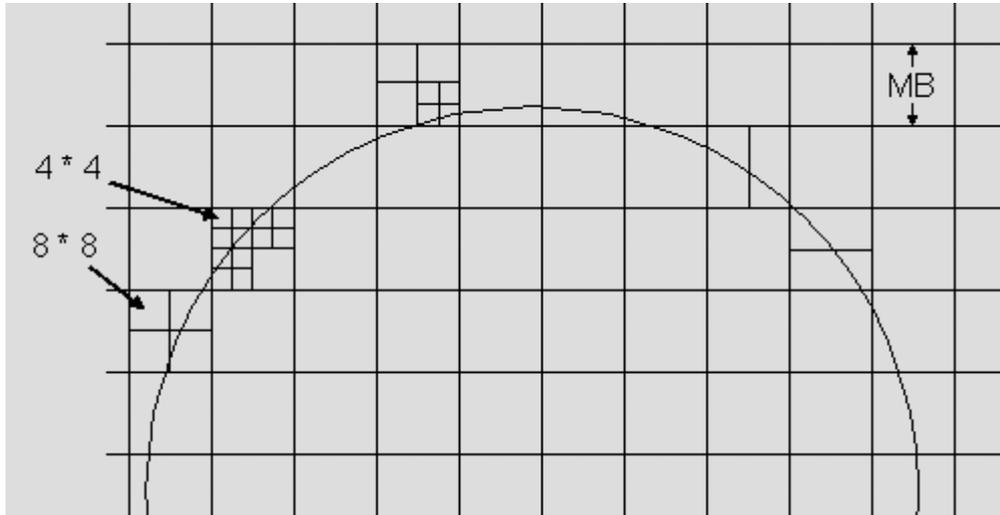


Abbildung 8: Möglichkeit der Makroblock-Unterteilung an Objektgrenzen

Quelle: Watkinson 2004, S.336

Bewegungskompensation auf $\frac{1}{4}$ Pixel genau gehört standardmäßig zum AVC-Codiervorgang. Die Vorgänger bieten dies als Option an.

Auch I-Frames werden einer Prädiktion unterworfen. Pixelwerte in Blöcken werden mit Hilfe von bereits codierten Nachbarblöcken geschätzt, lediglich die Differenzen werden übertragen. Dies entspricht einer Intraframe-Codierung nach DPCM.

P- und B-Frames werden nicht nur ein Bild weit zur Bewegungsschätzung herangezogen, sondern können beliebig viele Frames verwenden. Der Sinn liegt in der Identifikation von periodischen Bewegungen. Die Profile limitieren die Anzahl allerdings auf fünf Referenzbilder.

Werden mehrere Referenzframes verwendet, können diese verschieden gewichtet werden. Aus- und Überblendungen erfahren somit eine effizientere Codierung.

Die MPEG-Standards überlassen das Postprocessing externen Quellen. AVC hingegen verfügt über einen Deblocking-Filter. Dieser begutachtet Pixelblöcke, die

neben Objektgrenzen liegen. Weiters können die auftretenden Übergänge bzw. Unterschiede bei Pixelwerten aus dem Grad der Quantisierung abgeleitet werden. Wenn diese Werte übereinstimmen, filtert oder glättet der Deblocking-Filter die entsprechende Stelle.

Flexible Macroblock Ordering (FMO): Die Makroblöcke können innerhalb einzelner Frames bzw. Slices relativ frei angegeben werden. Daher ist es möglich, Makroblöcke – wie aus Abbildung 9 ersichtlich – z.B. entlang horizontaler oder vertikaler Objektgrenzen zu setzen. Wenn im linken Teil der Abbildung eine Slice-Gruppe verloren geht, ist immer noch ein weniger gut aufgelöstes Bild erkennbar. Im rechten Teil wird bei Verlust zumindest der wichtigere Ausschnitt des Frames bzw. das Objekt vollständig übertragen.

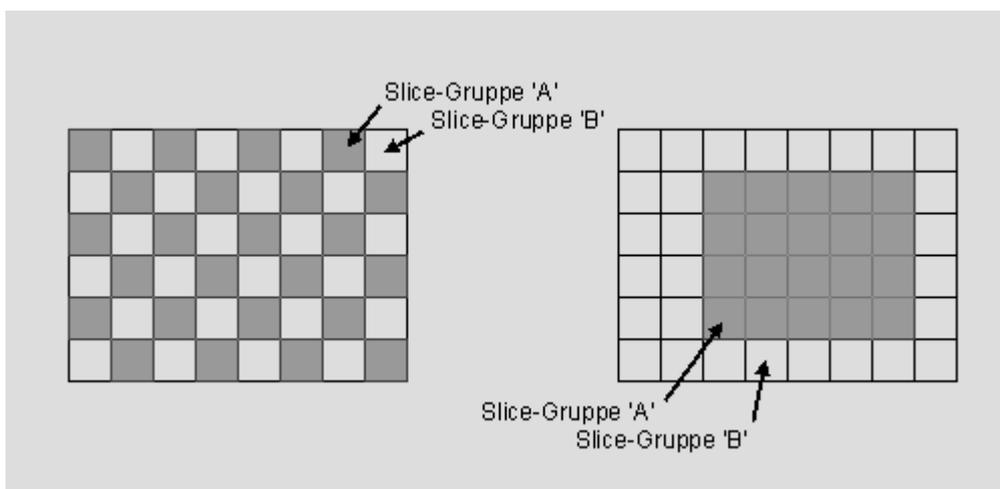


Abbildung 9: Übertragungsmöglichkeiten von Slice-Gruppen

Quelle: Watkinson 2004, S.330

AVC ist bei der Codierung etwa dreimal so effizient wie H.262 und zwei bis zweieinhalbmal wirksamer als MPEG-2. Jedoch steigt der Rechenaufwand um den Faktor 2 bis 3.³⁶

³⁶ vgl. <http://de.wikipedia.org/wiki/H.264> und Watkinson 2004, S.329ff

4.4. Profiles und Levels

Überaus wichtig bei der Entwicklung von MPEG-4 – wie auch schon bei MPEG-2 – war die Kompatibilität des Standards mit verschiedenen Systemen (Interoperabilität). Dafür wurden Profiles (Profile) und Levels (Auflösungen) geschaffen.

Profile verwenden in verschiedenen Ausführungen Tools (Werkzeuge) aus dem umfangreichen Sortiment von MPEG-4. Alle Tools auf einmal dem Anwender anzubieten, wäre zu umfassend, undurchschaubar und einfach ineffizient. Den ersten Schritt zu einer Anpassung an den Endbenutzer bilden Conformance Points (Übereinstimmungspunkte), die eine Untergruppe der Profiles darstellen. Als Beispiel wäre das Advanced Simple Visual Profile zu nennen, das bei DivX 5 zum Einsatz kommt. Dieses Profil verwendet einerseits eine auf ein Viertel-Pixel genaue Bewegungskompensation und B-Frames, andererseits kein Shape Coding. Um eben mehrere Möglichkeiten zu bieten, gibt es bei MPEG-4 zahlreiche Profile, während bei MPEG-2 nur vier Auflösungen bei fünf Profilen zur Verfügung stehen. Profilen zugrunde liegen Objekttypen, die die Erstellung vereinfachen sollen. Folgende Tabelle gibt einen Einblick:

MPEG-4 Video-Werkzeuge	Simple	Simple Scalable	Core	Main	N-Bit	ACE*
I- und P-VOP, Koeffizientenvorhersage	X	X	X	X	X	X
Kurzer Header	X		X	X	X	X
B-VOP		X	X	X	X	
Quantisierung nach Methode 1 / 2			X	X	X	X
P-VOP-basierte zeitliche Skalierbarkeit			X	X	X	X
Binary Shape			X	X	X	X
Alpha Shape				X		
Interlaced Modus				X		
Sprites				X		
Zeitliche, räumliche Skalierbarkeit		X				X
N-Bit					X	
Skalierung von willkürlichen Formen						X

Tabelle 2: Auswahl von MPEG-4 Video-Objekttypen

Quelle: Heyna/Briede/Schmidt 2003, S.90

Ein höherer Objekttyp übernimmt dabei die Fähigkeiten eines niedrigeren. Werden alle Eignungen übernommen, handelt es sich um ein Superset. Ein Objekttyp, der Teile von höheren Typen beinhaltet, ist ein Subset.

Den Grundstein legt das Simple Natural Visual Object, das nur ein rechteckiges Objekt verkörpert und für niedrigste Datenraten gedacht ist. Dies reicht hin bis zum Main Object, das unter anderem mittels des Shape Coding-Einsatzes für die beste Qualität sorgt. Für die Verwendung von synthetischen und/oder hybriden Inhalten stehen das Animated-2-D-Object, das Basic-Animated-Texture-Object und das Simple-Face-Object zur Verfügung.

	Simple	Simple Scalable	Core	Main	N-Bit	ACE
MPEG-4 Video-Objekttypen						
Simple	X	X	X	X	X	X
Simple Scalable		X				
Core			X	X	X	X
Main				X		
N-Bit					X	
ACE						X

Tabelle 3: Auswahl von MPEG-4 Video-Profiles

Quelle: Heyna/Briede/Schmidt 2003, S.91

Wie aus Tabelle 3 ersichtlich, werden bestimmte Objekttypen in Profiles verwendet, die wiederum abhängig von der Codiersituation zum Einsatz kommen. In etwa leistet das Main Profile in der Distribution im Broadcasting-Bereich oder bei interaktiven DVDs seinen Dienst. MPEG-4 Version 2 kann mit einigen neuen Profilen wie dem Advanced Real-Time Simple Profile (mit einem Rückkanal für die Videotelefonie ausgestattet) aufwarten. Außerdem wurden 2001 das Simple Studio Profile und das Core Studio Profile fertig gestellt, die den Videoschnitt im Studio mit bis zu 2 Gbps und reiner I-Frame-Codierung ermöglichen.

Levels repräsentieren verschiedene Bildauflösungen. Bei MPEG-4 reicht die Palette von Sub-QCIF (Sub Quarter Common Interchange Format), das mit einer Auflösung von 128 * 96 Pixeln arbeitet, bis hin zum professionellen Studiobetrieb (4000 * 4000 Pixel).

Profile	Level	Typische Szenengröße	Datenrate (bps)	Maximale Anzahl von Objekten
Simple	L1	QCIF	64 k	4
	L2	CIF	128 k	4
	L3	CIF	384 k	4
Core	L1	QCIF	384 k	4
	L2	CIF	2 M	16
Main	L2	CIF	2 M	16
	L3	ITU-R 601 (720 * 576)	15 M	32
	L4	1920 * 1080	38,4 M	32

Tabelle 4: Auswahl von Profiles mit dazugehörigen Levels

Quelle: In Anlehnung an: http://www.chiariglione.org/MPEG/tutorials/papers/icj-mpeg4-si/07-natural_video_paper/7-natural_video_paper.htm

Die Datenrate reicht bei den in Tabelle 4 veranschaulichten Profilen von 64 kbps bis 38,4 Mbps. Möglich sind allerdings auch Datenraten von 5 kbps für GSM (Mobilfunknetz-Standard) bis in den Gigabit-Bereich, der z.B. beim digitalen Kinofilm seine Anwendung findet. MPEG-4 unterstützt dabei sowohl den Halbbild-, als auch den Vollbild-Modus, außerdem konstante und variable Bitraten (CBR/VBR) und die Sampling-Formate 4:2:2, 4:2:0, 4:0:0.³⁷

4.5. Skalierbarkeit

Essentiell bei der Videoübertragung ist, auf den Endbenutzer einzugehen und sich dessen Verhältnissen anzupassen. Bei MPEG-4 geschieht dies in sehr effizienter Form. Mit Hilfe der Skalierbarkeit muss ein Video nur einmal codiert werden und kann anschließend trotzdem mit geänderten Parametern wiedergegeben werden – je nachdem, wie viel der Endanwender verarbeiten kann. Man unterscheidet drei Skalierungstypen:

³⁷ vgl. http://www.chiariglione.org/MPEG/tutorials/papers/icj-mpeg4-si/11-Profiles_paper/11-Profiles_paper.htm

- Örtliche Skalierung (Spatial Scalability)

Dieser Skalierungstyp sieht vor, alle Objekte in einer reduzierten Bildauflösung darzustellen. Es wird also z.B. der Teil des Bitstroms decodiert, der von einem Objekt nur die Form, nicht aber die Textur beinhaltet.³⁸

Dies geschieht im Base Layer. Gleichzeitig werden diese VOPs dazu verwendet, Video Object Planes aus den höher auflösenden Enhancement Layers vorherzusagen. Unumgänglich ist dabei die Kombination mit der Bewegungsschätzung aus dem Enhancement Layer.

- Zeitliche Skalierbarkeit (Temporal Scalability)

Ein Clip wird bei der zeitlichen Skalierbarkeit einer Eliminierung einzelner Frames unterworfen. VOPs bilden zusammen mit dem jeweiligen VOP aus dem zuletzt generierten Enhancement Layer den Grundstock zur Bewegungsschätzung. Die Verknüpfung der zeitlichen mit der feinkörnigen Skalierung ist nicht möglich, während die örtliche Skalierung mit den beiden anderen Typen kombiniert werden kann.

- Feinkörnige Skalierung (Fine Grain Scalability)

Diese Skalierung wurde eingebaut, um zwischen dem Codier- und dem Übertragungsprozess unterscheiden zu können. Im Codierbereich werden Daten in einem einzigen Bitstrom im Enhancement Layer zusammengefasst. Die Anwendung der feinkörnigen Skalierung kürzt den Strom und passt diesen dann an die Übertragungsgegebenheiten an.³⁹

Im Base Layer wird die Grundqualität eines Clips zur Verfügung gestellt. Die zur Erstellung verwendeten Objekte werden vorerst unterabgetastet (Downsampling). Alle zur Erhöhung der Qualität nötigen Informationen befinden sich in den Enhancement Layers der VOLs. Diese Informationen können als Differenz

³⁸ vgl. Heyna/Briede/Schmidt 2003, S.93f

³⁹ vgl. Pereira/Ebrahimi 2002, S.332ff

zwischen dem Original und dem unterabgetasteten Objekt angesehen werden, folglich kommen hier nur P- und B-VOPs zum Einsatz. Base Layer dagegen beinhalten I- und P-VOPs.⁴⁰

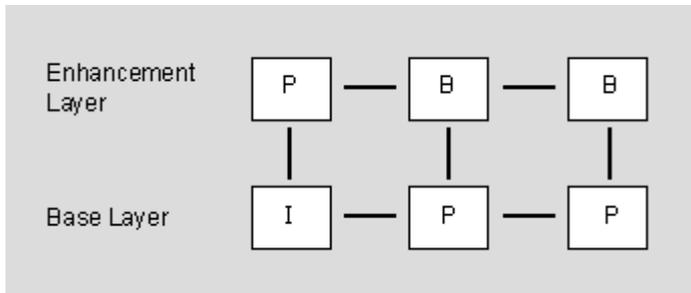


Abbildung 10: Bezüge zwischen Base und Enhancement Layers

Quelle: Heyna/Briede/Schmidt 2003, S.95

4.6. Audio

Das bekannteste Audio-Format MP3 erfuhr 1991 seine Einführung und kommt somit schon beim ersten Standard von MPEG zum Einsatz. Die Abkürzung steht für MPEG-Audio Layer-3 und ist die dritte Veröffentlichung in dieser Produkt-Reihe. Layer-1 wird praktisch nicht mehr verwendet, Layer-2 beschränkt sich vor allem auf den professionellen Bereich wie z.B. den digitalen Rundfunk (DAB, DVB). Eine Weiterentwicklung bei MPEG-2 geschah insofern, dass die Möglichkeit des Mehrkanaltons bis 5.1 eingebaut wurde und niedrigere Sample-Frequenzen eine höhere Bitraten-Effizienz zulassen. Außerdem wurden die Abtastraten des digitalen Eingangssignals erweitert. Abgesehen von MP3 nützt MPEG-2 auch das neuere Advanced Audio Coding (AAC), das als Nachfolger von MP3 gilt. Natürlich kann auch MPEG-4 auf diese Codiervarianten zurückgreifen und stellt des weiteren Algorithmen für Datenraten bis lediglich etwa 2 kbps zur Verfügung.⁴¹

Wie auch im Videobereich zielt MPEG-4 bei Audio-Daten (Audio Objects) auf Interaktivität ab. Mit AAC wird der Audio-Bereich in unterschiedliche Typen geteilt und im Bitstrom gemeinsam übertragen. Am Decoder ist es anschließend möglich,

⁴⁰ vgl. Heyna/Briede/Schmidt 2003, S.94f

⁴¹ vgl. Heyna/Briede/Schmidt 2003, S.106f

einzelne Objekte herauszugreifen. Das bedeutet, dass in einer Szene z.B. die Lautstärke der Stimme verändert werden kann, während Neben- und Hintergrundgeräusche davon nicht betroffen sind.

Weiters wird zwischen natürlichen und synthetisch hergestellten Signalen unterschieden. Diese können auch je nach Kompressions-Wirkungsgrad mit verschiedenen Verfahren codiert werden.

Audio-Ströme, die keine Objekte enthalten, werden standardgemäß per AAC oder TwinVQ (Transform-Domain Weighted Interleaved Vector Quantisation) komprimiert. AAC bei MPEG-4 stellt grundsätzlich dieselben Funktionen wie bei MPEG-2 zur Verfügung, wurde aber um einige Funktionalitäten erweitert. Dazu gehört als Beispiel eine Verbesserung der Rauschunterdrückung (Perceptual Noise Substitution, PNS). Gewicht legten die Entwickler von MPEG-4 auf die Codierung von Sprachsignalen bis 2 kbps mit Prozeduren wie CELP (Code Excited Linear Prediction) oder HVXC (Harmonic Vector Excitation Coding). CELP z.B. konfrontiert ein ankommendes Audio-Signal mit einem bestehenden Wörterbuch und gibt nur einen Code weiter, wenn Aspekte eines Signals mit einem Eintrag übereinstimmen. Das Verfahren HILN (Harmonic and Individual Line plus Noise) prüft ein Signal auf sinusähnliche und Rauschanteile und übergibt die entsprechenden Parameter.⁴²

4.7. MPEG-4-Systems

MPEG-4 baut auf dem Systems-Bereich von MPEG-2 auf. Auch aufgrund der Echtzeit-Verarbeitung (z.B. Streaming) ist eine Unterteilung in Pakete mit Synchron-Angaben vonnöten. Darüber hinaus muss wegen der Objekt-Hierarchie die Datenfluss-Struktur aufwändiger sein. Es kann nicht mehr von AV-Datenströmen gesprochen werden, da MPEG-4-Objekte in den Elementarströmen oder Kanälen separiert werden. Das Prinzip ist mit „Create once, access everywhere“ – „Einmal erstellen, überall zugreifen“ definierbar. Nach der

⁴² vgl. http://www.tnt.uni-hannover.de/project/mpeg/audio/documents/w1903_over.html und <http://www.tnt.uni-hannover.de/project/mpeg/audio/documents/w2803.html>

Erstellung gelangen die Objekte in Puffer, was sowohl Variabilität als auch alternierendes Aussehen der Datenströme garantiert. Unumgänglich ist, dass alle erzeugten Object Descriptors mitgeliefert werden, damit der Decoder alle Bezüge wiederherstellen kann.

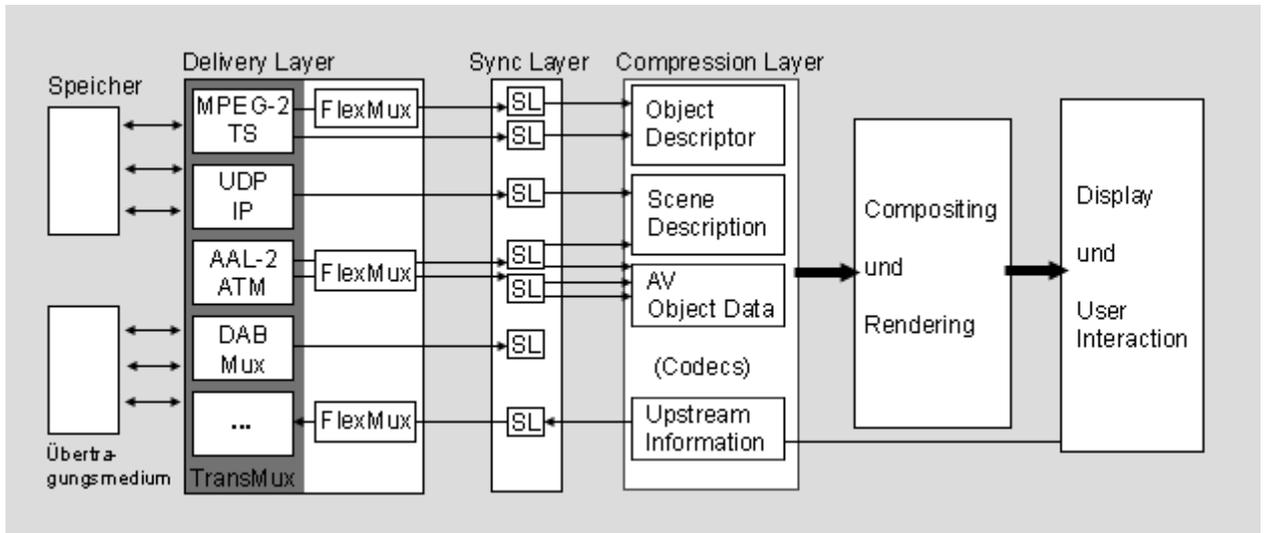


Abbildung 11: MPEG-4-Systems Architektur

Quelle: Heyna/Briede/Schmidt 2003, S.131

Abbildung 11 zeigt den Aufbau von MPEG-4-Systemen und den Weg vom Ankommen der Datenströme bis zur Ausgabe – also die Demultiplexierung. Zu erwähnen ist, dass nicht nur Einweg-Kanäle vorkommen, sondern auch ein Rückkanal für den Benutzer vorhanden ist. Die einzelnen Ebenen werden im Folgenden näher beleuchtet.⁴³

4.7.1. TransMux

Die unterste Ebene bildet der Delivery Layer, der wiederum aus dem TransMux-Layer (kurz für Transport Multiplex) und dem im Bedarfsfall verwendeten FlexMux-Layer (Flexibler Multiplex) besteht. Der TransMux-Layer muss mit zahlreichen

⁴³ vgl. <http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm> und http://www.chiariglione.org/MPEG/tutorials/papers/icj-mpeg4-si/03-systems_overview_paper/3-systems_overview_paper.htm

Übertragungsprotokollen zurecht kommen, da viele Unternehmen ihre Interessen bzw. Produkte einbringen wollen. Beispiele wären neben den MPEG-2-Transportströmen das UDP (User Daten Protokoll) über IP (Internet Protokoll) und ATM AAL2 (Asynchroner Transfermodus Adaption Layer 2). Die TransMux-Ebene ist daher transport-agnostisch, d.h. das eingesetzte Verfahren ist nicht bekannt. Den Übergang zwischen dem Eingang von Daten und dem Sync Layer stellt das DMIF (Delivery Multimedia Integration Framework) dar. Für die Kommunikation ist das DMIF Application Interface (DAI) zuständig.

4.7.2. FlexMux

Bei FlexMux handelt es sich um ein von MPEG entwickeltes Verfahren zur Demultiplexierung bzw. Multiplexierung. Steht ein derartiges Verfahren bei Anwendungen bereits zur Verfügung, wird diese Ebene übersprungen und gleich der Sync-Layer anvisiert. FlexMux wird also nur bei Bedarf aufgerufen und arbeitet auf kapazitätsschonende Weise.

4.7.3. Sync-Layer

Am Sync-Layer (Synchronisationsebene) liegen die der Demultiplexierung unterzogenen Daten wieder als elementare Ströme vor. Hier werden alle zur Synchronisation notwendigen Informationen bestimmt, die auf Coder-Seite von Access Units repräsentiert werden. Mit Hilfe von Zeitstempeln wird dem Sync-Layer vermittelt, wann dieser Datenströme aus dem Decoder-Puffer dem Darstellungspuffer zu übergeben hat. Diese Zeitstempel heißen Object Time Base (OTB) oder Object Clock Reference (OCR). Eine andere Möglichkeit, eine Zeitbasis zu erhalten, ist die Verknüpfung mit den Abstraten der AV-Datenströme. Selten, z.B. bei Slide-Show-Präsentationen, orientiert sich der Sync-Layer an keiner Zeitbasis, sondern handelt die ankommenden Aufgaben einfach nacheinander ab.

4.7.4. Präsentation

Nach dem Sync-Layer gelangen die Elementary Streams zu den entsprechenden Coder-/Decoder-Paaren (Codecs). Es erfolgt die Auswertung der grundsätzlichen AV-Objekte, außerdem die der Objekt- und Szenenbeschreibungen. Von diesen muss zumindest jeweils eine vorliegen, damit der Prozess überhaupt gestartet wird. Die Auswertung geht in separaten Decodern vonstatten. Anschließend wird die gesamte Szene zusammengesetzt. Nach dem Rendering führt der Weg schließlich vom Zwischenspeicher zur aktuellen Präsentation. Letzteres wird nicht mehr von Systems übernommen, sondern dies muss eine externe Applikation bewältigen.⁴⁴

4.8. Objekt-Framework

Dieser Abschnitt ist für die Abwandlungen von MPEG-4 (Derivate) irrelevant, wird aber hier aufgrund des zukunftsweisenden Ansatzes ausführlich beschrieben. Die Erkennung von und Arbeit mit Objekten befindet sich in der Entwicklungsphase und wird noch in keiner Version verwendet. MPEG-4 stellt lediglich die Rahmenbedingungen zur Verfügung, also die Werkzeuge, mit denen Objekte von der Bewegungserkennung über den Transport im Datenstrom bis zur Darstellung erfasst werden können.

Das große Bestreben liegt also in der Zerlegung von multimedialen Inhalten in einzelne Objekte. Bei diesen Media Objects handelt es sich um Daten jeglicher audiovisueller Form, also Standbilder, Video- und Audioobjekte, Texte und Grafiken. Diese Inhalte sollen zukünftig vom User beliebig beeinflusst werden können, sodass weiterführend ein globaler Objektaustausch mit ganzen Datenbanken möglich wird.

Außerdem soll eine dynamische Steuerung von Datenströmen erreicht werden. Dies bedeutet, dass der Benutzer imstande ist, per User Event (z.B. Klicken auf

⁴⁴ Steinmetz 1999, S.172ff und http://www.chiariglione.org/MPEG/tutorials/papers/icj-mpeg4-si/03-systems_overview_paper/3-systems_overview_paper.htm

ein Objekt) eine Verzweigung auf andere Inhalte herbeizuführen. Darüber hinaus soll auch der Blickwinkel innerhalb einer Szene veränderbar werden, wodurch sich gleichzeitig die akustischen Signale abhängig vom Standort ändern.⁴⁵

4.8.1. *Media Objects*

Der Benutzer ist mit MPEG-4 nicht mehr nur Beobachter, sondern kann aktiv in das Geschehen eingreifen und es manipulieren. Natürlich hängen diese Handlungsmöglichkeiten von der Art der Erstellung – dem Authoring – ab.

Objekte sollen dadurch verschoben (Translation), in der Größe verändert (Skalierung) und gedreht (Rotation) werden können. Außerdem sollen Bildelemente nicht vom Original-Kontext zwingend abhängen, sondern global austauschbar sein.

Diese Objekte werden beim dritten Standard von MPEG begrifflich zusammengefasst und bilden die unterste Stufe einer Hierarchie. Zusätzlich ist es möglich, Avatare – künstlich erstellte Formen – als Media Objects anzusehen und somit sowohl zwei- als auch dreidimensionale Objekte auf Basis von Drahtgittermodellen (Meshes) zu identifizieren.

Die Codierung von natürlichen und synthetischen Objekten innerhalb einer Szene wird Synthetic Natural Hybrid Coding (SNHC) genannt. Für die Beschreibung eines Media Objects sind Object Descriptors zuständig, die die Form und das Verhältnis zur Szene festlegen.

Die zusammengesetzten Medien-Objekte (Compound Media Objects) bilden die übergeordnete Ebene. Dabei werden Art, Menge und die verschiedenen Zusammenhänge der Elemente gespeichert. Solche so genannten Kompositionen werden anschließend als Szene gruppiert. Die Datenübertragung in den Elementarströmen (Elementary Streams, ES) erfolgt bei den früheren MPEG-Versionen komprimierter. Bei MPEG-4 hingegen ist es Standard, dass einzelne Objekte in einem oder mehreren Elementarströmen – unabhängig von der

⁴⁵ vgl. <http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm>

Szenenbeschreibung – untergebracht werden. Es können auch nur Objektbeschreibungen in einem ES übertragen werden, damit dem Decoder die Möglichkeit gegeben wird, nur kleine Details herauszufiltern und zu verarbeiten. Dies ist aufgrund der unterschiedlichen Leistungsfähigkeit von Endgeräten nötig.⁴⁶

4.8.2. Strukturierung

Objekte sind in einer Hierarchie von mehreren Ebenen (Layer) eingebettet. Die wichtigste ist der Base Layer (Basisebene), der die grundlegenden Eigenschaften eines Objekts beinhaltet. Darüber hinaus existieren Enhancement Layers (Erweiterungsebenen), die übertragen werden je nachdem, ob sich dem Datenstrom Hürden wie z.B. Bandbreitenbeschränkungen in den Weg stellen. Dies soll garantieren, dass Daten – wenn auch mit geringerer Qualität – bei ihrem Bestimmungsort ankommen.

Nur auf den visuellen Bereich bezogen werden Media Objects auch Video Objects (VO) genannt. Inklusiv Audio spricht man von Audio-Visual Objects (AVO). Video-Objekte sind nicht die kleinste Einheit bei MPEG-4, sie basieren auf den Video Object Planes (VOP), bei denen es sich um die zeitlich diskretisierte Form handelt. Eine Ebene darüber befinden sich Group of VOPs (GOV), die mehrere VOPs zusammenfassen. Eine GOV beinhaltet Start Codes, also Punkte, die den Zugriff auf alle Objekte und deren Eigenschaften ermöglichen. Group of VOPs sind allerdings optional verfügbar, somit können VOPs auch direkt auf die nächste Stufe des Video Objekt Layers (VOL) gelangen. Alle Video-Objekte und die entsprechenden Layer ergeben schlussendlich eine gesamte Szene bzw. die Video Session (VS).⁴⁷

⁴⁶ vgl. Heyna/Briede/Schmidt 2003, S.79ff

⁴⁷ vgl. http://www.chiariglione.org/MPEG/tutorials/papers/icj-mpeg4-si/07-natural_video_paper/7-natural_video_paper.htm

4.8.3. Object Description

Bei der Objektbeschreibung dreht sich alles, wie die Bezeichnung bereits vermuten lässt, um die Darstellung der Eigenschaften eines Objekts. Dazu gehörige Object Descriptors (OD) enthalten alle entsprechenden Informationen zu dem Media Object, wobei ODs diese Beschreibungen im Bitstrom lokalisieren und übernehmen. Zu Hilfe kommen den ODs außerdem Subdescriptors, die auf Folgendes referenzieren:

- Individuelle Elementarströme
- Semantische Informationen eines Objekts
- Zugang zur Verwaltung von Multimedia-Inhalten

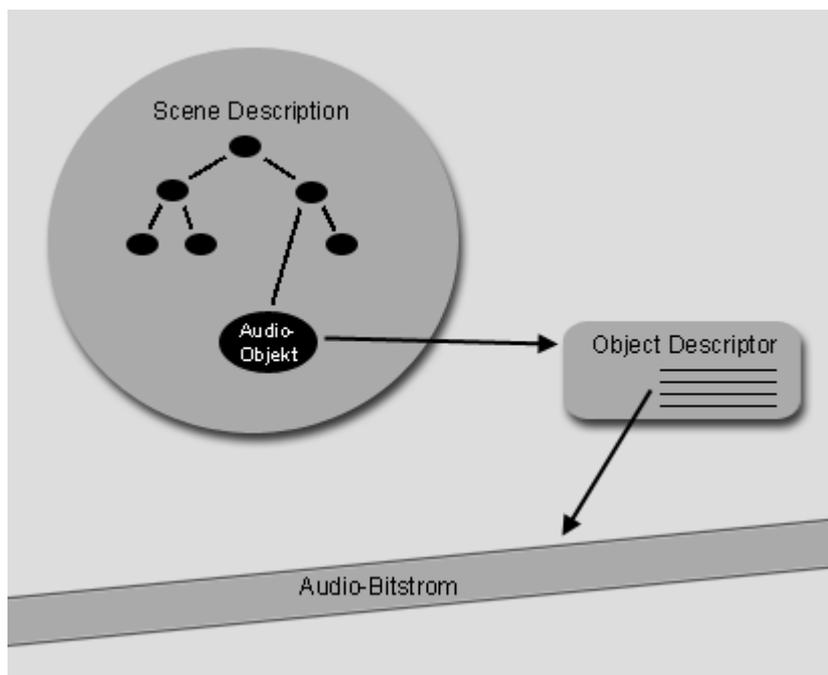


Abbildung 12: BIFS-Knotenpunkt referenziert mittels OD auf Audio-Bitstrom

Quelle: In Anlehnung an Pereira/Ebrahimi 2002, S.67

Abbildung 12 veranschaulicht die Aufgabe der Object Descriptors anhand eines Audio-Objekts. Die in Kapitel 4.8.4. beschriebene Scene Description, die jegliche Information über eine Szene enthält, beinhaltet auch ein Audio-Objekt. Welche Erscheinungsform dieses besitzt, weiß der OD, der wiederum die entsprechenden Daten im Audio-Bitstrom findet. Es können darüber hinaus mehrere Bitströme mit

einem Objekt verbunden werden. Die Syntax der ODs ist in einer gemeinsamen Klasse festgelegt, die Base Descriptor genannt wird. Darin ist unter anderem geregelt, dass jeder einzelne Object Descriptor mit einer Zahl versehen wird, um deren Einzigartigkeit zu garantieren.⁴⁸

Jedes Objekt lässt sich durch die Eigenschaften Form, Bewegung und Textur beschreiben. Bei MPEG-4 können diese separat codiert werden – und darin liegt wohl der größte Entwicklungssprung. Nicht mehr nur die Codierung rechteckiger Bilder, sondern die Erfassung willkürlicher Formen wird möglich. Allerdings fehlt es hier an konkreten Definitionen, wie Objekte voneinander separiert werden, dies wird an spezielle Anwendungen abgegeben. Adaptive Segmentierungen, die die Erkennung von Objekten automatisieren, befinden sich noch im Entwicklungsstadium. Beim Standard ist nur der Decodier-Prozess zu finden, die Einbindung von Encoder und Segmentierungssoftware wird dem freien Markt überlassen.

Formen werden bei der Objekt-Codierung als Bitmap angesehen, also eine Matrix aus binären Werten. Dies liegt in der Effizienz und im Rechenaufwand begründet. Weiters werden zwei Arten von beliebigen Formen (Shapes) unterschieden:

- Binäre Form (Binary Shape)
- Graustufenebene (Gray Scale, Alpha Shape)

Bei beiden Vorgangsweisen wird um das zu bestimmende Objekt eine rechteckige Maske gelegt, die um die Grenzen der Video Object Plane (VOP) verläuft und horizontal wie vertikal ein Vielfaches von 16 Pixeln groß sein muss. Diese Maske wird anschließend in Blöcke von 16 * 16 Pixeln (Binary Alpha Blocks) geteilt, die einzeln codiert werden.

Beim Binary Shape erfolgt eine Bewertung der Blöcke mit „on“ oder „off“ (sozusagen ja oder nein) – je nachdem, ob sie als zum Objekt gehörig befunden werden oder nicht. Das Content-Based Algorithm Encoding (CAE) untersucht

⁴⁸ vgl. Pereira/Ebrahimi 2002, S.66ff

dabei Ähnlichkeiten im örtlichen Nachbarbereich (IntraCAE). Alternativ dazu stellt das InterCAE Affinitäten bei einer sich zeitlich verändernden Form fest. Binary Shape bringt aber einen großen Nachteil mit sich – Alias-Effekte. Diese treten auf, wenn ein Objekt an seinen Rändern harte Kanten aufweist, es wirkt ausgefranst. Ein weiteres Problem kommt auf, wenn eine Form nicht vollständig vom Hintergrund gelöst wird, dann bleibt ein meist andersfarbiger Rand.

Abhilfe schafft der Einsatz von Alpha Shape. Bei diesem Verfahren erhalten die Blöcke einen Transparenzwert von 0 bis 255 – je höher die Zahl, desto transparenter. Daraus werden an den Übergängen weiche Kanten gewonnen, die auch im Interframe-Bereich nicht verloren gehen. Diese Vorgehensweise findet auch im Studiobereich seine Anwendung. Wo der Blue Screen bzw. Green Screen zum Einsatz kommt, werden mit Hilfe von Transparenzmasken Störeffekte z.B. rund um Nachrichtensprecher eliminiert.

Texturen unterwerfen sich bei der Codierung der hybriden DCT und werden getrennt von der Form verschlüsselt. Dabei wird mittels 8 * 8-Makroblöcken ein maximal enges Gitter geschaffen. Speziell bei statischen Texturen, die z.B. bei Polygon-Drahtgitter-Objekten (Meshes) angebracht werden können, wird auf die dafür besser geeignete diskrete Wavelet-Transformation zurückgegriffen.

Avatare spielten bei der Entwicklung von MPEG-4 eine große Rolle. Künstlich erstellte Repräsentanten werden vom Binary Format for Scenes (BIFS; Kapitel 4.8.4.) gesteuert und sollen folgendermaßen agieren: Beispielsweise wird ein Gesicht (Facial Animation Object) geschaffen und mit einem neutralen Ausdruck versehen. Emotionen und Reaktionen erhält der Avatar erst durch Parameter aus Datenströmen, die auf Decoderseite nur mehr richtig eingesetzt werden müssen. MPEG-4 bietet also auch die Möglichkeit, virtuelle Animationen zu erzeugen.⁴⁹

⁴⁹ vgl. Heyna/Briede/Schmidt 2003, S.84ff

4.8.4. Scene Description

Eine Ebene über der Codierung der einzelnen Objekte befindet sich die Szenenbeschreibung bzw. Scene Description, die Binary Format for Scenes genannt wird. BIFS basiert auf der Skriptsprache VRML (Virtual Reality Modeling Language), die für Interaktionsanwendungen im Internet verwendet wird.

Grundsätzlich setzt sich eine Szene aus mehreren Objekten zusammen. Beschreibende Elemente (Descriptors) wie z.B. Bewegungsvektoren oder Raumkoordinaten werden separat codiert und mit dem Datenstrom übertragen. Dem Benutzer bietet sich also die Möglichkeit, jeden einzelnen Descriptor per Bedienprogramm (Graphical User Interface, GUI) zu verändern. Szenenbeschreibungen können also dynamisch verändert, d.h. Elemente hinzugefügt, verschoben oder gelöscht werden.

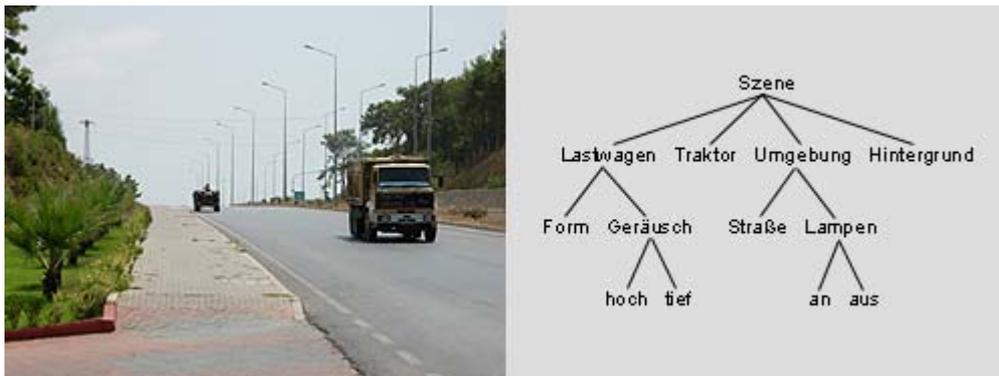


Abbildung 13: Beispiel für Szene aus mehreren Objekten

Quelle: In Anlehnung an Heyna/Briede/Schmidt 2003, S.81

Abbildung 13 stellt nur ein Beispiel für eine Anordnung von Objekten dar. Die einzelnen Elemente werden Nodes (Knoten) bezeichnet.⁵⁰

Ein Beispiel aus der großen Palette wäre der Shape Node, der Informationen über die Geometrie und Form eines Objekts enthält. Knoten bestehen wiederum aus verschiedenen Fields (Feldern). Wert, Art des Wertes (z.B. Integer), Art des Verhaltens und Name sind die Bestandteile, die die Knoten beschreiben. Essentiell für Verbindungen sind Routes (Routen), die mit Quell- und Zielfeldern

⁵⁰ vgl. Heyna/Briede/Schmidt 2003, S.83f

ausgestattet sind und Informationen transportieren. Ändert sich auf einer Seite ein Parameter wird auch auf der anderen Entsprechendes vollzogen.

Der Hauptszene untergeordnet können Subscenes eingebaut werden. Diese haben die Struktur einer normalen Szene, sind allerdings sozusagen eine Teilmenge davon. Z.B. kann die Hauptszene ein Haus sein, während die Subscene aus einem Raum besteht. Diese Aufteilung kann folgende Intentionen haben:

- Die komplette Szene ist zu groß um leicht gehandhabt zu werden.
- Teile einer Szene müssen öfters als andere modifiziert werden.
- Einzelne Szenen werden von verschiedenen Verfassern geschaffen.
- Die Subscene darf im Gegensatz zur Hauptszene nicht verändert werden oder umgekehrt.

Die Subscene erfährt mit Hilfe des Inline Knotens den Einbau in eine Szene. Eine andere Möglichkeit ist die des Hyperlinks. Wie bei einer HTML-Seite muss der Benutzer nur klicken, um die Verbindung zu einer neuen Szene bzw. Subscene herzustellen.⁵¹

Der zeitliche Gleichlauf der Scene Description mit allen Medien-Bitströmen benötigt eine Kontrolleinheit. Diese Aufgabe übernimmt der Synchronization Layer (oder Sync Layer). Zeit spielt dabei eine entscheidende Rolle und ist immer relativ. Als Referenz wird bei MPEG-4 jener Zeitpunkt (Time Stamp) herangezogen, zu dem ein BIFS-Befehl zur Verarbeitung bereitsteht.

⁵¹ vgl. Pereira/Ebrahimi 2002, S.104ff

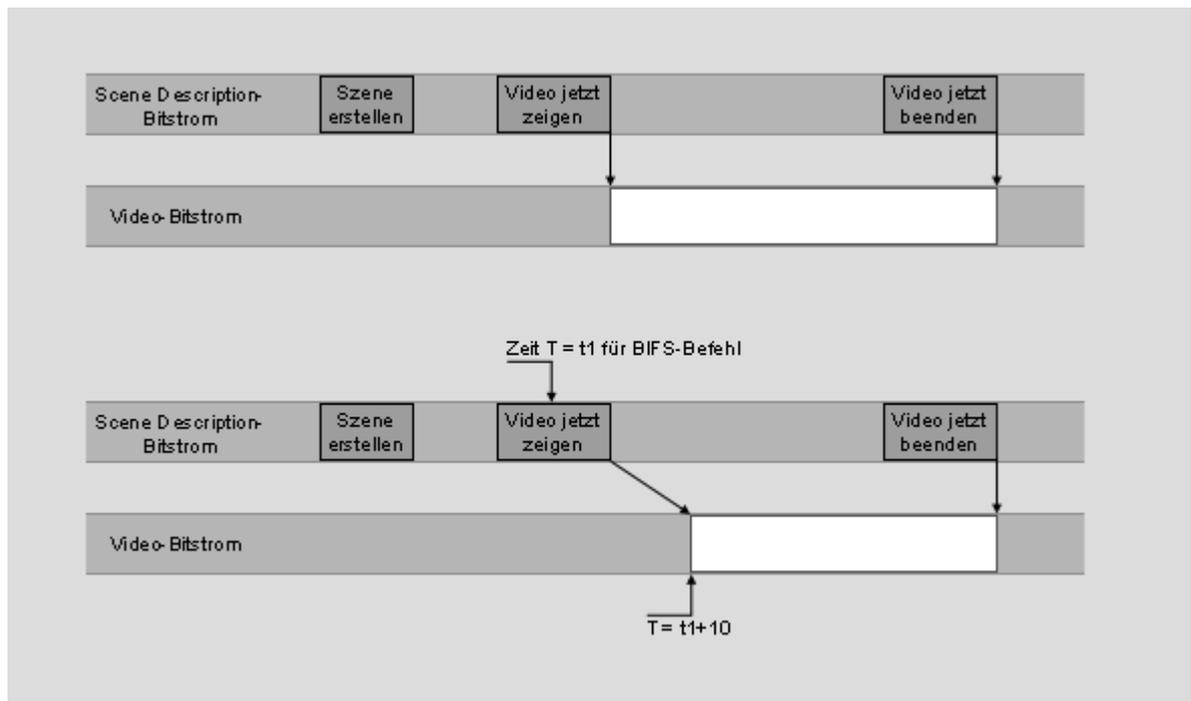


Abbildung 14: Zeit relativ zur internen Zeit eines BIFS-Befehls gesehen

Quelle: In Anlehnung an Pereira/Ebrahimi 2002, S.91/92

Time Stamps gewährleisten die Synchronisation von Daten aus verschiedenen Bitströmen. Dazu gehören auch Access Units (Zugangseinheiten), die kleine Datenpakete in den Elementarströmen sind und auf bestimmte Zeitpunkte referenzieren. Elementarströme bestehen dementsprechend aus einer Folge von Access Units. Es existieren zwei Arten von Time Stamps, die mit Access Units verbunden werden können:⁵²

- Decoding time (Decodierzeit): Zu diesem Zeitpunkt müssen alle Informationen des Time Stamps dem Empfänger zur Verfügung stehen und sollten sofort decodiert werden.
- Composition time (Kompositionszeit): Die decodierte Access Unit wird der darauf folgenden Verarbeitung (Komposition und Präsentation) bereitgestellt.

⁵² vgl. Pereira/Ebrahimi 2002, S.90ff

4.8.5. Bewegungsschätzung und -kompensation

Eine Vorhersage von Bewegungen geht auch bei MPEG-1 und MPEG-2 vonstatten, nur ist bei MPEG-4 der Unterschied, dass der Schätzung Objekte bzw. Video Object Planes (VOP) zugrunde liegen. Dies erfolgt mit Hilfe der Shape-Adaptive DCT (SA-DCT). Wie bei der DPCM kommen auch hier IntraVOPs (I-VOP), unidirektionale (P-VOP) und bidirektionale VOPs (B-VOP) zum Einsatz, die je nachdem nur das Objekt in einem Bild codieren oder Nachbarbilder als Schätzung zur Hilfe nehmen. Zur Erkennung der Formen wird ein Gitter von Makroblöcken herangezogen, wobei jeder Block Luminanz- (16 * 16 Pixel) und Chrominanzblöcke (8 * 8 Pixel) beinhaltet. Das Blockmatching-Verfahren bestimmt bei MPEG-4 Version 2 Ähnlichkeiten zwischen Frames auf ein Viertel-Pixel Genauigkeit und vergibt für den ganzen Makroblock einen Bewegungsvektor. Zusätzlich erhalten alle unterteilten Blöcke jeweils einen Vektor. Wenn die Formcodierung nicht vonnöten ist und Objekte von einer rechteckigen Form umschlossen werden, bedient sich MPEG-4 der von früheren Versionen angewandten Standard-Block-DCT.⁵³

4.9. Vergleich zu MPEG-2

MPEG-4 basiert im Grunde auf MPEG-2, kann allerdings mit wesentlichen Neuerungen aufwarten. Die wichtigste davon ist, dass bei MPEG-4 Rahmenbedingungen für den Zugriff auf einzelne Objekte (Media Objects) innerhalb eines Bildes geschaffen wurden, während bei MPEG-2 nur ganze Frames betrachtet werden können. Darüber hinaus spielt zukünftig Interaktivität eine tragende Rolle, welche beim Vorgänger keine Beachtung findet.

Die bereits beschriebenen Techniken und Anwendungen werden nachfolgend kurz zusammengefasst. Während MPEG-2 im Zuge der Codierung die hybride DCT zur

⁵³ vgl. <http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm> und http://www.chiariglione.org/MPEG/tutorials/papers/icj-mpeg4-si/07-natural_video_paper/7-natural_video_paper.htm

Verfügung steht, erweitert MPEG-4 sein Sortiment mit der Wavelet-Transformation bzw. der Integertransformation.

Eine wichtige Aufgabe beim neuesten MPEG-Standard war die Verringerung der Bitrate, um z.B. auch im Bereich der Mobilfunknetze aktiv zu werden. So ist eine Übertragungsrate bis zu 5 kbit/s möglich. Beim Vorgänger vermindert sich mit abnehmender Bitrate auch die Videoqualität drastisch.

Auch bei den Bildauflösungen erweitert MPEG-4 seine Spanne. Die niedrigste liegt bei 128 * 96 Pixeln, während MPEG-2 eine Auflösung von 352 * 288 bis 1920 * 1152 Pixeln (HDTV) erlaubt. Der Nachfolger ermöglicht eine Größe von 4k * 4k für den professionellen Schnittbetrieb.

Die Anwendbarkeit von Profilen ist beim älteren Standard auf fünf bei jeweils vier Levels (Auflösungen) beschränkt, außerdem sind diese nur für den Videobereich ausgelegt. MPEG-4 bietet dagegen über 30 Profile, die sowohl für Video, als auch für Audio, 2D- und 3D-Grafik ausgelegt sind.

Tabelle 5 gibt einen Überblick der Unterschiede in verschiedenen Bereichen.

	MPEG-2	MPEG-4
Standardisiert seit:	November 1994	Januar 1999
Typische Anwendungsgebiete:	Multimedia-CD-ROM, DVD, VCD, S-VCD, Digitales Kabel- und Satelliten-TV (DVB)	Streaming Video (Video-Konferenzen, Bildtelefonie, mobile A/V-Kommunikation wie UMTS u.a., Broadcasting-Anwendungen), Internet Multimedia, interaktive Videospiele
Vorteile:	Mehrkanal-Audio, bei höheren Bitraten gute Videoqualität, weite Verbreitung und hohe Kompatibilität mit gängiger Video- und PC-Hardware	extrem hohe Kompression, Skalierbarkeit; Flexibilität in Bezug auf Bildformate, Auflösung und Tonsuren; bietet vielfältige Voraussetzungen für zukünftige interaktive A/V-Anwendungen
Nachteile:	mäßige Codier-Effizienz, in geringen Bitraten schlechte Videoqualität	vergleichsweise rechenintensiv; Vielfalt an unterschiedlichen technischen Umsetzungen und inkompatible Codecs verzögern den industriellen Einsatz
Typische Profile für Filme:	Simple Profile (I-Frames & P-Frames) / Main Profile (zusätzlich B-Frames)	Simple Profile (I-VOPs & P-VOPs) / Advanced Simple Profile (zusätzlich B-VOPs, Global Motion Compensation, Viertel-Pixel)
Bildauflösung:	352x288 bis 1.920x1.152 (HDTV)	bislang implementiert bis 1.920x1.088 (DivX 5)
Bitraten:	typisch: 500 kbit/s bis 10 Mbit/s, erweiterbar bis 80 Mbit/s	typisch: 20 kbit/s bis 4 Mbit/s, erweiterbar bis über 1 Gbit/s
Kompressions-Techniken:	DCT, Motion Estimation, Quantisierung, GOPs, CBR-/VBR-/Multipass-Encoding	DCT, Motion Estimation, Quantisierung, adaptive VOP-Codierung, Wavelet-Kompression, CBR-/VBR-/Multipass-Encoding
Typisches Format für Filme:	DVD	DivX 5.03 Home Theater Profil
Zugehörige Bildauflösung:	720x576 (PAL)	720x576 (PAL)
Zugehörige Videodatenrate:	bis 10 Mbit/s	bis 9 Mbit/s
Audio-Formate:	MPEG1 Layer 2 (MP2), AC3 bis 768 kbit/s bei 48 kHz Sample-Rate	nicht spezifiziert, üblicherweise MPEG1 Layer 3 (MP3) zwischen 96 kbit/s und 160 kbit/s, auch OggVorbis und AAC
Mehrkanal-Audio:	bis zu acht Spuren (Dolby Digital AC3 / DTS)	bis zu sechs Spuren (AAC, AACplus, Dolby Digital AC3, OggVorbis)
Durchschnittliche Spielzeit pro Medium:	bis zu 120 Minuten pro Layer (4,7 GByte)	zwischen 60 und 130 Minuten pro CD-R (700 MB)
Unterstützte Hardware:	PC / DVD-Player	PC / nur wenige DVD-Player / modifizierte Xbox

Tabelle 5: Chip Online Archiv: Standard-Vergleich MPEG2 versus MPEG4

Quelle: http://archiv.chip.de/artikel/c1_archiv_artikelunterseite_17143025.html?tid1=33105&tid2=0

5. Derivate von MPEG-4

Ein Derivat, aus dem lateinischen „derivare“ – „ableiten“, bezeichnet eine Abwandlung eines Produkts. Ähnlichkeiten sind in der Regel gegeben, jedoch spricht man erst bei „genügend“ Unterschieden von einem Derivat.⁵⁴ Wie auch MPEG-4 erfüllen Derivate den Sinn, dass AV-Material um ein Vielfaches komprimiert wird. Besonders bei Filmen erfreuen sich diese großer Beliebtheit, da Daten in DVD- auf CD-Größe (700 MB) verkleinert werden können.⁵⁵

Den nachstehenden Formaten liegt allesamt (zumindest) MPEG-4 Part 2 zugrunde.

5.1. DivX

DivX steht für Digital Video Extreme und ist ein Produkt der Firma DivX Inc. (vormals DivXNetworks Inc.). Erste Versionen von DivX basierten auf Microsofts MPEG-4-Codec, der von einem Franzosen namens Jérôme Rota gehackt wurde. Genauer gesagt, entschlüsselte er Microsofts MP43C32.dll, aus der der MPEG-4-Codec Version 3 für Windows entstehen sollte. Außerdem wollte der Konzern den Codec in das Advanced Streaming Format (.asf) packen. Rota patchte gemeinsam mit einem deutschen Hacker den Code und brachte das Produkt DivX;-) 3.11 Alpha für AVI-Dateien heraus. Aus Patentgründen kam es später zur Entwicklung eines völlig neuen, autonomen Codecs, dem Nachfolger DivX 4. Diesen entwickelten DivX Inc. zusammen mit dem Institut für theoretische Nachrichtentechnik und Informationsverarbeitung (TNT) der Universität Hannover und dem Heinrich-Hertz-Institut (HHI) Berlin. Aktuell ist DivX 5 verfügbar, das seinen Vorgängern nicht zugrunde liegt. DivX ist Open Source, steht also als Decoder (nicht als Encoder) frei zur Verfügung. Außerdem unterstützt der Codec nur das MPEG-4-Simple Profile vollständig. Das Advanced Simple Profile verwendet das kostenpflichtige DivX Pro.⁵⁶

⁵⁴ vgl. http://de.wikipedia.org/wiki/Derivat_%28Software%29

⁵⁵ vgl. <http://de.wikipedia.org/wiki/Divx>

⁵⁶ vgl. <http://www.heise.de/ct/02/18/094/default.shtml>

5.2. XviD

Das MPEG-4-Derivat XviD baute anfangs auf dem OpenDivX-Quellcode auf, der wiederum aus der MPEG-4 Referenzimplementierung des EU-Projekts MoMuSys. Als OpenDivX jedoch geschlossen wurde, arbeitete eine Gruppe von Freiwilligen an der Neuentwicklung des Open Source-Codecs. Mit dem Basiswissen aus OpenDivX wurde XviD in vielen Bereichen verändert und optimiert. Die Bezeichnung (XviD ist rückwärts DivX) ist eine Anspielung auf den mittlerweile kommerziell gewordenen Kontrahenten. Die neueste Version unterstützt das Advanced Profile von MPEG-4 und arbeitet somit mit Tools wie B-Frames, Quarter Pixel und Global Motion Compensation. Der nächste Schritt ist der Übergang zu XviD AVC, das auf dem Advanced Video Coding basiert.⁵⁷

5.3. HDX4

Bei HDX4 handelt es sich um einen MPEG-4-Codec der deutschen Firma Jomigo. Bei Tests sticht der Codec vor allem durch seine Geschwindigkeit heraus, allerdings wird dies mit einer niedrigeren Codiereffizienz bezahlt. Wie XviD beinhaltet auch HDX4 das Advanced Simple Profile. Eingesetzt wird das kostenpflichtige Produkt eher im kommerziellen Bereich. Lufthansa oder Fraunhofer IGD, die damit Videokonferenzen realisieren, vertrauen auf HDX4.⁵⁸

5.4. 3ivx

Ein weiterer MPEG-4-Codec ist 3ivx von der australischen Firma 3ivx Technologies. Das Erzeugnis ist vor allem für eingebettete Systeme, d.h. hardwarenahe (Alltags-)Fabrikate, mit geringem Rechenaufwand gedacht.⁵⁹ Bemerkenswert ist die Entwicklung eigener Filter, mit denen im Software-Paket

⁵⁷ vgl. <http://de.wikipedia.org/wiki/Xvid>

⁵⁸ vgl. <http://de.wikipedia.org/wiki/HDX4>

⁵⁹ vgl. <http://de.wikipedia.org/wiki/3ivx>

codiert und decodiert, aber auch Elemente extrahiert oder geteilt werden können. Mit dem 3ivx Media Muxer ist es möglich, MP4-Dateien zu erzeugen.⁶⁰

5.5. Nero Digital

Nero Digital ist eine Ko-Produktion von Nero AG (Deutschland) und Atome (Frankreich). Das Produkt vereint jeweils zwei Video- und Audiocodecs:

- MPEG-4 Part 2 ASP (Advanced Simple Profile)
- MPEG-4 Part 10 AVC

- MPEG-4 Part 3 LC-AAC
- MPEG-4 Part 3 HE-AAC

Die Codier-Software Nero Recode packt die komprimierten Daten in MP4-Container.⁶¹

⁶⁰ vgl. <http://www.3ivx.com/technology/filters/index.html>

⁶¹ vgl. http://de.wikipedia.org/wiki/Nero_Digital

6. Empirische Studie

6.1. Vorgehensweise

Die empirische Studie soll der Frage nachgehen, welches von den fünf MPEG-4-Derivaten am leistungsfähigsten ist. Dafür wird der Film „From Dusk Till Dawn“ (aus dem Jahr 1996) herangezogen. Mit Hilfe von geeigneter Codier-Software und möglichst gleichen Parametern erfährt der Film eine Komprimierung durch DivX, XviD, HDX4, 3ivx und Nero Digital. Im Anschluss werden die Endprodukte mittels Programmen analysiert und bewertet.

	Länge	Frames	Fernsehnorm
„From Dusk Till Dawn“	1'43"33	155325	PAL

Tabelle 6: Eckpfeiler des Films

Quelle: eigene Darstellung

Der Film wird nach folgenden Kriterien codiert:

	Video-Bitrate		Audio-Bitrate	Zielgröße
„From Dusk Till Dawn“	DivX	783 kbit/s	128 kbit/s	700 MB (1 CD-R)
	XviD	788 kbit/s	MP3, Variable	
	HDX4	798 kbit/s	Bitrate	
	3ivx	880 kbit/s		
	Nero Digital	859 kbit/s	64 kbit/s AAC (HE)	692 MB (interne Vorgabe)

Tabelle 7: Grundangaben zur Codierung

Quelle: eigene Darstellung

Als Zielgröße wird das vorhandene Volumen auf einer üblichen CD-R gewählt, also 700 Megabyte. Den Unterschieden bei den Video-Bitraten liegen software-spezifische Einstellungen zugrunde. Zu DivX wird z.B. im Gegensatz zu XviD ein Frame-Overhead gerechnet. Dabei handelt es sich um geringe Datenmengen, die

in regelmäßigen Intervallen angefügt werden, um zusätzliche Informationen zu übertragen.

Das Video-Material durchläuft die Komprimierung jeweils ein Mal (One Pass). Als Bildgröße wird jedes Mal das Format 640 * 512 Pixel gewählt.

6.2. Hard- und Software

Der Codec-Vergleich wird auf nachstehendem System vollzogen:

- AMD Athlon 64 3000+
- ATI Mobility Radeon X700 PCI Express
- 512 MB DDR RAM

Folgende Programme kommen während des Codierens zum Einsatz:

Für DivX und XviD:

- DGIndex 1.0.12 und DGDecode
(Die Tools decodieren einen MPEG-2-Strom, z.B. von einer DVD-Quelle, und erstellen eine Projektdatei mit Index-Verweisen auf den Datenstrom, damit der Encoder Referenzpunkte erhält)
- AviSynth 2.55
(für Frameserving; AviSynth erstellt ein Skript mit Hilfe der DGIndex-Projektdatei, das gewünschte Filter, z.B. Resize, enthält, um das Codierprogramm damit speisen zu können)
- VirtualDub 1.6.14
(für das Codieren; erhält die Informationen vom AviSynth-Skript)

Für HDX4:

- HDX4 Movie Creator 1.5.7.512

Für 3ivx:

- AviSynth 2.55
- VirtualDub 1.6.14

Für Nero Digital:

- Nero Recode 2 CE

Diese Codec-Versionen werden verwendet:

- DivX 5, 6.2.2.3 (Erschienen am 01.06.06)
- XviD 1.1.0 (Erschienen am 30.12.05)
- HDX4 1.5.4.419 (Erschienen am 19.04.06)
- 3ivx D4 4.5.1 (Erschienen am 26.01.04)
- Nero Digital (Erschienen am 26.11.05)

6.3. Codec-Implementierung

Um vergleichbare Resultate zu schaffen, müssen bei den einzelnen Codecs bestimmte Einstellungen vorgenommen werden. Es wird dabei darauf geachtet, eine ausgewogene Qualität bei daraus resultierender durchschnittlicher Geschwindigkeit zu erreichen. Beispielsweise werden B-Frames eingesetzt, jedoch keine Suche auf ein Viertel-Pixel genau. Die nachstehenden Punkte präsentieren die wichtigsten Einstellungen, wenn auch aufgrund von Einschränkungen je nach Codec keine vollständige Gleichstellung geschaffen werden kann.

- DivX
 - Profil: Home Theater Profile
 - Ausgewogene Qualität
 - Einzelne B-Frames
 - Quantisierung: H.263 optimiert
(Dieser Quantisierungstyp liefert ein leicht geglättetes Bild und verhindert bei niedrigen Bitraten weitgehend Blockartefakte)
 - Psychovisuelle Verbesserungen: Formen
(Die Option macht sich die visuelle Wahrnehmung des Betrachters zunutze, die dazu neigt, bestimmte Teile eines Bildes stärker zu erfassen)

- XviD
 - Profil: unbeschränkt
 - Bewegungssuche: durchschnittlich
 - MPEG-Quantisierung
(Die MPEG-Quantisierung erzeugt im Gegensatz zu H.263 bei höheren Bitraten ein schärferes Bild und behält somit mehr Details)
 - Trellis-Quantisierung
(Diese alternative Art ermöglicht eine bessere Komprimierbarkeit und somit ein hochwertigeres Bild bei gleicher Datenrate. In diesem Fall wird das Ergebnis der MPEG-Matrix etwas glatter und sauberer)
 - Einzelne B-Frames
 - Turbo (Schnellere Bewegungsschätzung bei B-Frames)

- HDX4
 - Mittlere Qualität
 - MPEG-Quantisierung
 - Einzelne B-Frames
 - Psychovisuelle Verbesserung: Ausgewogen

- 3ivx
 - Mittlere Qualität
 - MPEG-Quantisierung

- Nero Digital
 - Profil: Standard-AVC
 - Qualität: Normal
 - Einzelne B-Frames
 - Psychovisuelle Erweiterungen: Niedrig

6.4. Qualitätsbestimmung

Zur objektiven Qualitätsfeststellung kommen die beiden Algorithmen PSNR und SSIM zur Anwendung. Als Software wird dafür Video Quality Studio 0.4 RC3 verwendet, das beide Tools zur Verfügung stellt. Getestet werden zwei Ausschnitte aus dem Film, die jeweils eine Länge von 2000 Frames aufweisen und anspruchsvoll zu verarbeitende Inhalte bieten:

- 1. Ausschnitt (Frame 12000 – 14000)



Abbildung 15: Bilderfolge aus dem 1. Ausschnitt

Quelle: eigene Darstellung

Das zweite Frame aus Abbildung 15 zeigt eine Szene mit Feuer, das aufgrund seiner schnellen Bewegungen die Codecs vor schwierige Aufgaben stellt. Die optische Analyse in Kapitel 7.1.3.2. behandelt diese Szene ausführlich.

- 2. Ausschnitt (Frame 119000 – 121000)



Abbildung 16: Bilderfolge aus dem 2. Ausschnitt

Quelle: eigene Darstellung

Der zweite Ausschnitt ist von raschen (Kamera-)Bewegungen geprägt und soll die Leistungsfähigkeit der Codecs offen legen.

- PSNR

PSNR (Peak signal-to-noise ratio; Signal-Rausch-Verhältnis) ermittelt den Unterschied zwischen dem Original und den Störungen im codierten Material. Die mathematische Rechnung gibt das Ergebnis in Dezibel (dB) aus.

Das Signal-Rausch-Verhältnis definiert sich über die aus der mathematischen Statistik stammenden mittleren quadratischen Abweichung (Mean Squared Error; MSE), deren Formel lautet:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \| I(i, j) - K(i, j) \|^2$$

Die Variablen repräsentieren m*n-große Bilder I und K, die jeweils Helligkeitswerte beinhalten. Eines von beiden stellt eine komprimierte Annäherung an das andere dar.

Daraus kann die Berechnung für PSNR erstellt werden:

$$PSNR = 10 * \log_{10} \left(\frac{MAX_I^2}{MSE} \right) = 20 * \log_{10} \left(\frac{MAX_I}{\sqrt{MSE}} \right)$$

MAX_I steht für den maximalen Pixelwert des Bildes. Wenn die Werte z.B. mit 8 Bit dargestellt werden, beträgt MAX_I 255.

Bei Farbbildern mit drei RGB-Werten pro Pixel wird dieselbe PSNR-Formel angewandt – mit dem Unterschied, dass sich MSE aus der Summe aller Abweichungen, geteilt durch die Bildgröße und durch drei ergibt.⁶²

Um einen Bezug zu den Ergebniswerten herstellen zu können, wird PSNR mit dem MOS (Mean Opinion Score) gleichgesetzt. MOS-Bewertungen geben wörtlich übersetzt die „durchschnittliche Meinung“ von menschlichen Betrachtern wieder und ergeben sich folglich aus subjektiven Eindrücken. Folgende Tabelle zeigt die Empfehlung P.800.1 der ITU (International Telecommunication Union) ergänzt mit den zugehörigen PSNR-Werten:

Wert	MOS-Bezeichnung		Vergleich zum Original	PSNR (in dB)
5	Excellent	Exzellent	Keine wahrnehmbare Verzerrung	≥ 37
4	Good	Gut	Wahrnehmbar, aber nicht störend	[31 ; 37)
3	Fair	Mittel	Leicht störende Verzerrungen	[25 ; 31)
2	Poor	Gering	Störende Verzerrungen	[20 ; 25)
1	Bad	Schlecht	Kaum erkennbar, stark störend	< 20

Tabelle 8: MOS in Beziehung zu PSNR

Quelle: In Anlehnung an <http://www.tfh-berlin.de/~mixdorff/dvt/Kenngroessen.pdf>

- SSIM

SSIM (Structural SIMilarity) dagegen erkennt nicht die Fehler, sondern sucht nach Ähnlichkeiten zwischen originalem und komprimiertem Clip. Dieser Ansatz wird wie folgt umschrieben:

“Die Hauptfunktion der menschlichen visuellen Wahrnehmung ist, dem Blickfeld strukturelle Informationen zu entnehmen. Darum kann die Messung des Verlustes an strukturellen Informationen eine gute Annäherung der bemerkbaren Bildstörungen sein.“

⁶² vgl. <http://en.wikipedia.org/wiki/PSNR>

Struktur-Daten sind jene, die den Aufbau von Objekten in Szenen repräsentieren.⁶³

Die Ermittlung des SSIM-Index geht von einem Pixel und seiner gewichteten Umgebung aus, die 11 * 11 Pixel groß ist. Dies bedeutet, dass der Index vorrangig vom Pixel abhängig ist und absteigend von den Pixeln in diesem Betrachtungsfenster.

Der erste Teilvergleich betrifft die Helligkeiten zwischen den Clips, die gespeichert und anschließend gleichgesetzt werden. Es folgen ein Kontrastvergleich und wieder eine Anpassung. Im letzten Vergleich wird die Struktur der Objekte, die von Helligkeit und Kontrast nicht abhängt, gegenübergestellt.⁶⁴

Das Ergebnis wird per YUV-Farbmodell präsentiert. Diese Komponenten erfahren eine Gewichtung, um zum SSIM-Index zu gelangen, die folgendes Aussehen aufweist:

$$\text{SSIM} = 0,8 * \text{SSIM}_Y + 0,1 * \text{SSIM}_U + 0,1 * \text{SSIM}_V$$

Diese Berechnung ist notwendig, da das menschliche Auge auf Helligkeitsinformationen (Y-Komponente) besonders anspricht und weniger auf Farbwerte.

⁶³ vgl. <http://www.cns.nyu.edu/~zwang/files/research.html>

⁶⁴ vgl. <http://www.cns.nyu.edu/pub/eero/wang03-reprint.pdf>

7. Ergebnisse der Studie

7.1. Darstellung und Interpretation

Das erste Bewertungskriterium ist die Codier-Geschwindigkeit der Derivate. Tabelle 9 zeigt die gerundeten, durchschnittlichen Geschwindigkeiten. Die Abkürzung „fps“ bedeutet „frames per second“ und bezeichnet die Anzahl der verarbeiteten Bilder in einer Sekunde.

Codier-Geschwindigkeit	DivX	XviD	HDX4	3ivx	Nero Digital
„From Dusk Till Dawn“	16 fps	16,6 fps	21,5 fps	13,8 fps	8,6 fps

Tabelle 9: Codier-Geschwindigkeiten der Codecs

Quelle: eigene Darstellung

HDX4 geht als schnellster Codec des Tests hervor, während sich DivX und XviD etwa gleichauf von den übrigen distanzieren können. Nero Digital erreicht einen Mittelwert von nur 8,6 fps. Diese Wertung bedeutet jedoch nicht zwingend, dass schnelle Codecs qualitativ nicht mithalten können und umgekehrt.

Ein weiterer Prüfstein ist, ob die Derivate die vorgegebenen Dateigrößen einhalten können. Bei Übertretung der Größe ist es möglich, dass die Datei auf dem gewählten Speichermedium CD-R nicht mehr Platz hat. Aber auch eine zu kleine Datei spricht nicht für den Codec, da beim Codieren noch Raum für AV-Daten übrig gewesen wäre.

Größe	DivX	XviD	HDX4	3ivx	Nero Digital
„From Dusk Till Dawn“	718.060 KB = 701,2 MB	720.938 KB = 704 MB	718.464 KB = 701,6 MB	711.054 KB = 694,4 MB	711.992 KB = 695,3 MB

Tabelle 10: Dateigrößen der Codecs

Quelle: eigene Darstellung

DivX und HDX4 sind der Vorgabe am nächsten und überziehen 700 MB im akzeptablen Rahmen. XviD, 3ivx und Nero Digital (Vorgabe 692 MB) verfehlen das Ziel relativ klar.

7.1.1. PSNR

Als objektives Qualitätsmerkmal werden die Derivate mit geeigneter Software dem PSNR-Test unterzogen. Dabei wird jedes einzelne codierte Frame mit dem Original verglichen und dessen Unterschiede festgestellt.

7.1.1.1. DivX

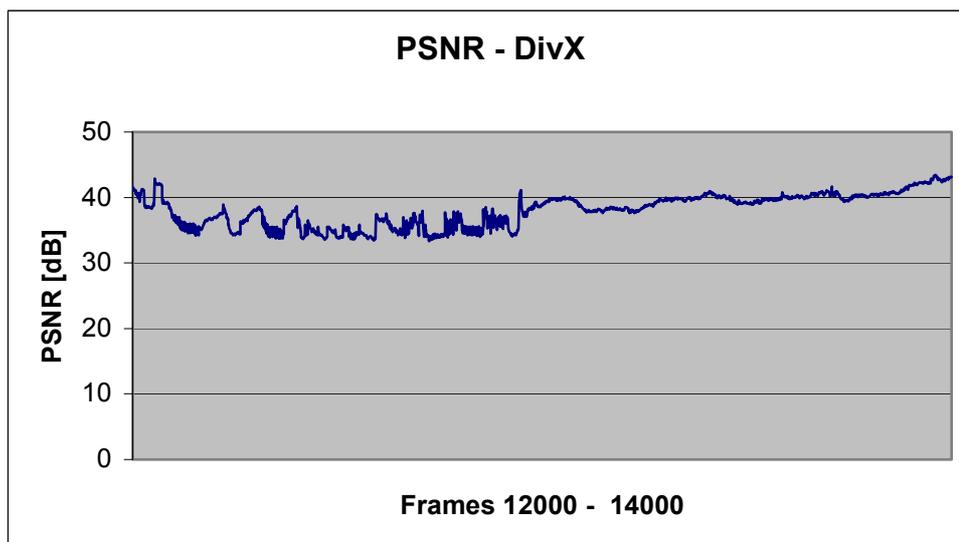


Abbildung 17: PSNR des 1. Ausschnittes bei DivX

Quelle: eigene Darstellung

Der PSNR-Durchschnitt aus den Frames 12000 bis 14000 beträgt laut Abbildung 17 gerundete 38 dB und liegt damit nach dem Mean Opinion Score aus Tabelle 8 knapp im höchsten Bereich mit der Bewertung „Exzellent“. Die größere Variation bei den ersten etwa 1000 Frames liegt wie bei den anderen Testergebnissen in den Szeneinhalten begründet. Es erfolgen hier rasche Kameranews,

Bewegungen und Bildwechsel. In der zweiten Hälfte des Abschnittes kommen offensichtlich ruhigere Szenen vor.

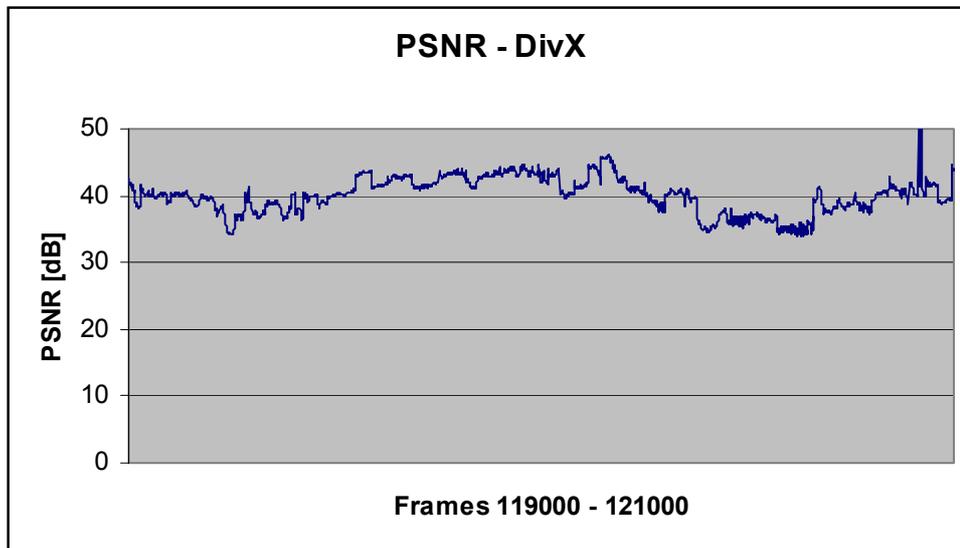


Abbildung 18: PSNR des 2. Ausschnittes bei DivX

Quelle: eigene Darstellung

Der PSNR-Durchschnitt macht beim zweiten Ausschnitt rund 40 Dezibel aus. Die unruhigere Linie in Abbildung 18 bedeutet viel Bewegung im gesamten betrachteten Bereich. Der Sprung an die 50-dB-Grenze nahe dem Abschnittsende ist mit einheitlich gefärbten Frames zu erklären. Diese Farbfläche erfährt bei der Codierung nahezu keinen optischen Verlust.

7.1.1.2. XviD

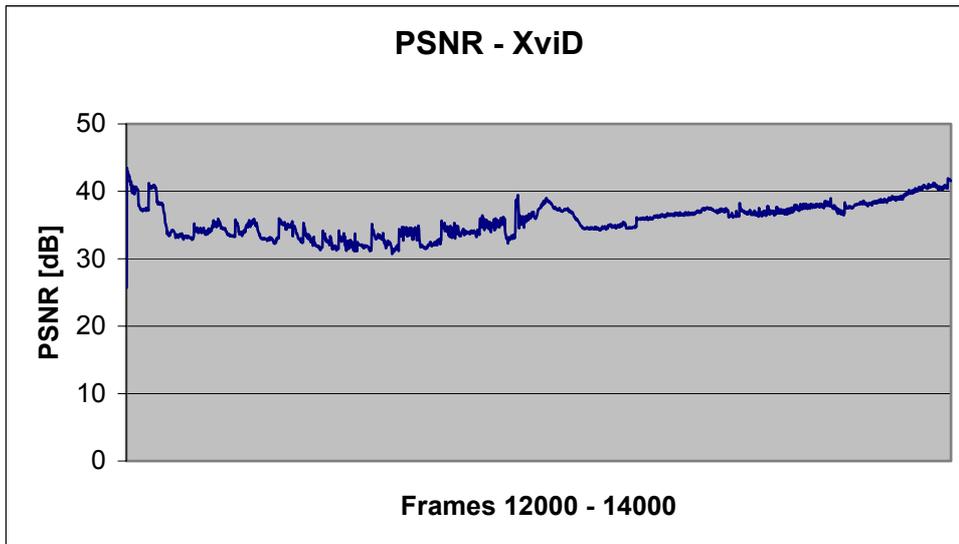


Abbildung 19: PSNR des 1. Ausschnittes bei XviD

Quelle: eigene Darstellung

Der Gesamtdurchschnitt liegt beim ersten XviD-Ausschnitt bei 35,7 dB, was die MOS-Wertung „Gut“ bedeutet.

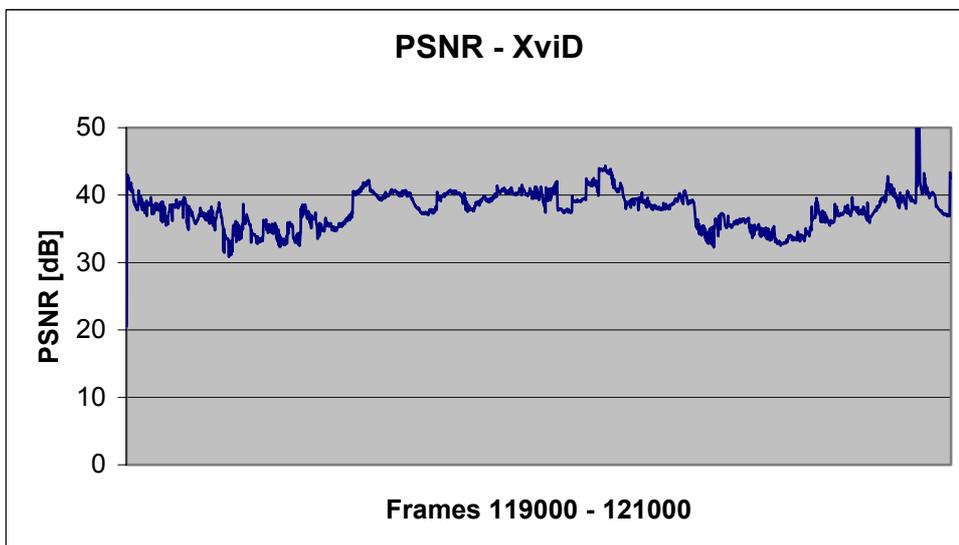


Abbildung 20: PSNR des 2. Ausschnittes bei XviD

Quelle: eigene Darstellung

Die Steigerung vom ersten auf den zweiten Ausschnitt ist bei jedem Derivat beobachtbar. XviD kommt auf 37,9 dB.

7.1.1.3. HDX4

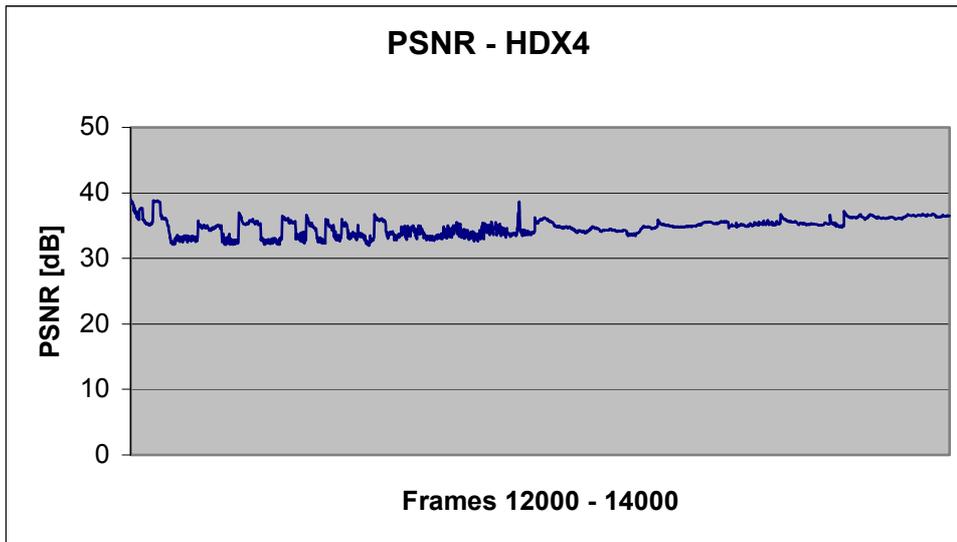


Abbildung 21: PSNR des 1. Ausschnittes bei HDX4

Quelle: eigene Darstellung

Der Codec aus dem Hause Jomigo fällt durch seine konstanten PSNR-Werte auf. Mit 34,8 dB reiht sich HDX4 knapp hinter XviD ein.

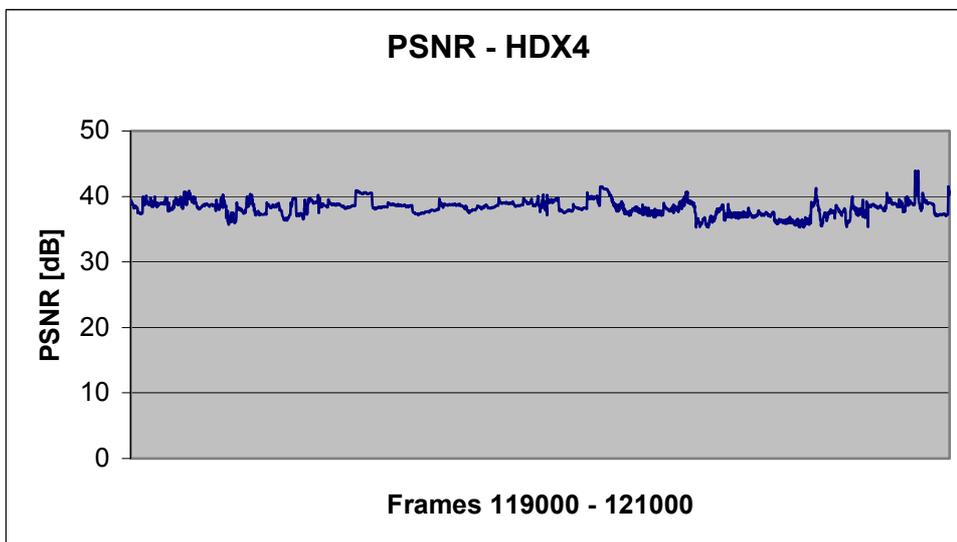


Abbildung 22: PSNR des 2. Ausschnittes bei HDX4

Quelle: eigene Darstellung

Die beständigen Werte bringen einen Mittelwert von 38,2 dB. XviD hat beim zweiten Ausschnitt das Nachsehen gegenüber HDX4.

7.1.1.4. 3ivx

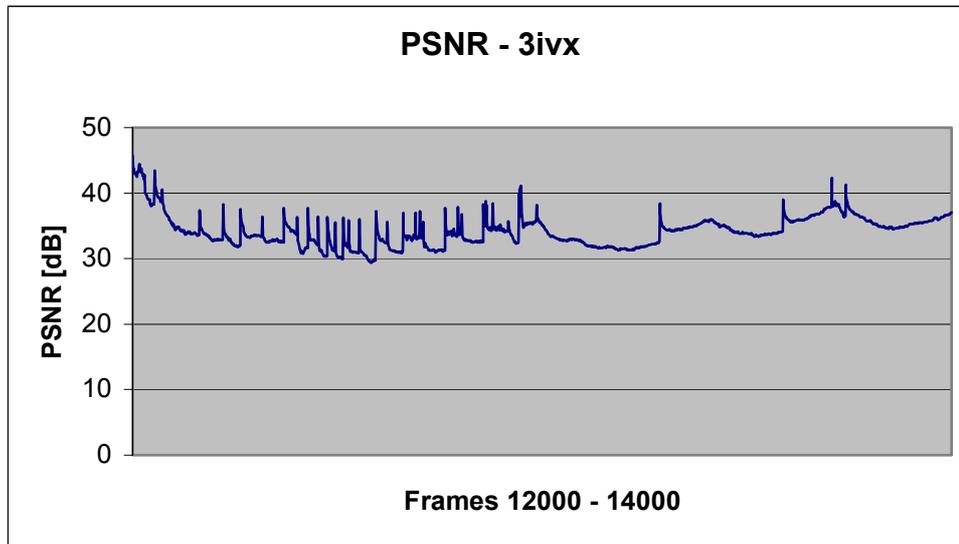


Abbildung 23: PSNR des 1. Ausschnittes bei 3ivx

Quelle: eigene Darstellung

Mit 34,2 dB liegt dieser Codec relativ deutlich hinter den anderen Derivaten, erhält allerdings trotzdem die MOS-Wertung „Gut“.

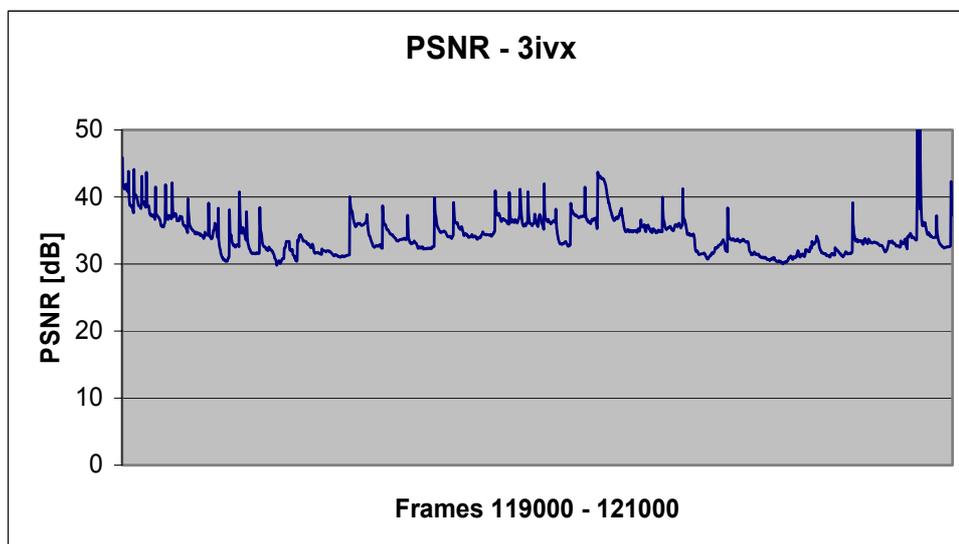


Abbildung 24: PSNR des 2. Ausschnittes bei 3ivx

Quelle: eigene Darstellung

3ivx hält mit einem Gesamtdurchschnitt von 34,3 dB das Niveau. Der PSNR-Index des zweiten Ausschnittes ist unbedeutend höher als der des ersten.

7.1.1.5. Nero Digital

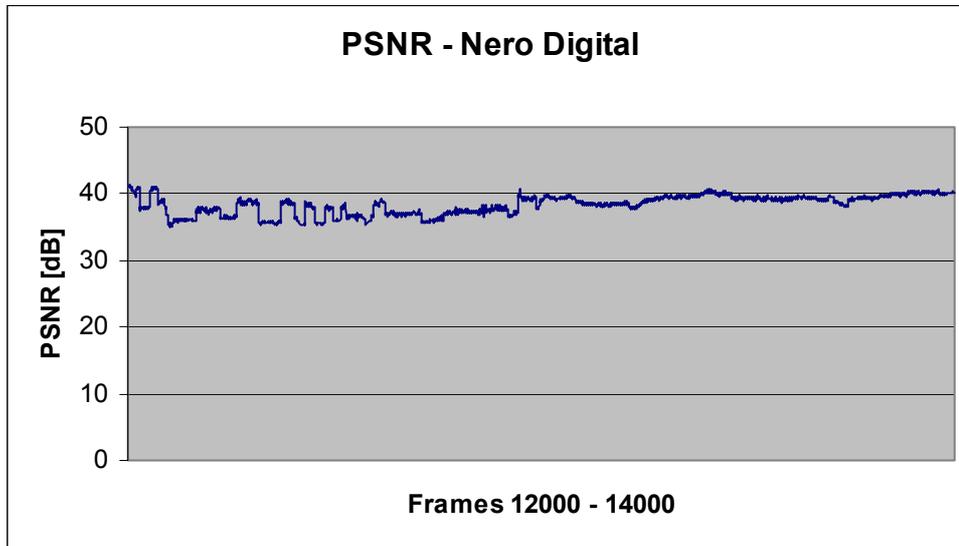


Abbildung 25: PSNR des 1. Ausschnittes bei Nero Digital

Quelle: eigene Darstellung

Einen noch etwas höheren Wert als DivX erreicht Nero Digital. Der PSNR-Durchschnitt beträgt 38,3 dB.

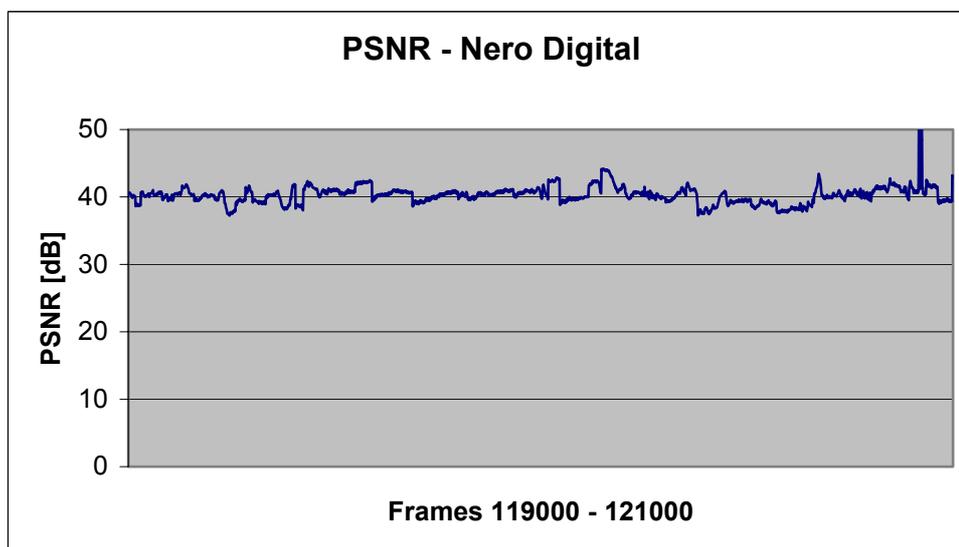


Abbildung 26: PSNR des 2. Ausschnittes bei Nero Digital

Quelle: eigene Darstellung

Der Codec steigert sich bei den Frames 119000 bis 121000 auf 40,3 Dezibel.

7.1.2. SSIM

Structural SIMilarity erkennt strukturelle Ähnlichkeiten zwischen dem Original und dem komprimierten Clip. Die Werte bewegen sich zwischen 0 und 1 – je höher, desto besser. Ein Wert von 1 bedeutet vollständige Übereinstimmung zwischen Original und codiertem Material.

7.1.2.1. DivX

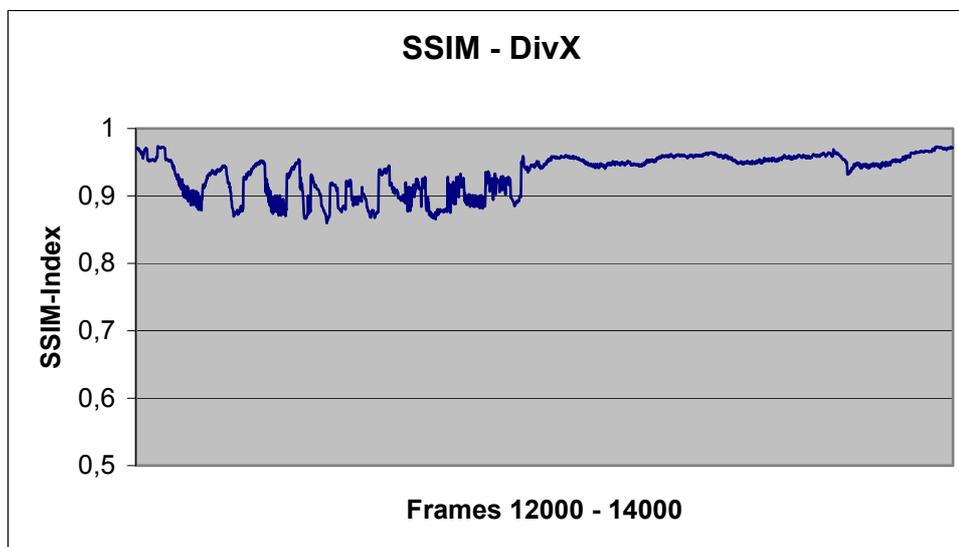


Abbildung 27: SSIM des 1. Ausschnittes bei DivX

Quelle: eigene Darstellung

Mit einem durchschnittlichen Wert von 0,93 kommt der DivX-Codec schon relativ nahe an das Original heran. Feststellbar ist wieder der Unterschied zwischen schnellen und ruhigen Szenen.

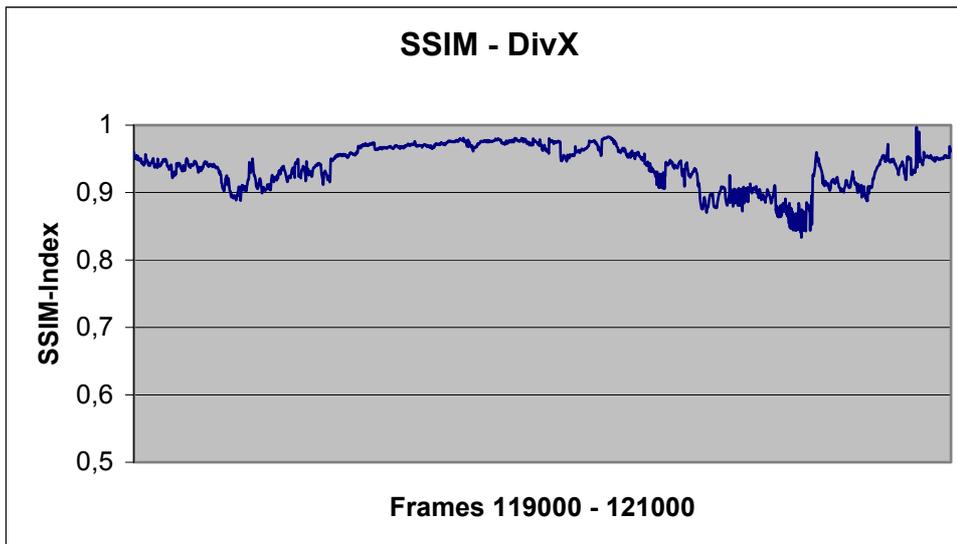


Abbildung 28: SSIM des 2. Ausschnittes bei DivX

Quelle: eigene Darstellung

Wie in Abbildung 28 erkennbar erreicht DivX beim zweiten Ausschnitt einen Wert von 0,94.

7.1.2.2. XviD

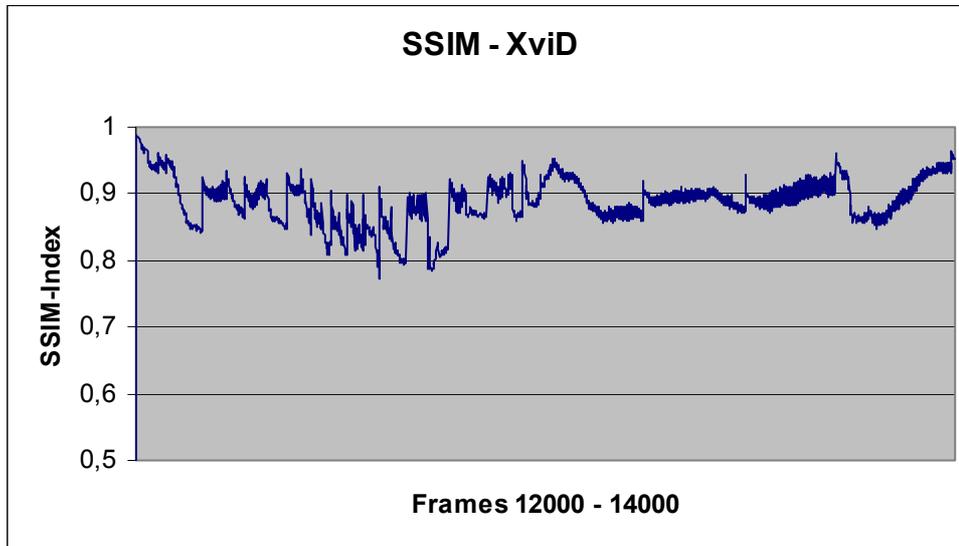


Abbildung 29: SSIM des 1. Ausschnittes bei XviD

Quelle: eigene Darstellung

Um einiges gestreuter als bei DivX erzielt XviD dessen ungeachtet einen Mittelwert von 0,89.

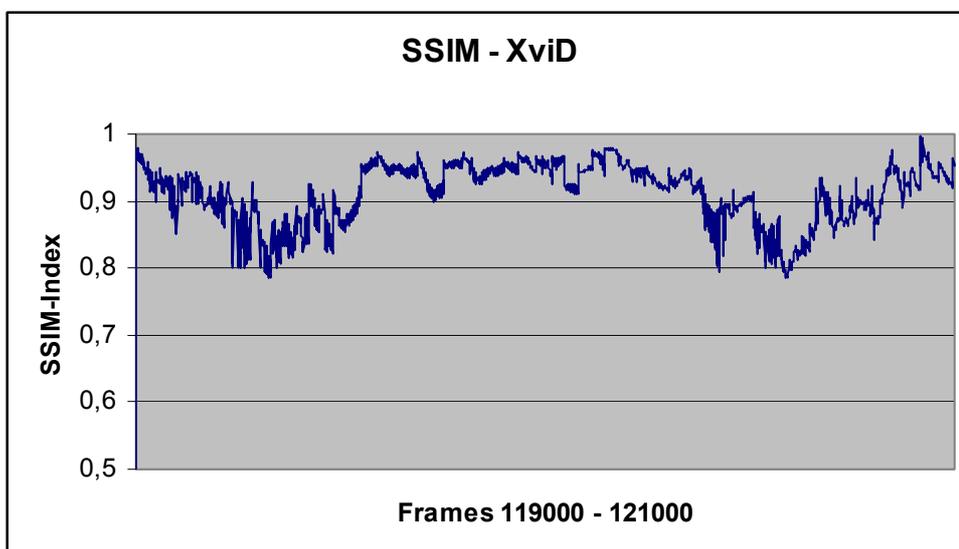


Abbildung 30: SSIM des 2. Ausschnittes bei XviD

Quelle: eigene Darstellung

Der Sprung über die 0,9-Grenze bringt das Derivat auf einen Wert von 0,91.

7.1.2.3. HDX4

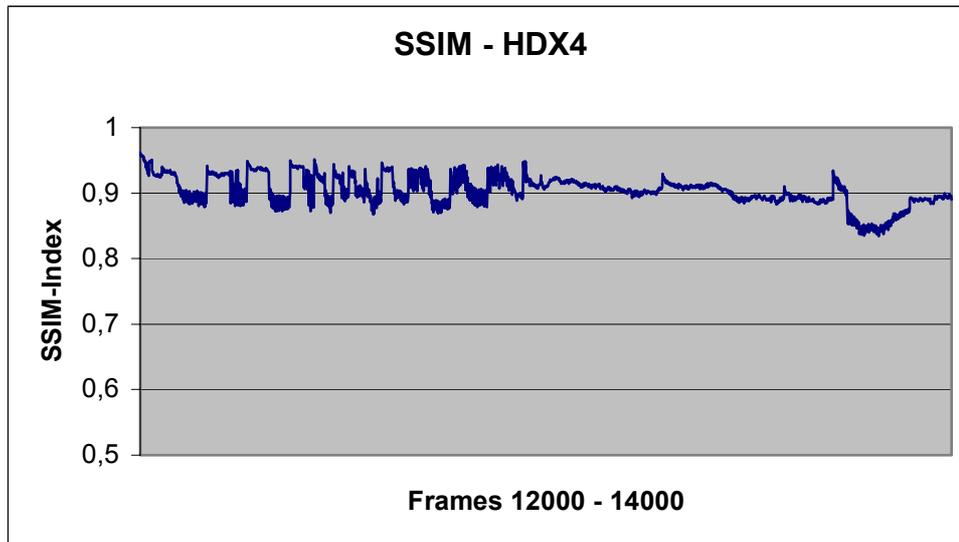


Abbildung 31: SSIM des 1. Ausschnittes bei HDX4

Quelle: eigene Darstellung

HDX4 erreicht einen durchschnittlichen SSIM-Index von 0,9. Auffallend ist der Rückfall in der zweiten, ruhigeren Hälfte.

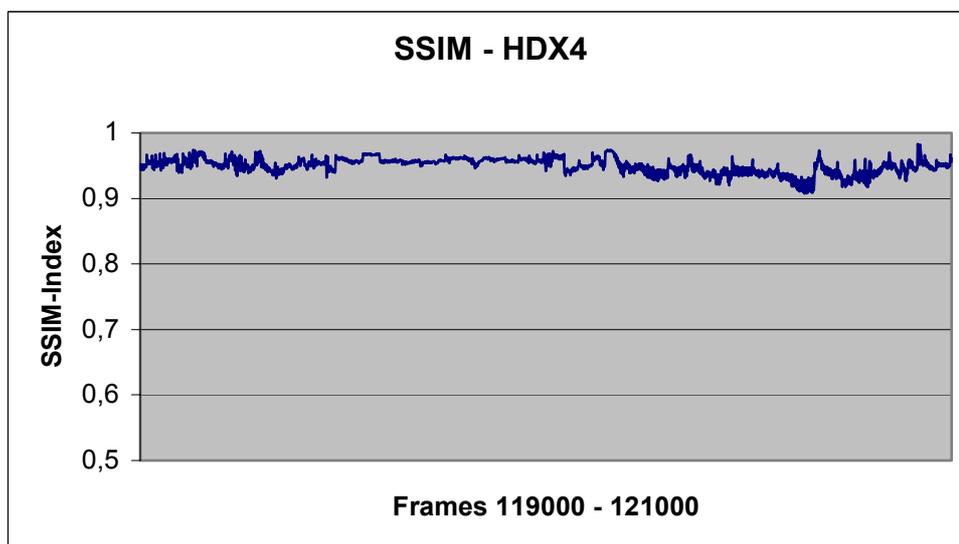


Abbildung 32: SSIM des 2. Ausschnittes bei HDX4

Quelle: eigene Darstellung

Gleichmäßige Werte hieven den Codec auf einen Mittelwert von 0,95.

7.1.2.4. 3ivx

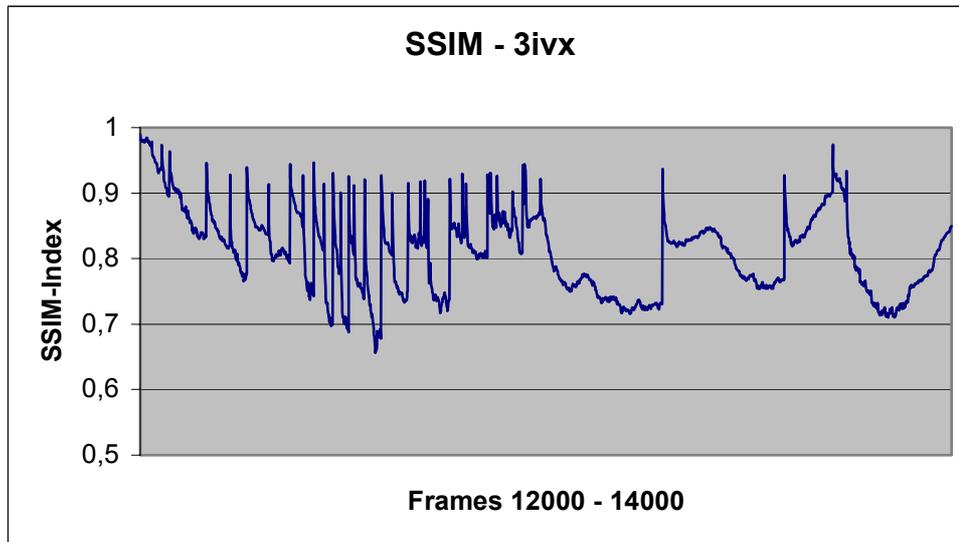


Abbildung 33: SSIM des 1. Ausschnittes bei 3ivx

Quelle: eigene Darstellung

Der aus Australien stammende Codec stößt hier wohl auf seine Grenzen. Der Durchschnitt liegt bei 0,81.

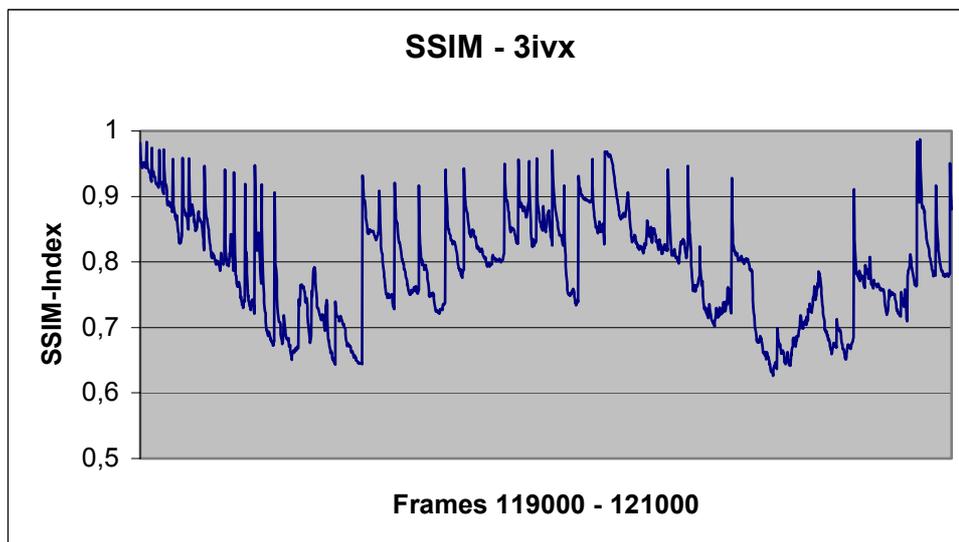


Abbildung 34: SSIM des 2. Ausschnittes bei 3ivx

Quelle: eigene Darstellung

3ivx hinkt mit 0,79 auch im zweiten Abschnitt nach.

7.1.2.5. Nero Digital

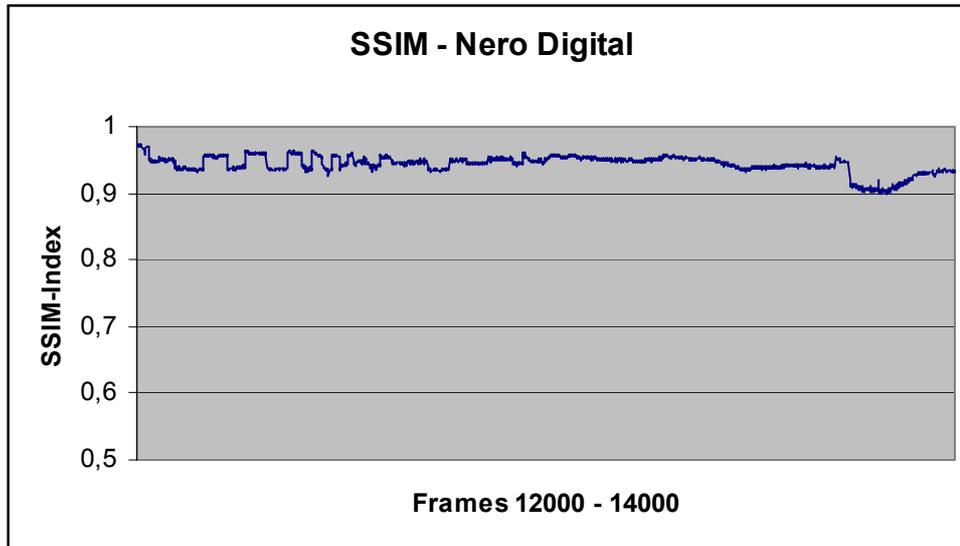


Abbildung 35: SSIM des 1. Ausschnittes bei Nero Digital

Quelle: eigene Darstellung

Auch Nero Digital weist im zweiten, „leichter“ zu codierenden Teil keinen markanten Anstieg auf. Dennoch kann sich der Codec mit 0,94 mit den Besten messen.

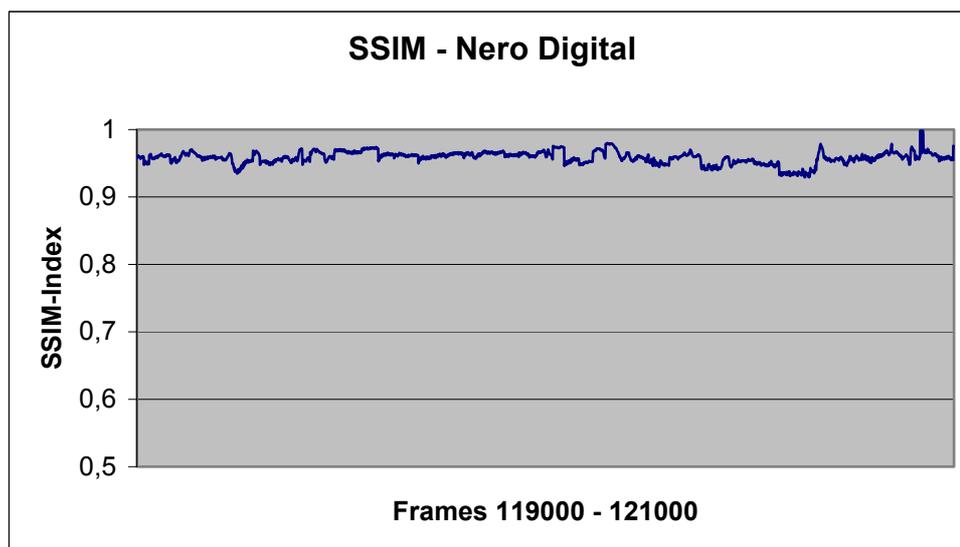


Abbildung 36: SSIM des 2. Ausschnittes bei Nero Digital

Quelle: eigene Darstellung

Ein noch etwas höherer Wert ergibt sich aus der SSIM-Messung der Frames 119000 bis 121000 – 0,96.

7.1.3. Direktvergleich

Um einen direkten Vergleich herstellen zu können, wird aus den Ergebnissen der objektiven Qualitätsbeurteilung jeder 50. Wert herausgenommen. Die vereinfachten Graphen geben darüber Aufschluss, wie die einzelnen Codecs bei den Tests abschneiden.

- PSNR

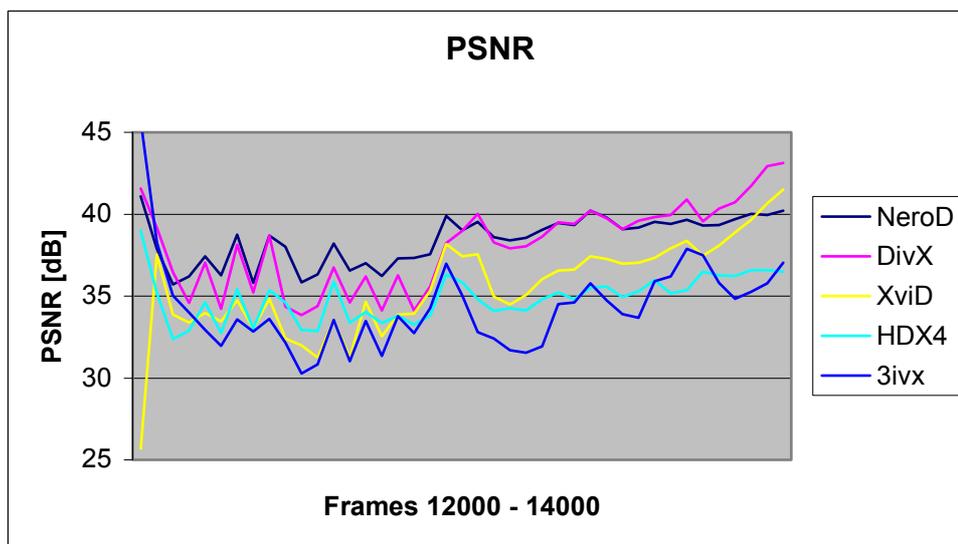


Abbildung 37: PSNR-Vergleich aller Codecs des 1. Ausschnittes

Quelle: eigene Darstellung

Aus dem ersten Direktvergleich zeichnen sich zwei Gruppen ab. Nero Digital (NeroD) erreicht in der ersten Hälfte Spitzenwerte, während DivX im ruhigeren Teil den besseren Eindruck macht. Dasselbe Bild bei den Derivaten XviD und HDX4, wo letzteres bei schnellen Szenen bessere PSNR-Werte erzielt und danach hinter XviD zurückfällt. An diese beiden kommt der Codec 3ivx nur zeitweise heran.

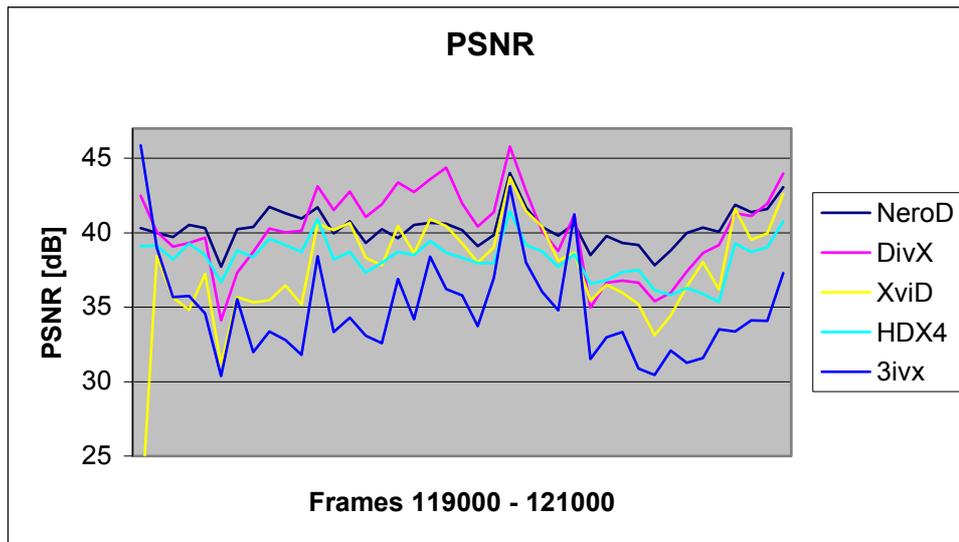


Abbildung 38: PSNR-Vergleich aller Codecs des 2. Ausschnittes

Quelle: eigene Darstellung

Ausgeglichenener ist das Ergebnis des zweiten PSNR-Vergleichs. Die wenigsten Unterschiede zwischen Original und Codec sind abwechselnd bei Nero Digital und DivX erkennbar. Aber auch XviD und HDX4 können teilweise mithalten.

- SSIM

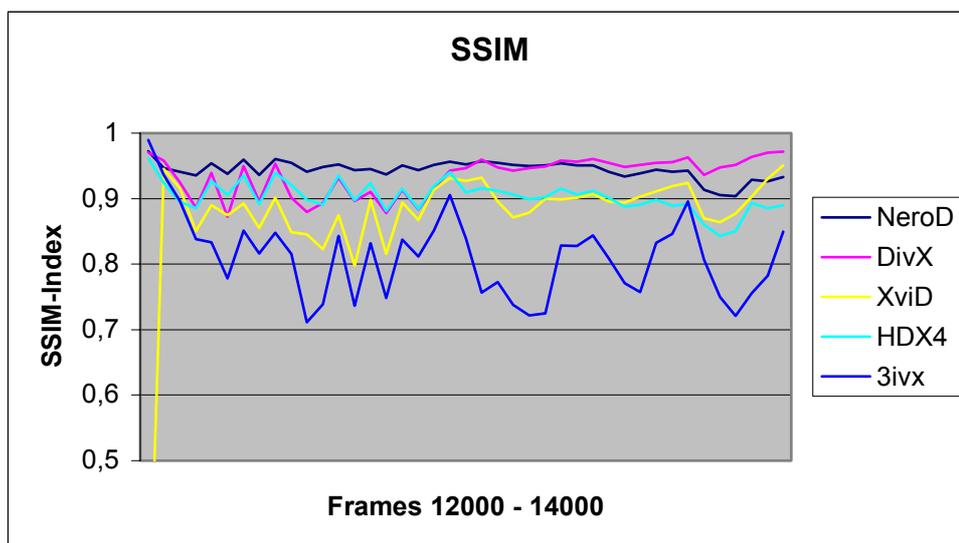


Abbildung 39: SSIM-Vergleich aller Codecs des 1. Ausschnittes

Quelle: eigene Darstellung

Bei der Suche nach strukturellen Ähnlichkeiten ergeben sich bis auf die zweite Hälfte der 2000 Frames aus Abbildung 39 Vorteile für Nero Digital. Der Kontrahent DivX erreicht in ruhigeren Szenen höhere SSIM-Werte. HDX4 kann im Gegensatz zu XviD bei den ersten 1000 Frames zu DivX aufschließen, fällt danach allerdings zurück. 3ivx kommt wieder auf verhältnismäßig niedrige Werte.

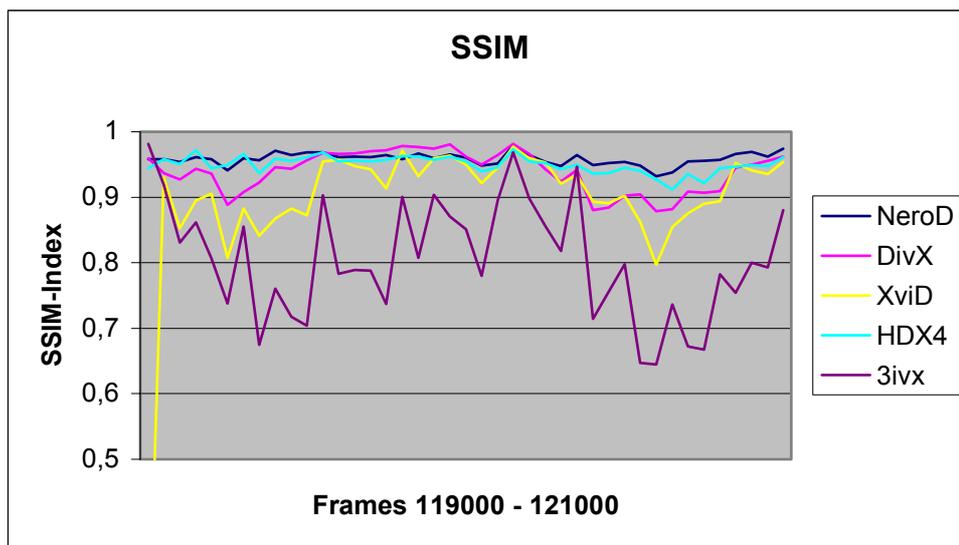


Abbildung 40: SSIM-Vergleich aller Codecs des 2. Ausschnittes

Quelle: eigene Darstellung

HDX4 überrascht beim zweiten SSIM-Vergleich mit hohen Werten und kann mit Nero Digital mithalten. DivX bleibt vor allem im zweiten Teil zurück und XviD erlangt nur teilweise gute SSIM-Werte.

7.1.3.1. Überblick

Im Überblick werden die Ergebnisse der einzelnen Tests einander direkt gegenübergestellt. Tabelle 11 zeigt alle Resultate auf einen Blick.

	Derivate (jeweils absteigend gereiht)				
Geschwindigkeit	HDX4 21,5 fps	XviD 16,6 fps	DivX 16 fps	3ivx 13,5 fps	Nero Digital 8,6 fps
Größe	DivX 701,2 MB (700 MB)	HDX4 701,6 MB (700 MB)	Nero Digital 695,3 MB (692 MB)	XviD 704 MB (700 MB)	3ivx 694,4 MB (700 MB)
PSNR (1. Teil)	Nero Digital 38,3 dB „Exzellent“	DivX 38 dB „Exzellent“	XviD 35,7 dB „Gut“	HDX4 34,8 dB „Gut“	3ivx 34,2 dB „Gut“
PSNR (2. Teil)	Nero Digital 40,3 dB „Exzellent“	DivX 40 dB „Exzellent“	HDX4 38,2 dB „Exzellent“	XviD 37,9 dB „Exzellent“	3ivx 34,3 dB „Gut“
SSIM (1. Teil)	Nero Digital 0,94	DivX 0,93	HDX4 0,9	XviD 0,89	3ivx 0,81
SSIM (2. Teil)	Nero Digital 0,96	HDX4 0,95	DivX 0,94	XviD 0,91	3ivx 0,79

Tabelle 11: Direktvergleich der Derivate

Quelle: eigene Darstellung

Auffällig sind die vergleichsweise unterschiedlichen Ergebnisse bei HDX4. Der SSIM-Test des zweiten Ausschnittes bringt fast den höchsten Durchschnittswert. Zu bemerken ist auch die Tatsache, dass Nero Digital in den objektiven Versuchen zwar stets die besten Werte erreicht, dies allerdings mit einer relativ gemächlichen Codiergeschwindigkeit bezahlt. Hinter Nero Digital kann sich beim PSNR-Test DivX behaupten, 3ivx bildet das Schlusslicht. Die SSIM-Versuchsreihe ordnet DivX und HDX4 abwechselnd nach Nero Digital ein.

7.1.3.2. Screenshots

Abseits der mathematischen Qualitätsbeurteilung sind eine Betrachtung und ein Vergleich einzelner Frames angebracht. Im folgenden Abschnitt werden die optischen Ergebnisse der Derivate gegenübergestellt. Anhand von objektiven Merkmalen können eindeutige Unterschiede zwischen den Konkurrenten ausgemacht werden.

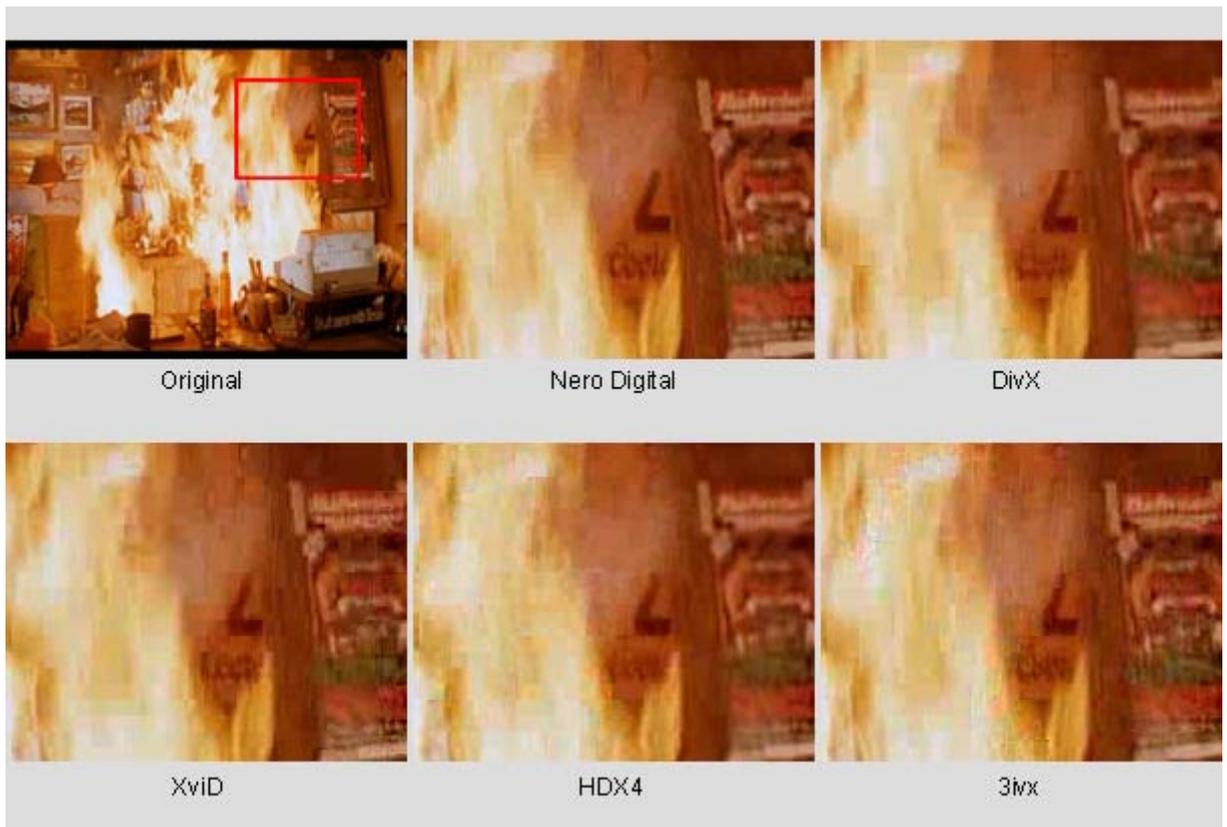


Abbildung 41: Bildvergleich bei Frame 12125

Quelle: eigene Darstellung

Abbildung 41 zeigt als erstes Bild das Original und im roten Rahmen den Ausschnitt, der unter den einzelnen Codecs verglichen wird.

Nero Digital verwendet die in Kapitel 4.9. behandelte Integertransformation, die mit einer Blockgröße von $4 * 4$ Pixel arbeitet. Von diesem Vorteil gegenüber den anderen Derivaten abgesehen, erscheint das Bild bei Nero Digital schärfer, die Übergänge zwischen den Blöcken sind originalgetreuer. Auffallend ist auch der Schriftzug in der Mitte des Frames, der deutlicher erkennbar ist.

DivX und XviD bewegen sich etwa auf dem gleichen Niveau. Allerdings tauchen im rechten Bilddrittel bei XviD harte Kanten auf, während DivX hier auf keinerlei Probleme trifft.

Das Bild bei HDX4 wirkt sehr körnig, mehr noch der 3ivx-Screenshot. Punkten kann HDX4 hingegen im rechten Bilddrittel, das den Vergleich mit DivX zulässt.



Abbildung 42: Bildvergleich bei Frame 13490

Quelle: eigene Darstellung

Die Personen im Vordergrund werden bei Nero Digital am besten aufgelöst. Im Gegensatz dazu ist der Hintergrund zu körnig geraten.

Bei DivX wirkt dieser weicher und auch ansehnlicher. HDX4 ist bezüglich des Hintergrunds zwischen den beiden anzusiedeln, zwischen körnig und weich.

XviD kann nicht mitziehen, sowohl die Personen, als auch die Szenerie ist unscharf und die Blöcke zeichnen sich deutlich ab.

Auf große Probleme stößt 3ivx, Farbflächen und Details werden mangelhaft codiert.



Abbildung 43: Bildvergleich bei Frame 119982

Quelle: eigene Darstellung

Das Frame aus Abbildung 43 bringt bei allen Derivaten außer 3ivx gute Ergebnisse.

Wieder sind bei Nero Digital aber Kanten und grobe Stellen sichtbar. Auch bei HDX4 haben sich in der linken Bildhälfte Blöcke gebildet.

XviD erzeugte ein gutes Bild, wobei das Frame leicht unscharf ist. Übertroffen wird es nur von DivX, es sind keine Blöcke bzw. Kanten erkennbar.

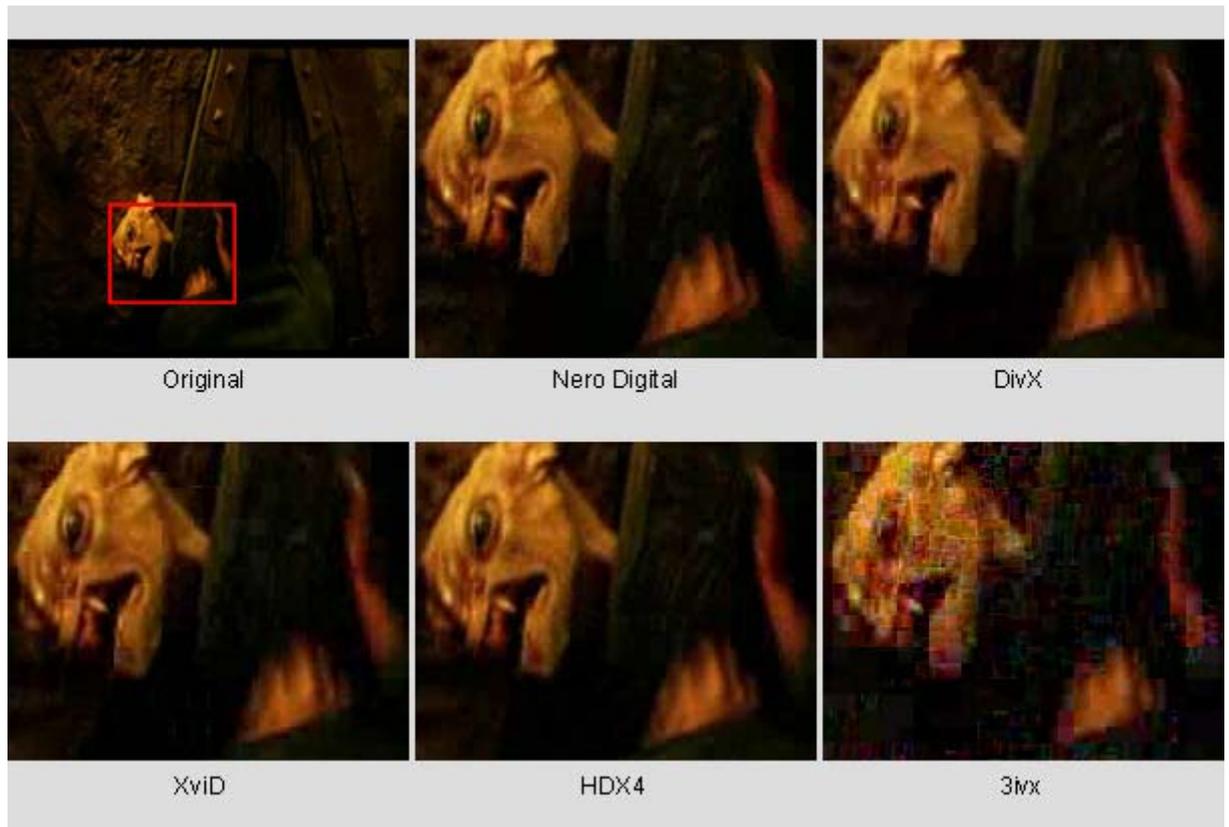


Abbildung 44: Bildvergleich bei Frame 120870

Quelle: eigene Darstellung

Den Abschluss bildet Frame Nummer 120870, das bei dem codierten Bild von Nero Digital und HDX4 nahezu einwandfreie Ergebnisse bringt. Bei letzterem Codec ist ein leichter Ansatz von Blockbildung erkennbar.

DivX und XviD können die Grenzen der Blöcke nur wenig kaschieren. Block- und Farbprobleme treten wieder bei 3ivx auf.

8. Schlussbetrachtung

Videokompression findet immer häufiger und auch zunehmend im Alltag ihre Anwendung. Da stets neue Komprimierungsfacetten erforscht werden, existieren zahlreiche Varianten von Codierprodukten. Die wichtigste Rolle bei der „Qual der Wahl“ spielt die Qualität der Resultate und eine möglichst hohe Annäherung an das Original.

Eine objektive Ermittlung der Qualität beschäftigt Experten seit Beginn der Videokomprimierung. Es gibt Algorithmen, die auf mathematischem Weg die Güte von Videodaten in Zahlen fassen wollen. Daraus kann allerdings nur ein Ansatz erstellt werden, der keine endgültige Bewertung zulässt.

Die Entscheidung über Qualität ist immer noch der Subjektivität überlassen. Es obliegt letzten Endes dem Betrachter, also dem menschlichen Auge, die beste Komprimierungsart zu erfassen.

9. Literaturverzeichnis

Dicks, Andreas/Götting, Detlef (2001): JPEG2000: Eigenschaften und Aufbau eines neuen Standards. In: Fernseh- und Kinotechnik, Vol. 6, S.363-370.

Heyna, Arne/Briede, Marc/Schmidt, Ulrich (2003): Datenformate im Medienbereich. München: Fachbuchverlag Leipzig im Hanser-Verlag.

Meyer, Martin (1999): Kommunikationstechnik. Braunschweig: Vieweg Verlag.

Pereira, Fernando/Ebrahimi, Touradj (2002): The MPEG-4 book. Upper Saddle River, NJ: Prentice Hall PTR.

Riempp, Roland/Schlotterbeck, Arno (1995): Digitales Video in interaktiven Medien. Berlin: Springer-Verlag.

Schmidt, Ulrich (2003): Professionelle Videotechnik. 3. Auflage, Berlin: Springer-Verlag.

Steinmetz, Ralf (1999): Multimedia-Technologie. 2. Auflage, Berlin: Springer-Verlag.

Watkinson, John (2004): The MPEG Handbook. MPEG-1, MPEG-2, MPEG-4. 2. Auflage, Oxford: Focal Press.

Internetquellen:

Avaro, Olivier/Eleftheriadis, Alexandros/Herpel, Carsten/Rajan, Ganesh/Ward, Liam: MPEG-Systems: Overview. Online im Internet: http://www.chiariglione.org/MPEG/tutorials/papers/icj-mpeg4-si/03-systems_overview_paper/3-systems_overview_paper.htm (Stand: 20.10.05; Abfrage: 23.06.06; [MEZ] 13:03 Uhr).

Ebrahimi, Touradj/Horne, Caspar: MPEG-4 Natural Video Coding - An overview. Online im Internet: http://www.chiariglione.org/MPEG/tutorials/papers/icj-mpeg4-si/07-natural_video_paper/7-natural_video_paper.htm (Stand: 20.10.05; Abfrage: 11.06.06; [MEZ] 22:21 Uhr).

Heising, Guido (2002): Nachrichtentechnische Kenngrößen. Online im Internet: <http://www.tfh-berlin.de/~mixdorff/dvt/Kenngroessen.pdf> (Stand: 24.04.02; Abfrage: 15.07.06; [MEZ] 12:42 Uhr).

Koenen, Rob (2002): Overview of the MPEG-4 Standard. Online im Internet: <http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm> (Stand: 04.07.03; Abfrage: 13.06.06; [MEZ] 19:14 Uhr).

Koenen, Rob: Profiles and Levels in MPEG-4: Approach and Overview. Online im Internet: http://www.chiariglione.org/MPEG/tutorials/papers/icj-mpeg4-si/11-Profiles_paper/11-Profiles_paper.htm (Stand: 20.10.05; Abfrage: 16.06.06; [MEZ] 11:14 Uhr).

Malvar, Henrique/Hallapuro, Antti/Karczewicz, Marta/Kerofsky, Louis (2003): Low-Complexity Transform and Quantization in H.264/AVC. Online im Internet: <http://research.microsoft.com/~malvar/papers/MalvarCSVTJuly03.pdf> (Stand: 30.07.03; Abfrage: 11.07.06; [MEZ] 10:50 Uhr).

Mandau, Markus (2003): Standard-Vergleich: MPEG2 versus MPEG4. Online im Internet:http://archiv.chip.de/artikel/c1_archiv_artikelunterseite_17143025.html?tid1=33105&tid2=0 (Stand: 28.05.03; Abfrage: 08.06.06; [MEZ] 16:23 Uhr).

Wang, Zhou/Bovik, Alan/Sheikh, Hamid/Simoncelli, Eero (2004): Image Quality Assessment: From Error Visibility to Structural Similarity. Online im Internet: <http://www.cns.nyu.edu/pub/eero/wang03-reprint.pdf> (Stand: 01.04.04; Abfrage: 14.07.06; [MEZ] 11:47 Uhr).

Wiegand, Thomas (2004): Video Coding Standards. Online im Internet: http://iphome.hhi.de/wiegand/assets/pdfs/video_coding_standards_03.pdf (Stand: 13.02.04; Abfrage: 25.08.06; [MEZ] 16:12 Uhr).

Zota, Volker (2002): DivX im Griff. Online im Internet: <http://www.heise.de/ct/02/18/094/default.shtml> (Stand: 30.06.06; Abfrage: 30.06.06; [MEZ] 13:34 Uhr).

O.V. (2006): MPEG Industry Forum. Online im Internet: <http://www.mpegif.com/mpeg4/> (Stand: 28.05.06; Abfrage: 29.05.06; [MEZ] 14:50 Uhr).

O.V. (1998): MPEG-4 Audio. Online im Internet: http://www.tnt.uni-hannover.de/project/mpeg/audio/documents/w1903_over.html (Stand: 15.01.1998; Abfrage: 21.06.06; [MEZ] 21:05 Uhr).

O.V. (1999): MPEG-4 Audio Version 2. Online im Internet: <http://www.tnt.uni-hannover.de/project/mpeg/audio/documents/w2803.html> (Stand: 29.07.1999; Abfrage: 21.06.06; [MEZ] 21:36 Uhr).

O.V. (2006): H.264. Online im Internet: <http://de.wikipedia.org/wiki/H.264> (Stand: 24.06.06; Abfrage: 26.06.06; [MEZ] 20:59 Uhr).

O.V. (2003): Structural Similarity Based Image Quality Assessment. Online im Internet: <http://www.cns.nyu.edu/~zwang/files/research.html> (Stand: 24.07.03; Abfrage: 06.07.06; [MEZ] 13:39 Uhr).

O.V. (2006): 3ivx. Online im Internet: <http://de.wikipedia.org/wiki/3ivx> (Stand: 05.07.06; Abfrage: 08.07.06; [MEZ] 19:02 Uhr).

O.V. (2006): Derivat (Software). Online im Internet: http://de.wikipedia.org/wiki/Derivat_%28Software%29 (Stand: 03.05.06; Abfrage: 07.07.06; [MEZ] 22:41 Uhr).

O.V. (2006): Divx. Online im Internet: <http://de.wikipedia.org/wiki/Divx> (Stand: 08.07.06; Abfrage: 08.07.06; [MEZ] 12:24 Uhr).

O.V. (2006): Xvid. Online im Internet: <http://de.wikipedia.org/wiki/Xvid> (Stand: 11.06.06; Abfrage: 08.07.06; [MEZ] 15:51 Uhr).

O.V. (2006): HDX4. Online im Internet: <http://de.wikipedia.org/wiki/HDX4> (Stand: 16.05.06; Abfrage: 08.07.06; [MEZ] 17:58 Uhr).

O.V. (2006): Nero Digital. Online im Internet: http://de.wikipedia.org/wiki/Nero_Digital (Stand: 24.04.06; Abfrage: 08.07.06; [MEZ] 21:34 Uhr).

O.V. (2004): The 3ivx Filter Suite. Online im Internet: <http://www.3ivx.com/technology/filters/index.html> (Stand: 08.04.04; Abfrage: 08.07.06; [MEZ] 19:29 Uhr).

O.V. (2006): Peak signal-to-noise ratio. Online im Internet: <http://en.wikipedia.org/wiki/PSNR> (Stand: 16.06.06; Abfrage: 06.07.06; [MEZ] 14:33 Uhr).

O.V. (2006): MPEG-4 Part 2. Online im Internet: http://en.wikipedia.org/wiki/MPEG-4_Part_2 (Stand: 27.08.06; Abfrage: 28.08.06; [MEZ] 12:14 Uhr).

O.V. (2006): DVB-S. Online im Internet: <http://de.wikipedia.org/wiki/DVB-S2> (Stand: 26.08.06; Abfrage: 28.08.06; [MEZ] 12:24 Uhr).

O.V. (2006): DVB – Digitales Fernsehen. Online im Internet: <http://www.alm.de/index.php?id=244&L=> (Stand: 28.08.06; Abfrage: 28.08.06; [MEZ] 12:26 Uhr).

O.V. (2006): MPEG-2. Online im Internet: <http://en.wikipedia.org/wiki/MPEG-2> (Stand: 29.08.06; Abfrage: 29.08.06; [MEZ] 11:27 Uhr).