

**Konzeption eines
digitalen Videoeditiersystems
auf Basis des
internationalen Standards
ISO/IEC 11172 (MPEG-1)**

**Arnd Steinmetz
Matthias Hemmje**

**Gesellschaft für Mathematik und Datenverarbeitung
(GMD)
Institut für integrierte Publikations- und Informationssysteme
(IPSI)**

REFERAT

DiVidEd „Distributed Video Editing System“

In der folgende Studie wird ein Konzept für ein Videoschnittsystem auf Basis der internationalen Norm ISO/IEC 11172 (MPEG) für digitales, komprimiertes Video entwickelt. Dazu wird zu Beginn eine Analyse der Aufgabenstellung und des Arbeitsablaufes des Videoschnittes durchgeführt deren Ergebnisse dann in der Konzeption des Systems berücksichtigt werden.

Innerhalb dieser Studie werden sowohl Konzepte zur Lösung der technischen Probleme des Editierens komprimierten digitalen Videos, als auch Ideen zur Unterstützung der über den rein „technischen Schnitt“ hinausgehenden kreativen Aufgaben des Cutters entwickelt.

Für Karin und Diana

An dieser Stelle möchte ich mich bei all denjenigen bedanken, die das Entstehen dieser Studie ermöglicht haben:

Zu erst danke ich meiner Frau Karin und meiner kleinen Tochter Diana, die mir all die Zeit und Unterstützung gaben, die ich brauchte.

Dann geht mein Dank an Matthias Hemmje und Dieter Böcker, die diese Studie erst ermöglichten und besonders an Matthias, der mir bei allen Problemen unermüdlich mit Rat und Tat zur Seite stand.

Mein Dank gilt auch Fr. Evi Ritzschel, die das ganze professionell korrigiert hat. Alle Fehler die jetzt noch vorhanden sein sollten, stammen sicherlich von meinen Nachbearbeitungen.

Danken möchte ich auch Prof. Dr. Erich Neuhold, der für die Veröffentlichung dieser Studie verantwortlich zeichnet.

INHALT

Inhalt	5
Abbildungsverzeichnis	7
1. Einleitung	8
1.1. Die „neue Medienlandschaft“	9
1.2. Übersicht	11
2. Fernsehtechnik	12
2.1. Historischer Überblick	13
2.2. Die gesellschaftliche Bedeutung	15
2.3. Die Funktionsweise	16
2.4. Digitales Fernsehen und digitale Aufzeichnung	18
3. Stand der Technik	21
3.1. Die Videoproduktion	22
3.2. Die herkömmliche Arbeitsumgebung	24
3.2.1. Der elektronische Schnitt	24
3.2.2. Das Studio	25
3.2.3. Player und Recorder	27
3.2.4. Timecode	30
3.3. Videoediting	33
3.3.1. Der Arbeitsprozeß	33
3.3.2. Linear Editing	36
3.3.3. Nonlinear Editing	36
3.4. Die herkömmlichen Systeme	38
3.4.1. Online Editing	38
3.4.2. Offline Editing	39
3.4.3. Analoge Systeme	40
3.4.4. Digitale Systeme	40
3.4.5. Unterstützung durch herkömmliche Systeme	41
3.5. Anforderungen an einen verbesserten Videoarbeitsplatz	44
4. MPEG	45
4.1. Warum MPEG?	46
4.2. Gründe für Datenkompression	47
4.3. Verfahren zur Bitratenreduktion	50

4.3.1.	Irrelevanzreduktion	50
4.3.2.	Redundanzreduktion	50
4.4.	International Standard ISO/IEC 11172	54
4.4.1.	Historie	54
4.4.2.	Überblick	56
4.4.3.	Systems	56
4.4.4.	Video	59
5.	DiVidEd	79
5.1.	Vorteile digitaler Integration	80
5.2.	Motivation des Systems	81
5.3.	System-Konzepte	82
5.4.	Methoden zum Editieren vom MPEG	84
5.4.1.	Wahlfreier Zugriff	84
5.4.2.	Wahlfreie, direkte Darstellung	84
5.4.3.	Variable Geschwindigkeit: Vorwärts	87
5.4.4.	Variable Geschwindigkeit: Rückwärts	91
5.4.5.	Schnitt von MPEG-Sequenzen	97
5.4.6.	Umcodierung einzelner Bilder	101
5.4.7.	Komposition inkompatibler MPEG-Streams	104
5.4.8.	Einhaltung der Constraints	106
5.5.	Verwaltung und Verwendung von Metainformation	107
5.5.1.	Automatisch zu gewinnende Informationen	107
5.5.2.	Zusätzlich zu speichernde Informationen	110
5.5.3.	Benutzerdaten Speichermodell	112
5.6.	Systemarchitektur	113
5.6.1.	Die Videodatenbank (Videobase)	113
5.6.2.	Die Source-Video-Clips	114
5.6.3.	Der Destination-Clip	115
5.6.4.	Der Video-Editor	116
6.	Ausblick	118
	Glossar	120
	Literaturverweise	127

ABBILDUNGSVERZEICHNIS

Abb. 2.1.a Der erste für die Massenproduktion konstruierte Fernseher von Loewe 1932[Denn74])	13
Abb. 2.1.b Die Entwicklung der Bildauflösung von 30 Zeilen je Bild 1929 bis 180 Zeilen je Bild 1936 [Denn74])	14
Abb. 2.3.a Die prinzipielle Funktionsweise der Fernsehtechnologie.	16
Abb. 3.2.a Die typische Konfiguration eines Videostudios	25
Abb. 3.2.b Die BBE 600 Editing Control Unit von BTS, ein typischer Videocontroller	26
Abb. 3.2.c Das Schrägspuraufzeichnungsverfahren für Magnetbänder	28
Abb. 3.2.d Probleme bei doppelten Timecodes	31
Abb. 3.2.e Probleme bei unzeitigem Timecode	32
Abb. 3.3.a Der generelle Arbeitsablauf des Videoediting	34
Abb. 3.4.a Typische Situation im Schnittstudio: Schnittsystem und Informationsverwaltung fin- den getrennt statt [Pott94]).	42
Abb. 4.3.a Die DPCM im Differenzbildverfahren	51
Abb. 4.3.b Die Bedeutung der Transformationskoeffizienten	52
Abb. 4.4.a Der prototypische MPEG-Codec	58
Abb. 4.4.b Die Phasen der MPEG Codierung/Decodierung	59
Abb. 4.4.c Die Korrelation der Chrominanz- mit den Luminanzpixeln nach CCIR 601.	62
Abb. 4.4.d Die Korrelation der Chrominanz- und Luminanzpixel nach MPEG	63
Abb. 4.4.e Die verschiedenen Bildcodierungsarten in MPEG	64
Abb. 4.4.f Darstellungs- und Speicherungsreihenfolge einer typischen MPEG-Sequenz	65
Abb. 4.4.g Das Verhalten der Intraquantisierung mit schmalen Nullband[MPEG93])	70
Abb. 4.4.h Das Verhalten der Interquantisierung mit breitem Nullband [MPEG93])	72
Abb. 4.4.i Die Reihenfolge der DCT-Koeffizienten zur Linearisierung	73
Abb. 4.4.j Codierungsablauf je Bildtyp	74
Abb. 4.4.k Der Aufbau der MPEG-Video-Hierarchie	76
Abb. 4.4.l Die Zerlegung des Picture in Slices und Macroblocks	77
Abb. 4.4.m Die Anordnung der Blocks innerhalb eines Macroblocks	77
Abb. 5.4.a Die Referenzvektoren	86
Abb. 5.4.b Die Bildqualität bei Verwendung weniger DCT-Koeffizienten [Wall91])	89
Abb. 5.4.c Die Modellierung des MPEG-Schnittes	98
Abb. 5.4.d Schnittmodell: Kürzen einer Sequenz	100
Abb. 5.4.e Schnittmodell: Einfügen einer Teilsequenz	101
Abb. 5.5.a Die Differenzbildcodierung ist nicht auf Bewegungserkennung ausgelegt	108
Abb. 5.5.b Ungeeignete Bildtypenfolge: Szenenwechsel ist nicht intracodiert	109
Abb. 5.6.a Die Systemarchitektur	113
Abb. 5.6.b Der Aufbau eines Source-Clips	114
Abb. 5.6.c Der Aufbau des Destination-Clips	115
Abb. 5.6.d Der interne Aufbau des Editors	116

1. EINLEITUNG

Dieses Kapitel gibt einen knappen Überblick über die künftigen Entwicklungen im Bereich der Informations-, Unterhaltungs- und Kommunikationstechnik und somit gleichzeitig die Motivation dieser Arbeit.

1.1. DIE „NEUE MEDIENLANDSCHAFT“

Trotz der unbestreitbar großen Relevanz des Fernsehens für unsere Gesellschaft gibt es nur eine äußerst rudimentäre Unterstützung durch geeignete „Werkzeuge“ zum Editieren und Redigieren von Fernsehbeiträgen und -sendungen.

Ganz im Gegensatz zu den heute an zweiter Stelle stehenden Printmedien beginnt sich im Bereich der Videotechnik erst jetzt eine breite Anwendung der Computertechnologie durchzusetzen. Worte wie „Desktop Video“ und „Nonlinear (Video-)Editing“ erwecken heute in der Videomedienwelt ähnliche Erwartungen und Befürchtungen wie noch in der letzten Dekade „Desktop Publishing“ und „Computersatz“ in der Printmedienwelt.

So schreiben denn auch H. Krug und T. Letocha:

„Inwieweit sich durch Desk Top Video die Qualität der erstellten Filme und damit unsere eigenen Ansprüche an professionelle Videobeiträge verändert, werden wir erst in einigen Jahren feststellen können. Eines allerdings ist jetzt schon sicher, mit [...] Produkten auf dem DTV-Markt wird die Arbeitswelt in der professionellen Videobearbeitung ähnlich revolutionär verändert werden, wie vor 10 Jahren Druck und Publishing durch die Einführung der DTP-Programme“

[KRUG93b]

Auf der anderen Seite findet das Medium Video immer weitere Verbreitung im Bereich der Computeranwendungen. Wie die Definition des weltweiten Standards MPEG(vgl. Kap. 4.) zeigt, hat sich Video schon als ein zusätzlicher Datentyp zu Text, Graphik und Audio etabliert.

Kaum eine zukunftsorientierte Softwareneuentwicklung wird an der Einbindung von Videosequenzen in das Informationsmanagement vorbeikommen. Dies gilt für alle Bereiche der Software-Entwicklung und nicht nur für die offensichtlichen Anwendungsgebiete wie Schulung, Präsentation etc. Die Verwendung und Akzeptanz von Audio und Video wird in (beinahe) allen Anwendungssparten der Computer mit der steigenden Leistungsfähigkeit der Rechner einen ganz ähnlichen Verlauf nehmen, wie die Verwendung von graphischen Darstellungen und Eingabegeräten sie schon durchlaufen hat.

Dennoch existieren auch hier, im Bereich der Bearbeitung des Datentyps Video, noch kaum geeignete Tools zur Unterstützung der Anwender und Entwickler. Das läßt sich durch den im hohen Grade spezialisierten Expertenstatus der bisherigen Videotechniker sowie durch den uns allen inhärenten eher passiven Umgang mit Video erklären. Um so nötiger wird es aber, das langjährig angesammelte Expertenwissen der Videofachleute in Tools zur Anwendung auf Rechnern umzusetzen und damit einer breiteren Anwendungsschicht zugänglich zu machen.

Ein weiterer Aspekt der kommenden Medienlandschaft muß dabei auch Berücksichtigung finden: Die Kommunikation über Netzwerke.

Überall in den Industrienationen wird die Idee der „Informations-Autobahnen“ in die Realität umgesetzt. Diese Computernetzwerke zur Übertragung riesiger Datenmen-

gen werden besonders zur Übertragung von Videokommunikation und „Fernsehprogrammen“ vorgesehen.

Somit wachsen also drei bisher getrennte Entwicklungsgebiete zusammen:

- Informationstechnologie
- Unterhaltungstechnologie
- Kommunikationstechnologie

Daraus ergeben sich faszinierende Möglichkeiten der Informationsverbreitung und -auswertung, besonders durch die Kombination verschiedener Informationsmedien. Als Beispiele seien hier nur genannt:

- Interaktives Fernsehen
- Individuelle Nachrichtenauswahl
- Flexible Informationsdarstellung
- Visuelles Telefon
- Video on Demand Service

(Für Details sei auf [HERT90] verwiesen.)

Um einen effektiven Umgang mit der zu erwartenden Informationsflut zu ermöglichen, müssen Tools erstellt werden, die das Wissen von Experten aller Teilgebiete kombinieren und einer breiten Anwenderschaft zugänglich machen. Ohne solche Werkzeuge ist eine sinnvolle und effektive Anwendung einer „Informations-Autobahn“ nicht möglich!

1.2. ÜBERSICHT

Ziel dieser Arbeit war die Konzeption eines Tools zur Bearbeitung des Datentyps Video in der Codierung nach dem ISO/IEC 11172 (MPEG) Standard.

Dazu wurde zunächst eine Studie über die Fähigkeiten konventioneller, dem heutigen Stand der Technik entsprechender Videoschnittsysteme erstellt (Kap. 2 und 3). Sehr schnell verifizierte sich allerdings, was auch R. Driemeyer in [Müll92] schreibt:

”Im Bereich der Film- und Fernsehproduktion gibt es Literatur über Dramaturgie und Urheberrecht, über Verträge und vieles mehr – aber [...] Literatur über die Praxis des elektronischen (Video) Schnitts ist äußerst selten.”

Daher dienten als Grundlagen der Studie hauptsächlich aktuelle Artikel aus Fachzeitschriften sowie persönliche Gespräche mit MAZ-Technikern (Videocuttern) beim Hessischen Rundfunk und in der Fa. CVS in Frankfurt.

Auch zur Norm ISO/IEC 11172 (MPEG) existiert bisher noch kaum Zusatzliteratur. Daher mußten bei der Erstellung des Kap.4 einige Direktkontakte zu MPEG-Entwicklern und an der Standardisierung Beteiligten (über Internet) hergestellt werden, um die nötigen Zusatzinformationen zu noch nicht publizierten Details zu erhalten.

Die Ergebnisse der Studie wurden in Kap.5 in das Konzept eines MPEG-Video-Editier-Systems eingebracht, das zum einen die Editierung von MPEG-codierten Videos überhaupt erst ermöglicht und zum anderen die Fähigkeiten der konventionellen Systeme, nicht jedoch deren Schwachstellen übernimmt. Das entstandene Konzept ist auch auf andere Videocodierungen erweiterungsfähig und berücksichtigt die oben genannten Aspekte einer zukünftigen Medien- und Kommunikationslandschaft, da es auch in einer verteilten Umgebung realisierbar ist.

2. FERNSEHTECHNIK

Dieses Kapitel erklärt die Funktionsweise der herkömmlichen analogen Fernseh-, also Videotechnologie sowie die Ansätze, in diesem Bereich auf eine digitale Speicherung und Übertragung überzugehen. Dieses Wissen ist notwendig, um den Aufbau und die Funktionalität der Videoschnittsysteme verstehen zu können.

2.1. HISTORISCHER ÜBERBLICK

- 1897 Entwicklung der ersten Kathodenstrahlröhre durch Ferdinand Braun.
- 1929 Erster Fernsehsendebetrieb in Schwarzweiß anlässlich der Internationalen Funkausstellung in Berlin. Bildauflösung: 30 Zeilen.



Abb. 2.1.a Der erste für die Massenproduktion konstruierte Fernseher von Loewe 1932 [Denn74])

- 1953 Inbetriebnahme des ersten Farbfernsehsystems in den USA. Entwickelt vom National Television System Committee (NTSC) lieferte dieser Standard 30 Vollbilder (60 Halbbilder) pro Sekunde mit einer Zeilenzahl von 525 Zeilen je Vollbild. Die Übertragung der Farbinformation erfolgte zusätzlich zu der Helligkeit, so daß Kompatibilität zum schon vorhandenen Schwarzweiß-Standard M der FCC (Federal Communications Commission) gewährleistet war.
- 1957 Entwicklung des auf dem NTSC basierenden SECAM (Système en couleur avec mémoire) durch Henri de France, der einige Schwächen des NTSC-Verfahrens bei Übertragungsstörungen ausgleichen wollte. Das SECAM-System arbeitete mit einer Bildfrequenz von 25 Bildern/sec und einer Zeilenzahl von 625 Zeilen/Bild.
- 1963 Entwicklung des PAL (Phase alternating line) Systems durch Walter Bruch. Es basiert ebenfalls auf dem NTSC System und arbeitet mit den gleichen Bildparametern wie das

SECAM-System, also 25 Bilder/sec mit 625 Zeilen. Es sollte die Übertragung der Farbinformation verbessern, deren Auflösung im SECAM-System geringer war als bei NTSC, ohne wieder dessen Störungssensibilität einzuführen.

- 1988 Erste Übertragung im HDTV-Format in Japan. HDTV zeichnet sich durch eine bessere Bildauflösung (1125 Zeilen, 60 Halbbilder/sec) und ein Bildseitenverhältnis von 16:9 (im Gegensatz zu den herkömmlichen 4:3) aus. Ein internationaler Standard konnte bis heute noch nicht geschaffen werden, obwohl schon seit 1992 täglich ein acht-stündiges Fernsehprogramm in Japan gesendet wird. Statt dessen existieren wieder verschiedene nationale Realisationsentwürfe.
- 1994 Einführung von PAL+ in Deutschland. Es ist zum PAL-System kompatibel, d.h., alte PAL-Empfänger können auch PAL+ darstellen, bringt aber das Bildseitenverhältnis auf 16:9.

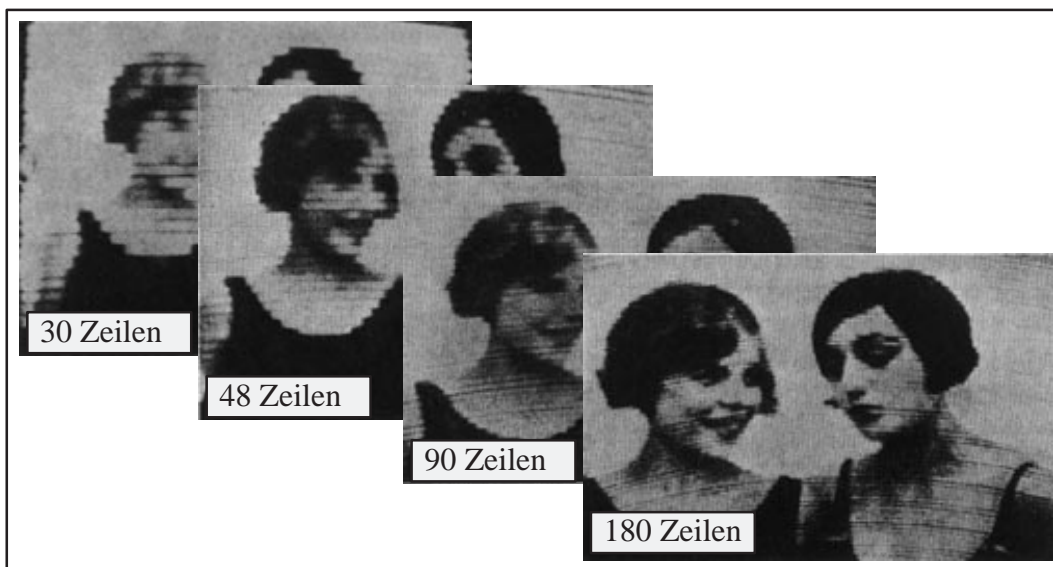


Abb. 2.1.b Die Entwicklung der Bildauflösung von 30 Zeilen je Bild 1929 bis 180 Zeilen je Bild 1936 [Denn74]

2.2. DIE GESELLSCHAFTLICHE BEDEUTUNG

Die Fernsehtechnik ist also noch gar nicht so alt, was sie mit der Computertechnologie gemeinsam hat. Allerdings fand sie schon wesentlich früher eine starke Verbreitung und Akzeptanz in der Bevölkerung und stellt heute die Medientechnologie schlechthin dar.

Die Existenz eines Mediums wie des Fernsehens hat unsere moderne Gesellschaft entscheidend mitgeprägt und die Informationsfreiheit des Einzelnen erheblich unterstützt, wie neuere Gerichtsurteile des Bundesverfassungsgerichtes belegen, die jedem das Recht auf den ungehinderten Empfang von Fernsehsendern zugestehen und ein Fernsehgerät pro Haushalt als so essentiell ansehen, daß es nicht pfändbar ist.

Zitat aus der Neuen Juristischen Wochenschrift:

„[...] Zum anderen sind die vom BVerwG im Urteil vom 22.5.1975 (BVerwGE 48, 237 [239]) aufgezeigten Alternativen der Kommunikation, Information und Unterhaltung jedenfalls teilweise für weite Bevölkerungsteile nicht (mehr) realisierbar [...]. Es ist daher dem VG Oldenburg (NJW 1991, 2921) zuzustimmen, daß dem elektronischen Bildmedium Fernsehen im Vergleich zu anderen Kommunikationsmedien eine überragende Bedeutung bei der Informations- und Unterhaltungsfunktion sowie bei der Vermittlung der Teilhabe am Prozeß der politischen Meinungsbildung zukommt.

[...] Nach alledem ist davon auszugehen, daß nach den heute herrschenden Lebensgewohnheiten ein Schwarz-Weiß-Fernsehgerät zum notwendigen Lebensunterhalt [...] zu zählen ist. [...] Ein Fernsehgerät ist als Sache des persönlichen Gebrauchs unpfändbar.“

[NJW93]

2.3. DIE FUNKTIONSWEISE

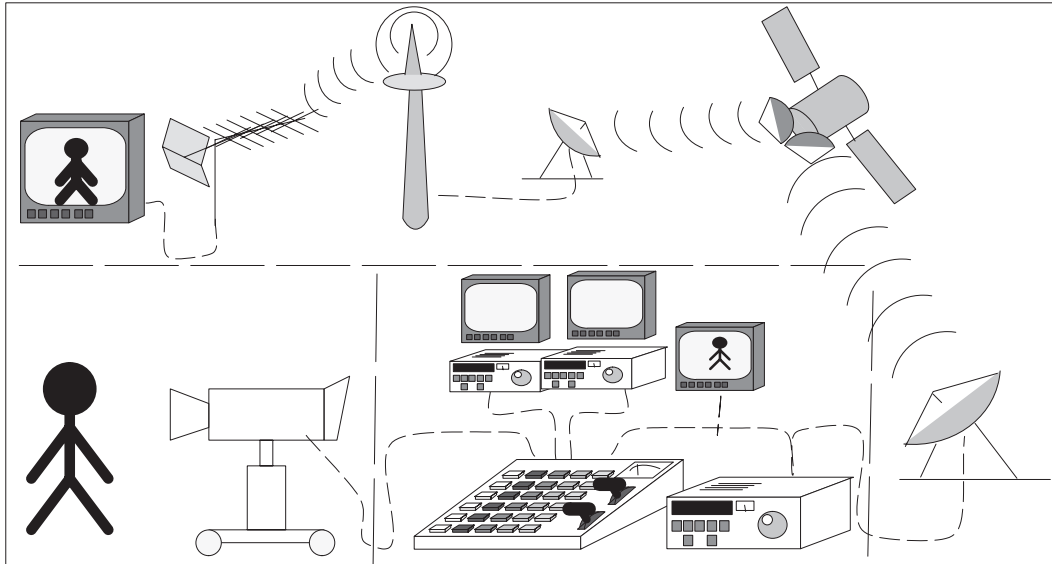


Abb. 2.3.a Die prinzipielle Funktionsweise der Fernsehtechnologie.

Bei der herkömmlichen Fernsehtechnologie nimmt eine Kamera das Bild auf und wandelt es über einen (oder mehrere) sog. CCD-Chip (Charge coupled device) in elektrische Impulse um. Das Bild wird dabei zeilenweise abgetastet, in Europa 25 mal pro Sekunde mit 625 Zeilen. Dieses Signal hat eine große Amplitude bei einem hellen Bildpunkt und eine kleine, wenn der entsprechende Bildpunkt dunkel ist (vgl. Abb. 2.3.b).

Das Bildsignal wird zusammen mit sogenannten Synchronisationssignalen (syncs), die jeweils das Ende einer Zeile bzw. das Ende eines Bildes anzeigen weitergegeben zur Regie. Da ein solches Signal Frequenzen zwischen 10 Hz und 5 MHz aufweist (je nachdem, wie dunkle und helle Bildpunkte verteilt sind), wird eine Frequenzmodulation auf einem Zwischenfrequenz-(ZF)-Träger ausgeführt, um das Signal besser verarbeiten zu können. Das Signal kann nun auf einer MAZ (magnetische Aufzeichnungsanlage) aufgezeichnet werden oder über den Sender oder Satelliten direkt (live) abgestrahlt werden.

Zunächst zur „analogen“ Aufzeichnung. Das von der Signalquelle kommende Videosignal wird auf dem Video-Recorder analog aufgezeichnet. Auf dem Magnetband entspricht dann ein dunkler Bildinhalt einer niedrigen Frequenz, ein heller Bildinhalt einer hohen Frequenz (vgl. Abb. 2.3.b). Gleiches gilt für die Audioaufzeichnung: Ein leiser Ton wird mit einer niedrigen Amplitude und ein lauter Ton mit einer hohen Amplitude geschrieben.

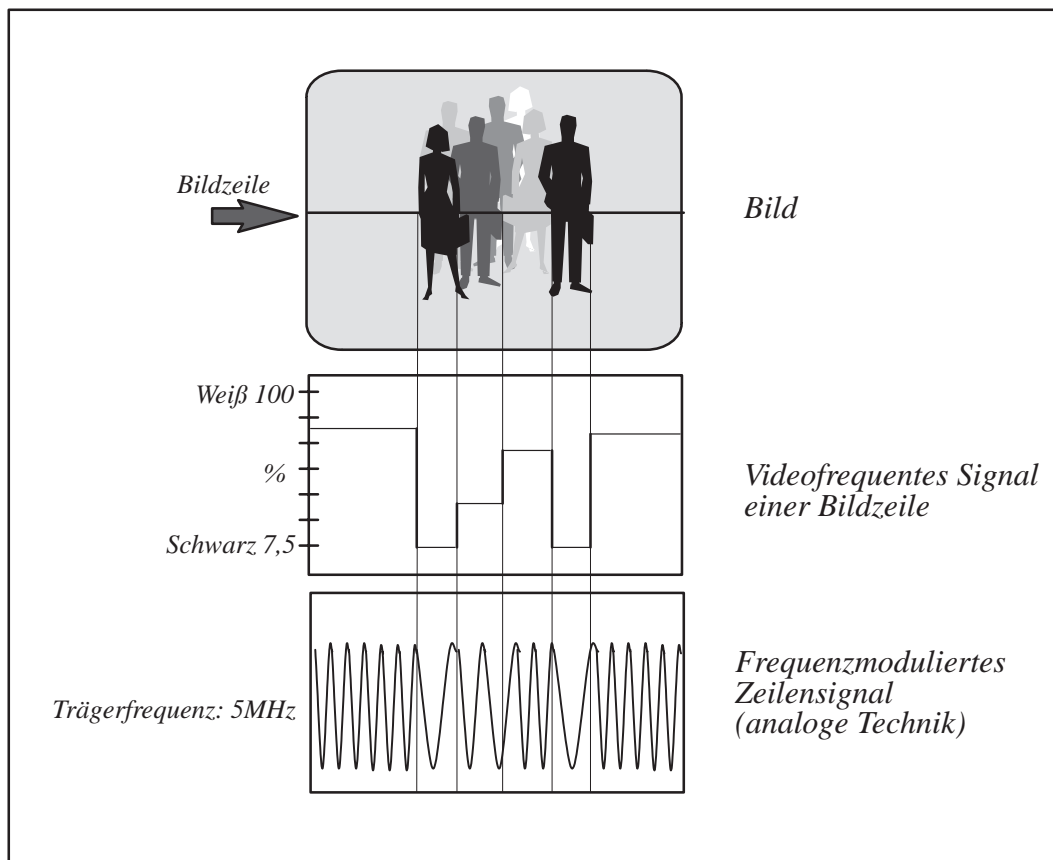


Abb. 2.3.b Die Codierung der Bildpunkte in analoger Technik

Während die Sende- und Empfangstechnik schon zu den Frühzeiten der Fernsehentwicklung zur Verfügung stand (vgl. Kap. 2.1.), wurde die MAZ erst 1956 von der Firma AMPEX entwickelt. Bis dato konnte tatsächlich nur live oder durch Abtastung von Filmmaterial gesendet werden (vgl. Kap. 3.2.1., Kap. 3.2.3.1.).

Zum Senden wird das Signal erneut auf einen noch höherfrequenten Träger modelliert. Dieses UHF (Ultra High Frequency) Signal wird über Sendemasten und Satelliten an die Empfänger weitergegeben, im Empfänger wieder demoduliert und mit Hilfe der Syncanteile die Position und Intensität eines Elektronenstrahls innerhalb einer Bildröhre gesteuert, der nun eine Matrix von Bildelementen wieder zeilenweise bestrahlt und zum Leuchten anregt. Diese Bildelemente leuchten lange genug nach, um bis zum nächsten „Refresh“ durch das nächste Bild einen optischen Eindruck zu hinterlassen. Der Bildaufbau erfolgt so schnell, daß das menschliche Auge ihn nicht wahrnehmen kann. Zudem reagiert das Auge so träge, daß bei einer Bildrate von 25 Bildern pro Sekunde der Eindruck einer flüssigen Bewegung in einem stehenden Bild wahrgenommen wird.

(vgl. [CCIR 624], [Prit80])

2.4. DIGITALES FERNSEHEN UND DIGITALE AUFZEICHNUNG

Die Zukunft des Fernsehens wird zweifellos digital sein, da es sehr viele Vorteile bietet. In der Studiotechnik ist die Digitalisierung schon viel weiter fortgeschritten, als vielfach angenommen wird (vgl. [Wolf93],[Krug94]).

Bei der digitalen Aufzeichnung wird das Videosignal nicht frequenzmoduliert, sondern durch Pulsmodulation (PCM) in einer Folge von Impulsen codiert, die einem Zahlenwert zwischen Null und 256 entsprechen. Diese wird aufgezeichnet. Die Information steckt in diesem Fall in einer Impulsfolge und nicht in einem Frequenz- oder Amplitudenwert (vgl. Abb. 2.4.a).

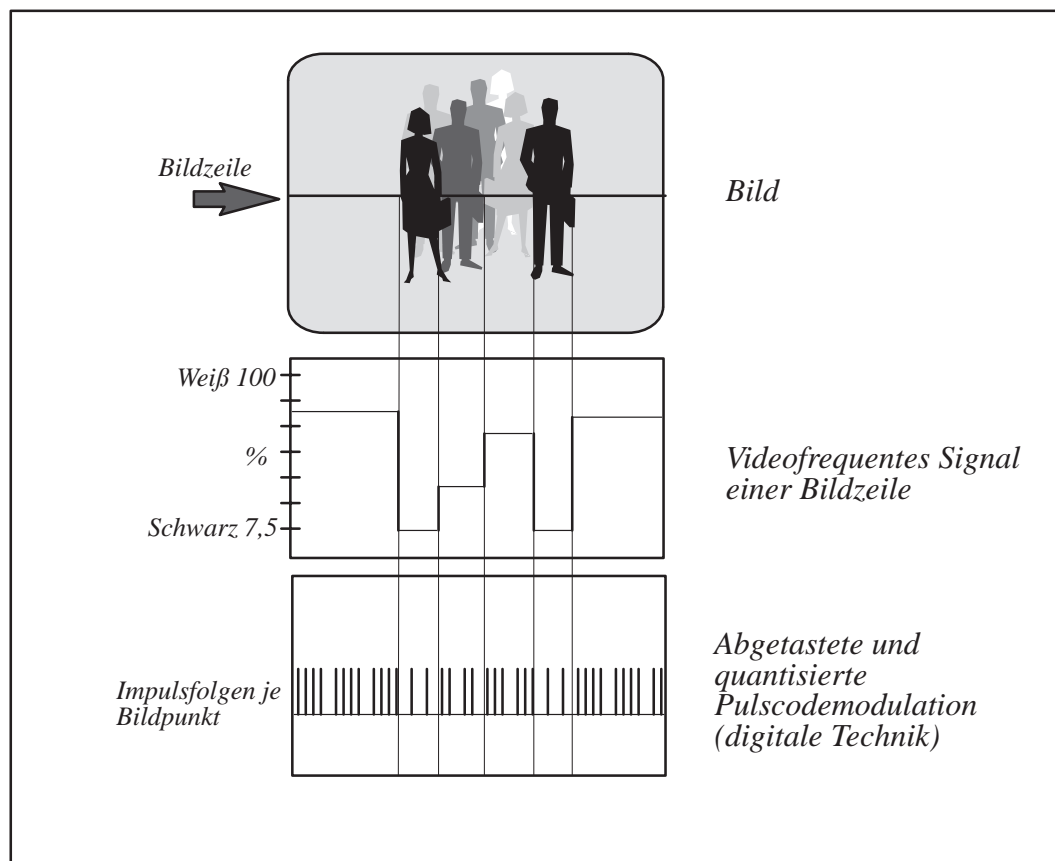


Abb. 2.4.a Die Codierung der Bildpunkte in digitaler Technik

Vorteile der Digitalisierung

Die Digitalisierung von Information und deren Weiterverarbeitung hat generell Vorteile:

Die Information liegt nun als Zahl und nicht mehr als reiner Spannungswert vor, und entsprechende Codierungen der Impulsfolge machen diese Information unempfind-

licher gegen Pegelschwankungen, Rauschen und anderen Störungen der Übertragung bzw. der weiterverarbeitenden Elektronik. D.h. insbesondere, daß eine Kopie der ursprünglichen Bildsequenz keine Verluste mehr erleidet und daher beliebig viele Kopiergenerationen erzeugt werden können.

Die Zahl kann zudem mathematischen Transformationen unterzogen werden, wodurch vielfältige bewußte Manipulationen der Information möglich sind: Alle Arten der analogen Filterung und der Spezialeffekte sind ebenso durch Berechnungen auf einer digitalen Repräsentation des Bildes erreichbar. Sie sind zudem leicht austausch- oder verbesserbar, da sie nur auf Software beruhen und nicht wie ihre analogen Gegenstücke in Hardware realisiert werden müssen. Damit bekommt der Anwender der digitalen Technologie ein gutes Stück mehr Unabhängigkeit von einem bestimmten Hersteller.

Darüber hinaus ist die Qualität des Bildes je nach Bedarf skalierbar, d.h., daß es nicht in allen Fällen nötig ist, mit der vollen Orts-, Frequenz- oder Farbauflösung zu arbeiten.

Mit einer digitalen Darstellung der Information werden auch alle Methoden der Informatik durch Computer anwendbar. Somit lassen sich Kompressionsverfahren mit und ohne Datenverlust, Datenübertragung über Computernetze, intelligente Unterstützung des Editierprozesses, Information Retrieval auf Videoquellen usw. realisieren.

Ebenso ist man dann in der Lage, neben der reinen Bild- und Toninformation auch andere mit dem Video verknüpfte Information zu speichern und auch wieder synchron abzurufen (z.B. den Text zum Ton ...)

Mittels hierarchischer Übertragungsmodelle ist es möglich, ein Video in mehreren Qualitäts- und/oder Komplexitätsstufen gleichzeitig zu übertragen, der Empfänger kann dann entscheiden, welche Qualitätsstufe er darstellen möchte.

Allgemein bietet die digitale Technik die Möglichkeit, Sender und Empfänger viel weitgehender zu entkoppeln, als das mit analoger Technik möglich wäre.

Die zur Digitalisierung notwendigen Schaltungen sind zunächst komplexer als die entsprechenden analogen Schaltungen. Sind diese Schaltungen jedoch einmal fixiert, können sie integriert, verkleinert und entsprechend „verdichtet“ werden. Es sind somit komplizierte, aufwendige Systeme in überschaubarer Größe und mit hoher Betriebssicherheit möglich.

(vgl. [Fox91], [Hoffm92], [Lipp91], [Liou91], [Münz93], [Wolf93])

Die Anwendungsgebiete

Verfolgt man den Weg des Videosignals einmal vom Empfänger zurück bis zum Studio, ergeben sich auf den verschiedenen Stationen immer wieder gute Gründe, das Signal zu digitalisieren:

Die Elektronenstrahlröhre des Fernsehgerätes benötigt ein analoges Videosignal. Also müßte ein digitales Signal hier auf jeden Fall wieder mittels eines Digital-Analog-Wandlers in ein analoges gewandelt werden. Vor diesem Punkt jedoch kann das Signal durchaus digital sein, da dadurch der Schaltungsaufwand der Fernsehgeräte

integrierbar würde und zudem zuverlässiger und einfacher zu bedienen wäre. Außerdem wäre eine Abstimmung des Signals nicht mehr erforderlich. Darüber hinaus wären Features wie Standbild, mehrere Bilder verschiedener Programme gleichzeitig auf dem Schirm und vieles andere mehr sehr leicht zu realisieren. Nicht zuletzt wird schon heute durch den Einsatz digitaler Bildspeicher das Bild flimmerfrei, wenn es mit doppelter Bildfrequenz auf dem Bildschirm erscheint. In digitale Fernsehgeräte könnten zudem leichter Normwandler integriert werden. Damit könnten sowohl Signale in PAL- als auch im SECAM- und im NTSC-Modus verarbeitet werden.

Vor dem Fernsehgerät liegt die Übertragungsstrecke. Digitale Signale lassen sich weniger stör anfällig übertragen als analoge.

Im Studio ist die Digitaltechnik bereits am weitesten vorgedrungen. Sie ermöglicht eine aufwendige Bildverarbeitung mit Standbildern über Festspeicher, Tricks, Computerbildern und vieles andere mehr – Dinge, die mittlerweile aus den täglichen Sendungen mit unzähligen Variationen von Überblendtechniken geläufig sind.

Die Aufzeichnungstechnik – als letztes Glied in der Kette – bietet gewaltige Vorteile, wenn digitalisierte Signale aufgezeichnet werden. Es entstehen z.B. keine Kopierverluste. Es sind also beliebig viele Kopiegenerationen möglich. Der Störabstand ist unabhängig vom Bandmaterial und Bildjitter, d.h., das Wackeln des Bildes gehört der Vergangenheit an. Nicht zuletzt ist ein genau definiertes Scheiden möglich. Digitale Studioaufzeichnungssysteme sind in verschiedenen Recordertypen realisiert und schon heute im Einsatz, z.B. D-1, D-2, D-3, DCT, Digital Betacam.

(vgl. [Gall91],[Krug93e],[Krug94],[PP93b],[PP93c],[Sijs91])

3. STAND DER TECHNIK

Dieses Kapitel definiert den Begriff des „Videoeditings“ und setzt es in sein Umfeld. Die typischen Arbeitsabläufe eines Videocutters werden analysiert und die konventionellen Schnittsysteme auf ihre Unterstützung dieser Abläufe hin untersucht.

3.1. DIE VIDEOPRODUKTION

Um den Begriff des Videoeditings in sein Umfeld einordnen zu können und eine Abgrenzung zu anderen Phasen der gesamten Videoproduktion zu ermöglichen, sei hier kurz der Prozeß der Produktion eines Videos skizziert.

Unter Videoproduktion wird hier der Vorgang des Planens, Vorbereitens, Ausführens und Aufzeichnens einer Sendung, eines Beitrages, eines Spielfilmes, eines Schulungs- oder Demonstrationsvideos, ... sowie dessen Nachbearbeitung in der sog. Postproduktion verstanden. Dieser Produktionsprozeß läuft in mehreren Phasen ab, die sich jedoch gegenseitig überlappen oder gar miteinander verschmelzen können:

- Preproduktion
- Vorbereitung
- Aufzeichnung
- Nachbearbeitung

Preproduktion

In der Preproduktionsphase ist, wie der Name schon andeutet, noch gar nicht sicher, ob es überhaupt zu einer Produktion kommen wird.

Hier werden Ideen erzeugt und konkretisiert (z.B. Drehbücher, Recherchen, ...) und letztendlich die Entscheidung darüber getroffen, ob die Ideen in eine Produktion umgesetzt werden.

Vorbereitung

Ist dann die Projektierung beschlossen, beginnt die Vorbereitungsphase. Nun werden Mittel und Personal ausgewählt und aquiriert (Studios, Drehorte, Schauspieler, ...), Termine gesetzt – kurz, die gesamte Planung gemacht.

Aufzeichnung

Die Aufzeichnungsphase ist die eigentliche Realisation der Ideen. Die Vorgaben des Drehbuches (Spielfilm) oder die Ergebnisse der Recherchen (Dokumentation) werden filmisch festgehalten. Üblicherweise wird heute als Aufzeichnungsmedium die MAZ (vgl. Kap. 2.3.) verwendet. In Ausnahmen wird auch heute noch auf Film aufgezeichnet (besonders bei großen Kinoproduktionen), aber der Trend geht bereits in vielen Bereichen zu einer direkt digitalen Aufzeichnung auf Speichermedien mit wahlfreiem Zugriff (Festplattenlaufwerke). Dies ist besonders deutlich im Bereich des Nachrichtenwesens zu erkennen, wo „Digital News Gathering“ (DNG) eine nahtlose, volldigitale Verarbeitungskette von Aufzeichnung über Bearbeitung bis zur Sendeabwicklung von der Festplatte anstrebt (vgl. [PP93b]).

Postproduktion

In der Postproduktionsphase werden die Aufzeichnungen ausgewertet, und zum endgültigen Produkt, dem Video, zusammengesetzt. Hier ist der Bereich des Videoeditings angesiedelt, der einen erheblichen Einfluß auf die Gestalt des Endproduktes hat.

Zur Postproduktion gehört aber auch das Durchführen der Werbekampagne bzw. das Kopieren und Verteilen an die Kunden im Falle eines Demonstrations- oder Schulungsvideos (Promotion & Distribution). Letztendlich kann auch die Ausstrahlung über einen TV-Sender, so denn eine solche stattfindet, zur Postproduktion gerechnet werden.

Diese Darstellung des Produktionsprozesses läßt sich natürlich nicht auf Live-Produktionen übertragen, da hier keine Nachbearbeitung in der Postproduktion stattfindet. Es gibt in diesem Sinne auch gar keine Aufzeichnung zur Weiterverarbeitung, es wird ja direkt – live – gesendet. Die Postproduktion beschränkt sich hier meist auf die Archivierung des aufgezeichneten Materials.

Live-Produktionen sollen hier nicht weiter betrachtet werden.

3.2. DIE HERKÖMMLICHE ARBEITSUMGEBUNG

Zunächst sei hier der aktuelle Stand der Technik beschrieben, wie er in den meisten Videostudios heute anzutreffen ist(vgl. [Spie93][Müll92]).

3.2.1. DER ELEKTRONISCHE SCHNITT

Der Unterschied zwischen Film- und Videoproduktionen liegt im Aufzeichnungsmedium. Während die Filmproduktion eben auf Filmnegativmaterial aufzeichnet, findet im Videobereich die MAZ (magnetische Aufzeichnung) (vgl. Kap. 2.3.)ihren Einsatz. Der Trend zur magnetischen Aufzeichnung selbst beim Film läßt sich durch eine Gegenüberstellung der Vor- und Nachteile beider Methoden verständlich machen.

	Vorteile	Nachteile
Film	Archivierbarkeit Einfaches mechanisches Schneiden und Kleben zur Nachbearbeitung möglich Sehr gute Bildqualität	Nicht direkt zu betrachten, muß erst entwickelt werden Schneiden in der Nachbearbeitung zerstört das Originalmaterial. Es hinterläßt neben dem Ergebnis nicht mehr verwendbare Filmstückchen. Jede Szene kann nur einmal verwendet werden . Kopien sind sehr aufwendig.
MAZ	Direkt nach der Aufnahme zu betrachten Evtl. auch wieder direkt zu löschen und wiederzuverwenden Einfach zu kopieren Kopie statt Schnitt, daher bleiben Originale unversehrt erhalten Szenen können mehrfach verwendet werden. Hohe Informationsdichte → Materialeinsparung	Notwendigkeit zur Benutzung komplexer elektronischer Technik bei der Nachbearbeitung Kürzen oder Verlängern eines Videos nur durch Umkopieren möglich Verminderte Bildqualität Archivierbarkeit zweifelhaft

Bei der MAZ erfolgt also kein echter Schnitt des Trägermaterials, sondern eine zeitlich exakt geplante Überspielung, eine Kopie der gewünschten Szene, die nahtlos an die davorliegende angefügt wird. Dieses Vorgehen nennt man „elektronischen Schnitt“. Durch das Kopieren beim Schnitt verschlechtert sich allerdings der Signal-/Rauschabstand, d.h. die Qualität des Bildes.

Ein elektronischer Schnitt erfordert erheblichen steuerungstechnischen Aufwand. Damit später der Übergang von einer Szene zur nächsten sauber und ohne Störungsstreifen wiedergegeben wird, muß die „neue“ Aufzeichnung exakt die Spuren der „alten“ überschreiben. Player und Recorder (vgl. Kap. 3.2.3.) müssen genau im gleichen Tempo laufen, so genau, daß sie die Aufzeichnung des Bildes stets an exakt derselben Stelle des Bildes ausführen.

3.2.2. DAS STUDIO

In einem typischen Videostudio findet sich meist die folgende Konfiguration:

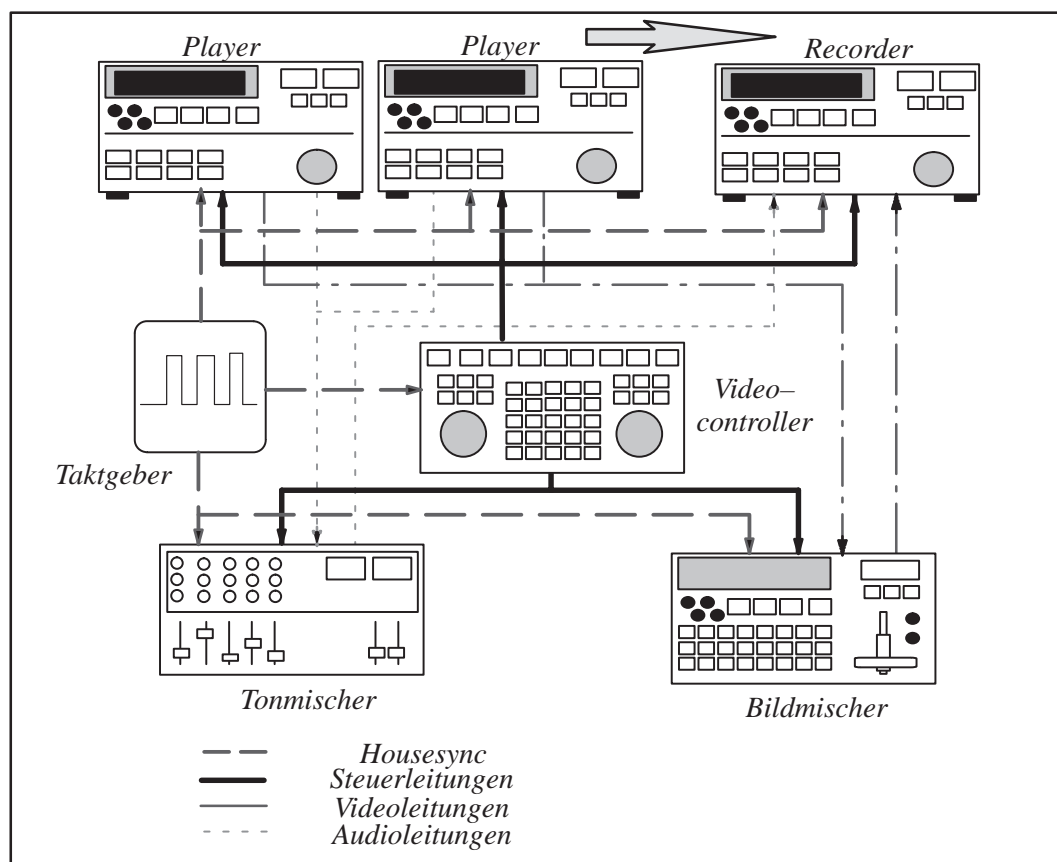


Abb. 3.2.a Die typische Konfiguration eines Videostudios

Es existieren mehrere Zuspielgeräte (*Player*). Sie werden mit dem Quellmaterial, den Rohaufnahmen, geladen und können dann deren Inhalt auf je einem Monitor darstellen und natürlich an das *Bildmischpult* abgeben.

Die Zuspielgeräte sind in der Lage, eine Stelle bildgenau anzufahren und ein einwandfreies Standbild zu liefern. Sie können die Bildfolgen in wählbarer Geschwindigkeit wiedergeben und zwar vorwärts wie rückwärts, dabei ist ihre Laufgeschwindigkeit überaus genau. Sie sind über eine definierte Schnittstelle fernsteuerbar, so daß sie bequem vom *Videocontroller* aus bedient werden können.

Jedes Zuspieldgerät ist also mit einem *Kontrollmonitor* und dem zentralen Videocontroller verbunden. Weiterhin gibt es ein oder evtl. auch mehrere Aufzeichnungsgeräte (*Recorder*). Für sie gilt das gleiche wie für die reinen Abspielgeräte. Zusätzlich müssen sie in der Lage sein, Schnitte bildgenau durchzuführen, d.h., sie können

1. an ein schon vorher aufgezeichnetes Signal eine neue Folge von Bildsignalen so anfügen, daß bei einer späteren Wiedergabe ein glatter Übergang garantiert ist (*assemble cut*).
2. falls eine neue Bildfolge in eine alte, schon auf dem Recorder vorhandene, eingefügt werden soll, so muß auch das letzte Bild der neuen Folge exakt so plaziert werden, daß ein glatter, synchroner Übergang zum nächsten Bild der alten Folge stattfindet (*insert cut*).

Der *Videocontroller* (oder das „Kontrollpult des Videoschnittsystems“) ist mit den Steuerleitungen aller Player und Recorder verbunden und ermöglicht so deren Fernbedienung entweder direkt über Bedienelemente, ähnlich denen an den Geräten selbst, oder indirekt über die Eingabe von sog. Timecodes (eindeutige Bildadressen, (vgl. Kap. 3.2.4.)), die ihn dann veranlassen, ein oder mehrere Zuspieldgeräte „auf diesen Timecode zu fahren“.



Abb. 3.2.b Die BBE 600 Editing Control Unit von BTS, ein typischer Videocontroller

Der Videocontroller ist ebenfalls mit dem Mischpult über eine Steuerleitung verbunden und kann Komandos an dieses absetzen. Das Mischpult erhält an den Eingängen die Videosignale der Zuspieldgeräte und stellt das fertig gemischte Signal am Ausgang dem Rekorder zur Verfügung.

Evtl. ist noch ein *Spezialeffektgerät* vorhanden. Dieses wird ebenfalls durch den Videocontroller gesteuert und kann die Signale vor oder nach dem Mischpult oder in einer Ein-/Ausgangsschleife des Mischpultes verändern.

Alle Geräte sind extern synchronisierbar, d.h., sie können die Bildsignalwiedergabe nach einem externen, nicht im Gerät selbst eingebauten Taktgeber steuern. Erst dieser *zentrale Taktgeber (housesync)* ermöglicht es, daß alle Zuspielgeräte, Rekorder und sonstigen Studiogeräte synchron ihr Signal abgeben. Dies ist absolut notwendig, wenn man z.B. zwei Bildsignale mischen will (vgl. Kap. 3.2.1.).

Die Verbindungen der Geräte untereinander lassen sich somit in drei Kategorien unterteilen:

- Übertragung der Bildsignale.
Die Videosignale können unterschiedlichen Formats sein. Innerhalb eines Studios wird jedoch meist eine homogene Umgebung vorgezogen, um Qualitätsverluste durch die sonst notwendigen Signalwandlungen zu vermeiden. Verwendet werden hier normalerweise die getrennten Signale der einzelnen Komponenten des Gesamtvideosignales: YUV (vgl. Kap. 4.4.4.1.).
- Übertragung des Housesync-Signales.
- Übertragung der Steuersignale.
Hier gibt es sowohl serielle wie parallele, standardisierte oder herstellerspezifische Schnittstellen und Protokolle. Es scheint sich jedoch ein Trend zu den Standards der in der Computerindustrie üblichen Schnittstellen durchzusetzen. So besitzen die meisten der neueren Geräte fast aller Hersteller einen RS422 oder RS232 Anschluß.

(vgl. [Adob93], [Avid], [Fast93], [Fels93b], [Gebh94], [Krug93d], [Krug94], [Müll92], [PP93a-d])

3.2.3. PLAYER UND RECORDER

Da der Arbeitsablauf des Editierens eines Videos wesentlich von den Fähigkeiten der Ausrüstung und hier besonders der Player und Recorder abhängt, wird hier noch einmal detaillierter auf die verschiedenen Arten von Wiedergabe- und Aufzeichnungsgeräten eingegangen.

3.2.3.1. Bandmaschinen

Bandmaschinen sind in den Studios heute die Standardausrüstung. Mit Bandmaschinen bezeichnet man heute in der Regel Player und Recorder mit Magnetband-Kassetensystemen, ähnlich denen, wie sie aus dem Heimvideobereich bekannt sind. Wirkliche Bandmaschinen deren Magnetband auf lose Spulen gewickelt wird, sind fast nicht mehr in Verwendung.

Das verwendete Bandmaterial sowie die Aufzeichnungstechniken können je nach Hersteller und Gerätetyp sehr unterschiedlich sein. So gibt es z.B. 1“, 1/2“ und 1/4“ breite Bänder mit verschiedenen Magnetisierungsparametern und Aufzeichnungsqualitäten. In der Regel wird innerhalb eines Studios jedoch in einer möglichst homogenen Umgebung gearbeitet, um Formatkonvertierungen zu vermeiden und Geräteinkompatibilitäten verschiedener Hersteller aus dem Weg zu gehen.

Diese Bandmaschinen sind Präzisionsgeräte und dementsprechend teuer. Sie erfüllen alle schon genannten Bedingungen zu Ansteuerbarkeit, Laufgenauigkeit und Synchronisierbarkeit.

Obwohl die Bandmaschinen den Stand der aktuellen Technik darstellen und eigentlich *die* Player und Recorder sind, besitzen sie einige gravierende Nachteile:

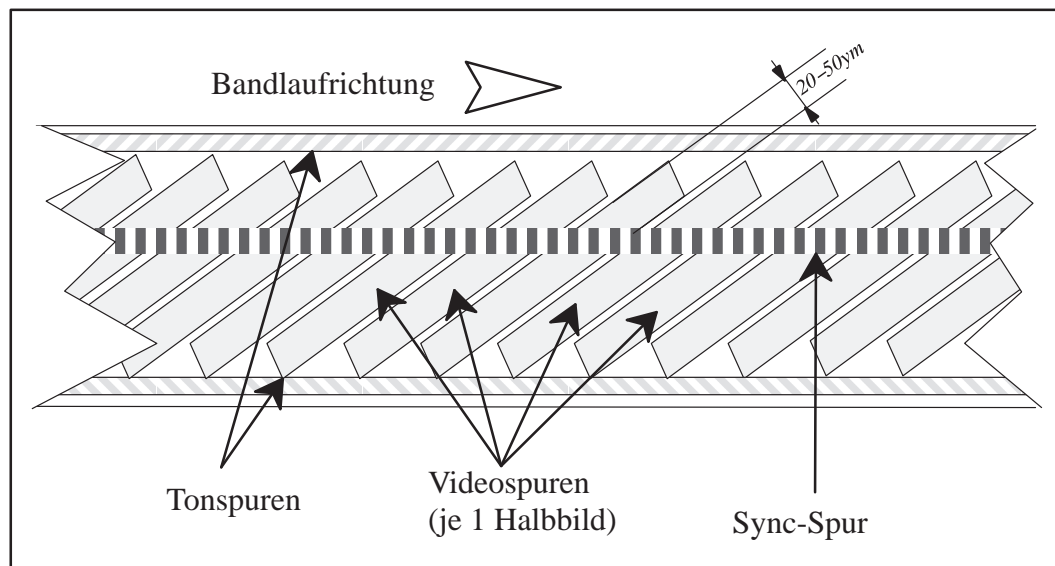


Abb. 3.2.c Das Schrägsपुरaufzeichnungsverfahren für Magnetbänder

Die Verwendung von Magnetbändern zur Aufzeichnung des Videosignales erlaubt kein echtes Schneiden und Kleben des Bandes mehr, wie es aus dem Filmschnitt und auch aus dem Tonschnitt bekannt ist. Die hohe Bandbreite des Videosignales macht eine Aufzeichnung im Schrägsपुरverfahren notwendig (vgl. Abb. 3.2.c) mit Spurbreiten im Bereich von 20 - 50 μm . Es ist klar, daß hier ein mechanisches Schneiden und Verkleben des Bandmaterials nicht ohne deutlich sichtbare Störungen bei der späteren Wiedergabe möglich ist.

Um eine konstante Bandgeschwindigkeit und Synchronität mit dem Housesync zu gewährleisten, muß vor jedem Schnitt ein sog. Preroll durchgeführt werden, d.h. Player und Recorder starten 10 Sekunden vor dem eigentlichen Schnittpunkt und synchronisieren ihre Bandgeschwindigkeit und ihr Videosignal. Zusätzlich muß der Recorder auch nach dem Schnittende noch min. 2-3 Sekunden weiter aufzeichnen, um den Anschluß eines weiteren Schnittes zu ermöglichen.

Bandmaschinen erlauben keinen „wahlfreien Zugriff“ (Random Access) auf das Material, sondern lediglich einen streng linearen. Um auf eine bestimmte Stelle zugreifen zu können, muß das Band erst an diese Stelle umgespult werden. Da das Magnetband nur eine begrenzte Belastbarkeit aufweist, ist eine Höchstgeschwindigkeit vom ca. 32-Fachen der Normalgeschwindigkeit nicht zu überschreiten.

Der streng lineare Zugriffscharakter erlaubt somit auch nur das „flüssige“ Abspielen von Sequenzen in der Reihenfolge, in der sie aufgezeichnet wurden. D.h., ein Ändern

der Abspielreihenfolge von verschiedenen Szenen erfordert vorher ein erneutes Aufzeichnen in dieser Reihenfolge, da die Umspulzeiten sonst eine stetige Wiedergabe verhindern würden.

Dies alles gilt ebenso für die immer weiter verbreiteten digital aufzeichnenden Bandmaschinen, die entweder direkt digitale Videosignale verwenden oder analoge zunächst digitalisieren. Einige dieser digitalen Bandmaschinentypen betreiben Datenkompression (vgl. Kap. 4.2., Kap. 4.3.) vor der Aufzeichnung, so daß auch hier eine Verschlechterung der Bildqualität über mehrere Kopiergenerationen zu beobachten ist.

(vgl. [Gebh94], [Krug93e], [Krug94], [Müll92], [Münz93], [PP93a-c], [Schu72], [Spie93])

3.2.3.2. Bildplatten

Bildplatten wurden schon 1970 in einem der ersten MAZ-Schnittsysteme überhaupt von CMX verwendet. Die Platten hatten damals eine Kapazität von ca. 5 fünf Minuten Film (vgl. [Müll92]).

Heutige Bildplatten arbeiten auf optischer oder magnetooptischer Basis und sind wiederbeschreibbar. Sie bieten den großen Vorteil des wahlfreien Zugriffs auf jedes einzelne Bild und ermöglichen so im nonlinearen Editing ein schnelles und effektives Arbeiten.

Dennoch ist auch heute eine echte wahlfreie Wiedergabereihenfolge mit Bildplatten nicht zu realisieren:

Als Beispiel sei hier der Pioneer Video-Disk-Recorder VDR 1000p genannt, ein in Videostudios durchaus gebräuchliches Gerät. Hier kostet eine Bildplatte in der Anschaffung ca. 2300,- DM und faßt ca. 32 min. Film. Die Zugriffszeit liegt beim nonlinearen Zugriff bei ca. 0,3 Sekunden, das entspricht 7,5 Bildern nach PAL-Rate und ist damit für wahlfreie Wiedergabereihenfolge nicht geeignet. Zudem sind sowohl die Geräte als auch das Aufzeichnungsmaterial wesentlich teurer als bei der Verwendung von Bändern.

(vgl. [Krug93d], [Krug94], [Müll92], [Spie93])

3.2.3.3. Speicher

Die Verwendung digitaler RAM-Speicher (Random Access Memory) zum „Aufzeichnen“ des Videomaterials zum Zweck des Editierens bietet außer dem wahlfreien Zugriff auch den Vorteil einer sehr schnellen Zugriffszeit, da hier keine mechanischen Bewegungen innerhalb des Aufzeichnungsgerätes erforderlich sind. somit ist im Prinzip auch ein Abspielen in beliebiger Reihenfolge gewährleistet. RAM-Systeme sind damit für die Verwendung im nonlinearen Editing prädestiniert.

Die Nachteile liegen in der Notwendigkeit zur Digitalisierung des Materials, was immer auch einen gewissen Qualitätsverlust mit sich bringt, sowie in der geringen Kapazität des Aufzeichnungsmediums. Die anfallenden großen Datenmengen (ca. 28.9 MByte/sec, (vgl. Kap. 4.2.)) erlauben selbst bei den heute möglichen RAM-Ka-

pazitäten, nur wenige Sekunden hochqualitativen Videos gleichzeitig im Speicher zu halten. Daher wird eine Kompression der Datenmenge unumgänglich. Da angesichts der nötigen Kompressionsraten von 10:1 und mehr verlustfreie Kompression auf beliebigem Bildmaterial nicht mehr realisierbar ist, kommen RAM-basierte Systeme zur Zeit nur als Simulationssysteme für den Offline-Schnitt in Betracht.

(vgl. [Gebh94], [Krug93a], [Krug94], [Müll92], [Ohan93], [PP93d], [Spie93], [Voig94])

3.2.4. TIMECODE

Jeder Rekorder zeichnet zusätzlich zur Bildinformation noch einen sog. Timecode (TC) mit auf. Dieser Timecode ist die Nummer oder „Adresse“ eines jeden Bildes und ist durch die SMPTE und EBU genormt (vgl. [TC86]). Es handelt sich dabei um ein 80 Bit breites Informationswort, das zusammen mit jedem Bild aufgezeichnet wird. Dieses Informationswort setzt sich zusammen aus einer achtstelligen Bildnummerierung

SS:MM:SS:BB

für Stunden, Minuten, Sekunden und Bild der Sekunde,

die während der Aufzeichnung durch einen sogenannten „Timecodegenerator“ fortlaufend erzeugt und mit dem Videosignal verkoppelt wird, und einer ebenfalls achtstelligen Benutzerinformation, den „User-Bits“, die frei belegt werden können, z.B.

01:24:12:93



Band 01, 24.12.1993

Der Timecode ist daher auch schon bei der Erstellung des Quellmaterials mit aufgenommen worden. Somit lassen sich über die Angabe des Timecodes ganz einfach bestimmte Stellen des Materials referenzieren:

- ein einzelnes Bild über seinen Timecode,
- eine Sequenz über die Timecodes des ersten und des letzten Bildes.

Für einen Cutter ist der Timecode ein Mittel, mit dem Bilder exakt lokalisiert sowie Bild- und Tonelemente genau synchronisiert werden können.

Auch eine Synchronisation verschiedener Quellaufnahmen ist mit Hilfe des Timecodes möglich. Wird z.B. eine Sendung mit mehreren Kameras aufgezeichnet, so werden alle Rekorder mit einem einheitlichen Timecode versorgt, der die wirkliche Uhrzeit der Aufzeichnung wiedergibt. Der Zugriff ist somit hinterher im Schneiderraum direkt über die Aufzeichnungszeit möglich. Zudem können alle verschiedenen Kamerapositionen parallel und synchron betrachtet werden, indem man die verschiedenen Zuspielderäte über den aufgezeichneten Timecode synchronisiert. Dadurch kann z.B. nach der geeignetsten Kameraperspektive oder dem besten Umschnitt gesucht werden.

Der Zugriff über den Timecode stellt einen gewaltigen Fortschritt gegenüber dem manuellen Filmschnitt dar. Hiermit ist es zum erstenmal möglich geworden, eine zu-

sätzliche Information zusammen, online, mit der Bildinformation zu speichern. In der Tat ermöglichte der Timecode erst die Automatisierung des Videoschnittes.

Allerdings hat die Verwendung des Timecodes auch ihre Grenzen.

1. Auf dem Quellmaterial eines Players darf keine Timecodeadresse zwei- oder mehrfach erscheinen. Das Schnittsystem wüßte sonst nicht, welche Stelle gemeint ist, wenn eine solche Adresse referenziert würde.

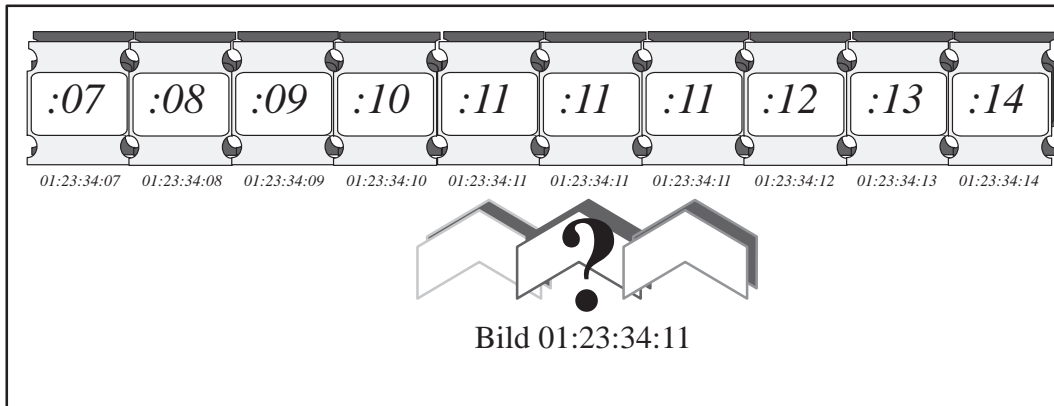


Abb. 3.2.d Probleme bei doppelten Timecodes

2. Der Timecode auf dem Material eines Geräts muß stetig sein, d.h., er darf keine Lücken oder Wertsprünge aufweisen. Das könnte bei der Suche nach einer Adresse zu einem Deadlock des Schnittsystems führen.

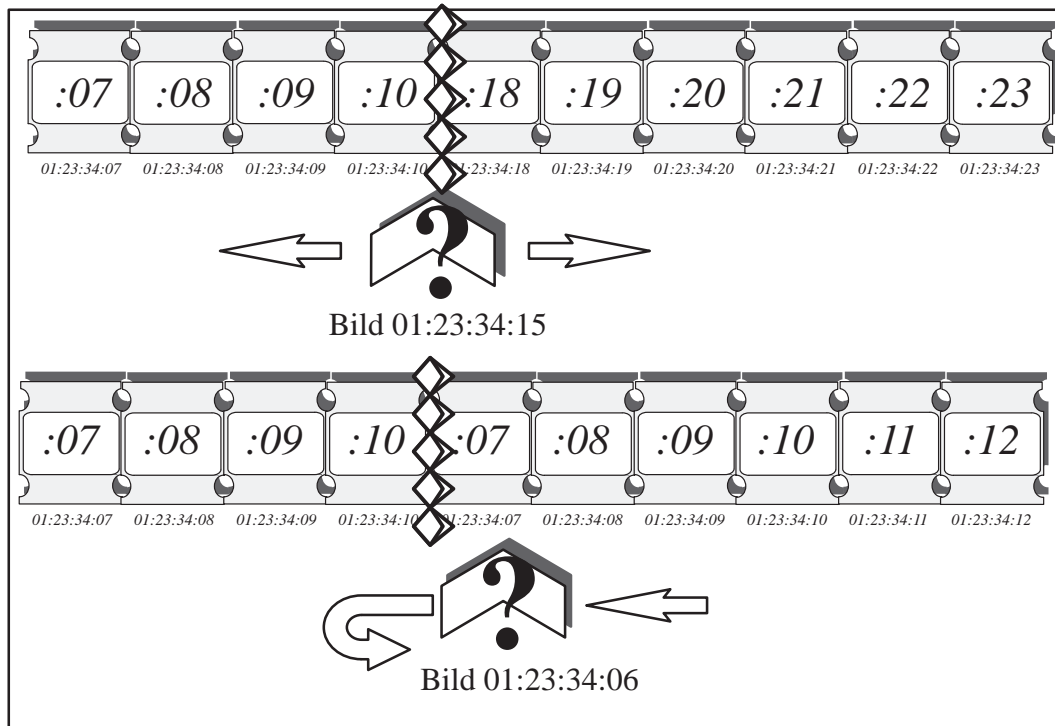


Abb. 3.2.e Probleme bei unstetigem Timecode

3. Alle Player und Recorder müssen einen (aufgezeichneten) Timecode lesen (bzw. schreiben) können, da sonst eine Synchronisation und Referenz nicht möglich ist. Das bedeutet insbesondere, daß das Speichermedium eines Recorders vorher quasi „formatiert“ werden muß, in dem es mit einem TC versehen wird, um jede mögliche Adresse direkt ansprechen zu können.
4. Der Timecode liefert keinerlei Information über den Inhalt des Videos an einer bestimmten Stelle, d.h., er hat keinen Kontextbezug

Der Timecode spielt also eine zentrale Rolle innerhalb des heutigen Videoeditierszenarios. Er ist *die* Referenz, die der Videocontroller, der im Zentrum der ganzen Konfiguration steht, benutzt, um die Geräte anzusteuern (vgl. [Adob93], [Fast93], [Müll93]).

3.3. VIDEOEDITING

Unter Videoediting soll hier der Teil der Postproduktion verstanden werden, der im allgemeinen unter der Bezeichnung Filmschnitt oder einfach unter Schnitt bekannt ist. Allerdings ist diese Bezeichnung nicht ganz korrekt, denn die Aufgabe des Cutters (so die Berufsbezeichnung, von engl.: to cut = schneiden) ist weit umfangreicher und kreativer, als sie suggeriert:

” Der eigentliche Film- und Videoschnitt besteht im Ordnen, Kürzen und Zusammenfügen ausgewählter Einzelteile zum Gesamtprodukt, dem Film oder der Dokumentation. Im nicht-szenischen Bereich (Dokumentation, Feature, Magazinbeitrag, Aktuelles) wählt die Cutterin nach vorgegebener Konzeption und eigenen Vorstellungen Bild- und Tonmaterial aus, und verdichtet es so entsprechend der redaktionellen Konzeption, im szenischen Bereich (Fernsehspiel oder Film) entsprechend den Vorstellungen der Regie.” [FILM94]

Es gibt also grundsätzlich zwei verschiedene Aspekte der Arbeit eines Videocutters:

- den technischen Aspekt,
den handwerklichen Umgang mit Material und Maschinen
- den kreativen Aspekt,
das „Schreiben“ einer Videogeschichte mit vorhandenem Material

3.3.1. DER ARBEITSPROZEß

3.3.1.1. Generell

Detaillierter läßt sich der typische Arbeitsprozeß des Editierens eines Videos wie folgt darstellen:

Nachdem die vorangegangenen Produktionsphasen von der Planung bis zur Aufzeichnung abgeschlossen sind, muß eine (oft sehr umfangreiche) Menge von Quellaufnahmen durchgesehen und bewertet werden. Regisseur und/oder Cutter machen sich Notizen, um ein späteres Wiederauffinden interessanter Szenen zu erleichtern. Danach werden Szenen ausgewählt und Editierentscheidungen getroffen. Die gewählten Szenen müssen zur Definition des Schnittes wiedergefunden werden oder, falls keine Notizen über eine passende Szene existieren, das Quellmaterial muß nach einer geeigneten Szene durchsucht werden, indem es vollständig angeschaut wird.

Der Schnitt wird definiert. Dazu sucht sich der Cutter die nächste Szene durch Spulen und Betrachten aus dem Quellmaterial aus, definiert die Ein- und Ausstiegspunkte bzw. die Länge der gewünschten Szene sowie die Umschnittart von der letzten zur aktuellen Szene. Evtl. muß er noch die Spezialeffekte definieren, die verwendet werden sollen.

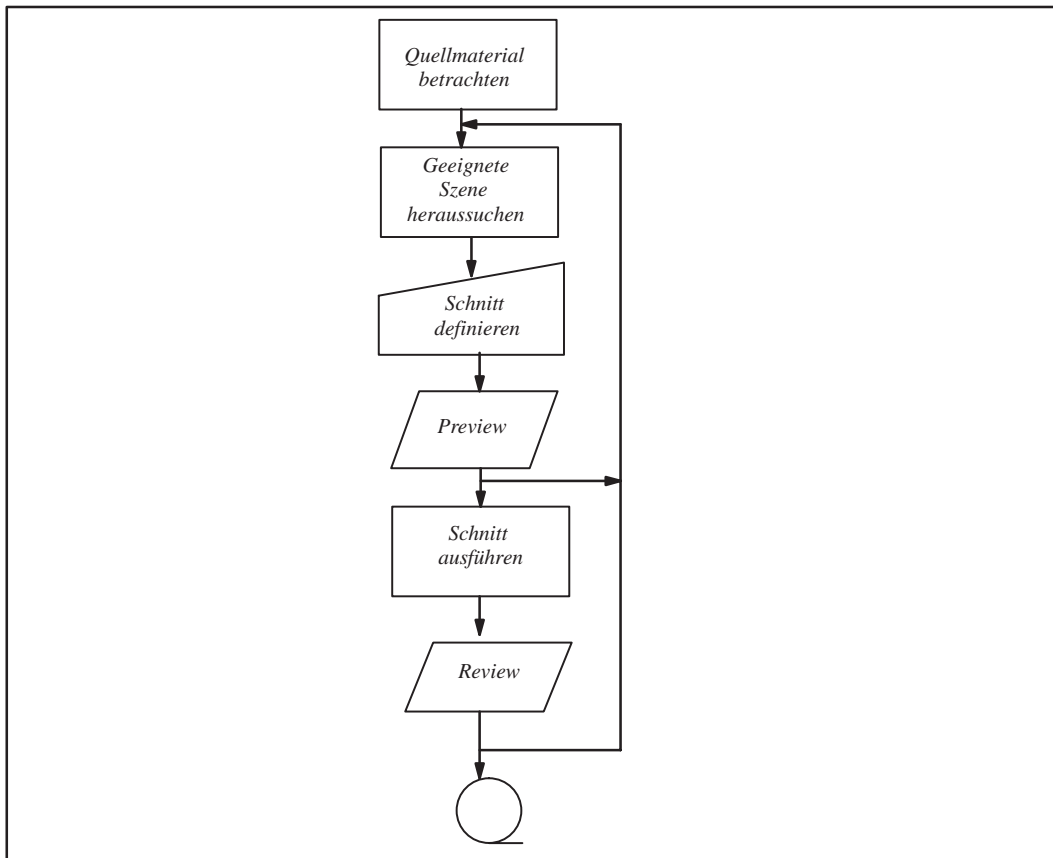


Abb. 3.3.a Der generelle Arbeitsablauf des Videoediting

Der Schnitt wird nun evtl. in einem „Preview“ kontrolliert und, falls er nicht den Vorstellungen des Cutters/Regisseurs entsprach, korrigiert und erneut betrachtet.

Ist das Ergebnis dann zufriedenstellend, wird die Aufzeichnung des Schnittes durchgeführt und danach noch einmal im „Review“ kontrolliert.

Danach wird der nächste Schnitt definiert und durchgeführt.

Dieser Ablauf ist in Abb. 3.3.a noch einmal als Diagramm zusammengefaßt.

Oft wird in einem ersten Arbeitsgang nur eine „Rohschnitt“-Version erstellt, die den an der Produktion beteiligten Personen zur Bewertung und Korrektur vorgelegt wird. Erst in einem zweiten, dritten oder weiteren Arbeitsgang wird die entgültige Version erstellt. Oder es werden gleichzeitig verschiedene Versionen desselben Videos geschnitten, z.B. eine Kino- und eine TV-Version oder eine kurze und eine ausführliche Fassung. Die bei der Erstellung einer Version getroffenen Entscheidungen bezüglich Szenenauswahl und Schnitt müssen also nachvollziehbar bleiben und die Schnitte selbst reproduzierbar sein.

Diese allgemeine Beschreibung des Arbeitsablaufes beim Schneiden eines Videos läßt sich weiter spezialisieren in zwei Produktionsstile, denen sich fast alle Produktionen unterordnen lassen, der dramaturgische und der dokumentarische Stil (vgl. [Dave89]).

3.3.1.2. Dramaturgische Produktionen

Als dramaturgisch sind die erzählenden Produktionen im Sinne des Theaters und des Schauspiels zu verstehen. Dramatische Produktionen lassen sich an Hand folgender Punkte abgrenzen:

Die Produktion richtet sich nach einem klaren und detailliert ausgearbeiteten Drehbuch, in dem alle Szenen schon vor der Aufnahme beschrieben sind. Ebenso definiert es die Abfolge der Szenen im Endvideo.

Das Quellmaterial besteht aus den jeweils mehrfachen Aufzeichnungen der im Drehbuch beschriebenen Szenen, aus denen nun die beste Version ausgesucht wird. Jede Szenenaufzeichnung ist mit einer Nummer und/oder einem Namen versehen, die selbst auf das Bildmaterial aufgezeichnet wurden. Es gibt eine genaue Buchführung über die Position der Szenen auf dem Quellband sowie über zusätzliche Daten zu den einzelnen Aufzeichnungen (etwa Länge, erste Bewertungen noch vom Drehort, Angaben zu Drehort und -umständen, ...).

Die Aufgabe des Editierens liegt hier in der „Verschönerung“ oder Perfektionierung der zu erzählenden Geschichte und ihrer Dramatik. Es gilt die beste Aufnahme einer Szene auszuwählen und sie in der im Drehbuch geforderten Reihenfolge anzuordnen. Dabei sind die einzelnen Entscheidungen weitgehend unabhängig voneinander, da jede Szene für sich beurteilt werden kann.

Dieser Editierstil ermöglicht ein *streng lineares Vorgehen* beim Aufbau der Zielsequenz, da die Reihenfolge der Szenen ja schon von vornherein bekannt ist. Zudem wird stets nur ein *sehr kleiner Teil des Quellmaterials zur Bearbeitung* benötigt, oft nur die verschiedenen Aufnahmen einer Szene. *Der Zugriff auf das Quellmaterial kann genau spezifiziert werden*, da umfangreiche Daten darüber vorhanden sind.

3.3.1.3. Dokumentarische Produktionen

Unter diesen Stil fallen z.B. die klassischen Tierberichte oder auch aktuelle Reportagen. Die dokumentarische Produktion kennzeichnet sich durch folgende Eigenheiten:

Es existiert oft bis zur Zeit des Schnittes noch kein fertiges Skript oder Storyboard. Der Produzent/Regisseur hat zwar schon während der Aufnahmen Vorstellungen über den Aufbau des Videos entwickelt, aber die „Geschichte“ entsteht erst beim Durchsehen und Editieren des Materials.

Jede Szene des Quellmaterials ist einzigartig, da die Aufnahmen nicht gestellt waren. Es existiert also normalerweise keine Szene doppelt. Dies muß man bei der Szenenauswahl für den Schnitt beachten. Um später beim Editieren das Wiederfinden bzw. das Suchen nach geeigneten Szenen überhaupt erst zu ermöglichen, müssen sehr genaue Notizen gemacht werden, da es meist keine definierbare Reihenfolge oder Anordnung der Szenen im Rohmaterial gibt. Die Szenen wurden so aufgezeichnet, wie sie sich gerade ergaben.

Die Aufgabe des Editierens besteht also darin, ein Arrangement von einzigartigen Szenen zu erstellen und dabei eine Geschichte aufzubauen. Dabei kann eine Ent-

scheidung über die Verwendung und Platzierung einer wesentlichen Szene dutzende anderer Entscheidungen beeinflussen oder revidieren.

Das Editieren ist hier also ein *hochdynamischer, iterativer und sicherlich nichtlinearer* Prozeß, der oftmals die *Revision getroffener Entscheidungen* verlangt und der stets *das volle Spektrum des Quellmaterials zur Bearbeitung* benötigt. Der Zugriff auf das Quellmaterial erfolgt oft über inhaltliche Spezifikationen.

3.3.2. LINEAR EDITING

Jede Editiertechnik, die ein lineares Aufbauen des Zielbandes erlaubt wird als „Linear Editing“ bezeichnet. Wie oben beschrieben ist dies im Bereich der Spielfilmproduktion oft der Fall.

Ein linearer Editierstil kann aber auch durch die Fähigkeiten der verwendeten Aufzeichnungs- und Wiedergabegeräte erzwungen werden.

Besonders bei der Verwendung von Bandmaschinen ist nur lineares Editing möglich (vgl. Kap. 3.2.3.1.):

Das Zielband kann nur linear, also Szene für Szene in der gewünschten Reihenfolge aufgebaut werden. Während das Auslassen einer Szene zum späteren Ausfüllen durch Freilassen einer entsprechend langen Stelle auf dem Zielband noch möglich ist (wenn man weiß, wie lange diese Szene sein soll), ist das nachträgliche Umsortieren des Zielbandes oder gar das Einfügen oder Löschen einer neuen Szene nicht möglich. Liegen die Quellszenen ungünstig auf dem Quellband, ist zwischen dem Aufzeichnen jeden Schnittes ein langes Spulen der Quellbandes nötig, da es keinen wahlfreien, sondern nur einen sequentiellen, linearen Zugriff auf die Szenen erlaubt.

In der Praxis erweist es sich, daß das Spulen und das Durchführen des Schnittes wesentlich mehr Zeit in Anspruch nimmt als das Treffen der Editierentscheidungen selbst.

3.3.3. NONLINEAR EDITING

Als Nonlinear Editing bezeichnet man eine Editiertechnik, die ein Erstellen und Ändern der Zielsequenz in beliebiger Reihenfolge und mit wahlfreiem Zugriff auf das Quellmaterial erfordert.

Die Produktion von Dokumentarfilmen ist oft auf die Möglichkeit zum Nonlinear Editing angewiesen. Besteht diese Möglichkeit nicht, verlängern sich die Produktionszeiten im Postproduktionsbereich erheblich.

Nonlinear Editing ist bekannt aus dem Filmschnitt, wo es ja jederzeit möglich ist, an jeder beliebigen Stelle des Filmes eine Szene einzufügen oder zu entfernen, da hier quasi das „Band“ in Form des Negativmaterials wirklich „geschnitten“ und „eingeklebt“ wird. Das ist aufgrund des magnetischen Aufzeichnungsverfahrens nicht mehr möglich (vgl. Kap. 3.2.3.1.). Man muß daher versuchen, einen wahlfreien Zugriff auf jedes einzelne Bild zu ermöglichen und die Zugriffszeit so kurz zu halten, daß es möglich ist, die Bilder in beliebiger Reihenfolge abzuspielen. Dieser Ansatz wird in vereinzelt Systemen durch den Einsatz von Laserbildplatten (read only)

oder magnetooptischen, also wiederbeschreibbaren Bildplatten realisiert. Diese sind jedoch als Speichermedien extrem teuer, und die Zugriffszeiten lassen einen wirklich wahlfreien Zugriff auch nicht zu (vgl. Kap. 3.2.3.2.).

Eine mögliche Lösung ist daher, den Computer mit seinem RAM (Random Access Memory) als Bildspeichermedium einzusetzen (vgl. Kap. 3.2.3.3.). Dieser Ansatz unterliegt aufgrund der relativ geringen Bildspeicherkapazität natürlich gewissen Einschränkungen, ist aber durch den Einsatz entsprechender Kompressionsverfahren dennoch erfolgversprechend.

3.4. DIE HERKÖMMLICHEN SYSTEME

Die heutigen Systeme entstanden als Computererweiterung des Videocontrollers, um die Schnittlistenverwaltung und -editierung zu vereinfachen. Sie sind zu unterscheiden in mehreren Kategorien:

Zunächst unterscheidet man zwischen Online- und Offline-Schnittsystemen, die sich in der Art des Ergebnisses eines Editierprozesses unterscheiden.

Dann gibt es analoge und digitale Systeme in Bezug auf die Verarbeitung und die Aufzeichnung der Videosignale.

Ihre Korrelation läßt sich in Form einer Matrix beschreiben:

	digital	analog
online	don	aon
offline	doff	aoff

Während in der Vergangenheit digitale Systeme mit nonlinearem Offline-Betrieb gleichgesetzt wurden, findet zur Zeit eine weitgehende Umstellung auf digitale Online-Schnittsysteme statt, die die Vorteile der Nonlinearität mit dem sofortigen Schnittergebnis des Online-Editings verbinden können.

(vgl. [AVID], [Gebh94], [Fels93b], [Krug93a], [Krug93b], [Krug93e], [Krug94], [PP93d], [Voig94])

3.4.1. ONLINE EDITING

Der Online-Schnitt entspricht dem Vorgehen des generellen Arbeitsprozesses (vgl. Kap. 3.3.1.1.)

Dabei wird der Schnitt direkt ausgeführt, d.h., das *Ergebnis des Editierens ist eine fertige Aufzeichnung* des Filmes.

Der Cutter kann dazu einen „Preview“ des von ihm gewählten Schnittes ansehen, indem der Schnitt durchgeführt und auf einem Kontrollmonitor dargestellt wird, ohne daß er auf dem Rekorder aufgezeichnet wird. Dazu werden alle beteiligten Zuspielderäte sowie der Recorder vom Videocontroller auf die jeweiligen Schnitteinstiegspositionen gespult und dann gestartet. Man sieht die vorhergehende, schon auf dem Recorder aufgezeichnete Szene. Am Schnitteinstiegs punkt wird der Umschnitt auf die neue Szene durchgeführt (der Rekorder zeichnet aber nicht auf) und diese angezeigt.

Entsprach der Schnitt nicht den Vorstellungen des Bedieners, muß er die Definitionen ändern und erneut „previewen“.

Ist das Ergebnis dann zufriedenstellend, wird die echte Aufzeichnung des Schnittes durchgeführt. Sie läuft genau so ab, wie der Preview, nur daß der Recorder diesmal auch aufzeichnet.

Nachdem der Schnitt durchgeführt wurde, wird nun im Review die Aufzeichnung noch einmal überprüft, d.h., der Rekorder spielt die Aufzeichnung noch einmal ab.

Danach wird der nächste Schnitt definiert und durchgeführt.

Setzt man einmal die Benutzung der traditionellen Bandmaschinen voraus, ist im Online-Schnitt also nur ein lineares Editing möglich!

3.4.2. OFFLINE EDITING

Das Offline-Editing stellt die Realisation einer nonlinearen Editiermöglichkeit dar. Allerdings werden hier nicht direkt die Bildsequenzen bearbeitet, sondern nur die Schnittanweisungen. Man schneidet quasi „virtuell“.

Im Offline-Editing werden die von Cutter definierten Schnittanweisungen in einer sog. Edit Decision List (EDL) festgehalten. Diese EDL beschreibt am Ende Schnitt für Schnitt den Aufbau des Filmes aus dem Quellmaterial. Sie kann, wenn sie vollständig ist, an ein automatisches Schnittsystem übertragen werden, das dann selbständig die Player und Recorder sowie das restliche Studioequipment ansteuert und die Schnitte ausführt.

Der Vorteile einer EDL:

- Man hat die Möglichkeit, während ihrer Erstellung jederzeit früher getroffenen Entscheidungen ändern oder verwerfen zu können.
- Zudem hat man stets einen Überblick über alle bis zu einem Zeitpunkt getroffenen Einstellungen.
- Man ist nicht unbedingt auf die Verfügbarkeit eines großen (und teuren) Videostudios angewiesen. Zur Szenenauswahl genügt ein sogenanntes „Simulations-System“, das weniger hochwertige Geräte enthält, die evtl. nicht einmal den Timecode lesen können. Dieser wird dann auf der Arbeitskopie ins Bild eingeblendet. Oft reicht sogar ein einzelner Player. Man kann dann die EDL per Hand erstellen und später ins Schnittsystem eingeben. Kennt man das Quellmaterial gut genug, ist nicht einmal der Player nötig.
- Man spart beim Editieren die Zeit ein, die sonst zur Durchführung der Aufzeichnung nötig wäre.
- Es ist sehr viel schneller möglich, verschiedene Schnittversionen zu erstellen.
- Der Schnitt ist jederzeit reproduzierbar (solange das Quellmaterial vorhanden ist) und auch editierbar.

Die Nachteile der EDL sind jedoch auch nicht zu vernachlässigen:

- Benutzt man nicht mehrere Player, ist kein Preview des gedachten Schnittes möglich. Der Cutter ist auf seine Vorstellungskraft und Erfahrung angewiesen.
- Es ist ebenfalls nicht möglich, bereits definierte Sequenzen im Review zu betrachten, da ja keine Aufzeichnung gemacht wurde.
- Weiterhin ist während oder nach der Ausführung des Schnittes noch einiges an Korrekturarbeit an der Erstversion der Schnittliste zu erbringen, um das Ergebnis zu optimieren.

Die fertige EDL wird dann meist in elektronischer Form gespeichert und so zum Videocontroller übertragen. Der Videocontroller fordert nun alle benötigten Quellbänder an und arbeitet die Schnittliste Schritt für Schritt ab. Auch wenn zwischendurch ein Bandwechsel nötig werden sollte, falls nicht genügend Zuspielderäte für alle Bänder zur Verfügung stehen, kann er dies in der Regel handhaben.

Da die Länge aller ausgewählten Szenen bekannt ist, besteht nun die Möglichkeit, eine Schnittfolgenoptimierung durchzuführen. D.h., der Videocontroller kann auf dem „formatierten“ Zielband den nötigen Platz freilassen und zuerst eine andere, spätere Szene aufzeichnen, falls diese auf dem Quellband günstiger zu erreichen ist.

3.4.3. ANALOGE SYSTEME

Analoge Systeme verarbeiten und zeichnen analoge Signale auf, d.h. Frequenzen und Amplituden von Spannungen (vgl. Kap. 2.3.). Alle ihre Komponenten werden in Hardware erstellt und sind stark spezialisierte Geräte. Die Hersteller dieser Technologie betreiben (noch) eine Strategie der „Black Boxes“, von denen nichts über ihre genaue Funktionsweise bekannt ist. Oft entsprechen auch die „Schnittstellen“ dieser Black Boxes zur Signalein- und -ausgabe sowie Steuerung nur herstellerspezifischen Normen und halten den Benutzer damit in einer starken Herstellerabhängigkeit.

Aus diesem Grund sind Änderungen oder Verbesserungen der Konfiguration eines Studios in der Regel nur mit großem Aufwand (HW-Tausch) und hohen Investitionskosten zu realisieren. Dennoch handelt es sich bei der analogen Technik um ausgereifte und zuverlässige Lösungen.

3.4.4. DIGITALE SYSTEME

Digitale Systeme verarbeiten und zeichnen digitale Codierungen der Bildinformation auf, d.h. Impulsfolgen (vgl. Kap. 2.4.).

Ihre Komponenten bestehen aus Softwarepaketen, die auf Standard-HW-Plattformen lauffähig sind. Damit liegen die Investitionskosten sowie der Aufwand für eine Erweiterung oder Verbesserung des Systems deutlich unter denen einer analogen Anlage. Sie geben zudem eine größere Unabhängigkeit vom Hersteller, da dieser sich an die Standardschnittstellen der Rechner halten muß. Alle volldigitalen Systeme sind in der Lage, auch mit analogen Signalen zu arbeiten. Diese werden dazu erst digitalisiert.

Digitale Schnitttechnik ist in analoger Technik zum Teil gar nicht zu realisieren. So werden z.B. auch in analogen Umgebungen digitale Videoeffektgeräte (DVE) für den Bildschnitt eingesetzt. Eine Umsetzung der hier möglichen Bild- und Überblendtricks wäre – wenn überhaupt – nur mit einem enormen Aufwand an analoger Filtertechnik und Signalverarbeitung zu erreichen.

3.4.5. UNTERSTÜTZUNG DURCH HERKÖMMLICHE SYSTEME

Wie unterstützen nun die heute im Gebrauch befindlichen Schnittsysteme den Cutter/Regisseur in diesem Arbeitsprozeß?

Der Videocontroller erlaubt eine einfache Handhabung des Bildmaterials. Sämtliche Bandmaschinen sind von einem zentralen Punkt aus steuerbar. Die Systeme erlauben sogar eine gewisse Abstraktion von Begriffen wie Band und Kassette, was den Zugriff auf das Bildmaterial betrifft. Sind die Kassetten mit ihrem Timecodebereich erst einmal vom System erfaßt worden, genügt die Angabe des Timecodes, um eine bestimmte Aufzeichnungstelle anzufahren. Das System kann selbständig entscheiden, auf welcher Bandmaschine es suchen muß und in welcher Richtung.

Bei der Verwendung einer EDL ist es möglich, mehrere Versionen eines Schnittes oder gar einer ganzen Schnittfolge zu definieren und evtl. auch aufzuzeichnen. Eine solche EDL gibt dem Regisseur auch die Möglichkeit, sich schnell über die bisher verwendeten Umschnitte und Spezialeffekte zu informieren, um den nächsten Schnitt diesem Stil entsprechend definieren zu können.

Die Speicherung des Timecodes auf dem Magnetband entspricht einer Online-Speicherung von Zusatzinformationen zum eigentlichen Bildinhalt. Er ermöglicht eine bildgenaue Adressierung und Ansteuerung des Bildmaterials und ermöglicht außerdem die Automatisierung der eigentlichen Schnittaufzeichnung sowie deren Reproduzierbarkeit. Er erleichtert die Synchronisation verschiedener Videoquellen und ist stets zusammen mit der Bildinformation verfügbar, d.h., er braucht nicht über einen weiteren Datenträger oder Kanal übertragen zu werden.

Die heutigen Systeme unterliegen aber auch schwerwiegenden Einschränkungen:

Verfügbarkeit

Um ein Video schneiden zu können, muß ein Schnittstudio zur Verfügung stehen. Diese sind meistens angemietet, und somit sind die Schnittzeiten ein beträchtlicher Kostenfaktor. Ein verteiltes Schnittsystem, das die Bearbeitung und Erstellung einer Schnittliste auch außerhalb des Studios zu beliebigen Zeiten unterstützt und dann im Schnittraum selbst eine optimale Nachbearbeitung und Ausführung dieser Schnittliste erlaubt, wäre hier sicherlich eine Verbesserung.

Besonders im Bereich der Dokumentationen ist es oft der Fall, daß an der Produktion maßgeblich Personen beteiligt sind, die von den Möglichkeiten und Einschränkungen der Videotechnik bestenfalls eine rudimentäre Kenntnis besitzen (z.B. Auftraggeber, Wiss. Berater, ...). Ihnen fällt es auf Grund mangelnder oder gar nicht vorhandener Erfahrung sehr schwer, sich eine Vorstellung vom Endprodukt zu machen. Daher ist oft eine lange Phase des Experimentierens nötig, in der viele Schnittversio-

nen entstehen, bevor ein zufriedenstellendes Schnittergebnis erzielt werden kann. Dies ist mit den heutigen, im Prinzip immer linearen Editiersystemen nur sehr langsam zu bewerkstelligen, da immer erst eine komplette Schnittfolge aufgezeichnet werden muß (mit allen Wartezeiten, die durch Umspulen, Auswählen, Preview, Pre-roll, Aufzeichnen, Kontrolle entstehen), ehe eine Bewertung des Ergebnisses vorgenommen werden kann.

Sourcenauswahl

Eines der größten Probleme stellt heute noch die Auswahl geeigneter Szenen des Quellmaterials dar.

Die einzigen Informationen, die man bei diesem Vorgang zur Verfügung hat, sind das Wissen der beteiligten Personen über das Quellmaterial und ihre Notizen, die beim Durchsehen des Materials entstanden sind. Als Referenz dient hier einzig und allein die Angabe der Materialträgerbezeichnung (z.B. Kassetten Nr.) und des Timecodes.



Abb. 3.4.a Typische Situation im Schnittstudio: Schnittsystem und Informationsverwaltung finden getrennt statt [Pott94]).

Die auf dem Markt vorhandenen Systeme bieten so gut wie keine Unterstützung bei der Suche nach Bildmaterial an. Fast ausschließlich wird hier der Timecode als Referenz verwendet. Weitere Informationen lassen sich nicht auf dem Band zusammen mit dem Bildinhalt speichern. Der Regisseur und der Cutter sind auf Informationen und Daten auf externen Datenträgern angewiesen (Stichworte zur Szene, Kommentare, Namen, Drehorte, ... im PC oder auf dem Notizblock), die meist nicht ins System eingebracht werden können, um den Zugriff zu erleichtern.

Der Zugriff auf das Bildmaterial ist der Natur des Datenträgers Magnetband entsprechend nur sequentiell möglich, d.h., daß das Band erst umgespult werden muß, um auf eine andere Szene zugreifen zu können. Ebenso ist auch das Zielband dem linearen Zugriff unterworfen, also ist z.B. ein Umsortieren oder Verschieben bereits auf-

gezeichneter Szenen nur durch ein wirklich physikalisches Umsortieren der Speicherbereiche möglich. Ein wahlfreier Zugriff, der die Zugriffszeiten drastisch senken würde und zudem eine beliebige Reihenfolge beim Abspielen einmal aufgezeichneter Szenen erlaubte, ist z.Zt. nur äußerst selten realisiert, indem man statt Bandmaschinen, optomagnetische Platten verwendet.

3.5. ANFORDERUNGEN AN EINEN VERBESSERTEN VIDEOARBEITSPLATZ

Speicherung von Zusatzinformation auf dem Bilddatenträger.

Mit einer Speicherung von zusätzlicher (benutzerdefinierter) Information auf dem Bilddatenträger wäre gewährleistet, daß diese Information automatisch mit der Bildinformation und synchron zu dieser vorhanden wäre. Vorstellbar sind hier

- Szenennamen oder -bezeichnungen
- Szenenbeschreibungen
- Stichworte zur Suche
- Kommentare
- Daten zu Drehort und Drehumständen
- Angaben über die Kopiengeneration
- Angaben zu Eigentums- und Kopierrechten
- Überschriften über Teilsequenzen des Videos (Gliederung)

Verwendung von Zusatzinformationen zur Szenensuche und -auswahl.

Neben den online gespeicherten Informationen könnten zur Einbeziehung weiterer Informationen zusätzlich zum Timecode Bildanalysemethoden bei der Szenensuche und -auswahl angewandt werden, um bildinhaltsbezogene Zugriffe zu ermöglichen. Auch externe Informationen sollte das System berücksichtigen können und eine Schnittstelle für deren Eingabe anbieten.

Insgesamt wäre es dann möglich, Anfragen an das Schnittsystem zu stellen wie an eine Datenbank. Szenen ließen sich nach Name, Datum, oder Stichwort zur Betrachtung auswählen. Von einem fertigen Video könnte der Betrachter aufgrund einer mitgegebenen Gliederung speziell nur die Einleitung oder aber die Schlußfolgerung betrachten.

Wahlfreier Zugriff

Wichtig ist ebenfalls die Realisation eines wahlfreien Zugriffs auf das Bildmaterial um den kreativen Prozeß des Editierens stärker in den Vordergrund zu stellen und mit möglichst kurzen „Antwortzeiten“ des Systems besser zu unterstützen. Idealerweise läge die Zugriffs- und Verarbeitungszeit weit unter der Entscheidungsfindung des Benutzers, so wie es etwa heute schon im Bereich der Textverarbeitung der Fall ist. Nur mit Hilfe eines wahlfreien Zugriffes läßt sich echtes Nonlinear Editing in die Wirklichkeit umsetzen.

Ortsungebundenheit

Die Verwendung eines Videoeditiersystems sollte nicht an den Ort des Schneidestudios gebunden sein. Besonders im Bereich des Offline Editings ist der sehr hohe Qualitätsstandard der in einem solchen Studio geboten wird, oft nicht nötig.

4. MPEG

Dieses Kapitel beschreibt die grundlegenden Methoden der Videocodierung nach ISO/IEC 11172, die als Grundlage des im Kapitel 5 konzipierten Schnittsystems dient.

4.1. WARUM MPEG?

In dieser Arbeit werden besonders zwei Aspekte der Nutzung von MPEG betrachtet:

- Editieren von MPEG-Videos ist schwierig und muß grundsätzlich erst noch ermöglicht werden.
- MPEG bietet die Möglichkeit, zusätzliche Informationen zusammen und synchron mit den Video/Audio-Daten zu codieren. Die Definition und Codierung derartiger Metainformationen bleibt der jeweiligen Anwendung überlassen.

Der MPEG Standard wurde aus mehreren Gründen als Basisdatenformat des Systems ausgewählt:

1. Eine komprimierte Speicherung der Videodaten ist notwendig, um ein Arbeiten auf einem durchschnittlichen Rechner auch ohne besondere Speicherplatzanforderungen zu ermöglichen. Dies ist durch die Verwendung des MPEG-Standards gegeben(vgl. [Ohan93]).
 2. MPEG ist ein ISO/IEC Standard, d.h., er ist weltweit gültig. Zudem ist er in Zusammenarbeit mit der Industrie entstanden und daher weitgehend durch diese akzeptiert. Es ist zu erwarten, daß sich in vielen Bereichen, besonders der Unterhaltungsindustrie eine Anwendung von MPEG-Verfahren ergibt, wie sich schon bei Philips abzeichnet, die den MPEG-Standard in ihr bereits vorhandenes CD-I Format übernahmen, und nun Hard- (MPEG-CD-Player) und Software (Filme, Spiele, Lernprogramme) dafür anbieten(vgl. [Sijs91]).
 3. Der MPEG-Standard hat schon eine weite Verbreitung innerhalb der internationalen Computerdatennetze erfahren, da er z.B. als Datenübertragungsformat für Videodaten innerhalb des WWW-Netzes¹ etabliert wurde. Allein schon für dieses Anwendungsspektrum der Verteilung von Videoinformation über die Netzwerke ist die Entwicklung eines Editors nötig.
 4. Inzwischen ist auch klar, daß der nachfolgende und erweiterte MPEG-2 Standard für die Übertragung von öffentlichen Fernsehsendungen genutzt werden soll. Auch hier müssen erst noch einige Aspekte der in dieser Arbeit besprochenen Probleme gelöst werden, ehe man eine solche Nutzung realisieren kann. Von Hewlett Packard wurde inzwischen ein MPEG-Media-Server entwickelt, der als MPEG-Video-Datenserver dienen soll und bis zu 50 Stunden Videomaterial verwalten kann(vgl. [Krug94], [Lipp91], [PP94], [Rowe92]).
 5. Im MPEG-Standard ist bereits die Speicherung und Verarbeitung von Timecode vorgesehen (vgl. [MPEG93]).
1. WWW bedeutet World Wide Web und ist eine vom Kernforschungszentrum Cern ange-regte Initiative, die es erlaubt, weltweit beliebige Dokumente innerhalb des Internet zu referenzieren und zu übertragen. Dabei wurden auch multimediale Dokumente bedacht, unter anderem auch Dateien im MPEG-Videoformat.

4.2. GRÜNDE FÜR DATENKOMPRESSION

Wie schon mehrfach angesprochen, wird in der digitalen Videowelt häufig Datenkompression verwendet (vgl. Kap. 2.4., Kap. 3.2.3., Kap. 3.4.4.). An dieser Stelle soll kurz darauf eingegangen werden, warum eine solche Kompression nötig ist.

Geht man von den z.Zt. gültigen Vorgaben in [CCIR 601] für die digitale Beschreibung der TV-Standards (hier besonders PAL) aus, ergeben sich folgende Rahmendaten:

Bildauflösung:

$$625 \text{ Zeilen} * 720 \text{ Pixel}$$

wovon aber nur

$$576 \text{ Zeilen} * 702 \text{ Pixel}$$

auch Bildinformation tragen.

Farbauflösung:

Bei einer *Farbvollauflösung* in RGB oder YUV (vgl. [CCIR601], Kap. 4.4.4.1.), bilden

$$3 \text{ Byte} = 24 \text{ Bit ein Pixel.}$$

Selbst bei Anwendung der in [CCIR 601] definierten *Chromaunterabtastung* im Verhältnis Y:U:V = 4:1:1 erhält man noch 8 Bit Y und je 2 Bit Cr/Cb, also

$$12 \text{ Bit pro Pixel.}$$

Bildrate:

$$25 \text{ Bilder/sec}$$

Somit ergibt sich eine Pixelauflösung von

$$P_B = 576 \frac{\text{Zeilen}}{\text{Bild}} * 702 \frac{\text{Pixel}}{\text{Zeile}} = 404352 \frac{\text{Pixel}}{\text{Bild}} \quad (\text{Gl. 4.1})$$

$$P_t = P_B * 25 \frac{\text{Bilder}}{\text{sec}} = 10108800 \frac{\text{Pixel}}{\text{sec}} \quad (\text{Gl. 4.2})$$

Und somit eine Datenmenge von

$$D_V = P_t * 3 \frac{\text{Byte}}{\text{Pixel}} = 30326400 \frac{\text{Byte}}{\text{sec}} = 28,9 \frac{\text{MByte}}{\text{sec}} = 231,3 \frac{\text{MBit}}{\text{sec}} \quad (\text{Gl. 4.3})$$

$$D_{4:1:1} = P_t * 1,5 \frac{\text{Byte}}{\text{Pixel}} = 15163200 \frac{\text{Byte}}{\text{sec}} = 14,5 \frac{\text{Mbyte}}{\text{sec}} = 115,7 \frac{\text{MBit}}{\text{sec}} \quad (\text{Gl. 4.4})$$

wobei:

P_B = Anzahl der Pixel pro Bild

P_t = Anzahl der Pixel pro Sekunde

D_V = Bilddatenmenge pro Sekunde Video bei Farbvollauflösung

$D_{4:1:1}$ = Bilddatenmenge pro Sekunde Video bei Chromaunterabtastung

Damit ergibt sich selbst bei der schon farbreduzierten CCIR 4:1:1 Darstellung für z.B. 5 Minuten Video ein Speicherplatzbedarf von

$$D_{4:1:1} * 300\text{sec} = 4,24\text{GByte} \quad (\text{Gl. 4.5})$$

Dies liegt an der Obergrenze der z.Zt. verfügbaren Kapazitäten, zumal zur reinen Bildinformation normalerweise noch zwei Tonkanäle (CD-Qualität: 48KHz x 16 Bit = 750 KBit/sec pro Kanal) mit aufgezeichnet werden müssen.

Ganz ähnlich verhält es sich bei den notwendigen Datenübertragungsraten. Betrachtet man vorhandene Datenübertragungskanäle (s.u.) wird klar, daß auch hier die vorhandenen Kapazitäten nicht ausreichen, um hochqualitatives Video zu übertragen.

Anforderung:	231,3 MBit/sec Vollauflösung bzw. 115,7 MBit/sec CCIR 601 4:1:1
VBN	Maximal 140 MBit/sec
Ethernet	ca. 10 MBit/sec
Tokenring	ca. 16 MBit/sec
Winchester Festplatten Zugriffe	8-18 MBit/sec
CD-ROM Zugriffe	ca. 1,5 MBit/sec

(vgl. [Roed91], [Kröm91])

Will man also digitale Videoverarbeitung im Bereich der Computer- und Telekommunikationstechnologie ermöglichen, müssen die Videodaten komprimiert werden.

Werden z.B. CD-ROM- oder DAT-Laufwerke als Massenspeicher eingesetzt, darf die Gesamtdatenrate (incl. Audio-, Verwaltungs- und Synchronisationsdaten) nicht über 1.5 MBit/sec steigen. Setzt man pro komprimierten Audiokanal 128 KBits/sec und für Verwaltungs- und Synchronisationsdaten ca. 120 KBits/sec (vgl. [Gall91]), bleiben den reinen Videodaten ca. 1,15 MBit/sec. Somit beträgt die notwendige Videokompressionsrate

$$\frac{1,15 \frac{\text{MBit}}{\text{sec}}}{115,7 \frac{\text{MBit}}{\text{sec}}} = 1 : 100 \quad (\text{Gl. 4.6})$$

wobei eine möglichst hohe Bildqualität erhalten werden soll. Dies wird in MPEG realisiert.

(vgl. [Ohan93], [MPEG93])

4.3. VERFAHREN ZUR BITRATENREDUKTION

Die Menge der Zeichen in einer Nachricht läßt sich in vier Bereiche unterteilen:

- den irrelevanten (unwichtigen)
- den relevanten (wichtigen)
- den redundanten (bekannten)
- den nichtredundanten (unbekannten)

Die Grundidee der Datenreduktion besteht nun darin, die Irrelevanz und die Redundanz, also die unwichtigen und die schon bekannten Informationen aus dem Informationsfluß zu eliminieren.

4.3.1. IRRELEVANZREDUKTION

Die Irrelevanzreduktion entfernt aus dem Signal die Information, die vom Empfänger nicht wahrgenommen werden kann und damit überflüssig (irrelevant) ist. Sie wird in der Videotechnik schon immer angewandt:

So ist das Spektrum des Videosignals begrenzt auf 5 MHz, 575 sichtbare Zeilen pro Bild und 25 Bilder/sec.

Da das Auge empfindlicher auf Helligkeitsunterschiede reagiert als auf Farbunterschiede, wird die Farbinformation mit einer geringeren Bandbreite übertragen als die Helligkeit.

In der digitalen Videotechnik muß die Abtastfrequenz (somit die Anzahl der Abtastwerte) und die Quantisierungsstufenzahl (die Genauigkeit der Abtastwerte) festgelegt werden. Auch damit wird Irrelevanz reduziert.

Aus der Tontechnik ist bekannt, daß ein lauter Ton einen leisen Ton verdeckt. Damit braucht der leise Ton nicht mehr aufgezeichnet oder übertragen zu werden.

Das Kriterium dafür, in welchem Maße Irrelevanzreduktion angewendet werden darf, liegt beim Empfänger. In der Fernsehtechnik ist dies das menschliche Auge und Ohr. Informationen, die von unserem Auge oder Ohr nicht wahrgenommen werden, brauchen nicht übertragen zu werden. Bezogen auf die heutige Fernsehtechnik stellt man leicht fest, daß sich eine Bildvorlage von dem auf dem Fernsehschirm dargestellten Bild in der wiedergegebenen Qualität doch noch erheblich unterscheidet. Dies bedeutet nichts anderes, als daß die Irrelevanzreduktion in einem unzulässig hohen Maß angewendet wird.

Die Irrelevanzreduktion ist ein irreversibler und verlustbehafteter Prozeß, da Information, die im Original vorhanden ist, durch die Reduktion verloren geht und sich damit das decodierte Bild vom Original unterscheidet.

4.3.2. REDUNDANZREDUKTION

Die Redundanzreduktion unterscheidet sich von der Irrelevanzreduktion dadurch, daß sie ein umkehrbarer und nicht verlustbehafteter Prozeß ist. Sie reduziert nicht die

im Signal enthaltene Information, sondern lediglich die zu übertragende Datenmenge. Das gelingt dadurch, daß eine Information, die der Empfänger bereits kennt, nicht noch einmal zu übertragen werden braucht.

Beispiel:

Ist ein stehendes Bild zu übertragen, so genügt es, dieses Bild nur einmal zu übertragen. Es braucht nicht 25 mal pro Sekunde wiederholt zu werden. Erst wenn das Bild sich ändert, muß das dem Empfänger zusammen mit der Änderung mitgeteilt werden. Ändert sich das Bild nicht in seiner Gesamtheit, brauchen natürlich nur die relevanten Bildteile als Änderung übertragen zu werden.

Zur Redundanzreduktion gibt es viele unterschiedliche Verfahren. Die wichtigsten sollen hier kurz beschrieben werden:

4.3.2.1. DPCM

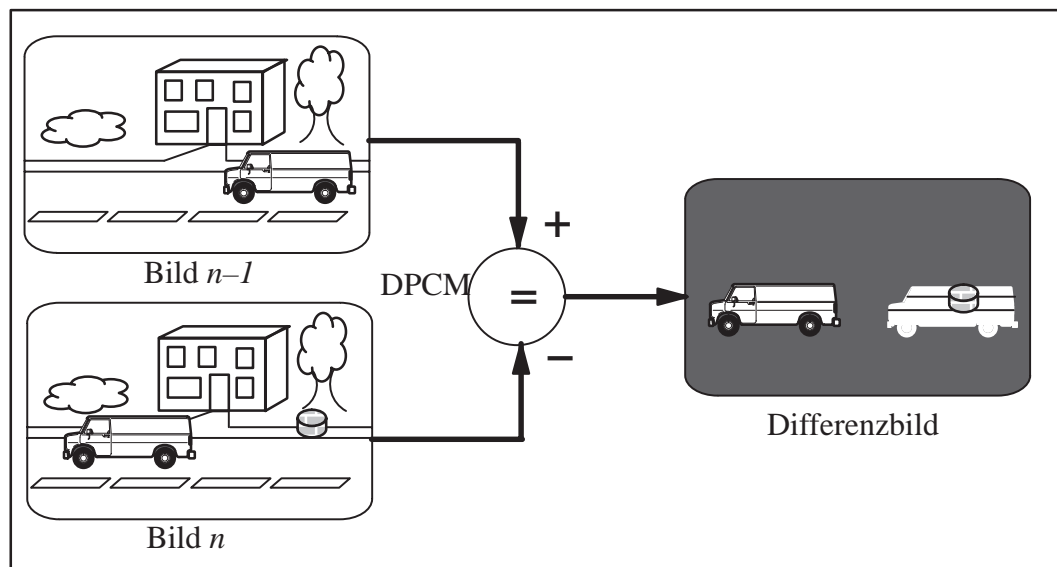


Abb. 4.3.a Die DPCM im Differenzbildverfahren

Das DPCM-Verfahren (Differenz Puls-Code Modulation) bildet die Differenz zwischen aufeinanderfolgenden Abtastwerten (z.B. Bildern). In Abb. 4.3.a wird eine Szene mit einer feststehenden Kamera aufgezeichnet. In der Szene steht ein Haus, neben dem sich rechts und links jeweils ein Baum und ein Strauch befinden. Davor fährt auf einer Straße ein Auto vorüber. Da die Kamera feststeht, ist das Auto die einzige Bewegung im Bild. Bildet man nun die Differenz zwischen zwei aufeinanderfolgenden Bildern, so erhält man als Ergebnis das Differenzbild, in dem das Haus mit Baum, Strauch und Straße nicht mehr enthalten ist. Lediglich an den Stellen, an denen sich das Auto befindet, entsteht relevante Information. Die zu übertragende Datenmenge ist damit erheblich geringer, als wenn beide Bilder komplett zu übertragen wären. Dieses Verfahren setzt im Sender zur Differenzbildung einen Bildspeicher voraus. Natürlich muß die Differenzbildung im Empfänger durch eine gegenläufige Maßnahme wieder rückgängig gemacht werden.

4.3.2.2. Transformationscodierung

Um einen 5kHz Ton einer bestimmten Amplitude zu übertragen, bedarf es eines Kanals mit einer Bandbreite von mindestens 5 kHz. Eine andere Methode, den Ton zu übertragen, besteht darin, ihn zu beschreiben und nur die Beschreibung zu übertragen anstatt den Ton selbst.

Dabei transformiert man den Ton vom Zeitbereich (analoges Tonsignal) in den Frequenzbereich und überträgt lediglich die spektralen Koeffizienten (in diesem Fall ein Zeichen für die Frequenz 5kHz mit bestimmter Amplitude).

Für Bilder funktioniert diese Transformation ähnlich, mit dem Unterschied, daß eine Bildvorlage nicht eindimensional ist, sondern zweidimensional. Damit gibt es spektrale Koeffizienten in horizontaler, vertikaler und allen diagonalen Richtungen, die die jeweiligen Änderungen in dieser Richtung repräsentieren. Mit der Transformationscodierung wird also nicht das Bild selbst übertragen, sondern seine spektralen Komponenten. In Abb. 4.3.b ist eine Koeffizientenmatrix aufgezeichnet. Der Koeffizient links oben repräsentiert den Gleichanteil, die Koeffizienten daneben stellen die Änderungen in horizontaler Richtung dar und die darunter die vertikalen Frequenzkomponenten. Dazwischen liegen die diagonalen Frequenzkoeffizienten. In der Regel unterzieht man nicht das Bild in seiner Gesamtheit der Transformationscodierung, sondern immer nur einzelne Blöcke mit einer Blockgröße von meist 8x8 Bildpunkten.

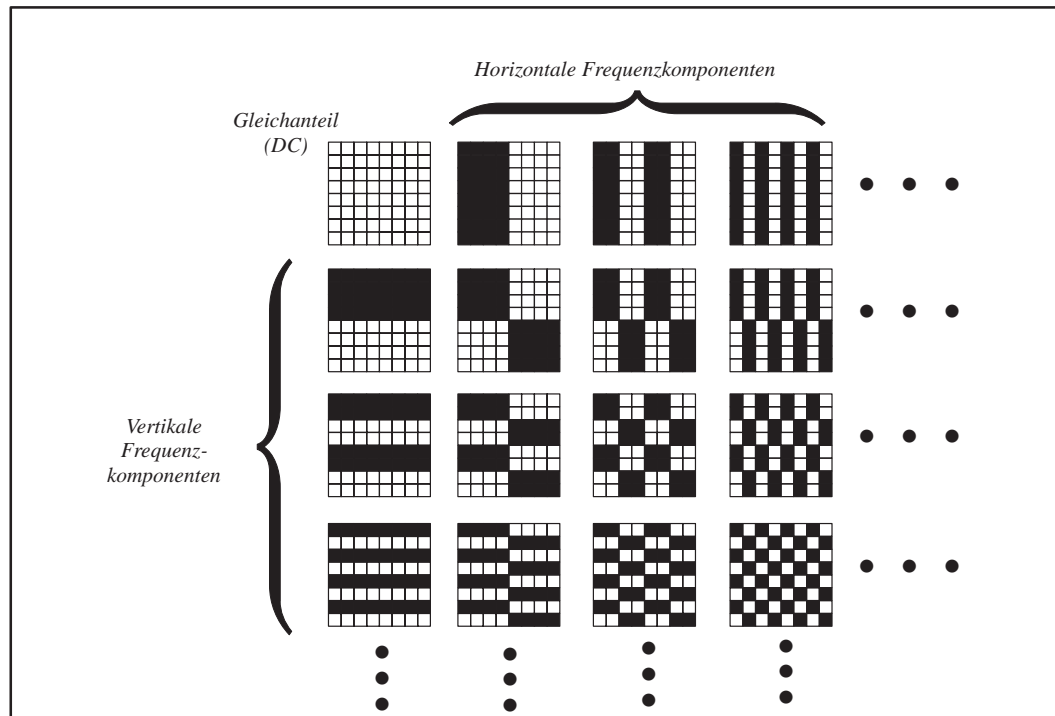


Abb. 4.3.b Die Bedeutung der Transformationskoeffizienten

(vgl. [DCT90], [Lege84], [Vett84])

4.3.2.3. Entropiecodierung

Der Informationsgehalt eines Zeichens läßt sich mit der Wahrscheinlichkeit, mit der das Zeichen auftritt, beschreiben. So liefert ein Zeichen, das immer wieder ausgewählt wird, keine neue Information. Dagegen ist der Informationsgehalt eines Zeichens, das nur sehr selten auftritt, entsprechend groß. Untersucht man die Buchstaben eines Textes auf die Häufigkeit ihres Auftretens hin, so wird man feststellen, daß die unterschiedlichen Buchstaben unterschiedlich oft auftreten. Sollen die Buchstaben in Form eines Dualcodes übertragen werden, ist es im Hinblick auf kleine Datenmengen sinnvoll, häufig auftretende Buchstaben mit einem kurzen Codewort zu belegen und selten auftretende mit einem langen. Das heißt die Codewortlänge richtet sich nach der Wahrscheinlichkeit, mit der das Zeichen (Buchstabe) auftritt (Einzelwahrscheinlichkeit). Dieses Verfahren hat schon Morse bei dem nach ihm benannten Alphabet angewandt. Bezogen auf die DPCM wird man feststellen, daß kleine Abweichungen vom Altwert häufiger auftreten als große, und diese dann auch mit kurzen Bitkombinationen codieren.

Bei unserem Buchstabenbeispiel wird man es nicht allein bei der Einzelwahrscheinlichkeit belassen, sondern untersuchen, wie groß die Wahrscheinlichkeit ist, daß nach einem Buchstaben ein bestimmter anderer folgt (Verbundwahrscheinlichkeit). Die Wahrscheinlichkeit, daß nach einem „q“ in der deutschen Sprache ein „u“ folgt, ist nahe 100 Prozent. Gemäß der Devise, was der Empfänger schon weiß, braucht nicht übertragen zu werden, kann man sich die Übertragung des „u“ sparen. Lediglich der selten auftretende Fall, daß kein „u“ folgt, wäre eine wichtige Information.

(vgl. [Huff53])

Natürlich ist es erlaubt, verschiedene Datenreduktionsverfahren hintereinander anzuwenden. So kann auf das Differenzbild der DPCM die Transformationscodierung angewandt werden und die Koeffizienten mittels Entropiecodierung auf optimale Codewortlänge gebracht werden.

Die hier beschriebenen Ideen und Verfahren sind alle seit langem bekannt. Während jedoch noch vor wenigen Jahren für die Bewältigung der Bildcodieralgorithmen stundenlange Simulationen und abenteuerliche Versuchsaufbauten erforderlich waren, sind sie heute durch den Fortschritt in der Mikroelektronik relativ preisgünstig anzuwenden und entsprechend weit verbreitet. In der Zukunft werden sicherlich neue Algorithmen gefunden und die bestehenden Verfahren verbessert werden, so daß mit wesentlich größeren Kompressionsfaktoren als heute zu rechnen ist.

(vgl. [Enc88], [Foley90], [Gall91], [Hilg93], [Hung93], [Kröm91], [Lipp91])

4.4. INTERNATIONAL STANDARD ISO/IEC 11172

”Information technology - Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s”

MPEG steht als Akronym für die Expertengruppe, die den Standard schuf (Moving Pictures Expert Group) und gleichzeitig als geläufiges Kürzel für den Standard selbst.

Der „MPEG-Standard“ definiert Methoden zur (verlustbehafteten) Bild- und Tondatenkompression, die eine Balancierung zwischen Kompressionsrate und Reproduktionsqualität ermöglichen. Er bietet einen Mechanismus zur Synchronisation der verschiedenen Bild- und Tondatenströme untereinander sowie ein Buffering-Management für die Datenübertragung und -decodierung (bei const. Datenrate).

4.4.1. HISTORIE

Die „MPEG“ wurde 1988 als ISO-IEC/JTC1/SC2/WG11 gegründet mit dem Ziel, innerhalb von zwei Jahren den Entwurf eines internationalen Standards zur Kompression und digitalen Speicherung von Video- und dazugehörigen Audiodaten zu erarbeiten.

Als Rahmenbedingungen waren vorgegeben:

- Ergebnisse innerhalb von zwei Jahren
- Leichte Realisierbarkeit in Hardware (VLSI)
- Datenrate von 1,5 Mbit/s bei annehmbarer Qualität
- Weitgehende Kompatibilität zu anderen, schon vorhandenen Standards

Die Rahmenbedingungen im Einzelnen:

Wollte man einen Standard schaffen, der allgemein akzeptiert wurde, mußte man sich den Entwicklungszeiten der Industrie anpassen, da schon zu dieser Zeit innerhalb der Unterhaltungs- sowie der Computerindustrie ein großer Bedarf nach Normen zur Speicherung und Verarbeitung digitaler Videoaufzeichnungen bestand. Eine lange Entwicklungszeit hätte einer „Internationalen Norm“ kaum eine Chance gegeben, sich gegen parallel entstehende Industrie- und Defactonormen durchzusetzen.

Eine hardwarenahe Lösung war notwendig, da bei den in der Videotechnik anfallenden hohen Datenraten eine (De-) Codierung in Echtzeit in Software nicht realisierbar erschien. Zudem waren die Entwicklungen in der VLSI-Chip-Technik soweit fortgeschritten, daß nun Video-Codec-Chips (Codec $\hat{=}$ Coder und Decoder) zu realistischen Preisen produziert werden konnten. So wurde denn auch schon nach der Veröffentlichung des ersten „Committee Draft“ 1990 mit der Erstellung erster MPEG-Chips begonnen.

Die Zieldatenrate von 1.5 Mbit/s für die codierten Video- und Audiodaten zusammen ergab sich aus der Anforderung, als digitale Speichermedien alle derzeit üblichen Geräte zuzulassen – z.B. CD-ROM, DAT, Bandlaufwerke, Winchestermagnetplatten – und auch bei der Übertragung möglichst viele Kanäle zu berücksichtigen, also z.B. ISDN und LAN.

In Anbetracht des Zeitdruckes und mit Blick auf die beteiligten Institutionen wurden eine Reihe von älteren Standards in die Realisierung des MPEG-Standards mit einbezogen, namentlich:

- JPEG: Joint Photographic Experts Group
ISO/IEC 10918
Ein Standard zur Kompression und Speicherung von stehenden Bildern.(vgl. [JPEG92])
- CCITT Recommendation H.261
Ein Standard zur Übertragung von Bewegtbildern über ISDN-Verbindungen mit Kapazitäten von $p \times 64 \text{Kbit/s}$ ($p = 1,2,3,\dots$).
- CCIR Recommendation 601
Ein Standard zur Codierung digitaler TV-Signale der Studioteknik.(vgl. [CCIR601])
- CCIR Recommendation 624-4
Ein Standard zur Beschreibung der analogen Videonormen.(vgl. [CCIR624])
- IEEE Draft Standard P1180/D2 1990
Ein Standard zur Implementation der 8×8 DCT.(vgl. [DCT90])

Die Entwicklung des Standards:

Die Entwicklung des MPEG-Standards gliederte sich in drei Phasen:

In der „Anforderungsphase“ (Requirement Phase) wurden die Anforderungen und die Entwicklungsziele der Standardisierungsbemühungen detailliert erarbeitet.

In der „Wettbewerbsphase“ (Competition Phase) entwickelten verschiedene Mitglieder des MPEG-Committees (Firmen und Institutionen) konkurrierende Lösungsvorschläge und reichten diese ein.

In der „Zusammenführungsphase“ (Convergence Phase) schließlich wurden Testverfahren entwickelt und (Teil-) Lösungen ausgewählt oder miteinander kombiniert.

Im September 1990 einigte man sich auf eine Committee Draft Version, die dann nur noch in unbedeutenden Details geändert wurde.

Im Frühjahr 1993 wurde der Standard ISO/IEC 11172 dann verabschiedet.

Inzwischen befaßt sich das MPEG-Committee mit dem Entwurf eines zweiten Standards (MPEG-2), der sich durch eine höhere Datenrate (geringere Kompression, bessere Qualität) sowie eine erweiterte Funktionalität auszeichnet. Zur Zeit (Mai 1994) existiert ein Committee Draft vom Dezember 1993.

(vgl. [Gall91], [MPEG93])

4.4.2. ÜBERBLICK

Der MPEG-Standard besteht aus drei Teilen, welche die verschiedenen Aspekte der Codierung beschreiben, und einem vierten, optionalen Teil, der die Testverfahren beschreibt:

Systems:

Beschreibt die Syntax und Semantik eines gemultiplexten Datenstromes, der die verschiedenen Datenströme (Video, Audio, Benutzerdaten) in einen gemeinsamen „Systemdatenstrom“ integriert. Es werden die Mechanismen zur Synchronisation und Packetierung sowie des Buffermanagements definiert.

Video:

Beschreibt die Syntax und Semantik eines „Videodatenstromes“ sowie ein prinzipielles Codierungs- und Decodierungsmodell. Es werden für viele Problemfelder (z.B. Codierungsverfahren) jedoch keine Algorithmen vorgegeben.

Audio:

Beschreibt die Syntax und Semantik eines „Audiodatenstromes“. Für die Codierung gibt es drei verschiedene Komplexitätsstufen, jedoch wird auch hier kein Algorithmus vorgegeben. Für Codierung und Decodierung wird ein prinzipielles Verfahrensmodell definiert.

Compliance testing:

Hier sind die Testverfahren gesammelt und beschrieben, die von den Mitgliedern des MPEG-Committees entwickelt und angewandt wurden, um die Qualität der verschiedenen Kompressions- und Codierungsverfahren zu bewerten. Man könnte hiermit eine Evaluierung der Implementationen verschiedener MPEG-Codecs vornehmen.

Im Rahmen dieser Arbeit wird nur auf die ersten beiden Teile - Systems und besonders Video - eingegangen. Das zu entwickelnde System beschäftigt sich (in seiner ersten Version) nur mit den Möglichkeiten der Bildeditierung, ohne eine Audiosignalverarbeitung prinzipiell auszuschließen. Daher ist ein tiefgehendes Verständnis der Codierung des Audiosignales nicht notwendig.

Auch eine Evaluierung verschiedener Kompressionsverfahren ist nicht Gegenstand dieser Arbeit, so daß auf eine Erklärung der Testverfahren verzichtet wird.

(vgl. [Gall91], [MPEG93])

4.4.3. SYSTEMS

Die MPEG-Norm beschreibt nirgends eine detaillierte Implementierung eines Coders oder Decoders, der in der Lage wäre, Video und Audiodaten zu verarbeiten. Stattdessen wird eine Art Datenaustauschformat definiert, das genau die Syntax und Semantik des Datenstromes vom Coder zum Decoder festlegt. Ein Coder und Deco-

der werden nur insofern beschrieben, als es zur Festlegung und Spezifikation von verwendeten Begriffen nötig ist. *Diese prototypischen Codec sind ausdrücklich nicht normativer Bestandteil der ISO/IEC 11172.* Ein MPEG-Encoder ist jedes „Gerät“ (Hardware, Software oder beides), das einen Datenstrom gemäß des Standards liefert. Entsprechend ist ein MPEG-Decoder ein „Gerät“, das einen solchen Datenstrom auswerten kann.

Abb. 4.4.a zeigt den prototypischen MPEG-Gesamt-Codec.

Der im Systems-Teil definierte System-Bit-Stream ist der eigentliche MPEG-Datenstrom. Er besteht aus den gemultiplexten Elementardatenströmen.

Elementardatenströme sind:

- bis zu 32 Videoströme
- bis zu 16 Audioströme
- zwei Arten Benutzerdatenströme,
die die Codierung beliebig vieler Subströme erlauben

Gleich beim Dateneingang wird die Systemzeit registriert und den jeweiligen Daten als „Presentation Time Stamp (PTS)“ mit 33 Bit Breite zugeordnet. Da die Systemzeit für alle Elementarcodierer von dem selben zentralen Zeitgeber kommt, ist somit eine spätere Synchronisation möglich.

Die Audio- und Videodaten werden im Coder nach den Vorgaben der entsprechenden Normteile codiert. Für die Codierung der Benutzerdaten gibt es keine Vorgaben.

Die codierten Daten der Elementardatenströme werden in „packets“ gepackt. Ein solches Paket enthält nur Daten eines Elementarstromes, der schon am Startcode des Paketes erkennbar ist. Die Pakete können unterschiedlich lang sein und zerschneiden den Elementardatenstrom an beliebiger Stelle. Die Reihenfolge der Daten bleibt jedoch erhalten.

Die Pakete der einzelnen Elementardatenströme werden nun in sog. „packs“ zusammengefaßt und übertragen oder gespeichert. In einem Pack können Pakete verschiedener oder auch nur eines Elementardatenstromes übertragen werden. Ihre Anordnung innerhalb des Packs und auch ihre Aufteilung auf evtl. mehrere Packs wird durch die Datenraten der einzelnen Elementardatenströme festgelegt.

Der Codierer muß Angaben über die maximal notwendige Größe des Puffers auf der Decodierseite machen und ist dann verantwortlich dafür, Pufferüber- oder -unterschreitungen zu verhindern.

Außer den Daten der Pakete selbst liefert jeder Pack noch eine 33 Bit breite „System Clock Reference (SCR)“, die den Stand des Codierer-Zeitgebers beim Versenden des Packs wiedergibt. Damit ist es möglich, im Decodierer einen zentralen Zeitgeber zu steuern und zu synchronisieren.

Die Übertragung der einzelnen Packs muß nicht stetig erfolgen. Zwischen den Packs kann es zu Pausen beliebiger Dauer kommen.

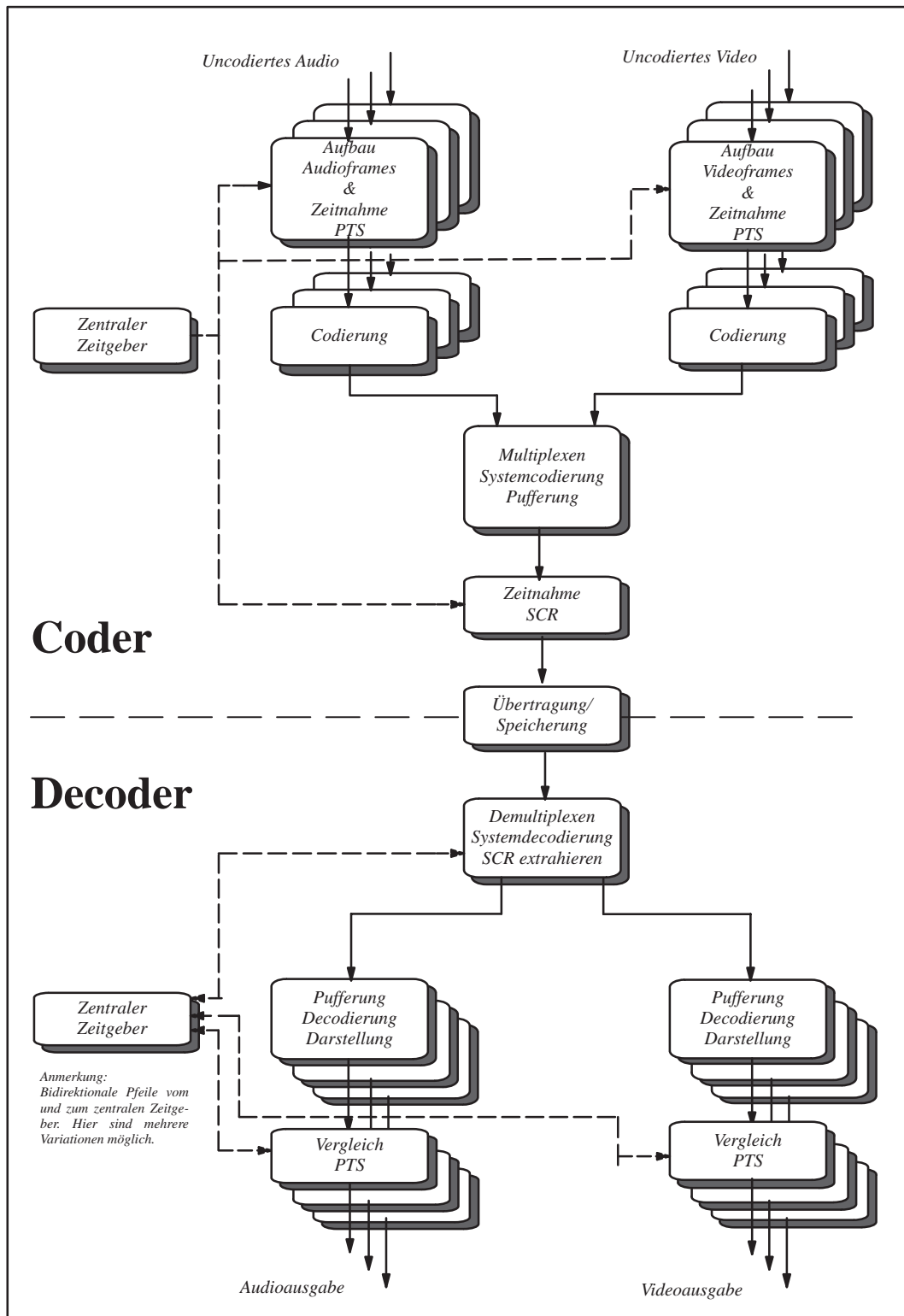


Abb. 4.4.a Der prototypische MPEG-Codec

Im Decoder werden die Packs entgegengenommen, die SCR extrahiert und damit der zentrale Zeitgeber gespeist. Prinzipiell ist es allerdings auch möglich, den Zeitgeber von einem der Elementardatenströme, mit Hilfe der PTS zu speisen. Somit kann z.B. ein Videostrom auf die Laufzeit des Audiostromes synchronisiert werden, in dem der Audiostrom den Zeitgeber steuert.

Das Demultiplexing der einzelnen Pakete sowie das Decodieren der Elementardatenströme erfolgt im einzigen Puffer des Decoders. Die Daten aller Pakete eines Elementardatenstromes ergeben aneinander gereiht wieder den kompletten Datenstrom. Ist eine „access unit“ (ein codierter Audio- oder Videoframe) komplett im Puffer, wird er durch den entsprechenden Elementardecoder entnommen, decodiert und nach einem Vergleich mit dem zentralen Zeitgeber dargestellt(vgl. [MPEG93]).

4.4.4. VIDEO

Das MPEG-Videocodierungsverfahren besteht aus sechs Schritten. Abb. 4.4.b zeigt den Datenfluß durch den Codierer bis zur Speicherung bzw. Übertragung der codierten Bildsequenz. Bei der Decodierung durchlaufen die Daten den Decodierer mit den inversen Funktionen in umgekehrter Reihenfolge(vgl. [Gall91], [Hung93], [Panc92], [Rowe93a]).

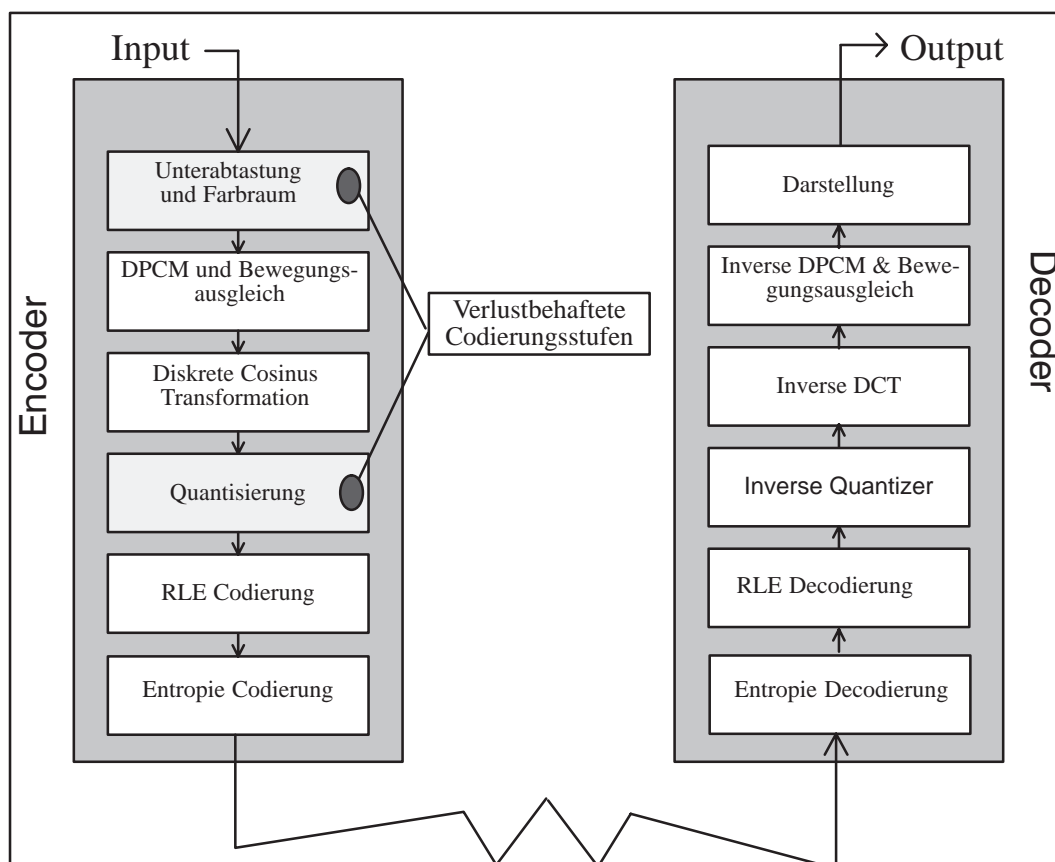


Abb. 4.4.b Die Phasen der MPEG Codierung/Decodierung

Das MPEG-Verfahren setzt Eingangsdaten entsprechend der CCIR 601 für digitale TV-Formate voraus. Eingangsdaten anderen Formates müssen erst gewandelt werden. Doch selbst die CCIR-Daten müssen in der ersten Stufe noch reduziert werden.

Die „motion compensation“ berechnet die Differenz zum Vorgängerbild, falls dies sinnvoll ist. Die „Transformation“ konzentriert die Information in die ersten wenigen transformierten Koeffizienten. Die „Quantisierung“ bewirkt einen kontrollierten Verlust von weniger wichtigen Informationen. Die beiden letzten Codierungsstufen komprimieren die Daten verlustfrei weiter.

Das MPEG-Codierungsverfahren wird als verlustbehaftet bezeichnet, weil das decodierte Bild nicht mehr mit dem Original identisch ist. Die Verluste sind jedoch kontrollierbar und so geschickt gewählt, daß für das menschliche Auge selbst bei hohen Kompressionsfaktoren kaum Unterschiede erkennbar sind.

Zu den einzelnen Codierungsstufen:

4.4.4.1. Farbraum und Unterabtastung

Farbraum

Seit Newton ist bekannt, das ein weites Spektrum von Farben aus drei Grundfarben (Primärvalenzen) gemischt werden kann. Der Wechsel der Primärvalenzen für eine Farbdarstellung entspricht mathematisch einer Hauptachsentransformation, wie sie aus der Linearen Algebra bekannt ist. Hierbei werden die Indizes, welche zu einer Normalbasis (den Primärvalenzen) des alten Farbraumes gehören, in Indizes bzgl. der neuen Normalbasis umgerechnet (vgl. [Hoffm92], [Hung93]).

Prinzipiell sind zwei Arten der Farbmischung zu unterscheiden: Additive und subtraktive Farbmischung. Sie teilen die Menge der möglichen Primärvalenzen in zwei disjunkte Mengen auf.

Beim Malen und in der Drucktechnik findet die subtraktive Farbmischung Verwendung, da die hier verwendeten Farben dem „weißen“ Umgebungslicht während der Reflektion Farbanteile - wie Filter - entziehen. Eine Mischung aller Primärvalenzen ergibt somit Schwarz. Die hier am häufigsten verwendeten Primärvalenzen sind:

Cyan, Magenta und Gelb (CMY)

In der TV-, Video- und Monitortechnik findet die additive Farbmischung Verwendung, da hier lichtemittierende Farbpigmente eingesetzt werden. Die in verschiedener Intensität leuchtenden Primärvalenzen addieren sich zur gewünschten Farbe. Eine Mischung aller Primärvalenzen führt hier zu Weiß. Fast ausschließlich arbeitet man hier mit:

Rot, Grün und Blau (RGB)

als Primärvalenzen.

Ein weiteres Farbmischmodell ist das von der CIE 1931 vorgestellte YUV Primärvalenzensystem, das nach den Ergebnissen einer Untersuchung dieser Kommission zur

Farbwahrnehmung des menschlichen Auges entwickelt wurde. Die Primärvalenzen sind hier

$$\begin{aligned} Y: & \quad \text{Luminanz (Helligkeit)} \\ U, V: & \quad \text{Chrominanz (Farbdifferenz)} \end{aligned}$$

Dieses System ermöglicht insbesondere die Berücksichtigung der unterschiedlichen Wahrnehmungsfähigkeiten des Menschen für Helligkeits- und Farbinformation (höhere Sensitivität im Luminanzbereich).

Das YUV-Farbmodell wurde als Basis der Codierung und Übertragung der verschiedenen TV-Normen der CCIR verwendet.

Die Umrechnungen der für den Monitor notwendigen RGB-Werte in YUV Koordinaten ist für z.B. den europäischen PAL-Standard (vgl. [CCIR624]) so definiert:

$$\begin{aligned} Y &= 0,299R + 0,587G + 0,114B & (\text{Gl. 4.7}) \\ U &= 0,493(B - Y) \\ V &= 0,877(R - Y) \end{aligned}$$

Die Gewichtungen der einzelnen RGB-Komponenten bei der Berechnung des Y-Wertes ergibt sich aus dem Wahrnehmungsmodell der CIE. Ebenfalls gemäß diesem Modell wurde die Reduktion des Informationsgehalts der U und V Komponenten um Faktoren von 0.493 bzw. 0.877 durchgeführt.

Auch das MPEG-Verfahren setzt eine Farbbildbeschreibung nach dem YUV-System voraus. Ein Eingangsbild muß also gegebenenfalls von einem anderen Primärvalenzsystem in die YUV-Normalbasis umgerechnet werden. Da es sich hier jedoch um einen digitalen Standard handelt, wird auch die digitale Entsprechung des Farbmodells, YC_bC_r , verwendet (vgl. [CCIR601]). C_b entspricht der analogen U-Komponente und C_r der analogen V-Komponente, allerdings mit leicht veränderten Reduktionsfaktoren.

Die CCIR 601 nutzt einen Wertebereich von 8 Bit je Komponente jedoch nicht vollständig. So liegt der Bereich der Luminanz zwischen 16 und 235 und der der Chrominanz zwischen -112 und +112 (zur Übertragung/Speicherung um +128 verschoben auf 16-240).

Die Umrechnung von je 8 Bit RGB-Komponenten ergibt sich also nach:

$$\begin{bmatrix} Y' \\ C_b' \\ C_r' \end{bmatrix} = \begin{bmatrix} 0,299 & 0,587 & 0,114 \\ -0,1687 & -0,3313 & 0,5 \\ 0,5 & -0,4187 & -0,0813 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (\text{Gl. 4.8})$$

mit einer anschließenden Skalierung

$$Y = Y' * \frac{219}{255} + 16 \quad (\text{Gl. 4.9})$$

$$C_b = C_b' * \frac{224}{255} + 128$$

$$C_r = C_r' * \frac{224}{255} + 128$$

Unterabtastung

Die CCIR 601 definiert für PAL eine Bildfrequenz von 25 Bildern pro Sekunde. Jedes Bild wird im Zeilensprungverfahren übertragen, d.h., ein Bild besteht aus zwei Halbbildern, die jeweils aus den geraden bzw. ungeraden Zeilen des Gesamtbildes bestehen. Die in der CCIR 601 vorgeschriebene Bildauflösung für PAL liegt bei 625 Zeilen a 720 Pixel, jedoch tragen nur 576 Zeilen x 702 Pixel auch Bildinformation. Da durch die YCbCr Farbcodierung die Bildinformation in der Y-Komponente konzentriert ist, können die einzelnen Komponenten mit unterschiedlichen Auflösungen übertragen werden. Als Eingabe ist im MPEG-Standard die 4:2:2 Auflösung definiert, das heißt:

$$Y: \quad 576 \times 720 \quad (\text{Gl. 4.10})$$

$$C_b/C_r: 576 \times 360$$

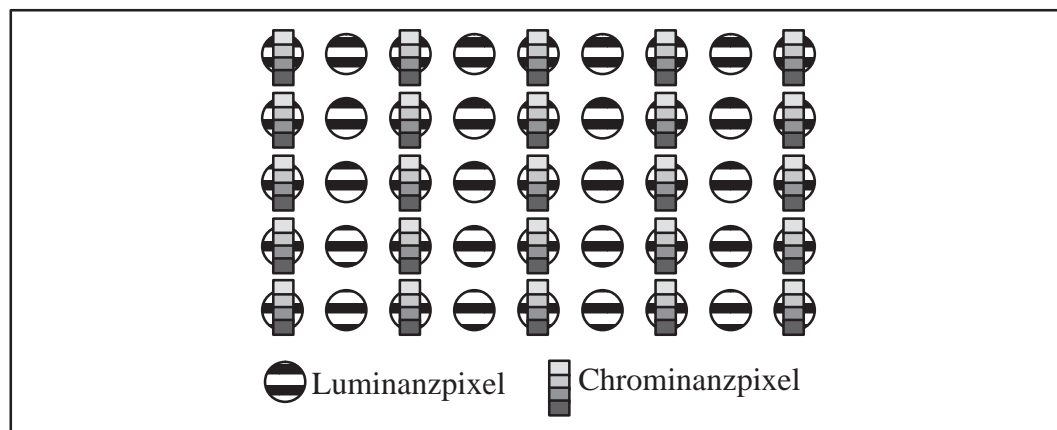


Abb. 4.4.c Die Korrelation der Chrominanz- mit den Luminanzpixeln nach CCIR 601.

Um eine effektive Codierung bei Datenraten bis zu 1,5 Mbit/s zu gewährleisten, wird die Auflösung in MPEG horizontal und vertikal auf die Hälfte weiter reduziert. Zudem wird das Zeilensprungverfahren nicht angewendet. Man muß also eine Unterabtastung der Eingangsdaten vornehmen. Vorgegeben ist nur die Zielgröße, der

Algorithmus ist frei wählbar. Abb. 4.4.d zeigt die Korrelation der Chrominanzpixel mit den Luminanzpixeln nach MPEG.

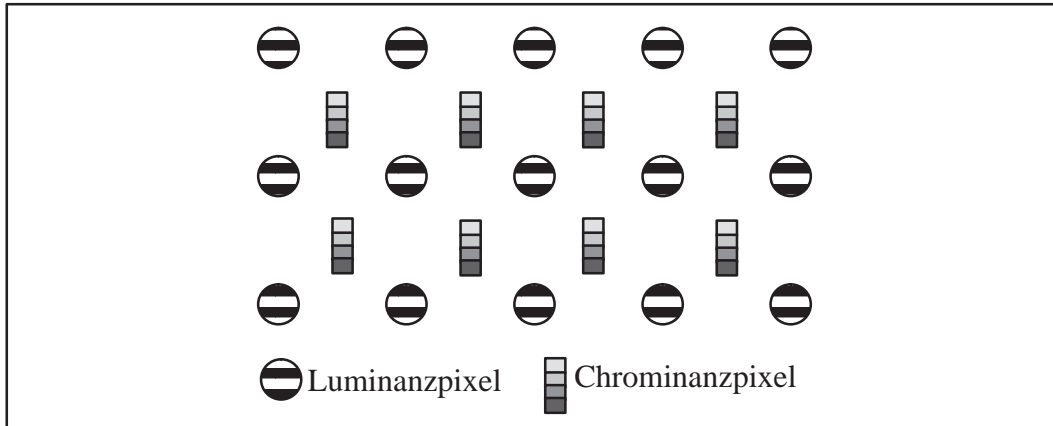


Abb. 4.4.d Die Korrelation der Chrominanz- und Luminanzpixel nach MPEG

Die Auflösung des tatsächlichen MPEG-Bildes beträgt dann

$$Y: \quad 352 \times 288 \quad (Gl. 4.11)$$

$$C_b/C_r: 176 \times 144$$

Diese sog. „significant pel area“ ergibt sich aus der Bedingung, daß alle Werte durch 16 teilbar sein müssen, um die weitere Verarbeitung zu gewährleisten.

(vgl. [CCIR601], [CCIR624], [MPEG93], [Prit80], [Gall91], [Hung93])

4.4.4.2. DPCM Codierung und Bewegungsausgleich

Grundsätzlich verwendet MPEG hier ein DPCM-Verfahren, wie dies in Kapitel 4.3.2.1. schon beschrieben wurde.

Da die meisten Bilder einer Bildfolge gleichen oder sehr ähnlichen Inhalts sind, ist es unnötig, diesen Inhalt mehrfach zu codieren. Er kann aus einem vorangegangenen Bild rekonstruiert werden, wobei nur eine geringere Informationsmenge wie Verschiebungsvektoren (falls der Bildinhalt lediglich verschoben wurde) oder Fehlerkorrekturen, falls der Inhalt sich nur geringfügig geändert hat, codiert werden müssen. Somit ist eine beträchtliche Datenreduzierung möglich, ohne eine Qualitätseinbuße in Kauf zu nehmen.

Eine weitere Datenreduzierung kann erreicht werden, indem man nicht nur vorangegangene Bilder zur Rekonstruktion des Inhaltes des aktuellen Bildes verwendet, sondern auch weiter vorne in der Bildfolge liegende, zukünftige Bilder berücksichtigt. Auf diese Weise ist es z.B. möglich, Bildinhalte, die erst durch eine Kamerabewegung langsam ins Bild kommen, einmal vollständig zu beschreiben (und zu übertra-

gen) und dann auch schon von früheren Bildern aus (evtl. teilweise) zu referenzieren. Dies ist nur dann möglich, wenn zum Zeitpunkt der Codierung bereits auf diese zukünftigen Bilder zugegriffen werden kann. Es ist also z.B. bei der Echtzeitcodierung einer Live-Aufnahme nicht möglich.

MPEG verwendet beide Richtungen des Bewegungsausgleichs und der Differenzbildung. Der Standard definiert drei Arten der Bildcodierung, die zu drei unterschiedlichen „Bildsorten“ innerhalb einer MPEG-Sequenz führen:

I-Bilder, P-Bilder und B-Bilder

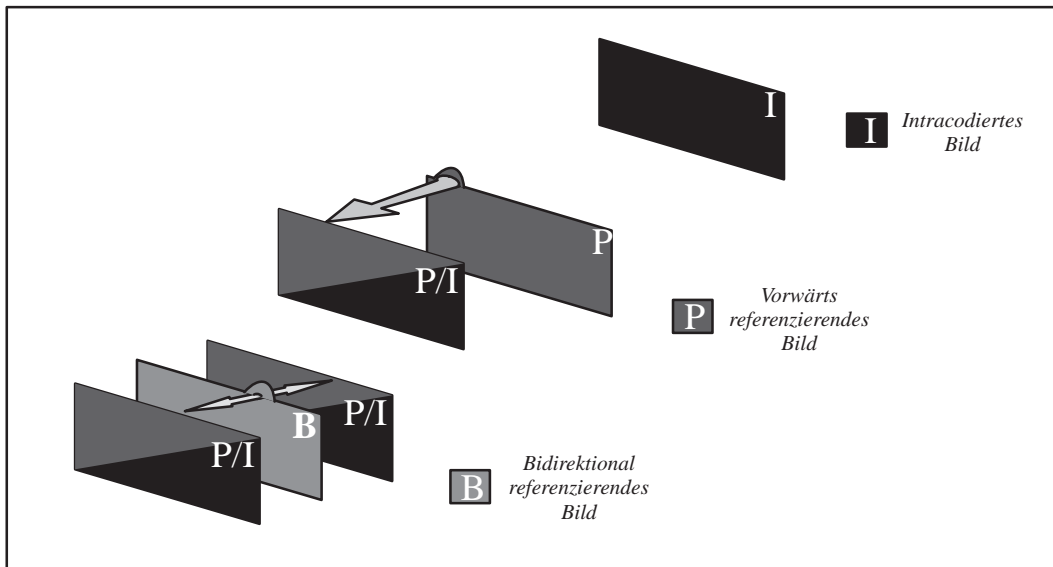


Abb. 4.4.e Die verschiedenen Bildcodierungsarten in MPEG

I-Bilder oder Intra-Codierte Bilder:

Ihr Bildinhalt wird ohne Bezug auf ein anderes Bild der Sequenz codiert, sie sind ohne Information über eines der anderen Bilder der Sequenz auch wieder zu decodieren.

I-Bilder erfüllen mehrere Zwecke:

1. Sie stellen den Initialzustand (man muß mit „etwas“ beginnen).
2. Sie definieren die Zugriffspunkte im wahlfreien Datenzugriff.
3. Sie ermöglichen eine Resynchronisation des Decodierers nach einem Übertragungsfehler.
4. Sie dienen als Referenzbilder für die beiden anderen Arten der Codierung.
5. Sie werden bei einem Szenen- oder Motivwechsel eingesetzt, wo keine Inhaltsreferenz möglich ist.

P-Bilder oder vorwärts referenzierende Bilder (forward predicted)

Sie referenzieren, wenn sinnvoll, das vorhergehende I- oder P-Bild (I/P-Bild) zur Rekonstruktion des Bildinhaltes. Ist dies nicht sinnvoll, wird der Inhalt vollständig (intra) codiert.

P-Bilder dienen selbst als Referenz für andere P-Bilder und für B-Bilder.

P-Bilder können auch dort codiert werden, wo ein Zugriff auf zukünftige Bilder nicht möglich ist.

B-Bilder oder bidirektional referenzierende Bilder (bidirectional predicted)

Sie referenzieren sowohl das vorhergehende I/P-Bild als auch das folgende I/P-Bild zur Rekonstruktion des Bildinhaltes. Falls sinnvoll kann die Referenz auch nur in eine Richtung gemacht werden oder der Inhalt intracodiert übertragen werden.

B-Bilder dienen selbst nicht als Referenz für andere Bilder und können einen evtl. Rekonstruktionsfehler nicht weitergeben. Es ist daher möglich, sie mit einer geringeren Genauigkeit zu codieren.

Die Reihenfolge

Die Reihenfolge der verschiedenen Bildsorten innerhalb einer MPEG-Sequenz ist beliebig und muß auch nicht regelmäßig sein. Die Codierungsart eines Bildes kann jederzeit nach Bedarf gewählt werden.

Eine MPEG-Sequenz wird jedoch in einer anderen Bildreihenfolge gespeichert/übertragen, als sie dargestellt wird. Abb. 4.4.f zeigt eine MPEG-Sequenz in *Darstellungs-* und *Speicherungsreihenfolge* (*Übertragungsreihenfolge*). Diese Umsortierung ist notwendig, um zu gewährleisten, daß zum Zeitpunkt der Darstellung eines B-Bildes dessen *beide* Referenzbilder bereits vorhanden sind. MPEG geht hierbei von einem rein sequentiellen Zugriff aus, wie dies z.B. bei einer Übertragung mit direkter Decodierung (Fernsehen) der Fall ist.

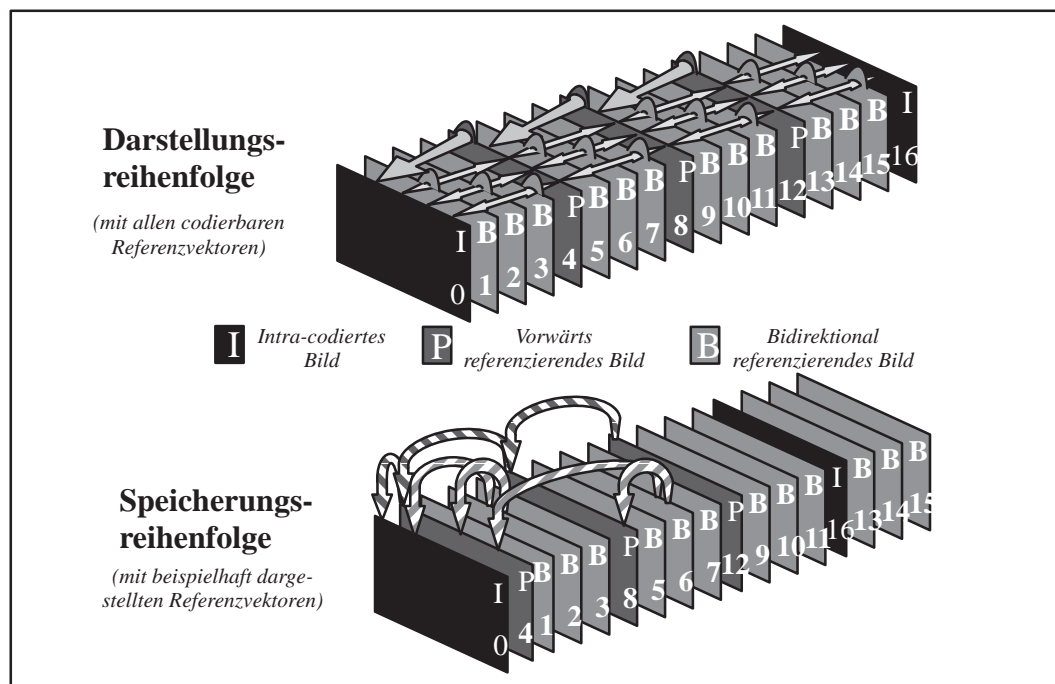


Abb. 4.4.f Darstellungs- und Speicherungsreihenfolge einer typischen MPEG-Sequenz

Wie wird der Bewegungsausgleich berechnet?

Um auch die Bewegung einzelner Objekte im Bild zu erfassen und um die Berechnung der Referenzvektoren zu vereinfachen, wird das Bild in sogenannte Macroblocks der Größe 16x16 Pixel der Luminanzauflösung zerlegt. Das entspricht einem 8x8 Pixel Block in einer der Chrominanzauflösungen. Jeder solche Macroblock wird nun mit Gebieten gleicher Größe auf dem Referenzbild verglichen und das Gebiet mit der besten Übereinstimmung ausgewählt.

Ist die Übereinstimmung so groß, daß die Codierung des Fehlerterms (der Differenz) weniger Bits beansprucht als die Codierung des Macroblockes selbst, werden der Fehlerterm und die Referenzvektoren codiert. Andernfalls wird der Macroblock selbst intracodiert.

Im Falle des bidirektionalen Bewegungsausgleichs gilt es noch eine Entscheidung über die Verwendung der Referenz auf ein zukünftiges Bild zu fällen oder eine Interpolation beider Referenzgebiete durchzuführen.

Zum Bestimmen des Referenzgebietes und der Bewertung der Übereinstimmung wird kein Algorithmus vorgegeben. Anhang D aus Teil 2 der [MPEG93] spezifiziert jedoch unverbindlich einige denkbare Verfahren, wie Teleskopsuche, Logarithmische Suche oder 2D Suche.

Die Genauigkeit der Referenzvektoren liegt bei 1/2 oder 1 ganzen Pixel und kann horizontal und vertikal unabhängig je Bild neu festgelegt werden.

Die Referenzvektoren können einen Bereich von -512 - 511,5 Pixel (bei 1/2 Pixel Genauigkeit) bzw. -1024 - 1023 Pixel (bei 1 Pixel Genauigkeit) beschreiben. Sie dürfen allerdings nicht über den Rand des Referenzbildes hinaus verweisen.

Die Referenzvektoren selbst werden dann wieder nach dem DPCM Verfahren codiert, indem nur die Differenz zu den Vektoren des vorangegangenen Macroblocks übertragen wird.

4.4.4.3. Transformation

MPEG definiert in der nun folgenden Codierungsstufe eine Transformation der Bilddaten, wie bereits in Kapitel 4.3.2.2. beschrieben:

Falls die Intensitäten des Bildes (die Pixelwerte) im unteren Ortsfrequenzbereich liegen, wie es bei „natürlichen“, nicht synthetischen Bildern sehr oft der Fall ist, läßt sich die Bildenergie mit einer Transformation des Bildes in sehr wenigen Transformationskoeffizienten konzentrieren. Um die Komplexität einer solchen Transformation zu verringern, werden die Macroblocks weiter zerlegt in Blocks der Größe 8x8 Pixel. (vgl. Abb. 4.3.b)

MPEG verwendet die zweidimensionale „diskrete Cosinustransformation (DCT)“ (vgl. [DCT90]) für diesen Zweck.

Die zweidimensionale DCT läßt sich kombinieren aus der Durchführung der eindimensionalen DCT über die Zeilen und über die Spalten des Blocks.

Die explizite Formel für die 8x8 zweidimensionale DCT ist definiert durch

$$F[u, v] = \frac{1}{4} \sum_i \sum_j C(i, u) C(j, v) f[i, j] \quad (\text{Gl. 4.12})$$

mit

$$C(i, u) = A(u) \cos \frac{(2i + 1)u\pi}{16} \quad (\text{Gl. 4.13})$$

$$A(u) = \begin{cases} \frac{1}{\sqrt{2}} & \text{für } u = 0 \\ 1 & \text{für } u \neq 0 \end{cases} \quad (\text{Gl. 4.14})$$

$$0 \leq i, j, u, v \leq 7$$

wobei

$F[u, v]$ ist der Transformationskoeffizient an Stelle u, v
 $f[i, j]$ ist der Pixelwert an Macroblokkordinate i, j

Das transformierte Ergebnis der 2D-DCT ist so geordnet, daß der Durchschnittswert, der sogenannte DC-Koeffizient in der oberen linken Ecke steht, also die Indizes (0,0) hat. Die übrigen Koeffizienten repräsentieren mit steigendem Abstand zum DC höhere Frequenzen, wobei die höheren vertikalen Frequenzen durch höhere Zeilenindizes repräsentiert werden und die höheren Horizontalfrequenzen durch größere Spaltenindizes (vgl. Abb. 4.3.b).

Die zur DCT inverse Operation ist definiert durch

$$f[i, j] = \frac{1}{4} \sum_u \sum_v C(i, u) C(j, v) F[u, v] \quad (\text{Gl. 4.15})$$

$$0 \leq i, j, u, v \leq 7$$

mit $C(i, u)$ wie in (Gl. 4.13) / (Gl. 4.14).

Die Transformation der Pixelintensitäten durch die DCT führt nicht zu einem Informationsverlust, abgesehen von den Rundungsfehlern und dem diskreten Charakter der Operation. Die Rücktransformation mit Hilfe der inversen DCT ergibt somit wieder den Originalblock.

Der Maximalwert eines Koeffizienten einer 8x8 DCT ist nach oben begrenzt auf das achtfache des Originalwertes. Also kann jeder 8 Bit Eingangswert durch einen 11 Bit Koeffizienten dargestellt werden.

(vgl. [DCT90], [Gall91], [Hung93], [Rowe93b])

4.4.4.4. Quantisierung

Die Quantisierung stellt, neben der Unterabtastung der Eingangsdaten, die verlustbehaftete Stufe der MPEG-Codierung dar. Hier wird auch die Balancierung vorgenommen zwischen Qualität und Kompressionsrate. Außerdem stellt sie damit ein wichtiges Instrument zur Regelung der Datenübertragungsrate.

Die Koeffizienten der DCT werden nun quantisiert, um ihren Wertebereich zu verkleinern. Dabei werden auch viele der Koeffizienten auf Null reduziert. MPEG nutzt eine einheitliche Quantisierung mit verschiedenen Schrittweiten je DCT Koeffizient (Vektor Quantisierung). Dazu wird eine 8x8-Quantisierungsmatrix $Q_{[u,v]}$ von 8Bit Werten definiert, die die Quantisierungsstufe für jeden der 64 Koeffizienten angibt. Zusätzlich existiert noch ein Quantisierungsfaktor Q_f , der je Macroblock definiert werden kann. Er skaliert die Quantisierungsmatrix, bevor sie angewendet wird ($Q_f/8 * Q_{[u,v]}$).

Die Wahl der Quantisierungsmatrix bestimmt entscheidend die Qualität des rekonstruierten Bildes und Q_f die der erzielten Kompression.

Da die hohen Frequenzen ein Rauschen repräsentieren und die niedrigen Frequenzen eine Struktur in dem Bild beschreiben, kann man davon ausgehen, daß in einem natürlichen Bild die niedrigen Frequenzen vorherrschend sind, während viele der hohen Koeffizienten kleine Werte aufweisen. Da das menschliche Wahrnehmungssystem strukturorientiert ist, kann die Quantisierung in den Bereichen der hohen Frequenzen größere Quantisierungsstufen ansetzen als in den sensibleren Bereichen der niedrigen Frequenzen, wo sonst sehr schnell „Blockeffekte“ wahrgenommen werden würden.

Probleme entstehen bei diesem Verfahren, falls das Bild tatsächlich einen hohen Informationsanteil in dem hohen Frequenzbereich enthält, wie es z.B. oft in synthetischen Bildern vorkommt, oder wenn sehr kleine Schrift im Bild ist. Hier muß die Quantisierungsmatrix entsprechend angepaßt werden, da sonst ein hoher Informations- und Inhaltsverlust des Bildes zu erwarten ist.

Durch die Quantisierung werden viele der kleineren Koeffizienten auf Null reduziert, so daß auf ihre Übertragung ganz verzichtet werden kann, was eine weitere Kompression zur Folge hat.

Tatsächlich existieren zwei Quantisierungsmatrizen, eine für intracodierte Blocks und eine andere für nicht intracodierte. Der Grund dafür liegt im Informationsgehalt und der Struktur der Blocks verborgen. Intercodierte, also DPCM-codierte Blocks

werden viel eher Koeffizienten im höheren Frequenzbereich aufweisen als die nicht-intracodierten (vgl. [Gall91], [MPEG93]), die ja nur den Fehlerterm, also die Differenz zu einem anderen, selbst schon quantisierten Block übertragen. Die Intra-Quantisierungsmatrix berücksichtigt also die erste Filterung des Originalfrequenzspektrums, während die Inter-Quantisierungsmatrix eher eine gleichmäßige weitere Quantisierung der Differenz vornimmt.

Für beide Matrizen ist eine Standardbelegung vorgegeben, die angenommen wird, falls die Anwendung nicht eigene Belegungen definiert. Eigenbelegungen der Quantisierungsmatrizen werden mit der MPEG-Sequenz übertragen.

Die Standard-Matrix der Intraquantisierung hat folgende Belegung:

$$Q_{intra}[u, v] = \begin{bmatrix} 8 & 16 & 19 & 22 & 26 & 27 & 29 & 34 \\ 16 & 16 & 22 & 24 & 27 & 29 & 34 & 37 \\ 19 & 22 & 26 & 27 & 29 & 34 & 34 & 38 \\ 22 & 22 & 26 & 27 & 29 & 34 & 37 & 40 \\ 22 & 26 & 27 & 29 & 32 & 35 & 40 & 48 \\ 26 & 27 & 29 & 32 & 35 & 40 & 48 & 58 \\ 26 & 27 & 29 & 34 & 38 & 46 & 56 & 69 \\ 27 & 29 & 35 & 38 & 46 & 56 & 69 & 83 \end{bmatrix}$$

Die Quantifizierungsstufe des DC, $Q(0,0)$ muß immer den Wert 8 erhalten.

In intracodierten Blocks werden der DC und die ACs unterschiedlich quantisiert mit:

$$C[0, 0] = \left\lfloor \frac{F[0, 0] \pm 4}{8} \right\rfloor \quad (Gl. 4.16)$$

$$C[u, v] = \left\lfloor \frac{A(u, v) \pm Q_f}{2Q_f} \right\rfloor \quad (Gl. 4.17)$$

mit

$$A(u, v) = \left\lfloor \frac{(F[u, v] * 16) \pm \frac{Q_{intra}[u, v]}{2}}{Q_{intra}[u, v]} \right\rfloor \quad (Gl. 4.18)$$

wobei

$$\pm = \begin{cases} + & \text{für } F[u, v] > 0 \\ - & \text{sonst} \end{cases}$$

$C[u, v]$ ist der quantifizierte Koeffizient

$F[u, v]$ ist der DCT-Koeffizient

$Q_{intra}[u, v]$ ist die Intra-Quantisierungsmatrix

Q_f ist der Quantisierungsparameter

Die inverse Quantisierung wird dann beschrieben durch:

$$F[0, 0] = 8 * C[0, 0] \quad (\text{Gl. 4.19})$$

$$F[u, v] = C[u, v] * Q_{intra}[u, v] * \frac{Q_f}{8} \quad (\text{Gl. 4.20})$$

Aufgrund dieser Berechnungen werden auch Koeffizienten mit Werten nahe Null nicht auf Null reduziert. Man erreicht somit eine feinere Quantisierung und weniger Informationsverlust. Abb. 4.4.g zeigt das Verhalten dieser Quantisierung.

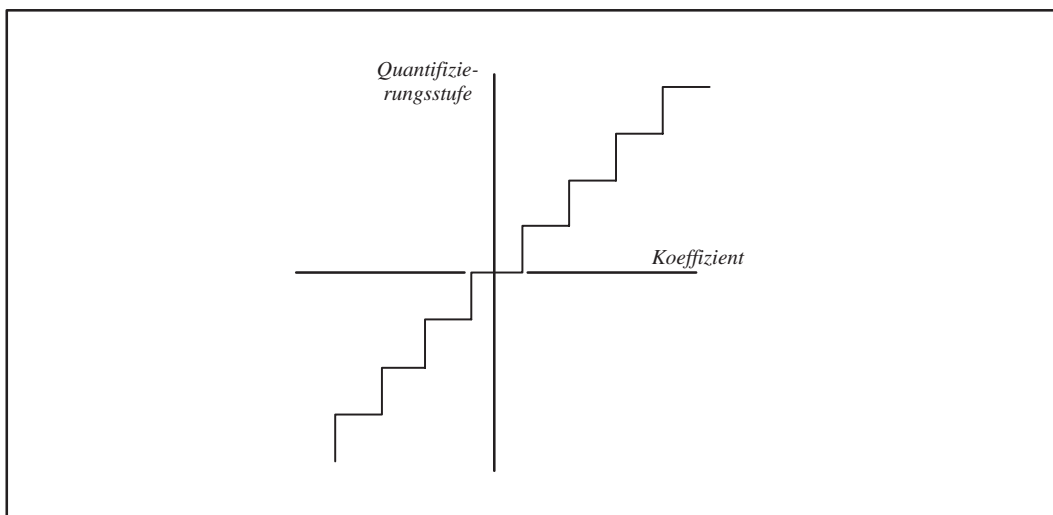


Abb. 4.4.g Das Verhalten der Intraquantisierung mit schmalen Nullband[MPEG93])

Die Standardbelegung der Inter-Quantisierungsmatrix ist

$$Q_{inter}[u, v] = \begin{pmatrix} 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \end{pmatrix}$$

Die nicht intracodierten Blocks werden, für DC und ACs gleich, quantisiert durch:

Falls Q_f ungerade:

$$C[u, v] = \frac{A(u, v)}{2Q_f} \quad (Gl. 4.21)$$

Falls Q_f gerade:

$$C[u, v] = \frac{A(u, v) \pm 1}{2Q_f} \quad (Gl. 4.22)$$

mit

$$A(u, v) = \left[\frac{F[u, v] * 16 \pm \frac{Q_{inter}[u, v]}{2}}{Q_{inter}[u, v]} \right] \quad (Gl. 4.23)$$

$$\text{wobei } \pm = \begin{cases} + & \text{für } F[u, v] > 0 \\ - & \text{für } F[u, v] < 0 \end{cases}$$

Die inverse Quantisierung ist hier:

$$F[u, v] = \frac{(2C[u, v] \pm 1) * Q_{inter}[u, v] * Q_f}{16} \quad (Gl. 4.24)$$

$$\text{wobei } \pm = \begin{cases} + & \text{für } C[u, v] > 0 \\ - & \text{für } C[u, v] < 0 \end{cases}$$

$C[u, v]$	ist der quantifizierte Koeffizient
$F[u, v]$	ist der DCT-Koeffizient
$Q_{inter}[u, v]$	ist die Inter-Quantisierungsmatrix
Q_f	ist der Quantisierungsparameter

Durch diese Art der Quantisierung werden viele Koeffizienten die einen kleineren Wert besitzen auf Null reduziert (vgl. Abb. 4.4.h).

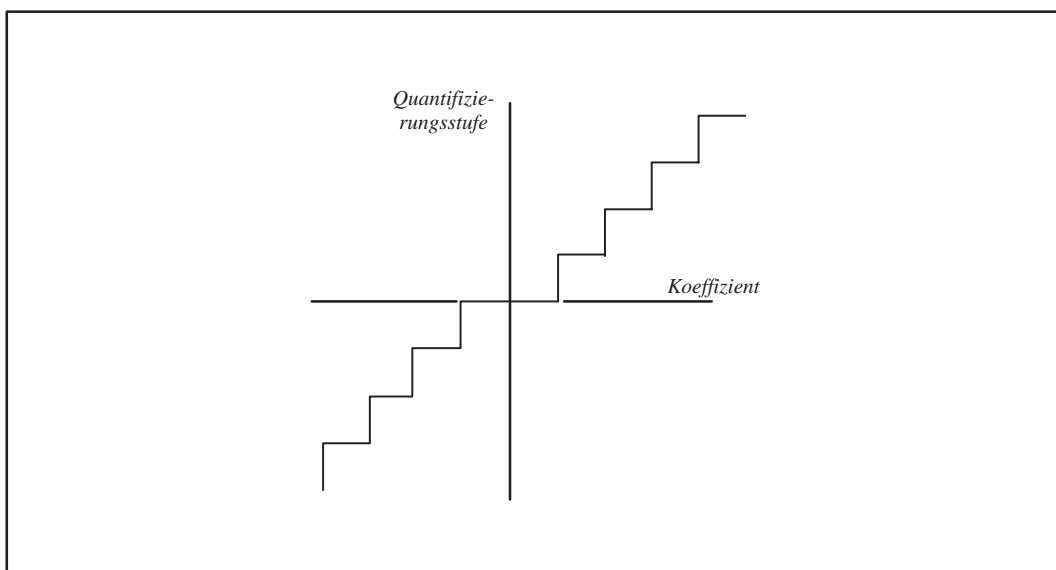


Abb. 4.4.h Das Verhalten der Interquantisierung mit breitem Nullband [MPEG93])

(vgl. [Hung93], [Gall91], [MPEG93], [Rowe93b])

4.4.4.5. Lauflängencodierung

Die DCT-Koeffizienten werden, niedrige Frequenzen zuerst, hohe Frequenzen zuletzt, linear angeordnet. Dazu entnimmt man die Werte der Matrix in einem Zickzackmuster nach Abb. 4.4.i .

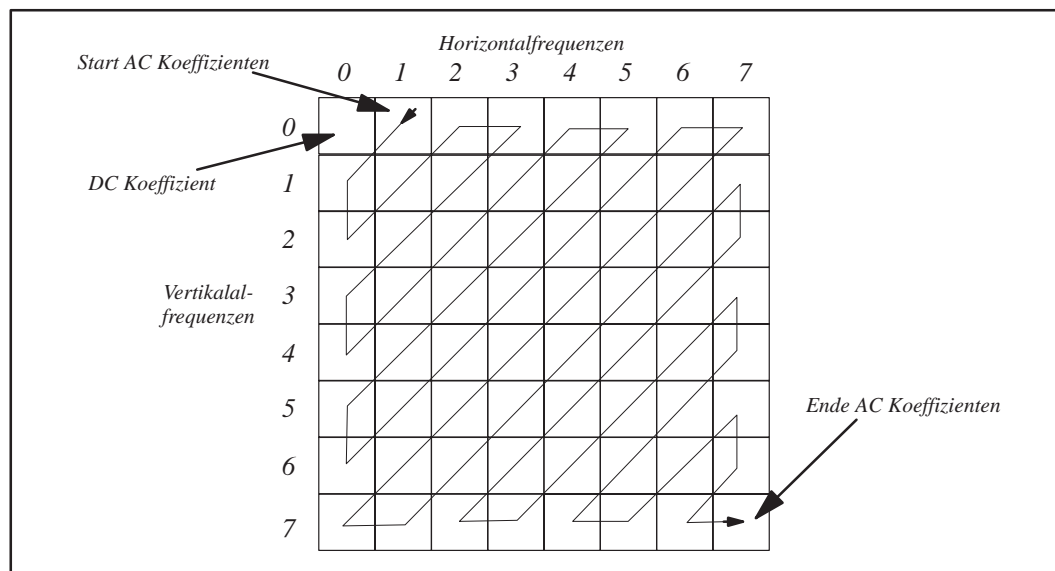


Abb. 4.4.i Die Reihenfolge der DCT-Koeffizienten zur Linearisierung

Da die Quantisierung zudem noch größere Quantisierungsstufen in den höheren Frequenzbereichen ansetzt und damit die Wahrnehmungseigenschaften des menschlichen Auges berücksichtigt (vgl. Kap. 4.4.4.4.), sind die ersten Koeffizienten der linearen Reihenfolge die wichtigsten. Sie tragen die meiste Energie, den meisten Informationsgehalt.

Diese lineare Anordnung der Koeffizienten wird nun mittels einer Lauflängencodierung (engl.: RLE (Run Length Encoding)) zusammengefasst, indem man nur die von Null verschiedenen Werte sowie die Anzahl der dazwischen liegenden Nullwerte überträgt.

Somit erhält man also eine Folge von Paaren

$$\{\text{Anzahl Nullen} , \text{Wert ungleich Null}\},$$

welche die vollständige Reihe der quantisierten DCT-Koeffizienten eines Blocks beschreibt.

(vgl. [Hung93], [Gall91], [MPEG93], [Rowe93b])

4.4.4.6. Entropiecodierung

In der letzten Stufe der MPEG-Codierung wird die Folge von Paaren sowie die übrigen zum Block gehörigen Informationen wie Referenzvektoren, Quantisierungsfak-

tor, ... mittels einer „Variable Length Coding (VLC)“ nach Huffman codiert. Der Huffman Entropiecodierer (vgl. Kap. 4.3.2.3.) wurde gewählt, da er leicht in Hardware zu realisieren ist. Allerdings wird kein echter Huffman-Algorithmus verwendet, da dieser die Generierung der Symboltabellen zur Codierungszeit erfordern würde. Stattdessen hat man einen Satz von Symbolen und ihrer Codierungen vorgegeben, ähnlich wie im Faxstandard (vgl. [FAX80]). Die Symbole wurden aufgrund von statistischen Untersuchungen für eine Reihe von Testfilmen ausgewählt (vgl. Teil 4 der [MPEG93]).

(vgl. [Huff53],[Hung93], [Gall91], [Lipp91], [MPEG93], [Rowe93b])

4.4.4.7. Ausgabe

Abb. 4.4.j zeigt nocheinmal alle Stufen der Codierung für die Bilder verschiedenen Typs.

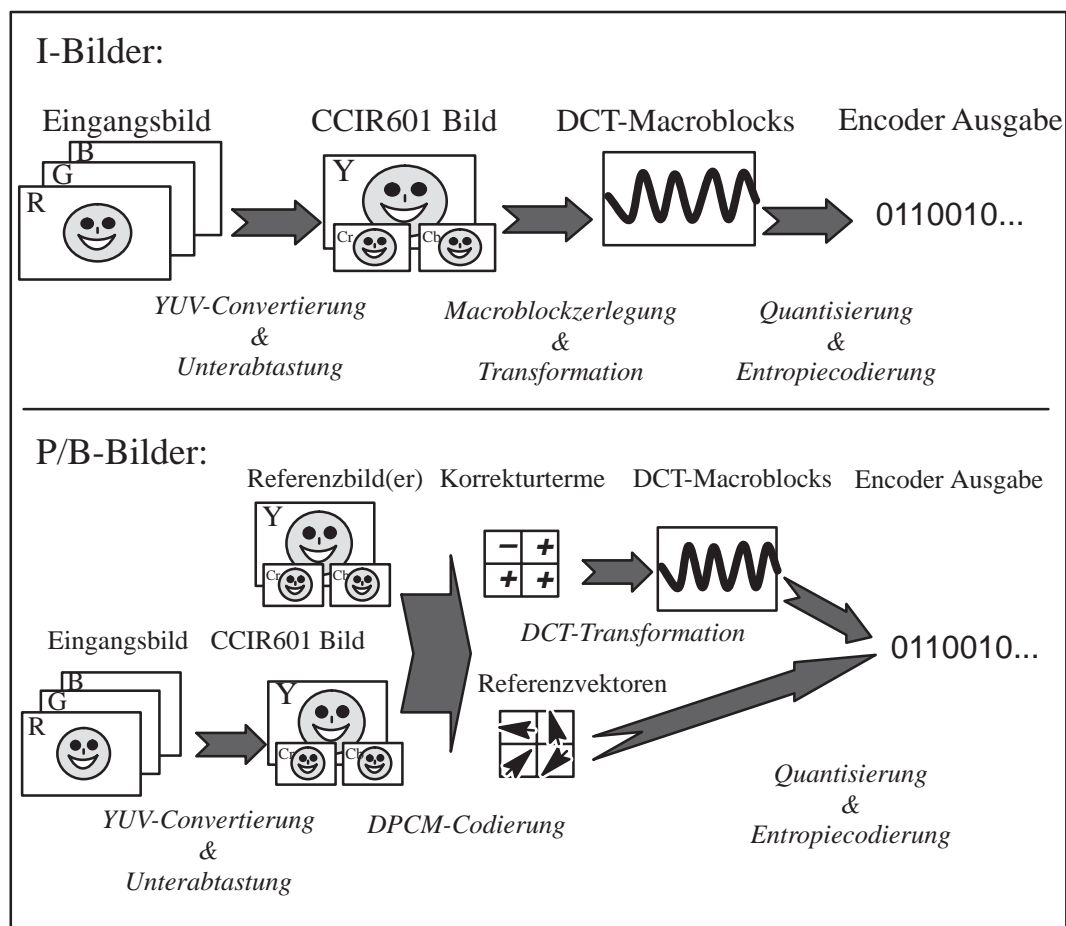


Abb. 4.4.j Codierungsablauf je Bildtyp

Die Ausgabe des Codierers entspricht einer Syntax und Semantik, die durch die Norm genau festgelegt ist. In der Tat beschreibt der Standard eigentlich nur die Struktur und Bedeutung dieses „Bitstreams“ sowie einiger Randbedingungen, die eine sichere Decodierung erlauben sollen.

Aufbau, Vorgehensweise und Funktion des Codierers sowie des Decodierers werden nur beispielhaft und unverbindlich beschrieben. Daher wird dieser Standard auch als „Generischer Standard“ bezeichnet, da er nur ein gemeinsames Basismodell sowie ein Datenaustauschformat spezifiziert. Die Anwendung und die Umsetzung dieser Spezifikationen bleibt dem Anwendungsentwickler überlassen.

4.4.4.8. Der Decodierer

Im Decodierer durchlaufen die Daten den umgekehrten Weg durch die inversen Operationen des Codierers. An der Eingabe steht also ein Bitstream gemäß ISO/IEC 11172 Definition und an der Ausgabe eine Bildsequenz.

4.4.4.9. Der Bitstream

Der MPEG-Video-Bitstream ist hierarchisch, aus mehreren Ebenen, aufgebaut. Dies erlaubt eine effiziente Gliederung der zusätzlich zum reinen Bildinhalt notwendigen Informationen.

Jede Hierarchieebene klammert die darunterliegenden Ebenen durch ihre *Startcodes*. Diese Startcodes sind 32-Bit-Worte, die nicht weiter codiert und komprimiert werden. Zudem sind die Codierungen aller anderen Daten im Videostream so definiert, daß sich kein Startcode „zufällig“ ergeben kann. Der MPEG-Videostream ermöglicht somit einen direkten Zugriff auf alle Hierarchieebenen.

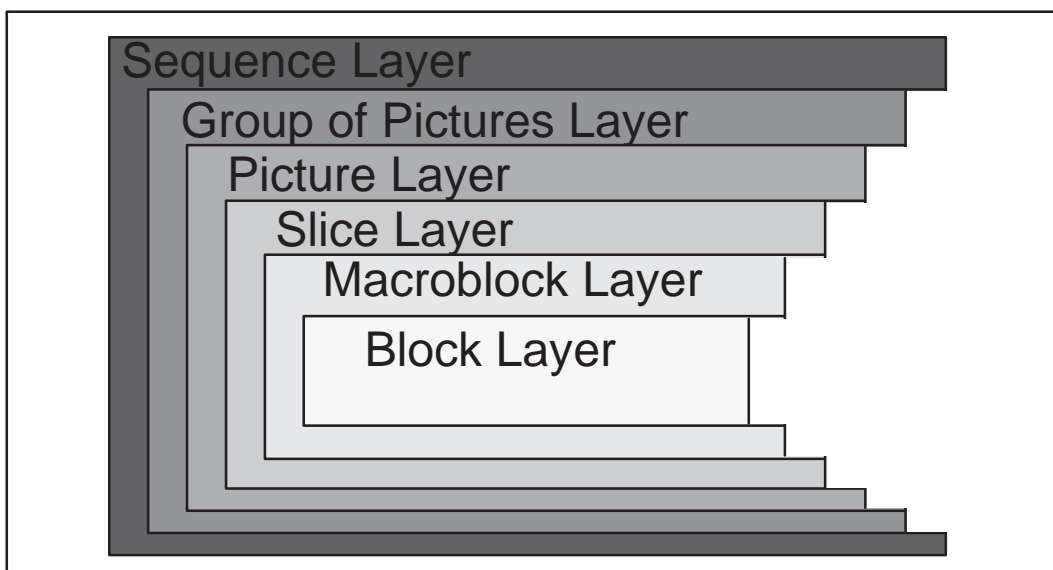


Abb. 4.4.k Der Aufbau der MPEG-Video-Hierarchie

An der Spitze der Hierarchie steht die „*Videosequence*“, die sog. Kontextebene. Sie enthält Informationen über Bildmaße, Bildseitenverhältnis, Bildwiederholrate, Quantisierungsmatrizen und Puffergrößen. Sie wird im Videostream repräsentiert durch einen „*Sequenceheader*“, der die Daten enthält, und einem „*Sequenceend-code*“, der die MPEG-Sequenz beendet. Der Sequenceheader darf beliebig oft wiederholt werden, bevor der Sequenceendcode gegeben wird. Es ist erlaubt, die Werte der Quantisierungsmatrizen jedesmal zu verändern. Alle anderen Daten müssen identisch bleiben. Damit ist eine Resynchronisierung bzw. ein Umschalten des Decodierers auf eine laufende MPEG-Sequenz möglich.

Die nächste Ebene stellt die „*Group of Pictures*“ (*GoP*) als sog. Einheit für den wahlfreien Zugriff. Eine *GoP* sollte für sich alleine, also ohne Informationen aus anderen *GoPs*, decodierbar sein. Der *GoP-Header* enthält Informationen zum TimeCode und zur (Un-) Abhängigkeit von den Nachbar-*GoPs*. Es gibt keine Information darüber, wieviele Bilder innerhalb dieser *GoP* liegen. *GoPs* sollten aus Gründen der Editierbarkeit (wahlfreier Zugriff) in regelmäßigen, nicht zu großen Abständen in der Videosequenz plaziert sein, doch darüber gibt es in dem Standard keine verbindli-

chen Vorgaben. Eine GoP muß immer mit einem I- oder B-Bild beginnen und mit einem I- oder P-Bild enden.

Eine Ebene darunter liegt das „Picture“, die primäre Codiereinheit. Es beschreibt die Position des Bildes innerhalb der GoP, die Codierungsart des Bildes (I-, P- oder B-Bild), Informationen zur Ausnutzung des Puffers und der Bewegungsausgleichscodierung (falls nötig).

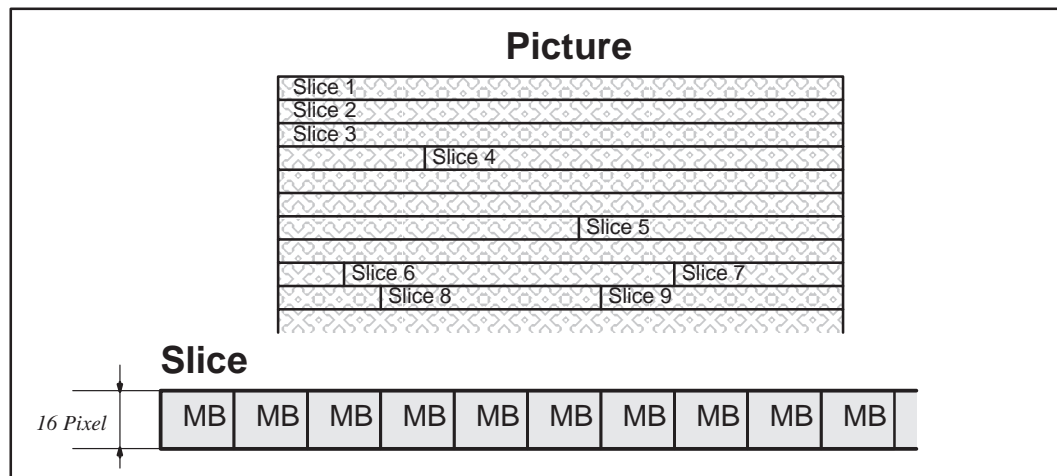


Abb. 4.4.l Die Zerlegung des Picture in Slices und Macroblocks

Eine weitere Ebene bildet das „Slice“, die Resynchronisationseinheit. Sie unterteilt das Picture in aneinander anschließende Streifen variabler Länge mit der Höhe eines Macroblocks. Zu Beginn eines jeden Slices kann die Decodierung des Bildes unabhängig von den vorangegangenen Slices wieder aufgenommen werden.

Unterhalb der Slices liegen die „Macroblocks“, die Bewegungsausgleichseinheiten. Ein Macroblock enthält seine Adresse (die Position innerhalb des Slices), evtl. einen Quantisierungsfaktor, die Referenzvektoren zum Bewegungsausgleich und Informationen darüber, welche Blocks des Macroblocks übertragen werden. Ein Macroblock besteht aus vier Luminanzblocks und zwei Chrominanzblocks á 8x8 Pixelwerten, da die Luminanzauflösung das vierfache der Chrominanzauflösung beträgt. Die Anordnung der Blocks zeigt Abb. 4.4.m .

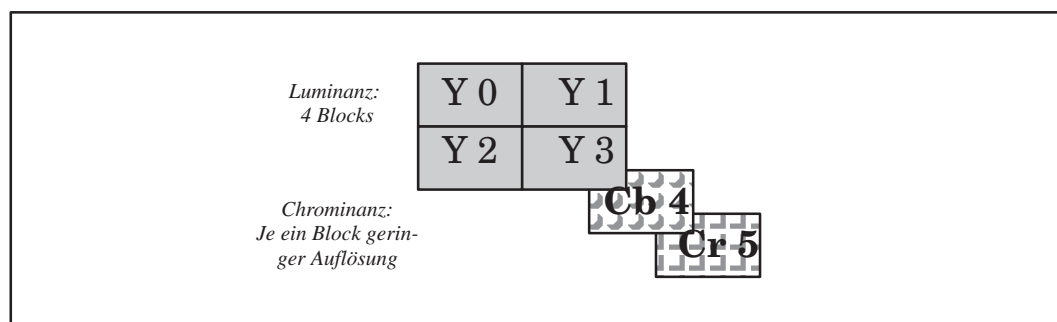


Abb. 4.4.m Die Anordnung der Blocks innerhalb eines Macroblocks

Die unterste Ebene bilden die „*Blocks*“, die DCT-Einheit. Sie enthalten den eigentlichen Bildinhalt. Da jeder Block eine genau festgelegte Position innerhalb des Bildes hat, wird er nur dann codiert und übertragen, wenn sich der Inhalt des entsprechenden Blocks des Vorgängerbildes verändert hat.

5. DIVIDED

Divided steht für

Distributed

Video

Editingsystem

und stellt ein Konzept für ein digitales Video-Editier- und -Schnittsystem auf Basis von MPEG dar.

Dieses Kapitel beschreibt die Lösungen grundsätzlicher Editier- und Zugriffsprobleme, die bei der Verwendung von MPEG bestehen, sowie die Umsetzung eines konventionellen Studio-Schnittsystemes mit seinen Komponenten in eine digitale Software-Entsprechung. Es gibt Hinweise auf Möglichkeiten zur Optimierung der Benutzerunterstützung durch die Verwaltung zusätzlicher Informationen und deren Integration in das Videodatenformat, die weit über die Möglichkeiten konventioneller Schnittsysteme hinaus gehen.

Das Konzept sieht eine Verwendung des Systemes im Online-Editing von MPEG-Videos sowie als Simulationssystem im Offline-Editing zum Erstellen von EDLs und deren spätere Verwendung in konventionellen, hochqualitativen Schnittsystemen vor. Es ermöglicht eine verteilte Realisierung des Systems und geht von einer Dienstleistungsorientierung aus, d.h., der Endbenutzer muß kein Experte sein.

5.1. VORTEILE DIGITALER INTEGRATION

Integriert man die Studiotchnik auf digitaler Basis in einem Computersystem, so bringt dies eine Reihe von Vorteilen:

Online- und Offline-Editing sind mit ein und demselben System möglich, da man sowohl die Verwaltung und Verarbeitung digital im Rechner gespeicherter Bilddaten vorsehen kann als auch die Ansteuerung externer Studiogeräte. Somit entfällt das Eingewöhnen auf zwei oder gar mehr Systeme.

Nonlineares Editing ist möglich, wenn man digitale Repräsentationen verwendet, die intern im Rechner gespeichert werden können. Somit sind ein wahlfreier Zugriff und kurze Antwortzeiten gesichert.

Orts- und Zeitunabhängigkeit, insbesondere von der Verfügbarkeit eines Videostudios sind gegeben, sobald die Daten einmal in maschinenlesbarer Form vorliegen. Sie können dann zusammen mit einem tragbaren Rechner transportiert und genutzt werden oder aber über ein Datennetz zu einem Verarbeitungsrechner nach Wahl übertragen werden. Das Videomaterial ist somit (fast) überall verfügbar, ähnlich wie heute schon Text und Photomaterial mit jedem Faxgerät übertragen werden können.

Die Qualität der verwendeten Videodarstellungen ist skalierbar, d.h., man kann die Darstellung und Speicherung dynamisch den aktuellen Erfordernissen anpassen.

Eine weitgehendere Herstellerunabhängigkeit als bei herkömmlichen Systemen, wird durch die Verwendung eines standardisierten Videodatenformates sowie durch eine auf Software basierende Systemintegration erreicht(vgl. Kap. 3.4.4.).

Dem Benutzer kann innerhalb eines volldigitalen Systems eine intelligenterere Unterstützung zur Durchführung seiner Aufgabe angeboten werden, indem nicht nur einfach eine breite Funktionalität zur Verfügung gestellt wird, sondern diese auch verständlich präsentiert und kontextorientiert ist.

Ein integriertes System bietet die Möglichkeit, neben den reinen Bild- (und Ton-) Informationen auch alle weiteren, beim Editieren eines Videos anfallenden und notwendigen Informationen (vgl. Kap. 3.5., Kap. 5.5.) zu bearbeiten und auszuwerten. Letztendlich bietet nur eine integrative Lösung die Möglichkeit des „Videoretrieval“, also der Wiedergewinnung videoteknisch gespeicherter Informationen.

Eine integrierte Computerlösung bietet also die größtmögliche Flexibilität, da sie die Möglichkeiten eines herkömmlichen Studios mit denen einer modernen Rechenanlage vereint. Dies ermöglicht z.B. das Offline-Editing und sein bequemes nichtlineares Zugriffsverhalten mit dem visuellen Feedback des Online-Editings zu kombinieren.

5.2. MOTIVATION DES SYSTEMS

Dem Entwurf des Konzeptes standen neben den konkreten Realisationskonzeptideen (vgl. Kap. 5.3.) auch einige wesentliche generellere Motivationen zugrunde.

Die Herabsenkung der Hemmschwelle zur Benutzung der Videotechnologie

Der Umgang mit den komplizierten und aufeinander abgestimmten Geräten in einem Videostudio erfordert einen gewissen Aufwand an Zeit und Energie zur Einarbeitung. Auch danach ist es sehr umständlich, jedesmal zur Bearbeitung eines Videos das Videostudio aufzusuchen.

Die Idee ist daher, das Studio auf den „Schreibtisch“ (also in die Workstation) zu bringen, um ein Arbeiten in gewohnter Umgebung und mit gewohnter Technik, zumindest als Vorbereitung auf den Umgang mit der im Ergebnis (vorerst noch) qualitativ besseren herkömmlichen Video-Technik, zu ermöglichen.

Die Ermöglichung intuitiveren Arbeitens mit Video

Aufgrund der linearen Arbeitsweise im Online-Schnittbetrieb und des fehlenden Feedbacks im Offline-Editing ist eine gewisse Erfahrung nötig, um schnell zu brauchbaren Ergebnissen zu kommen. Gewährleistet man einen schnellen Zugriff mit Wysiwyg-Effekt² ist ein schnelleres Erlernen der notwendigen Techniken möglich. Die Idee ist also hier die des „learning by doing“.

Die Reduzierung des Arbeitsumfeldes

Die dritte grundsätzliche Überlegung war, daß die Videobearbeitung für Präsentationszwecke in Zukunft in den Arbeitsbereich eines jeden Einzelnen fällt und zur alltäglichen Aufgabe, wie das Verfassen von Texten für Veröffentlichungen, wird.

Idealerweise soll dies ohne eine Anhäufung von weiteren kompliziert zu bedienenden und fehlerempfindlichen Gerätschaften erfolgen. Die Möglichkeiten zur Realisierung eines entsprechenden Werkzeuges sind durch das Vorhandensein der Arbeitsplatzrechner gegeben und erlauben somit eine Integration der neuen Präsentationstechniken in die vorhandene Arbeitsumgebung.

2. Engl.: Akronym für „What You See Is What You Get“ und bedeutet, daß der optische Eindruck der Bildschirmdarstellung der des Endproduktes entspricht

5.3. SYSTEM-KONZEPTE

Verteiltes System

Das System selbst muß in einem Netzwerk (homogen wie heterogen) lauffähig sein, da ja eine der Systemanforderungen ist, daß „das System zum Benutzer kommt“ und somit durch ein Netzwerk auf jeder Workstation gestartet werden kann.

Die Umsetzung der Studioarchitektur in ein integriertes System bietet die Möglichkeit der Parallelisierung der einzelnen Softwarekomponenten. Darauf aufbauend kann auch eine verteilte Ausführung auf verschiedenen Maschinen innerhalb eines Rechnernetzes realisiert werden.

Der Zugriffsmechanismus des Systems auf die Videodaten soll in jedem Falle eine verteilte Datenhaltung vorsehen. So steht sehr bald die Realisation verschiedener sog. Videoserverten an, die Videodaten über ein Datennetzwerk anbieten.

(vgl. [Krug94], [Rowe91], [Rowe92])

Videodaten

Idealzustand wäre, wenn es gelänge, alle Arten von Videoquellen zu nutzen, also

- digitales Video
- analoges Video
- aufgezeichnetes / gespeichertes Video
- Live-Video

Das bedeutete die Integration sehr verschiedener Technologien (vgl. [Lieb91]).

Einen guten Ansatz dazu stellt die Editierung von Videos nach dem MPEG-Standard dar. MPEG stellt in der Zukunft wohl den allgemeinen digitalen Videostandard (vgl. Kap. 4.1.). Somit ist eine Editierbarkeit von MPEG-Videos eine Grundvoraussetzung für diese Integration. Außerdem muß das System in der Lage sein, Timecode zu verarbeiten.

Unterstützung des „handwerklichen“ (technischen) Aufgabenteils des Cutters

Das System muß einen wahlfreien Zugriff auf die einzelnen Bilder einer Bildfolge gewährleisten. Da MPEG dies aufgrund seines Codierungsverfahrens nicht direkt zuläßt, muß eine Methode entwickelt werden, die eine möglichst effiziente Decodierung und Darstellung jedes beliebigen Bildes leistet.

Die Darstellung des Quellmaterials sowie der Zielsequenz soll in Echtzeit erfolgen. Sollte dies nicht möglich sein, z.B. aus mangelnder Rechenleistung, muß der Anwender die Möglichkeit erhalten, aus mehreren denkbaren Lösungen wie

- Reduzierung der zeitlichen Auflösung
- Reduzierung der Ortsauflösung
- Reduzierung der Bildqualität
- Reduzierung der Bildgröße
- oder auch gar keiner Maßnahme

die ihm am geeignetsten erscheinende zu wählen.

Es soll möglich sein, eine beliebige Anzahl von „Playern“, d.h. Quellsequenzen gleichzeitig darzustellen. Dabei muß es sowohl möglich sein, diese Player parallel und synchronisiert (z.B. nach dem Timecode), also mit einer Steuerung, zu bedienen, wie es zum Vergleich verschiedener Kamerapositionen oder Szeneneinstellungen nötig ist, als auch jeden Player separat bedienen zu können.

Das System muß in der Lage sein, die verschiedenen Probleme, die beim Editieren (d.h. dem Generieren eines neuen Videos aus Teilen von alten) von MPEG entstehen, zu handhaben. Insbesondere sind hier zu erwähnen:

- Bildgenaues Schneiden und Aneinanderfügen von MPEG-Sequenzen trotz DPCM-Codierung
- Gewährleistung der Bedingungen eines „Constraint Bitstreams“ für das generierte Video. Dies betrifft insbesondere die Datenübertragungsraten und das Buffermanagement.
- Das Zusammenschneiden von „inkompatiblen“ MPEG-Videos, z.B.
 - Videos mit verschiedenen Bildgrößen,
 - verschiedenen Bildwiederholungsraten,
 - unterschiedlicher Genauigkeit (und somit Reichweite) der Referenzvektoren.

(vgl. Kap. 5.4.)

Unterstützung des kreativen Teiles der Aufgabenstellung des Cutters

Das System soll eine Informationsverwaltung und Unterstützung anbieten. Alle auf das Videomaterial bezogenen Informationen sollen auch mit diesem zusammen abgespeichert werden und synchron mit diesem wieder abzurufen sein. MPEG bietet hierzu alle Möglichkeiten (vgl. Kap. 5.5.).

Die Suche nach geeigneten Szenen aus dem Quellmaterial muß durch geeignete Abfragemöglichkeiten unterstützt werden. Dazu sollen alle zur Verfügung stehenden Informationen genutzt werden. Insbesondere sind dies:

- die manuell eingegebene Information des Cutters (Notizen, Daten, ...), die mit dem Bildmaterial gespeichert wurde,
- automatisch aus dem Bildinhalt zu gewinnende Informationen wie:
 - Farbanteile
 - Kontraste
 - Ähnlichkeiten mit gegebenen Bildern
 - Änderungen des Bildinhaltes, also Erscheinen oder Verschwinden eines Objektes, oder Größe, Richtung und Geschwindigkeit eines sich bewegenden Objektes.

Das System soll Entscheidungshilfen für die Schnittfolgen geben, indem es den Überblick über bereits getroffene Entscheidungen ermöglicht, Previews erlaubt, eine Versionsverwaltung anbietet. Denkbar ist auch, daß grundlegende stilistische Kenntnisse über ein Expertensystem zu Ratschlägen an den Benutzer verarbeitet werden.

5.4. METHODEN ZUM EDITIEREN VOM MPEG

Im folgenden werden die entwickelten Methoden und Algorithmen, die speziell bei der Verwendung von MPEG zur Darstellung und Editierung von Videosequenzen notwendig sind, beschrieben.

5.4.1. WAHLFREIER ZUGRIFF

Beim Editieren eines Videos ist es immer nötig, bestimmte Stellen der Bildsequenz, d.h. bestimmte Bilder wie z.B. Cut-In oder Cut-Out Positionen, direkt anzusprechen und darzustellen. Auch bei der Materialdurchsicht ist ein solches Vorgehen nützlich, um „durch das Material zu springen“.

Der direkte wahlfreie Zugriff auf ein beliebiges Bild in einer MPEG-codierten Sequenz z.B. zur Darstellung des Bildes oder zum Schnitt ist nicht immer gewährleistet. Zwar ist die Position eines jeden Bildes innerhalb des Streams aufgrund der MPEG-Startcodes leicht zu ermitteln, doch viele der zum Decodieren benötigten Informationen sind an anderer Stelle (vorher, aber unabhängig von der Position des Bildes) des MPEG-Streams gespeichert.

Zur Lösung sei vorausgesetzt, daß die Position des gewünschten Bildes bekannt ist und in geeigneter Form angegeben werden kann, z.B. durch den Timecode, eine Relativposition (z.B. 10 Bilder weiter), die Angabe von Metainformation. Daraufhin wird der MPEG-Videostream nach Picture-Start-Codes durchsucht. Dazu muß man auch die zugehörigen Informationen der übergeordneten Hierarchieebenen auffinden. Dies sind:

- Die Quantisierungsmatrizen der Sequence-Headers
- Die Timecodeangaben der GoP-Headers

Eine deutliche Beschleunigung beim Auffinden eines bestimmten Bildes ist durch die Erzeugung einer Sequenz- oder Clip-Datenbasis zu erreichen. So werden nach der Auswahl der Sequenz aus der Videobase alle für den Zugriff auf die einzelnen Bilder relevanten Daten extrahiert und in der Clip-Datenbank abgelegt. Zusätzlich können auch weitere Metainformationen über den Videostream abgelegt und zum schnellen Zugriff bereit gehalten werden (vgl. Kap. 5.5., Kap. 5.6.).

5.4.2. WAHLFREIE, DIREKTE DARSTELLUNG

Selbst wenn der wahlfreie Zugriff auf ein beliebiges Bild der Sequenz garantiert ist, kann auf Grund der in MPEG verwendeten DPCM-Codierung der direkte Zugriff auf ein beliebiges Bild die Berechnung anderer Bilder erfordern, bevor eine Darstellung des gewünschten Bildes möglich ist (nämlich dann, wenn es sich um ein intercodiertes Bild, also P- oder B-Bild, handelt).

Beispiel: In der Bildfolge

$$I_0 B_1 B_2 P_3 B_4 B_5 P_6 B_7 B_8 I_9$$

macht der direkte Zugriff auf B_7 die vorherige Berechnung der Bilder

$$I_9, I_0, P_3 \text{ und } P_6$$

notwendig.

Das kann, insbesondere wenn in der MPEG-Sequenz sehr lange GoPs codiert wurden, zu langen Zugriffszeiten führen, falls kein intelligentes Zugriffsverhalten realisiert wird.

Handelt es sich bei dem gesuchten Bild um ein I-Bild, so ist es sofort darstellbar, da es ja keine Referenzen auf andere Bilder enthält. Da die Verteilung der Bildtypen in einem typischen MPEG-Video ungefähr einem Verhältnis von 1I:6P:12B entspricht (vgl. [Rowe93a]), wird es sich in der Mehrzahl der Fälle bei einem wahlfreien Zugriff um ein intercodiertes, also ein P- oder B-Bild handeln. Die somit notwendigen Berechnungen der Referenzbilder können durch Anwendung folgender Methoden beschleunigt werden:

- Interpolation
- Selektive Rückberechnung
- Kontextorientierung

5.4.2.1. Interpolation

Es ist nicht nötig, jedes Referenzbild vollständig zu decodieren.

Man kann hier die Voraussetzung zur Codierung von Videomaterial nach MPEG verwenden, nämlich, daß es sich um „natürliche, kontinuierliche“ Bilder handeln sollte, da nur hier gute Ergebnisse der DCT-Transformation und Quantisierung zu erwarten sind. Diese Kontinuität innerhalb eines Bildes erlaubt auch die Interpolation des Bildinhaltes von weniger Daten, wenn man dabei eine Qualitätseinbuße in Kauf nimmt.

Im Detail:

Jedes intercodierte Bild enthält auch nicht-intercodierte, also intracodierte Macroblocks. Dies kann auf zwei Weisen verwendet werden:

1. Je nach geforderter Bilddarstellungsqualität und der Verteilung solcher Macroblocks im Bild läßt sich eine Rückberechnung evtl. ganz vermeiden, indem man aus den vorhandenen intracodierten Macroblocks das Gesamtbild interpoliert. Dazu muß eine gleichmäßige Verteilung der Intra-Macroblocks über das Bild gegeben sein. Die Bestimmung der Dichte der Verteilung und damit der Qualität der Darstellung sollte dem Benutzer überlassen sein.
2. Stößt man bei der Rückverfolgung der Referenzvektoren auf einen intracodierten Macroblock, kann die Rückverfolgung auch dann abgebrochen werden, wenn die Referenz nicht exakt auf diesen Macroblock ausgerichtet ist, sondern nicht-intracodierte Teile aus seiner Umgebung mit einschließt. Dabei ist eine Extrapolation der im Macroblock vorhandenen Daten möglich.

Eine Kombination beider Techniken ist ebenfalls möglich. Ein Referenzbild wird aus seinen Intra-Blocken interpoliert und löst damit alle Referenzen des darzustellenden Bildes auf. Somit läßt sich die Referenzierungstiefe im wahlfreien Zugriff begrenzen.

Die so gewonnene Darstellungsqualität des Bildes ist auf jeden Fall für einen ersten Blick darauf geeignet. Sofort nach dieser ersten Darstellung kann mit einer exakten Nachberechnung und Neudarstellung des Bildes begonnen werden, falls dieses lange genug zu sehen ist.

5.4.2.2. Selektive Rückberechnung

Durch eine dezidierte Rückverfolgung der Referenzvektoren der intercodierten Macroblocks kann man die Berechnung nichtreferenzierter Bildteile der Referenzbilder vermeiden.

Zur Berechnung eines Macroblocks kann die Berechnung eines, zweier oder vierer Macroblocks der Referenzbilder (R-Bild) notwendig werden (vgl. Abb. 5.4.a).

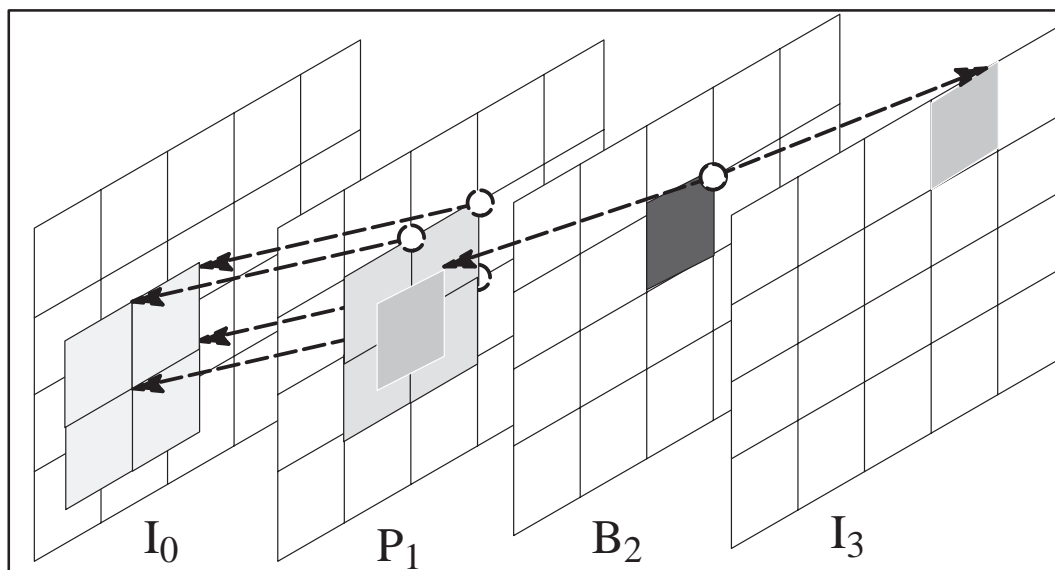


Abb. 5.4.a Die Referenzvektoren

Berechnet man im jeweiligen R-Bild nur die Macroblocks, die auch tatsächlich referenziert werden, ergibt sich eine Optimierung der Berechnungszeit sofort, falls dies nicht alle Macroblocks des R-Bildes sind.

Bei der Realisierung einer solchen selektiven Rückberechnung ist zu beachten, daß auch innerhalb eines Bildes Teile nebeneinanderliegender Macroblocks DPCM-codiert sind (z.B. Macroblockadresse und Referenzvektoren). Somit wird also für die Berechnung eines beliebigen Macroblocks des R-Bildes evtl. die (teilweise) Berechnung vorhergehender Macroblocks desselben Bildes notwendig. Da die Referenzvektoren eines Bildes jedoch auf beliebige Stellen des R-Bildes verweisen dürfen, wird eine zweistufige Referenzauflösung nötig.

In der ersten Phase wird das zu decodierende, aktuelle Bild (A-Bild) nach Referenzvektoren abgesucht und diese in der Reihenfolge der referenzierten Macroblocks des R-Bildes gespeichert. In der zweiten Phase werden nur diese benötigten Macroblocks nach dem gleichen Verfahren (rekursiv³), aber in der Reihenfolge ihrer Abhängigkeiten decodiert und stehen als Referenz zur Verfügung.

Danach können die Macroblocks des A-Bildes decodiert und dargestellt werden.

Eine weitere Optimierung ergibt sich durch eine Parallelisierung der Bildberechnung, da die Decodierung der benötigten Macroblocks des R-Bildes gleichzeitig mit der Decodierung der intra-codierten Macroblocks des A-Bildes erfolgen kann.

Im Rahmen einer Kontextorientierung (vgl. 5.4.2.3.) ist es sinnvoll, die bereits decodierten Macroblocks eines R-Bildes zu speichern und in einer Liste zu verwalten, so daß sie bei späteren Referenzen durch andere A-Bilder direkt verwendet werden können.

5.4.2.3. Kontextorientierung

Häufig ist das Arbeiten auf einen relativ kleinen lokalen Bereich des Videomaterials beschränkt, z.B. wenn ein Sourceclip definiert ist und Synchronisationspunkte gesucht werden. In einem solchen Fall kann man den Clip als Kontext definieren, in dem gearbeitet wird.

Innerhalb eines Kontextes kann das System von sich aus alle Referenzbilder decodieren und bereithalten. Damit wird ein schneller Zugriff gewährleistet. Als Varianten dieses Vorgehens können in einem sehr großen Kontext oder auf einer kleinen Maschine nur die I-Bilder in decodierter Form bereitgehalten werden sowie ein relativer Kontext in der direkten Umgebung des gegenwärtigen Editiervorgangs mit allen Referenzbildern geschaffen werden.

Das Erkennen eines Kontextes sowie das Anlegen der decodierten Referenzbilder kann vom System selbst vorgenommen werden, indem es auf die Häufigkeit und Art der Zugriffe auf das Videomaterial achtet.

5.4.3. VARIABLE GESCHWINDIGKEIT: VORWÄRTS

Beim Editieren eines Videos wird zum Auswählen geeigneter Stellen die Möglichkeit zum Betrachten mit variabler Geschwindigkeit benötigt. Wir betrachten hier zuerst den einfacheren Fall des vorwärts „laufenden Bandes“.

Dieser Fall ist deswegen einfacher zu handhaben, weil die Vorwärtsrichtung der DPCM-Codierungsrichtung entspricht. Man kann davon ausgehen, daß alle nötigen Informationen und Referenzbilder zur Darstellung eines bestimmten Bildes zu dessen Darstellungszeit bereits vorhanden sind. Außerdem ist die von MPEG vorgesehene Puffergröße vollkommen ausreichend zur Decodierung aller Bilder.

3. Als Abbruchbedingung der Rekursion steht hier das Auftreten eines intracodierten Macroblocks, der selbst keine Referenzvektoren definiert. Ein solcher Macroblock wird immer spätestens im nächsten referenzierten I-Bild gefunden.

Das einfache Abspielen wird als Spezialfall gesehen. Definiert man eine Skala, welche die Abspielgeschwindigkeit als Vielfaches der normalen Bildrate angibt, so erhält man folgende Tabelle

Bildrate		Geschwindigkeit
Normal	(25 Bilder/sec)	1
Geringer	(<25 Bilder/sec)]0,1[
Höher	(>25 Bilder/sec)]1,Obergrenze]

Im Gegensatz zur Analogtechnik ist die max. Geschwindigkeit im Prinzip nicht nach oben begrenzt. Jedoch macht eine allzugroße Geschwindigkeit keinen Sinn, da dann das Erkennen der Bildinhalte nicht mehr möglich ist. Ab einer gewissen Geschwindigkeit kann also auf die Darstellung verzichtet werden.

Wir gehen davon aus, daß die Wiedergabe mit normaler Geschwindigkeit ($\cong 25$ Bildern/sec) möglich ist. Dann ist selbsterklärend auch die Wiedergabe mit geringeren Geschwindigkeiten (und somit Bildraten) möglich.

Die heutigen Rechenanlagen sind nicht in der Lage, die Decodierung und Darstellung des MPEG-Streams für beliebig hohe Geschwindigkeiten zu leisten. Das gilt sowohl bei rein in Software realisierten MPEG-Systemen wie auch für Hardware-Systeme, wenngleich die Leistungsraten hier sicherlich wesentlich höher anzusetzen sind. Daher muß ein Betrachten des Videos mit Geschwindigkeiten $\gg 1$ durch geeignete Maßnahmen „simuliert“ werden.

Zur Lösung müssen Methoden angewendet werden, die den Decodier- und Darstellungsaufwand reduzieren.

Bei einer schnelleren Bildfolge, also einer schnelleren Bewegung, ist das menschliche Auge nicht mehr in der Lage, die volle Bildauflösung zu erfassen. Es interpoliert Gesehenes über den Bewegungseindruck. Das gilt sowohl für die Ortsauflösung der einzelnen Bilder als auch für die temporäre Auflösung der Bildfolge (vgl. [Hoffm92], [Kröm91]). Es ergeben sich somit zwei prinzipielle Lösungen:

- Reduzierung der Ortsauflösung
- Reduzierung der Zeitauflösung

5.4.3.1. Reduzierung der Ortsauflösung

Bei der Reduzierung der Ortsauflösung erhöht man die Bildrate. Anstatt jedoch die Bilder in voller Qualität darzustellen, verwendet man die Eigenschaft der DCT-Codierung, welche die meiste Information im ersten Koeffizienten, dem DC, codiert. Es werden nur die DC-Koeffizienten der Blocks im Bild decodiert und zur Darstellung verwendet. Bei nicht zu hohen Geschwindigkeiten und entsprechender Leistungsfähigkeit des Systems können auch noch weitere Koeffizienten hinzugenommen werden. Wie Wallace in [Wall91] gezeigt hat, ist auch bei der Nutzung nur eines Koeffizienten (DC), also der faktischen Reduktion der Bildauflösung um den Faktor 8 in beiden Richtungen, noch eine erstaunlich gute Bildqualität zu erzielen (vgl. Abb. 5.4.b).

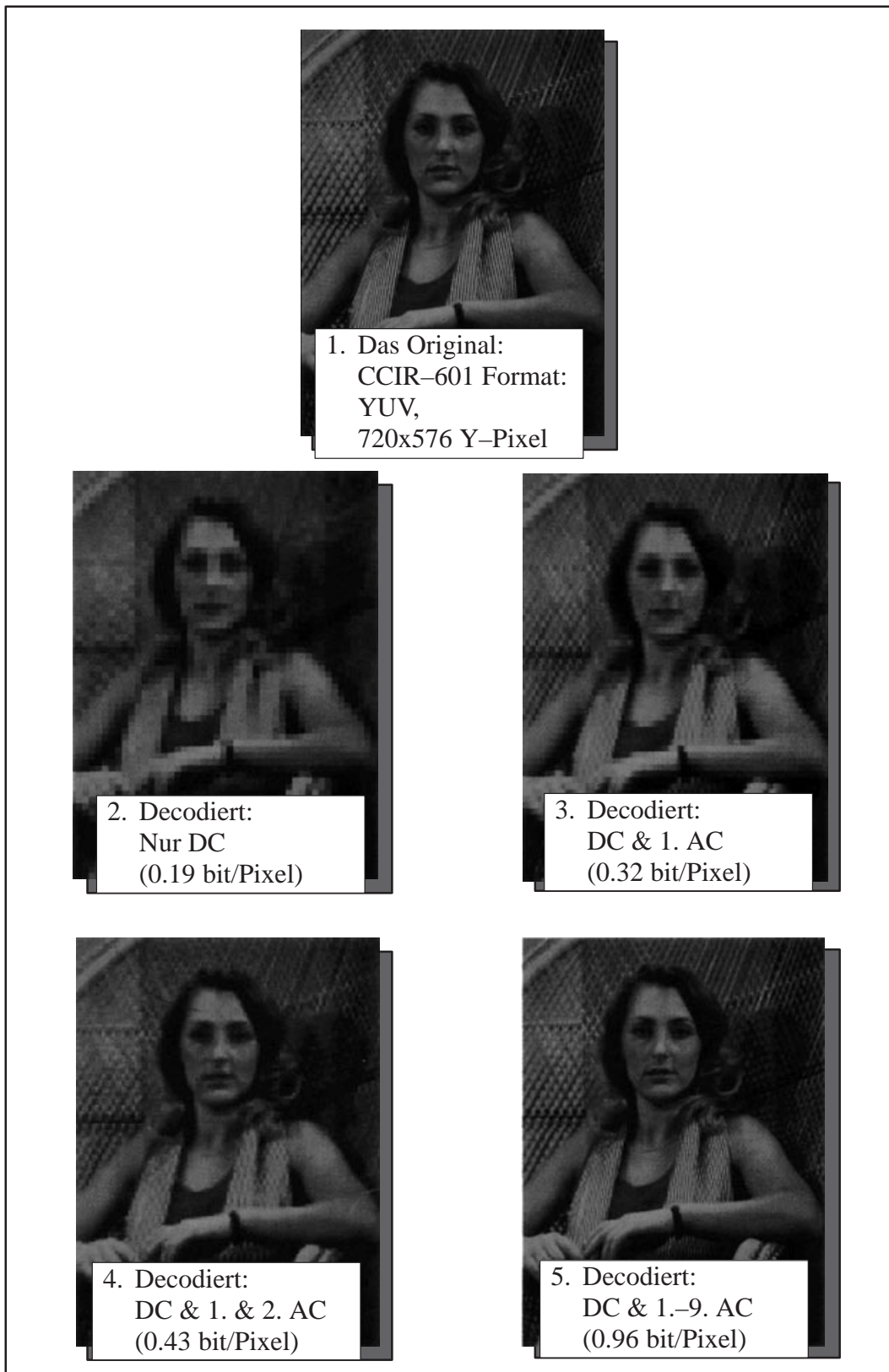


Abb. 5.4.b Die Bildqualität bei Verwendung weniger DCT-Koeffizienten [Wall91])

5.4.3.2. Reduzierung der zeitlichen Auflösung

Bei der Reduzierung der zeitlichen Auflösung hält man die Bildrate konstant, stellt aber nicht mehr jedes Bild dar. D.h., bei z.B. 3-facher Geschwindigkeit zeigt man nur noch jedes dritte Bild an. Dieses Verfahren birgt jedoch einige Nachteile, die es nur sehr eingeschränkt nutzbar machen:

Beim Überspringen von Bildern entsteht das *Problem des wahlfreien Zugriffs*, der im Konflikt steht mit der in MPEG angewandten DPCM-Codierung der Interbilder (vgl. Kap. 5.4.1., Kap. 5.4.2.). D.h., obwohl man die Bildrate auf dem Originalwert von 25 Bildern/sec hält und somit durchaus noch die Möglichkeit hat, einstufige Referenzen in Echtzeit aufzulösen (wie dies bei normaler, einfacher Geschwindigkeit ja auch nötig ist), kann es besonders bei höheren Geschwindigkeiten vorkommen, daß mehrstufige Referenzen zur Darstellung des nächsten Bildes aufgelöst werden müssen.

Beispiel: Gegeben sei folgende Bildfolge in Darstellungsreihenfolge:

$$B_{-1}I_0B_1P_2B_3P_4B_5P_6B_7I_8B_9P_{10}B_{11}P_{12}B_{13}P_{14}B_{15}I_{16}B_{17}P_{18}B_{19}P_{20}B_{21}P_{22}B_{23}I_{24}$$

Das entspricht folgender Speicherungsreihenfolge:

$$I_0B_{-1}P_2B_1P_4B_3P_6B_5I_8B_7P_{10}B_9P_{12}B_{11}P_{14}B_{13}I_{16}B_{15}P_{18}B_{17}P_{20}B_{19}P_{22}B_{21}I_{24}B_{23}$$

Bei 3-facher Geschwindigkeit ergibt sich folgendes Darstellungsmuster:

$$I_0B_3P_6B_9P_{12}B_{15}P_{18}B_{21}I_{24}$$

Zu decodieren sind nunmehr:

$$\langle I_0 \rangle \langle P_2P_4B_3 \rangle \langle P_6 \rangle \langle I_8P_{10}B_9 \rangle \langle P_{12} \rangle \langle P_{14}I_{16}B_{15} \rangle \langle P_{18} \rangle \langle P_{20}P_{22}B_{21} \rangle \langle I_{24} \rangle$$

Im Falle sämtlicher darzustellender B-Bilder bedeutet dies schon eine Referenzierung über 2 Ebenen in Echtzeit. Setzt man als Geschwindigkeit 5 an, ergibt sich folgendes Darstellungsmuster

$$I_0B_5P_{10}B_{15}P_{20}B_{25}$$

mit folgendem noch schlechterem Decodiermuster

$$\langle I_0 \rangle \langle P_2P_4P_6B_5 \rangle \langle I_8P_{10} \rangle \langle P_{14}I_{16}B_{15} \rangle \langle P_{18}P_{20} \rangle \langle I_{24}P_{26}B_{25} \rangle$$

das zur Darstellung von B₅ die Berechnung von 4 Referenzierungsebenen erfordert.

Je nach der Reihenfolge und Regelmäßigkeit der gegebenen MPEG-Sequenz können beliebig viele Referenzebenen aufzulösen sein, bevor das Bild darstellbar ist. Aus diesem Grund müßten auch in diesem Fall die Methoden der wahlfreien Darstellung (vgl. Kap. 5.4.2.) oder zusätzlich auch eine Ortsauflösungsreduktion (vgl. Kap. 5.4.3.1.) angewandt werden.

Das Überspringen vom Bildern birgt noch einen weiteren Nachteil:

Bei relativ großen Geschwindigkeiten, also bei großen Lücken in der dargestellten Bildfolge, werden die Bewegungen sprunghaft und der Zusammenhang wird schwerer zu erkennen, da die *Kontinuität der Bilder nicht mehr gegeben* ist.

5.4.4. VARIABLE GESCHWINDIGKEIT: RÜCKWÄRTS

Ebenso wie vorwärts wird auch eine Möglichkeit benötigt, die Bildfolge rückwärts mit variabler Geschwindigkeit zu betrachten. Dies ist zum genauen Justieren einer Schnittposition unerlässlich.

Sowohl die DPCM-Codierung der MPEG-Sequenz als auch die Speicherungsreihenfolge der einzelnen MPEG-Bilder ist auf eine strikt vorwärts gerichtete Wiedergabe des Videomaterials ausgelegt, d.h., daß

1. zur Decodierung eines DPCM-Bildes das Referenzbild immer *davor* liegt, also schon decodiert ist, und
2. das Referenzbild in der Speicherungsreihenfolge immer auch *vor* dem DPCM-Bild abgelegt ist, also auch zum richtigen Zeitpunkt zur Verfügung steht.

Ein Rückwärts-Abspielen einer MPEG-Sequenz stellt deshalb ein prinzipielles Problem dar. Dazu kommt die Anforderung einer variablen Geschwindigkeit, besonders einer erhöhten Geschwindigkeit.

Zur Lösung setzen wir voraus, daß auf alle Bilder wahlfrei zugegriffen werden kann (vgl. Kap. 5.4.1.).

Zur Darstellung eines beliebigen Bildes müssen alle seine Referenzbilder bereits decodiert sein, falls man keine Qualitätseinbuße in entweder Orts- oder Zeitauflösung in Kauf nehmen will (vgl. Kap. 5.4.2.). Zumindest für das Abspielen mit einfacher Geschwindigkeit soll dies auch rückwärts der Fall sein. Daraus ergibt sich, daß man quasi eine Umsortierung der Sequenz vornehmen muß, um die benötigten Bilder zur richtigen Zeit zur Verfügung zu haben. „Quasi“ deshalb, weil man aufgrund des wahlfreien Zugriffs nicht tatsächlich umsorthern muß, sondern lediglich eine geänderte Zugriffsreihenfolge festlegt.

Beispiel: Die MPEG-Sequenz

$$I_0 B_1 P_2 B_3 P_4 B_5 P_6 B_7 I_8 B_9 P_{10} B_{11} P_{12} B_{13} P_{14} B_{15} I_{16}$$

mit der Speicher- bzw. normalen Zugriffsreihenfolge

$$I_0 P_2 B_1 P_4 B_3 P_6 B_5 I_8 B_7 P_{10} B_9 P_{12} B_{11} P_{14} B_{13} I_{16} B_{15}$$

Soll nun rückwärts, also in Reihenfolge

$$I_{16} B_{15} P_{14} B_{13} P_{12} B_{11} P_{10} B_9 I_8 B_7 P_6 B_5 P_4 B_3 P_2 B_1 I_0$$

abgespielt werden. Dazu ist eine Decodierung und somit ein Zugriff in der Reihenfolge

$$I_{16} I_8 P_{10} P_{12} P_{14} B_{15} B_{13} B_{11} B_9 I_0 P_2 P_4 P_6 B_7 B_5 B_3 B_1$$

notwendig.

Dies ist leicht ersichtlich, weil zum Darstellen des ersten B-Bildes B_{15} bereits P_{14} decodiert sein muß, und damit auch dessen Referenzbilder, also P_{12} , P_{10} und I_8 .

Da alle diese Referenzbilder (P_{14} - I_8) später selbst noch zur Darstellung gelangen und von weiteren B-Bildern referenziert werden, müssen sie solange zwischengepuffert werden. Damit ist ein Einhalten der durch MPEG garantierten Maximalpuffergröße von drei Bildern nicht mehr einzuhalten.

Dennoch kann mit Hilfe des folgenden Puffermechanismus eine maximale Puffergröße in Abhängigkeit der Struktur der MPEG-Sequenz gegeben werden. Sie beträgt:

$$Buf_{max} = 2 + \max(PzI) + 1 \quad (Gl. 5.1)$$

wobei

PzI = Anzahl der P-Bilder zwischen 2 aufeinander folgenden I-Bildern

Beweis:

Es sind 3 Fälle zu unterscheiden:

1. Die MPEG-Sequenz enthält nur I-Bilder.

Damit ist jedes Bild sofort ohne Referenzauflösung decodierbar und darstellbar. Die benötigte Puffergröße ist somit

$$Buf_{Tat} = 1 \leq Buf_{Max}$$

2. Die MPEG-Sequenz besteht aus I- und P-Bildern.

Wir setzen o.B.d.A. die Indizes der Bilder so fest, daß in normaler Abspielreihenfolge, also vorwärts, die Indexzählung bei einem I-Bild mit 0 beginnt, und dann bis zum nächsten I-Bild mit jedem P-Bild um 1 steigt. Hat das zweite I-Bild also Index n , so liegen zwischen beiden Bildern $n-1$ P-Bilder (Indizes $1 - n-1$).

Zur Decodierung des Bildes P_i muss Bild $i-1$ ($1 < i < n-1$) bereits decodiert sein. Dies setzt sich fort bis zu Bild I_0 , das ohne Referenzen zu decodieren ist. Spielt man die Sequenz rückwärts ab, ist I_n direkt decodier- und darstellbar. Um P_{n-1} zu decodieren und darzustellen, müssen zunächst alle Bilder $I_0 - P_{n-2}$ decodiert und gepuffert werden, anschließend wird P_{n-1} decodiert und dargestellt, und dann kann die Sequenz der nun schon decodierten Bilder P_{n-2} bis I_0 dargestellt werden.

Kein Bild muß *nach* seiner Darstellung noch gepuffert werden, da nun bereits alle Referenzen aufgelöst sind.

Die benötigte Puffergröße Buf_{Tat} beträgt also ein I-Bild + $n-1$ P-Bilder

$$Buf_{Tat} = 1 + \max(PzI) \leq Buf_{Max}$$

3. Die MPEG-Sequenz enthält I-, P-, und B-Bilder.

Die Decodierung und Darstellung der I und P-Bilder kann unabhängig von der der B-Bilder erfolgen, da weder I- noch P-Bilder Daten der B-Bilder benötigen. Hier bleiben also die Überlegungen aus 2. gültig.

Das Indizierungsschema soll nun auf die B-Bilder erweitert werden. Dazu bleiben die Indizes aus 2. für die I- und P-Bilder erhalten. Die nun zwischen diesen eingetretenen B-Bilder erhalten einen Doppelindex, dessen erste Komponente den Index des letzten P- oder I-Bildes wiedergibt und dessen zweite Komponente bei 0 beginnend die B-Bilder bis zum nächsten P- oder I-Bild in normaler Abspielrichtung zählt.

Liegen also zwischen Bild P_{n-1} und I_n m B-Bilder, so haben diese die Indizes $B_{n-1,0} - B_{n-1,m-1}$

Beim Rückwärts-Abspielen der Sequenz wird also nach I_n $B_{n-1,m-1}$ dargestellt.

Zur Decodierung eines beliebigen Bildes $B_{i,j}$ wird das vorangehende und das nachfolgende P- bzw. I-Bild (I/P) benötigt, also I/P_i und I/P_{i+1} , d.h., beide müssen bereits decodiert und gepuffert sein. Somit wird die Decodierung des Bildes P_{n-1} nicht erst zu dessen eigener Darstellungszeit erforderlich, sondern es muß schon zur Darstellungszeit von $B_{n-1,m-1}$ decodiert und gepuffert sein.

Die B-Bilder selbst werden nicht referenziert, und somit ist eine Pufferung nach ihrer Decodierung und Darstellung nicht nötig. Bild I/P_{i+1} muß nun allerdings auch nach seiner eigenen Darstellung noch gepuffert werden, bis alle B-Bilder mit Referenzen darauf ($B_{i,j} \ 0 \leq j \leq m-1$) decodiert und dargestellt sind. Da P-Bilder sowieso alle gepuffert werden müssen, ergibt sich nur für den Fall, daß Bild $i+1$ ein I-Bild ist, eine Puffervergrößerung zusätzlich zu dem einen Pufferplatz zur Decodierung des B-Bildes.

Somit benötigt man Buf_{Tat} mit 2 I-Bildern $n-1$ P-Bildern und 1 B-Bild.

$$Buf_{Tat} = 2 + \max(PzI) + 1 \leq Buf_{Max}$$

qed \square

Zudem kann mit folgendem Decodierungs- und Pufferungsschema nach einer Vorlaufzeit von maximal

$$V_{Max} = 2 + \max(PzI) \text{ Bildcodierungen} \quad (Gl. 5.2)$$

eine fortlaufene Darstellung in Echtzeit erfolgen.

Wir gehen davon aus, daß ein Pufferbereich in ausreichender Größe, also entsprechend Buf_{Max} aus (Gl. 5.1) vorhanden ist, in dem die decodierten Bilder gespeichert werden. Zusätzlich existiert wie im normalen MPEG-Decodierverfahren noch ein Darstellungsspeicher, in dem das Bild, welches gegenwärtig dargestellt wird, abgelegt ist.

Zur Erläuterung sei o.B.d.A. angenommen, daß eine regelmäßige MPEG-Sequenz vorliegt, d.h. insbesondere, daß die Anzahl der P-Bilder zwischen zwei I-Bildern stets die gleiche ist. Im allg. Fall ist durch die Wahl des $\max(PzI)$ garantiert, daß der P-Pufferbereich nicht überlaufen kann.

Die rückwärts darzustellende Sequenz habe I/P Indizes im Bereich von $0-p$, und die Anzahl der P-Bilder zwischen 2 I-Bildern sei q , so daß p ein Vielfaches von $(q+1)$ ist. Somit laufen die B Indizes im Bereich von $0, 0-p-1, r-1$, mit $r = \text{Anzahl der B-Bilder zwischen zwei aufeinanderfolgenden I/P-Bildern}$.

Weiterhin wird davon ausgegangen, daß es wie im normalen MPEG-Verfahren möglich ist, während ein Bild angezeigt wird, ein weiteres zu decodieren.

Die Bilder der Sequenz werden ihrem Typ entsprechend in drei Listen in absteigender Reihenfolge sortiert, also jeweils das Bild mit dem höchsten Index zuerst.

Der Pufferbereich wird ebenfalls in drei Bereiche der vorgesehenen Größe (vgl. (Gl. 5.1)) unterteilt:

$$\begin{array}{ll} 2 & \text{I-Bilder} \\ \max(PzI) & \text{P-Bilder} \\ 1 & \text{B-Bilder} \end{array}$$

Nun werden zunächst in einem *Vorlauf* die I- und P-Pufferbereiche aufgefüllt, in dem Bilder von Anfang jeweils ihrer Liste entnommen und decodiert werden. Somit sind nach diesem Vorlauf die beiden I-Bilder I_p und I_{p-q-1} sowie alle dazwischen liegenden P-Bilder $P_{p-q} - P_{p-1}$ decodiert.

Während der Darstellung von I_p kann das B-Bild $B_{p-1,r-1}$ (das erste der B-Liste) decodiert werden. Direkt nach der Decodierung gelangt dieses B-Bild zur Darstellung und gibt den Puffer frei, so daß das nächste B-Bild, $B_{p-1,r-2}$ decodiert werden kann. So werden alle hintereinander liegenden B-Bilder decodiert und dargestellt. Während der Darstellung von $B_{p-1,0}$ kann I_p , das nicht weiter referenziert werden wird, durch das nächste I-Bild der I-Liste ersetzt und dieses decodiert werden. Dann erfolgt die Darstellung des bereits decodierten P_{p-1} . Nun werden die Bilder $B_{p-2,r-1}$ bis $B_{p-2,0}$ in Folge geladen und decodiert. Während der Anzeige von $B_{p-2,0}$ kann P_{p-1} , das ebenfalls nicht weiter referenziert werden wird, durch das nächste *decodierbare* Bild der P-Liste ersetzt werden. Dies wird für die übrigen Bilder fortgesetzt.

Also wird immer während der Darstellung eines B-Bildes mit der 2. Indexkomponente 0 das nicht mehr benötigte I- oder P-Bild gegen das nächste decodierbare Bild der jeweiligen Liste ausgetauscht und dieses decodiert. Das ist möglich, weil nach dem Bild $B_{x,0}$ stets das bereits decodierte I/P_x zur Anzeige kommt (alle $B_{x,j}$ hatten es ja referenziert).

Hier ein Beispiel dieses Ablaufes:

Gegeben sei eine regelmäßige MPEG-Sequenz mit je 3 P-Bildern zwischen den I-Bildern und je 3 B-Bildern zwischen den I/P-Bildern.

Der untere Index der Bilder gibt die durchlaufende Nummerierung der Bilder in Vorwärtsrichtung an, der obere Index richtet sich nach der oben beschriebenen Indizierungsmethodik.

Es soll also folgende Darstellungsreihenfolge realisiert werden:

$$I_{48}^{12} B_{47}^{11,2} B_{46}^{11,1} B_{45}^{11,0} P_{44}^{11} B_{43}^{10,2} B_{42}^{10,1} B_{41}^{10,0} P_{40}^{10} \dots B_{33}^{8,0} I_{32}^{8} \dots$$

Dazu wird folgende Zugriffs- und Decodierungsreihenfolge definiert:

$$I_{48}^{12} I_{32}^{8} P_{36}^9 P_{40}^{10} P_{44}^{11} B_{47\dots45}^{11,2\dots11,0} I_{16}^4 B_{43\dots41}^{10,2\dots10,0} P_{20}^5 B_{39\dots37}^{9,2\dots9,0} P_{24}^6 \dots$$

Dieser Ablauf ist in der folgenden Tabelle (vgl. Seite 96) noch einmal dargestellt. Die rechte Seite gibt den Inhalt des Decodierpuffers wieder, die linke Spalte gibt das jeweils dargestellte Bild an. Die Einträge der Tabelle geben jeweils erst den Bildindex nach obiger Methodik und dann die fortlaufende Numerierung in normaler Abspielreihenfolge an.

Jede Zeile entspricht der Darstellungszeit eines Bildes, stellt also somit eine Taktung der Darstellung und Decodierung dar. Pro Zeile kann also max. je ein Bild in den Puffer geladen und decodiert werden. Bilder werden in den umrahmten Einträgen in den Puffer geladen und decodiert. Sie verbleiben dann im Puffer, solange sie zur Darstellung oder Referenzierung benötigt werden. Sowie das letzte referenzierende Bild decodiert ist, kann noch am Ende dieses Taktes der Pufferbereich des nunmehr unnötigen Referenzbildes freigegeben werden (in der Tabelle mit dem durchgestrichenen Eintrag symbolisiert).

Dargestelltes Bild	B-Puffer	P-Puffer			I-Puffer	
						12 48
Vorlauf					8 32	
				9 36		
				10 40		
			11 44			
12 48 I	11,2 47					
11,2 47 B	11,1 46					
11,1 46 B	11,0 45					12 48
11,0 45 B						4 16
11 44 P	10,2 43					
10,2 43 B	10,1 42					
10,1 42 B	10,0 41	11 44				
10,0 41 B		5 20				
10 40 P	9,2 39					
9,2 39 B	9,1 38					
9,1 38 B	9,0 37		10 40			
9,0 37 B			6 24			
9 36 P	8,2 35					
8,2 35 B	8,1 34					
8,1 34 B	8,0 33			9 36		
8,0 33 B				7 28		
8 32 I	7,2 31					
7,2 31 B	7,1 30					
7,1 30 B	7,0 29				8 32	
7,0 29 B					0 0	
7 28 P	6,2 27					
6,2 27 B	6,1 26					
6,1 26 B	6,0 25	5 20				
6,0 25 B		1 4				
6 24 P	5,2 23					
5,2 23 B	5,1 22					
5,1 22 B	5,0 21		6 24			
5,0 21 B			2 8			
5 20 P	4,2 19					
4,2 19 B	4,1 18					
4,1 18 B	4,0 17			7 28		
4,0 17 B				3 12		
4 16 I	3,2 15					
3,2 15 B	3,1 14					
3,1 14 B	3,0 13					4 16
3,0 13 B						
3 12 P	2,2 11					
2,2 11 B	2,1 10					
2,1 10 B	2,0 9	1 4				
2,0 9 B						
2 8 P	1,2 7					
1,2 7 B	1,1 6					
1,1 6 B	1,0 5		2 8			
1,0 5 B						
1 4 P	0,2 3					
0,2 3 B	0,1 2					
0,1 2 B	0,0 1			3 12		
0,0 1 B					0 0	
0 0 I						

5.4.5. SCHNITT VON MPEG-SEQUENZEN

„Schneiden“ bedeutet, daß aus einer Bildsequenz eine Teilsequenz verwendet wird, um zusammen mit anderen Teilsequenzen eine neue Bildfolge aufzubauen. In MPEG existiert die Möglichkeit, eine Sequenz zu Beginn einer „Group of Pictures (GoP)“ aufzutrennen (vgl. Kap. 4.4.4.9.). Sie ist idealerweise eine abgeschlossene Einheit, die keine Referenzvektoren auf Bilder außerhalb der Gruppe enthält⁴.

Beim Videoediting wird es jedoch fast immer notwendig sein, an einer beliebigen Stelle schneiden zu können, sei es nun aus gestalterischen Gründen (Szenenwechsel an einer bestimmten Stelle), aus Sachzwängen (vorgegebene Länge einer Videosequenz) oder aus Synchronisationsproblemen (Bild und Ton).

Aufgrund der in MPEG verwendeten DPCM-Codierung läßt sich die Sequenz nicht immer zwischen zwei beliebigen Bildern auftrennen, um sie in einem neuen Kontext wieder zu verwenden. Sind in einem intercodierten Bild Referenzvektoren definiert und wird das Referenzbild dann weggeschnitten bzw. durch ein Bild anderen Inhalts ersetzt, ist das intercodierte Bild nicht mehr (fehlerfrei) zu decodieren.

Zur Lösung wird zunächst ein Schnittmodell definiert (vgl. Abb. 5.4.c):

Eine Videoteilsequenz, bestehend aus einer beliebigen Anzahl Bilder und beginnend an einer beliebigen Stelle, wird aus einer größeren Videosequenz ausgewählt. Die Teilsequenz wird durch Kopieren an das Ende einer anderen Videosequenz angefügt (assemble cut) oder an einer beliebigen Stelle in eine andere Videosequenz eingefügt (insert cut). Dabei sollen alle Bilder der Teilsequenz auch in der neuen Videosequenz decodierbar sein, so daß eine Wiedergabe der neuen Gesamtsequenz möglich ist. Die ausgewählte und zu kopierende Teilsequenz soll mit Source und der schon vorhandene Teil der neuen Videosequenz soll mit Target bezeichnet werden.

Die Source hat genau zwei Schnittstellen:

den *Cut-In*, der durch ihr erstes Bild beschrieben ist und genau vor diesem ausgeführt wird, und

den *Cut-Out*, der durch ihr letztes Bild beschrieben ist und genau nach diesem ausgeführt wird.

4. Zumindest ist dies die Intention bei der Schaffung einer solchen „Random Access Unit“ gewesen. Tatsächlich ist es jedoch möglich, die GoP als „Open GoP“ zu kennzeichnen, so daß ein Editor erkennen kann, daß doch Referenzen auf vor der GoP liegende Bilder innerhalb der GoP vorhanden sind (z.B. falls sie mit B-Bildern beginnt). Ebenso kann man kennzeichnen, daß zwar Referenzvektoren aus der GoP heraus existieren, diese jedoch nicht mehr gültig sind, da die Bilder vor der GoP nicht mehr den Referenzbildern entsprechen (z.B. weil ein Schnitt ausgeführt wurde). Es handelt sich dann um einen sogenannten „Broken Link“

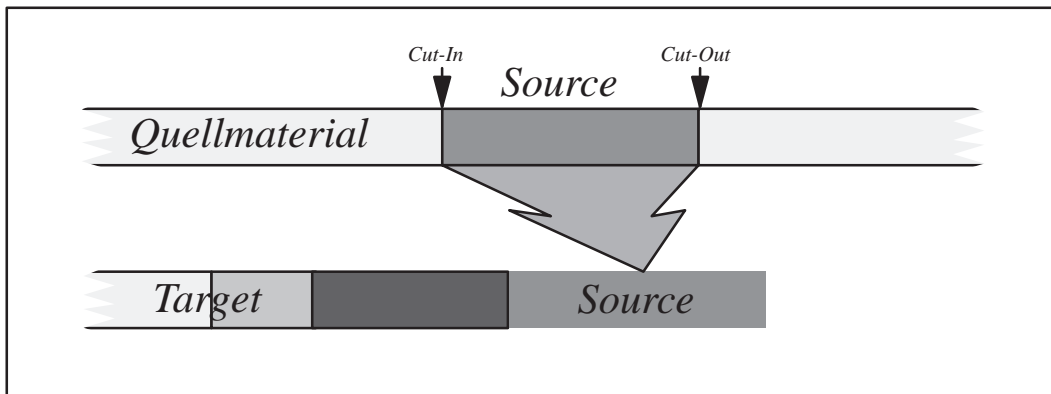


Abb. 5.4.c Die Modellierung des MPEG-Schnittes

Hier interessieren nur die Auswirkungen des Schnittes auf die ausgewählte Teilsequenz und nicht die Auswirkungen auf den verbleibenden Rest der Originalsequenz. Faktisch hat der Schnitt gar keine Auswirkungen auf das Original, da die Teilsequenz lediglich kopiert wird. Spezialoperationen wie das Kürzen einer Sequenz lassen sich mit dem Schnittmodell simulieren (vgl. Kap. 5.4.5.3. ff).

Der Cut-In kann nur „Vorwärtsreferenzen“ zerstören, d.h., sollten am Anfang der Source Bilder stehen, die ein Bild referenzieren, das vor dem Cut-In liegt, kann nicht davon ausgegangen werden, daß das entsprechende Bild des Target den gleichen Inhalt hat. Somit müssen diese Referenzen aufgelöst werden, und die entsprechenden Bilder müssen *ohne diese Referenzen neu codiert* werden, bevor sie in das Target eingesetzt werden.

Ebenso kann der Cut-Out nur „Rückwärtsreferenzen“ auf ein Bild nach dem Cut-Out zerstören. Hier gilt das o.g. entsprechend.

Alternativ könnte man versuchen, nach der Auflösung der zu schneidenden Referenzen, also der Decodierung der entsprechenden Bilder, diese *innerhalb des Targets* neu zu codieren. Jedoch kann sicherlich in der großen Mehrzahl aller Fälle davon ausgegangen werden, daß der Schnitt erfolgt, um einen Szenenwechsel zu realisieren, so daß die zur DPCM-Codierung nötige Korrelation der aufeinander folgenden Bilder an der Schnittstelle der Target-/Sourcesequenzen nicht gegeben ist und somit auch Referenzen zwischen diesen Bildern nicht sinnvoll sind.

Für beide Schnittstellen sind verschiedene Fälle zu betrachten:

5.4.5.1. Referenzvektoren am Cut-In

- Der Cut-In liegt auf dem Beginn der Originalsequenz.
Hier kann es keine „Vorwärtsreferenzen“ geben. Die Bilder können direkt kopiert werden.
- Der Cut-In liegt auf einem I-Bild.
Auch hier kann es keine Vorwärtsreferenzen vor diesem I-Bild geben, da alle nachfolgenden intercodierten Bilder sich nur auf dieses I-Bild beziehen können. Somit sind alle Referenzen inner-

halb der Source auflösbar, die Bilder können direkt kopiert werden.

- Der Cut-In liegt auf einem P-Bild.
Hier sind nur Vorwärtsreferenzen genau dieses Bildes betroffen, da alle nachfolgenden intercodierten Bilder sich auf dieses P-Bild beziehen müssen. Es ist also eine Auflösung der Vorwärtsreferenzen dieses P-Bildes nötig, was faktisch eine Umwandlung in ein I-Bild bedeutet (vgl. Kap. 5.4.6.).
- Der Cut-In liegt auf einem B-Bild.
Hier sind dieses B-Bild sowie alle unmittelbar folgenden B-Bilder betroffen, falls sie Vorwärtsreferenzen enthalten. Es besteht die Möglichkeit, entweder die Vorwärtsreferenzen aller betroffenen B-Bilder aufzulösen und sie somit in B-Bilder umzuwandeln, die nur Rückwärtsreferenzen enthalten (außer den Macroblocks die intracodiert sind), oder aber das erste B-Bild durch Auflösung aller Referenzen in ein I-Bild zu wandeln und die Vorwärtsreferenzen der übrigen B-Bilder auf dieses neue I-Bild umzurechnen. Dies ergibt dann einen Sinn, wenn man bedenkt, daß optimaler Weise nur dann B-Bilder codiert sein sollten, wenn die Änderung des Bildinhaltes zwischen den beiden umgebenden P- oder I-Bildern gleichmäßig und interpolierbar ist.

5.4.5.2. Referenzvektoren am Cut-Out

- Der Cut-Out liegt auf einem I-Bild.
Das I-Bild selbst referenziert kein anderes Bild, und alle intercodierten Bilder vor dem I-Bild können sich nur auf dieses beziehen. Somit existieren keine Referenzen über dem Cut-Out hinaus; die Source kann direkt kopiert werden.
- Der Cut-Out liegt auf einem P-Bild.
Das P-Bild kann nur Vorwärtsreferenzen enthalten. Davor liegende Interbilder können, wenn sie Rückwärtsreferenzen enthalten, nur dieses P-Bild referenzieren und nichts darüber hinaus. Somit existieren auch hier keine Rückwärtsreferenzen, die durch den Schnitt zerstört würden, und die Source kann direkt kopiert werden.
- Der Cut-Out liegt auf einem B-Bild.
Sowohl dieses B-Bild als auch alle direkt davor liegenden B-Bilder können Rückwärtsreferenzen auf ein P- oder I-Bild enthalten, welches nach dem Schnitt nicht mehr verfügbar ist. Alle diese Referenzen müssen aufgelöst und die Bilder neu codiert werden. Dies entspricht praktisch einer Umkodierung der B-Bilder in P-Bilder. Prinzipiell ist hier analog zum Fall „Cut-In auf B“ auch eine Auflösung der Rückwärtsreferenzen nur im letzten B-Bild und eine

Umrechnung der Rückwärtsreferenzvektoren der übrigen B-Bilder auf dieses neue Bild möglich.

Die Modellierung verschiedener spezieller Schnitte durch das Schnittmodell:

Mit Hilfe des oben definierten Schnittmodells des Kopierens einer Source in ein Target lassen sich auch alle speziellen Schnittvorgänge durchführen. Dazu hier einige Beispiele:

5.4.5.3. Löschen eines Teiles einer Sequenz (Kürzen)

Das Entfernen einer Teilsequenz aus der Originalsequenz wird in zwei Schritten modelliert. Zunächst wählt man als Source das Stück vom Beginn der Originalsequenz bis zum letzten Bild vor der Teilsequenz, die entfernt werden soll, und schneidet. Dies bedeutet, daß die Referenzen über den Cut-Out hinaus aufgelöst werden. Dann wählt man als Source das Stück vom ersten Bild nach der zu entfernenden Teilsequenz bis zum Ende der Originalsequenz, was die Vorwärtsreferenzen über den Cut-In hinaus auflöst. Diese Source wird an die erste Source angefügt. Abb. 5.4.d zeigt diese Vorgehen noch einmal.

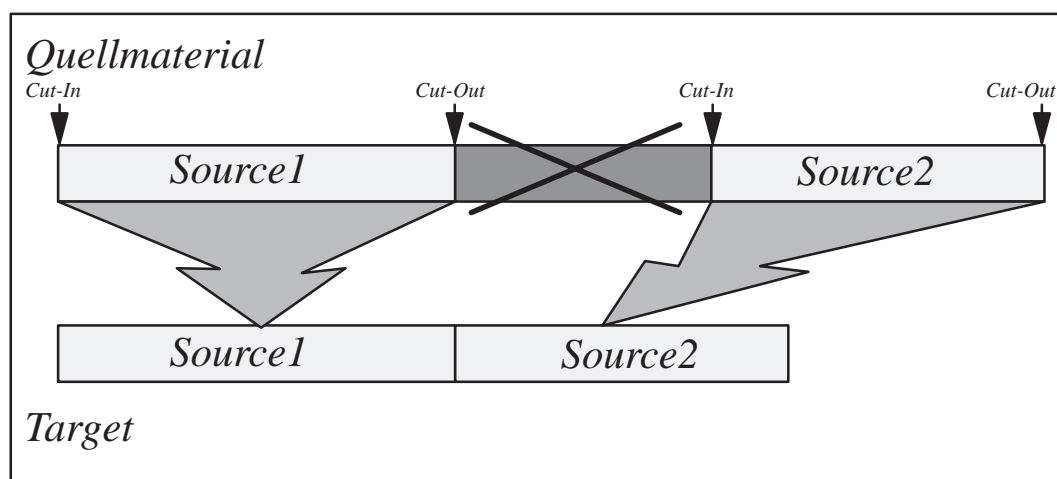


Abb. 5.4.d Schnittmodell: Kürzen einer Sequenz

Zusätzlich zu dieser allgemeinen Lösung gibt es noch einen Spezialfall, dessen Beachtung zu einem optimierten Vorgehen führt:

Sollen nur einzelne bzw. mehrere direkt hintereinanderliegende B-Bilder herausgeschnitten werden, brauchen keinerlei Referenzen aufgelöst zu werden. B-Bilder sind jederzeit aus der Sequenz zu entfernen, da sie nicht referenziert werden können.

5.4.5.4. Einfügen einer Teilsequenz in eine andere (Verlängern)

Hier ist die Einfügeoperation des klassischen Filmschnittes gemeint. Dazu wird in der Zielsequenz zunächst der Insert-Point als Stelle zwischen den beiden Bildern definiert, in die später die Teilsequenz eingefügt werden soll. Die erste Source

erstreckt sich nun vom Beginn der Zielsequenz bis zum Insert-Point, die zweite Source besteht aus der einzufügenden Teilsequenz, und die dritte Source wird aus den Bildern vom Insert-Point bis zum Ende der Zielsequenz gebildet. Somit sind an den Schnittstellen alle Referenzen aufgelöst.

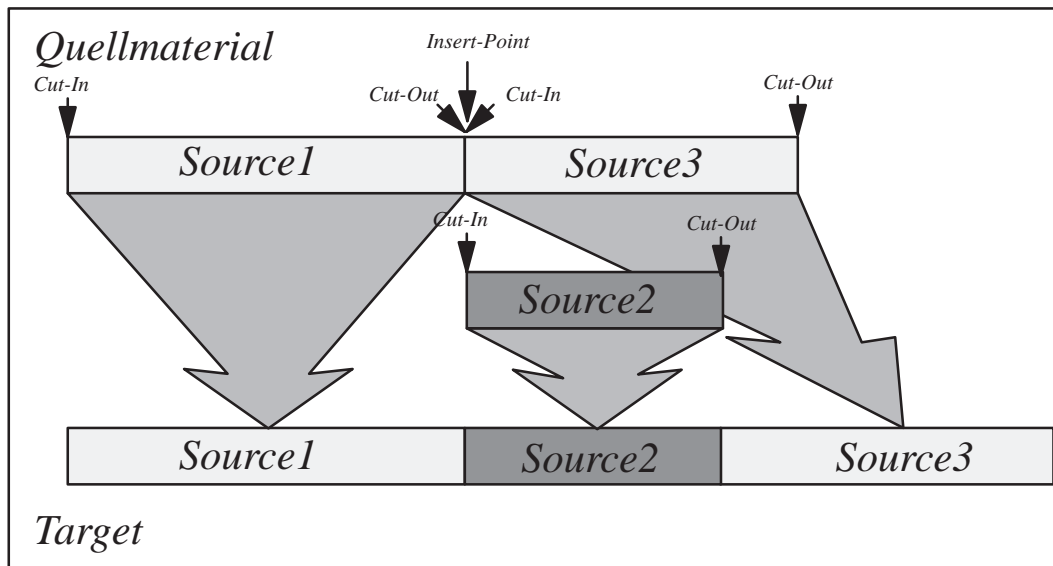


Abb. 5.4.e Schnittmodell: Einfügen einer Teilsequenz

5.4.5.5. Überschreiben einer Teilsequenz (Insertcut)

Hier ist die „Einfügeoperation“ des Videoschnittes gemeint. Ein Teil der Zielsequenz wird durch eine andere Sequenz überschrieben. Dies lässt sich durch das nacheinander ausgeführte Löschen der Teilsequenz und das Einfügen der anderen Sequenz realisieren.

5.4.6. UMCODIERUNG EINZELNER BILDER

Wie in Kap. 5.4.5. beschrieben, kann es beim Schnitt nötig werden, einzelne Bilder „umzucodieren“, d.h. Referenzen auf andere Bilder aufzulösen.

Zum Auflösen von Referenzen auf ein anderes Bild müssen evtl. mehrere Referenzebenen durchlaufen werden (vgl. Abb. 5.4.a). Hierbei kann eine Qualitätsminderung wie bei der Realisierung der wahlfreien Darstellung nicht akzeptiert werden, da diese Qualitätsverluste in die neue Codierung eingehen, ohne sie später wieder ausgleichen zu können. Allerdings ist auch hier die in Kap. 5.4.2.2. vorgestellte Methode der selektiven Rückberechnung anwendbar, da sie keine Qualitätseinbuße mit sich bringt.

Prinzipiell ist zur Auflösung der Referenzen eines inter-codierten Bildes die vollständige Decodierung und Neucodierung ohne Referenzen nötig. Dennoch können auch hier einige optimierende Methoden in Abhängigkeit der betroffenen Bild- und

Macroblocktypen entwickelt werden. Insbesondere wird für den Fall, daß die referenzierte Stelle wieder genau ein Macroblock ist, die Berechnung des neuen Macroblocks nach den von B. Smith für JPEG-codierte Bilder entwickelten Formeln anwendbar, welche nur die Huffman-Decodierung erforderlich machen. Dieser Fall tritt (neben der Zufälligkeit von Bildbewegungen genau in Macroblockschritten) logischer Weise immer dann auf, wenn im Bild unbewegte Gebiete vorhanden sind und somit die Macroblocks von einem zum nächsten Bild einfach erhalten bleiben. Smith's Berechnungen arbeiten auf den DCT-transformierten und quantisierten Blockdaten und ermöglichen somit eine Beschleunigung um das 50-100 Fache gegenüber Methoden, die auch noch die inversen Quantisierungen und DCT-Transformationen durchführen müssen [Rowe93b].

Die hier relevante Formel bezieht sich auf die Addition zweier DCT-codierter und quantisierter 8x8 Pixel Blocks. Sei also

$$n[i, j] = r[i, j] + e[i, j] \tag{Gl. 5.3}$$

die gewünschte Pixeladdition, dann ergibt sich aus den Gleichungen (Gl. 4.15), (Gl. 4.19), (Gl. 4.20) und (Gl. 4.24)

$$N_Q[u, v] = \frac{q_R(u, v)}{q_N(u, v)} R_Q[u, v] + \frac{q_E(u, v)}{q_N(u, v)} E_Q[u, v] \tag{Gl. 5.4}$$

mit:

- N_Q mit $Q_N[u, v]$ quantifizierter DCT-Block
- R_Q mit $Q_R[u, v]$ quantifizierter DCT-Block
- E_Q mit $Q_E[u, v]$ quantifizierter DCT-Block
- $q_x[u, v]$ Element $[u, v]$ der Quantifizierungsmatrix Q_x

Mit N sei im folgenden immer der *neue*, zu berechnende Block, mit R der *referenzierte* Block und mit E immer der aktuelle Block (dessen Inhalt die *Error terms* sind) bezeichnet. Gilt für alle Elemente $q[u, v]$ zweier Quantisierungsmatrizen R und N das

Verhältnis $\frac{q_R[u, v]}{q_N[u, v]} = \gamma$, so läßt sich (Gl. 5.4) vereinfachen zu

$$N_Q[u, v] = \gamma_R R_Q[u, v] + \gamma_E E_Q[u, v] \tag{Gl. 5.5}$$

Zur Auflösung der Referenzen werden die einzelnen Bildcodierungstypen betrachtet:

I-Bilder besitzen keine Referenzen auf andere Bilder, eine Umcodierung ist also nicht nötig.

P-Bilder können Vorwärtsreferenzen auf frühere P- oder I-Bilder enthalten. Zur Auflösung dieser Referenzen und Umcodierung der Macroblocks sind folgende Maßnahmen in Abhängigkeit vom Typ des Macrobloks möglich (vgl. [MPEG93]):

Intra_c und *Intra_q* Macroblocks bleiben unverändert erhalten, da sie keine Referenzen besitzen.

Macroblocks der Typen *Pred_c* und *Pred_qc* enthalten keinen Referenzvektor und beziehen sich damit auf den vollständigen Macroblock gleicher Adresse des Referenzbildes. Ist das Referenzbild ein I-Bild, so können nun mit Gl. xx die Referenzbilddaten des I-Bild-Blocks und die Korrekturdaten des P-Bild-Blocks addiert werden und ergeben direkt die quantisierten und DCT-transformierten Bilddaten des neuen, nunmehr intra-codierten P-Bild-Blocks. Setzt man dabei für den I-Bild-Block R_Q und für den P-Bild-Block E_Q und geht davon aus, daß der referenzierte I-Bild-Block und der neue Intra-P-Bild-Block die selbe Quantisierungsmatrix (nämlich die Intra-Quantisierungsmatrix) verwenden, ergibt sich aus (Gl. 5.3)–(Gl. 5.5):

$$N_Q[u, v] = \frac{Q_R}{Q_N} R_Q[u, v] + \frac{q_E(u, v)}{q_N(u, v)} E_Q[u, v] \quad (\text{Gl. 5.6})$$

mit Q_x Quantisierungsfaktor des Blocks X

Ist das Referenzbild ein P-Bild, wird eine weitere Rückverfolgung notwendig. Die Korrekturterme beider Blöcke können jedoch bereits im quantisierten Ortsfrequenzbereich summiert werden. Geht man hier davon aus, daß beide Blöcke dieselbe Quantisierungsmatrix, die Inter-Quantisierungsmatrix, benutzen und daß diese auch von dem summierten Korrekturblock verwendet werden soll, ergibt sich aus (Gl. 5.3)–(Gl. 5.5):

$$N_Q[u, v] = \frac{Q_R}{Q_N} R_Q[u, v] + \frac{Q_E}{Q_N} E_Q[u, v] \quad (\text{Gl. 5.7})$$

mit Q_x Quantisierungsfaktor des Blocks X

Macroblocks der Typen *Pred_mc* und *Pred_mcq*, die mit ihren Referenzvektoren wieder auf Macroblockadressen zeigen, also vollständige Macroblocks referenzieren, können ebenfalls nach obigen Gleichungen umcodiert werden. Macroblocks dieser Typen, deren Referenzvektoren nicht auf Macroblockadressen zeigen, die somit also mehrere andere Macroblocks referenzieren, erfordern nach [Rowe93b] die Decodierung der referenzierten Macroblocks zur Auflösung der Referenzen.

Macroblocks des Typs *Pred_m*, die nur über einen Referenzvektor, aber keine Korrekturdaten verfügen, übernehmen die Daten des referenzierten Blocks direkt. Es kann hier analog zu den Typen *Pred_mc* und *Pred_mcq* vorgegangen werden.

Skipped Macroblocks übernehmen die Daten des Macroblocks mit gleicher Adresse des Referenzbildes. Ein weiteres Vorgehen erfolgt dann gemäß dessen Typs.

B-Bilder können Vorwärts- und Rückwärtsreferenzen enthalten. Je nach dem, ob die Vorwärts- (Cut-In) oder Rückwärtsreferenzen (Cut-Out) aufzulösen sind (vgl. Kap. 5.4.5.), können die jeweils anderen erhalten bleiben. Ein Vorgehen erfolgt hier gemäß den Ausführungen zu den P-Bildern.

5.4.7. KOMPOSITION INKOMPATIBLER MPEG-STREAMS

Bei dem Zusammenschneiden von Teilen verschiedener MPEG-Videostreams können Inkompatibilitäten auftreten, deren Erscheinungen den konventionellen Schnittsystemen völlig unbekannt sind, da hier nur „kompatible“ Formate geschnitten werden. Z.B. werden Schnitte von PAL nach NTSC nicht durchgeführt, es muß vorher eine Formatwandlung erfolgen.

In MPEG können Bitstreams Parameter besitzen, die sie a priori als nicht zusammenpassend erscheinen lassen.

Es sind dies die Parameter:

- Bildgröße
- Bildrate
- Datenrate
- Puffergröße
- Constraint-Flag
- Quantisierungsmatrizen

Haben diese Parameter verschiedene Werte, ist ein direktes Schneiden nicht möglich, ohne eine spätere korrekte Wiedergabe zu gefährden.

Die entsprechenden Parameter müssen angepaßt werden.

Quantisierungsmatrizen

Da nach der Umcodierung der Cut-Ins und Cut-Outs sowohl des letzten Bildes des Targets als auch des 1. Bildes der Source (vgl. Kap. 5.4.5.) die Bedingungen zur Bildung von GoPs gegeben sind (das 1. Bild in Speicherreihenfolge ist ein I-Bild, das letzte Bild in Darstellungsreihenfolge ist ein P- oder I-Bild), kann nach dem letzten Bild des Targets ein GoP-End-Code und vor dem 1. Bild der Source ein GoP-Header eingesetzt werden. Dazwischen ist die Neudefinition der Quantisierungsmatrizen in einem eingefügten Sequence-Header nach MPEG erlaubt und möglich.

Datenrate

Haben beide Streams verschiedene Datenraten in ihrem Sequence-Header eingetragen bzw. ist einer der beiden Streams ohne Datenratenangabe (freie Datenrate), so ist für den Zielstream die höhere bzw. die freie Datenrate zu wählen. Die Teile des Zielstreams, die aus dem Stream mit der niedrigeren Datenrate stammen, werden durch Bitstuffing (Macroblock-Ebene) auf die höhere Datenrate gebracht. Dabei ist darauf zu achten, daß in jedem Picture-Header des Streams mit der geänderten Datenrate der vbv-Bufferdelay neu zu errechnen ist (vgl. [MPEG93]).

Puffergröße

Haben beide Streams verschiedene Puffergrößen vorgegeben, erhält der Zielstream die größere von beiden. Somit wird ein Pufferüberlauf auf jeden Fall vermieden. Ein

Unterlauf wird ja sowieso von beiden Streams kontrolliert. Auch hier ist der vbv-Bufferdelay der Picture-Header des Streams mit dem kleineren Original-Puffer neu zu berechnen.

Bildgröße

Da in MPEG alle Bildgrößen durch 16 teilbar sein müssen (vgl. [MPEG93]), um Vielfache der Macroblockauflösung zu erhalten, können sich die Bildgrößen zweier Bitstreams nur um ganze Macroblocks unterscheiden. Hier sind zwei Vorgehensweisen möglich, die aber beide automatisch eine Anpassung der Datenraten und Puffergrößen eines Streams erforderlich machen:

1. Die Bildgröße des größeren Streams bleibt erhalten. Bildsequenzen des kleineren Streams werden dann z.B. zentriert, mit einem Rahmen aus schwarzen Macroblocks, welche beim jeweils 1. Bild der kleineren Teilsequenz intracodiert werden müssen. Bis zum nächsten 1. Bild der größeren Sequenz können sie dann als skipped Macroblocks codiert werden.
2. Die Bildgröße des kleineren Streams wird gewählt. Dazu muß der größere Stream an einem gewählten Ausschnitt geklippt werden. Liegt dieser Ausschnitt nicht in der linken oberen Ecke des großen Bitstreams, sind hier ebenfalls die Neuberechnungen aller Macroblock-Adressen der verwendeten Bilder nötig. Der Ausschnitt sollte auf einer Macroblockgrenze liegen, also nur ganze Macroblocks enthalten, da sonst eine Decodierung und Neucodierung aller Bilder des größeren Streams notwendig wird, weil Teile von Macroblocks nicht übertragen werden können.
3. Werden die Bilder nicht geklippt, sondern skaliert, so ist eine komplette Decodierung und nach der Skalierung eine Neucodierung erforderlich. Für Methoden zur Skalierung von Pixelrepräsentationen sei auf [Enc88], [Foley90] und [Hoffm92] verwiesen.

Bildrate

Werden in den verwendeten Streams unterschiedliche Bildraten angegeben sein, so ist eine Convertierung nach den bekannten Mustern der TV-Normkonvertierung vorzunehmen (vgl. [CCIR 624]).

z.B. PAL NTSC \equiv 25 30 Bilder/sec.

Jedes 4 PAL-Bild wird verdoppelt, ausgenommen Nr. 24. Damit ergibt sich folgendes Umsetzungsschema:

NTSC: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
 PAL: 1 2 3 4 4 5 6 7 8 8 9 10 11 12 12 13 14 15 16 16 17 18 19 20 20 21 22 23 24 25

NTSC PAL \equiv 30 25 Bilder/sec.

Jedes 6. NTSC-Bild wird ausgelassen:

PAL: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
 NTSC: 1 2 3 4 5 7 8 9 10 11 13 14 15 16 17 19 20 21 22 23 25 26 27 28 29

Hierbei ist ebenfalls eine Anpassung der Datenrate und Puffergröße vorzunehmen.

Constraints

Ist bei einem der Streams das Constraint-Parameter-Flag gesetzt, und bei dem anderen nicht, so ist dieses Flag in der Zielsequenz zu löschen. Falls der unconstrained Stream nicht automatisch durch die evtl. Anpassung anderer Parameter wie Bildrate, Bildgröße, Puffergröße oder Datenrate wieder die Anforderungen an einen constraint-Bitstream erfüllt:

- Horizontale Bildgröße ≤ 768 Pixel
- Vertikale Bildgröße ≤ 576 Pixel
- Bildrate ≤ 30 Bilder/sec.
- Anzahl der Macroblocks pro Bild ≤ 396
- Anzahl der Macroblocks pro Sekunde $\leq 396 * 25$
- Forward_f_code ≤ 4 ⁽⁵⁾
- Backward_f_code ≤ 4 ⁽⁵⁾
- vbv-Puffergröße ≤ 40 KByte
- Bitdatenrate ≤ 1856 KBit/sec

5.4.8. EINHALTUNG DER CONSTRAINTS

Selbst wenn alle verwendeten Bitstreams in allen Parametern übereinstimmen und Constraint-Bitstreams sind, kann es aufgrund der Schnittfolge und der damit verbundenen Umcodierung dazu kommen, daß ein Pufferüberlauf oder -unterlauf des Decoders entsteht.

Da es sich bei dem im Schnittsystem vorhandenen Umcodierer um einen modifiziertem MPEG-Encoder handelt, ist er in der Lage, das Puffermanagement zu überwachen. Die adaptive Kompression zum korrekten Pufferungsverhalten erfolgt also automatisch, wie bei einer vollständigen MPEG-Codierung auch, durch Bitstuffing und Quantisierungsfaktorisierung.

5. Damit wird die erlaubte Länge der Referenzvektoren auf 64 Pixel bei Halbpixelgenauigkeit und auf 128 Pixel bei Vollpixelgenauigkeit beschränkt.

5.5. VERWALTUNG UND VERWENDUNG VON METAINFORMATION

Das System wird den Video Cutter auch bei seiner kreativen Arbeit unterstützen. Eine der zeitraubendsten Aufgaben ist das Auswählen und Wiederauffinden geeigneter Szenen aus der Masse des Quellmaterials (vgl. Kap. 3.3.). Daher verwaltet heute jeder Cutter und Regisseur Informationen über das Videomaterial selbst, die er während des Editierens in Timecodeangaben für das Schnittsystem umsetzen muß. Diese Informationen lassen sich grob in zwei Kategorien einteilen:

1. Informationen, die sich auf den Bild- (und Ton) -Inhalt beziehen, z.B. Szene mit Haus und Auto
2. Metainformationen, die sich nicht nur auf das Bildmaterial selbst beziehen, sondern auf z.B. Umstände seiner Entstehung, den Drehort, usw.

Informationen beider Kategorien werden vom Schnittsystem verwendet, um den Prozeß der Szenauswahl und die Wiedergewinnung gewählter Szenen zu vereinfachen. Dazu müssen sie dem Schnittsystem zugänglich gemacht werden.

5.5.1. AUTOMATISCH ZU GEWINNENDE INFORMATIONEN

Die bildinhaltsbezogenen Informationen lassen sich im Prinzip automatisch aus den Videodaten extrahieren. Somit wäre eine automatische inhaltsbezogene Indizierung der Videos möglich. Trotz großer Forschungsbemühungen auf dem Gebiet der Bildanalyse und Bilderkennung scheint eine derart komplexe Aufgabe noch nicht in voller Allgemeinheit lösbar (vgl. [Hoffm92], [Cher93], [Enc88], [Foley90], [Litt93]).

Im Bereich des Videoeditings liefern jedoch auch wesentlich weniger komplexe Indizierungen schon eine sehr gute Ergebnismenge, da die Anfragen der Editoren oft sehr allgemein gehalten sind. Szenen, deren Inhalt genauer spezifiziert werden kann, sind sowieso schon bekannt, und damit auch weitere zusätzliche Informationen, über die wesentlich schneller darauf zugegriffen werden kann (z.B. Drehort, Timecode,...). Anfragen nach Szenen bezüglich des Bildinhaltes haben oft den Hintergrund, das keine Szene geeigneten Inhalts bekannt ist, was nur bei unwichtigen stimmungsbildenden Szenen der Fall sein wird (vgl. Kap. 3.3.).

Andererseits lassen sich aber Anfragen wie z.B.:

- Szene mit starkem Rotanteil
- Szene mit blauem Himmel und sogar
- Szene mit Wolken im Himmel

sehr leicht durch statistische Auswertungen der Bilder oder Teilbilder beantworten. (vgl. [Hoffm92], [Enc88], [Foley90]).

MPEG-codierte Videosequenzen bieten darüber hinaus aufgrund der bei der Codierung durchgeführten Bildanalyse und Transformation weitere Ansatzpunkte für eine automatische Bildindizierung.

Zum einen ist in der Codierung der Videodaten selbst eine auswertbare Menge an Informationen enthalten:

- Die Farbbildcodierung in Luminanz- und Chrominanzkanälen macht eine Auswertung der Farbinformation (s.o.) einfacher. Die Farbkanäle können direkt ausgewertet werden. Auch alle weiteren unten genannten Analysemethoden können direkt auf die getrennten Luma- und Chromakanäle angewendet werden.
- Aufgrund der Differenzbildcodierung sind in den Referenzvektoren bereits Informationen über Kamera- und Objektbewegungen im Bild ausgewertet. Somit lassen sich Szenenanfragen über Größe, Richtung und/oder Geschwindigkeit sich bewegender Objekte sowie Richtung und Geschwindigkeit der Kamerabewegung einfach durch die Auswertung der Referenzvektoren ermitteln. Das Erscheinen und/oder Verschwinden von Objekten ist durch eine Analyse der codierten Macroblocks zu ermitteln – wie viele Intra/Inter-Macroblocks enthält ein Bild an einer bestimmten Stelle, wie groß sind die codierten Korrekturwerte. Ebenso läßt sich ein Szenenwechsel durch eine Analyse der Bildtypenfolge erkennen.

Dies sind leider keine vollständig verlässlichen Methoden. So ist die Referenzvektorcodierung nicht auf Bewegungserkennung ausgelegt, sondern rein auf eine Bildähnlichkeitsanalyse auf Macroblock-Ebene. Abb. 5.5.a zeigt einen Fall, in dem eine vorhandene Bewegung außer zu Beginn und Ende derselben nicht durch die Referenzvektoren intercodierter Bilder angezeigt wird.

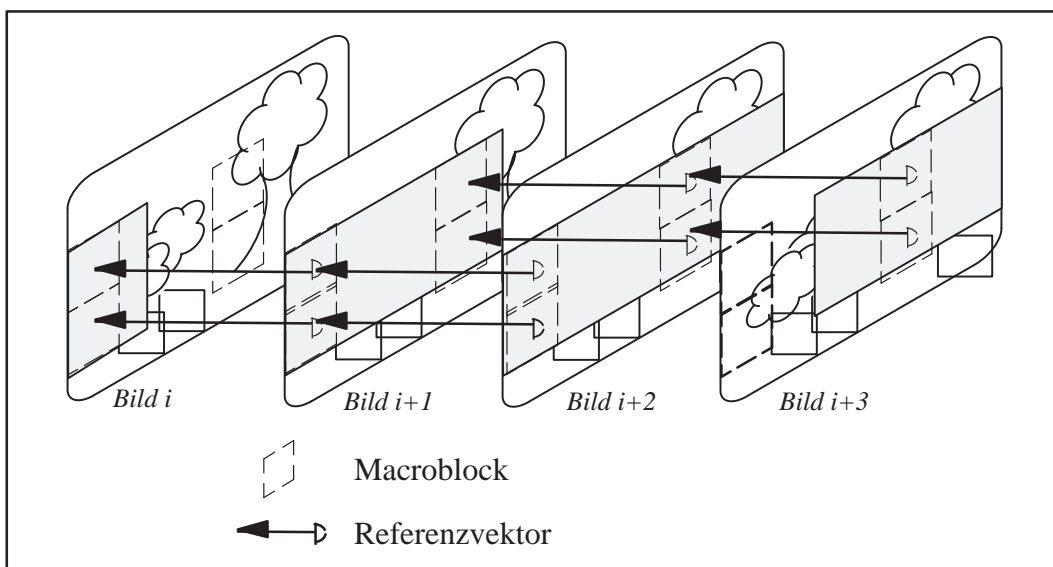


Abb. 5.5.a Die Differenzbildcodierung ist nicht auf Bewegungserkennung ausgelegt

Gleichfalls ist die Bildtypenfolge einer MPEG-Sequenz heute noch allzu häufig in einem dem Encoder vorgegebenen Muster

fixiert und richtet sich nicht nach den tatsächlichen Erfordernissen der inhaltsbezogenen Kompression. Abb. 5.5.b zeigt die Effekte einer solchen Fehlcodierung.

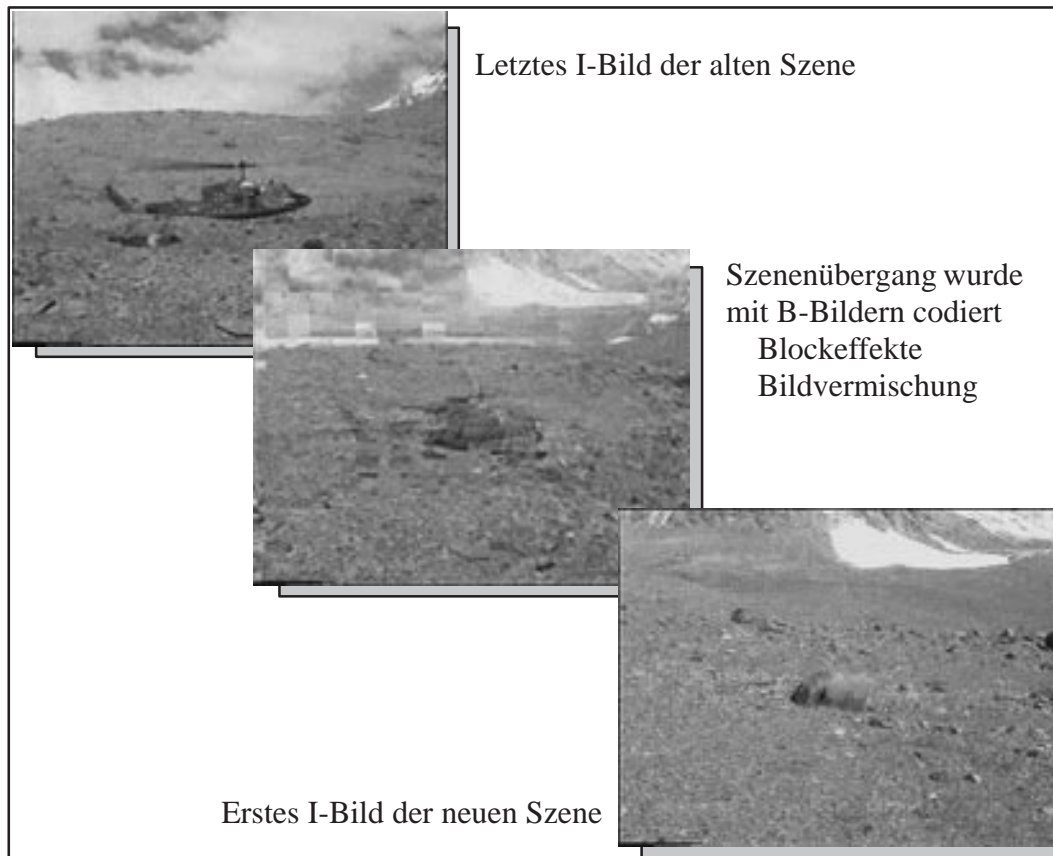


Abb. 5.5.b Ungeeignete Bildtypenfolge: Szenenwechsel ist nicht intracodiert

Da jedoch eine Beachtung derartiger Zusammenhänge bei der Codierung zu einer besseren Qualität bei höherer Kompression beiträgt, steht zu erwarten, daß zukünftige Encoder in dieser Beziehung verbessert werden und somit auch die zukünftigen MPEG-Videos mehr verwertbare DPCM Informationen enthalten.

Aber selbst unter den genannten Restriktionen erlaubt die Auswertung der in der Differenzbildcodierung enthaltenden Zusatzinformationen schon heute die Beantwortung von Anfragen auf Basis des Bildinhaltes.

- Die DCT-Transformation der Bildbeschreibung ermöglicht (nach der Quantisierung) zumindest eine grobe Auswertung der im Bild enthaltenen Frequenzen und somit Kontraste. Dies ist allerdings nur bei I-Bildern möglich, da die DCT-Transformierte der DPCM-codierten Bilder kaum Rückschlüsse auf die Frequenzverteilung des eigentlichen Bildinhaltes zuläßt. (vgl. [MPEG 93]).

Als eine weitere Möglichkeit der automatischen Informationsextraktion lassen sich die im MPEG-Encoder implementierten Methoden zur Auswertung weiterer Bildinformation nutzen. So ist z.B. die Referenzvektorbestimmung des Encoders anwendbar auf die Suche von Mustervorgaben. Damit können Bilder und Szenen aufgrund von Ähnlichkeiten mit vorhandenen Bildteilen bestimmt werden. Dazu stellt man lediglich ein vorhandenes Bild oder einen Bildausschnitt (Objekt) als Referenz zur Verfügung und sucht mittels der Differenzbildcodierung des Encoders Bilder, Bildteile und Objekte mit möglichst geringer Abweichung. Da ein reiner Encoder nur auf Vollbilddaten arbeitet, ist hier die Verwendung eines nach Kap. 5.4.6. modifizierten „Reencoders“ notwendig, um auch auf bereits codiertem Material suchen zu können.

5.5.2. ZUSÄTZLICH ZU SPEICHERNDE INFORMATIONEN

Außer der auf der automatischen Generierung von Informationen basierenden Indizierung des Videomaterials ist das System auch in der Lage, Informationen aus externen Quellen in den Retrieval-Prozeß mit einzubeziehen und diese Informationen darüber hinaus mit dem Videomaterial zu verkoppeln. Als externe Quellen können hierbei betrachtet werden:

- Der Timecode, der entweder von einer zentralen Stelle im Studio dem System zugeführt wird oder dem Source-Material entnommen ist;
- Erfassung verschiedener Aufzeichnungsparameter wie Kamera-
position, Blende, Beleuchtung, usw;
- manuelle Indizierung durch Verknüpfung mit Metainformationen
in verschiedenen Phasen der Produktion. Beispielhaft seien hier
genannt:
 - genaue Bezeichnung der Szene
 - Drehort der Szene
 - Schauspieler, Regisseur, Cutter
 - Umstände
 - erste Bewertung
 - Stichworte
 - Daten, Uhrzeit
 - Eigentums- und Urheberrechte
 - Kameraeinstellungen
 - Beleuchtung
 - Relation zu anderen Szenen
 - Szene bereits verwendet ?
 - Inhaltsbeschreibung

Diese Liste ist keineswegs vollständig und kann es auch nicht sein, da die verfügbare Information, ihre Formulierung und ihre Verwertbarkeit sehr stark von der Art des Materials, des Produktionsstils und der beteiligten Personen abhängt (vgl. Kap. 3.3.).

All diese Informationen zum Videomaterial dürfen nicht getrennt von diesem gespeichert und verarbeitet werden. Nur wenn das Bildmaterial gemeinsam mit den es betreffenden Metainformationen und, synchron verkoppelt mit diesen, dem Schnittsystem zur Verfügung steht, ist eine optimierte Unterstützung des Videoeditings gewährleistet.

MPEG bietet dazu die Möglichkeit. Neben den Video- und Audiostreams können in einem MPEG-System-Stream auch Benutzerdaten-Streams enthalten sein. Eine Synchronisation der Benutzerdaten mit den Audio- und Videodaten ist durch den Standard gewährleistet. Jede Hierarchieebene des MPEG-Video-Streams bietet zusätzlich noch die Möglichkeit, benutzerdefinierte Daten innerhalb des Video-Streams zu speichern und zu übertragen (vgl. [MPEG93], Kap. 4.4.3.).

Somit kann zusätzlich zum Timecode in MPEG jede Art der o.g. manuellen Indizierung durchgeführt werden. Diese Daten stehen auf Wunsch jedem sofort zur Verfügung, der das MPEG-codierte Video zum Weiterverarbeiten oder auch nur zum Betrachten erhält.

Die Codierung dieser Art der Information innerhalb des Streams wird vom Standard in keiner Weise vorgegeben. Es ist jedoch offensichtlich, daß eine möglichst effiziente Codierung notwendig ist, was sowohl den Speicherplatzbedarf als auch den Verarbeitungszeitbedarf betrifft. Die Auswertung und Darstellung einer MPEG-Sequenz unterliegt einer strengen zeitlichen und räumlichen Vorgabe, d.h., der Decoder muß in der Lage sein,

- alle 40ms ein komplettes Bild zu empfangen und zu decodieren;
- in einem endlichen (sogar fixen) Pufferbereich zur Speicherung und Decodierung zu arbeiten.

Die Codierung von Daten innerhalb des MPEG-Video-Streams bedeutet automatisch die Reduzierung der für Videodaten vorhandenen Bitraten und somit eine höhere Kompression der Videodaten.

Eine Codierung innerhalb eines eigenen Benutzerdaten-Streams dagegen beeinflußt nur die vorgegebene Gesamtdatenrate des MPEG-System-Streams. Hier ist allerdings schon die Übertragung zusätzlicher Daten miteingerechnet. So beträgt die mögliche Gesamtdatenrate eines Constraint-System-Streams tatsächlich 1.856 MBits/sec, obwohl der Standard für Video/Audio-Codierungen mit bis zu 1.5 MBits/sec definiert ist (vgl. [MPEG93]).

Die Synchronisation mit den Videodaten kann, wie in (vgl. Kap. 4.4.3.) beschrieben, über PTS und DTS Zeitmarken erfolgen.

5.5.3. BENUTZERDATEN SPEICHERMODELL

Die Benutzerdatenbereiche des Videostreams bleiben aktuellen und temporären Eintragungen des Schnittsystems vorbehalten. Diese Bereiche sind direkt zu beschreiben oder zu löschen, da sie nicht weiter codiert oder komprimiert werden. Als Einträge sind hier vorzusehen:

- Marken zum Justieren von Schnitten
- Cut-In und Cut-Out Markierungen
- Wiedergabeinformationen wie Beginn/Ende der Zeitlupe, Zeitraffer, oder Videoeffektbeschreibungen (z.B. Ab- oder Aufblende)

Die Benutzerdaten-Streams werden verwendet um die o.g. Metainformationen zu speichern. Diese können ebenso wie der Videostream hierarchisch strukturiert sein, um globale Informationen wie Regisseur, Eigentumsrechte usw. an den Anfang zu stellen (Sequence-Ebene) und szenen-, bild- oder bildausschnittspezifische Informationen wie Szenenname oder Benennung einzelner Bildobjekte mit den Videodaten zu synchronisieren. Die Benutzerdaten werden durch einen eigenen Elementardecoder ausgewertet (vgl. Kap. 4.4.3.) und stehen dem Schnittsystem dann zur Verfügung.

Die Informationen der Benutzerdaten-Streams müssen von Schnittsystem auch wieder in das Ergebnisvideo übernommen werden, so daß sie dort wieder zur weiteren Verarbeitung zur Verfügung stehen.

Ebenso kann dem Endbetrachter des Videos mehr Information über das Video mitgegeben werden, indem man es gemäß einem Inhaltsverzeichnis indiziert. Somit ist es ihm möglich, sich gezielt bestimmte Teile des Videos, also z.B. die Einleitung, das Schlußwort oder eine Kurzversion anzeigen zu lassen.

5.6. SYSTEMARCHITEKTUR

Das System wird unabhängig von der zur Verfügung stehenden Hardware arbeiten können, aber trotzdem alle vorhandenen Ressourcen nutzen. Es soll die Nutzung einer Datenbank vorsehen, um auch Videoserver als Datenquellen akzeptieren zu können. Die Architektur ist objektorientiert, was einer Parallelisierung und späteren Wartung und Erweiterung sehr dienlich ist.

Das System soll ein „Desktop-System“ sein, d.h., es muß ein möglichst intuitives Benutzungsmodell und eine ebenso intuitive (graphische) Benutzungsoberfläche besitzen. Es muß also leicht zu erlernen und zu bedienen sein.

Realisation:

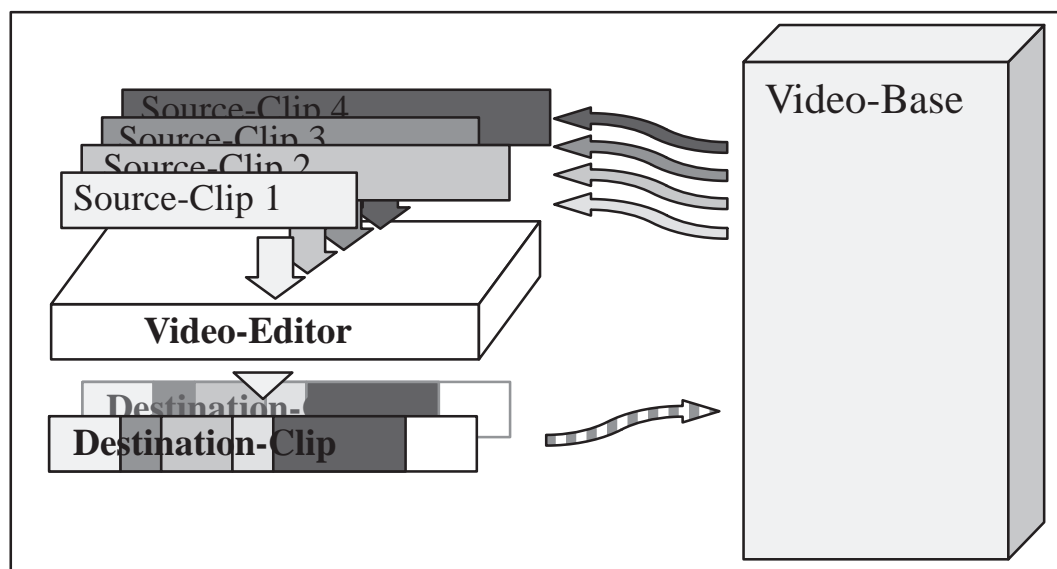


Abb. 5.6.a Die Systemarchitektur

Das Editiersystem besteht im Wesentlichen aus vier verschiedenen Komponenten:

5.6.1. DIE VIDEODATENBANK (VIDEOBASE)

Die Videodatenbank muß durchaus keine Datenbank im engsten Sinne des Begriffes sein, d.h., sie muß nicht über ein Datenbankmanagementsystem verfügen. Obwohl dies wünschenswert wäre, soll auch die Verwendung loser Dateien mit einer eigenen

Verwaltung derselben realisiert werden. Die Videobase ist also durchaus ein echter Bestandteil des Systems, wenngleich die Vorteile eines DBMS offensichtlich sind:

- Die Zugriffsverwaltung und -reihenfolge ist geregelt.
- Die Datenkonsistenz ist auch bei Mehrfachzugriffen gewährleistet.
- Ein Transaktionsmanagement ist vorhanden.
- Die Wartung und der Betrieb der Datenhaltung sind unabhängig vom Schnittsystem.
- Die Abfrage der Videos erfolgt über eine definierte Querylanguage.

Die Anbindung des Systems an die Videobase muß so allgemein sein, daß sie jederzeit ausgetauscht werden kann.

5.6.2. DIE SOURCE-VIDEO-CLIPS

Der Benutzer wählt aus der Videobase Video(-Teile) zur Verwendung als Quellmaterial aus. Dazu muß es geeignete Abfragemechanismen geben (s.o.). Das System erzeugt daraus ein eigenständiges Objekt, das im Wesentlichen die Fähigkeiten eines Wiedergabegerätes hat. Es ist möglich, daß mehrere dieser Source-Clips auf das gleiche Videomaterial, d.h. auf die gleiche Datei, zugreifen und evtl. jeweils andere Abschnitte daraus darstellen. Es ist auch möglich, den Start (Cut-In) und das Ende (Cut-Out) eines solchen Clips jederzeit zu verschieben, solange dies innerhalb des Quellmaterials möglich ist.

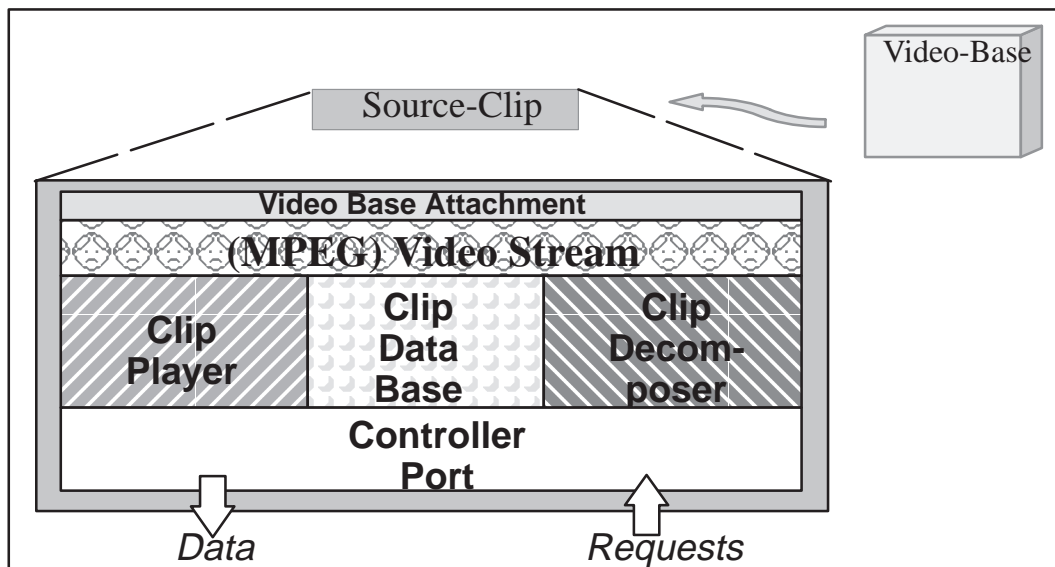


Abb. 5.6.b Der Aufbau eines Source-Clips

Der Source-Clip enthält die Videodaten (bzw. eine Referenz darauf) sowie eine Einheit, welche die Kommunikation mit der Videobase leistet. Dies ist beim Erzeugen (In-

initialisieren) und beim Ändern der Start- und Endbilder des Clips erforderlich. Sehr große Clips werden dynamisch geladen und verwaltet.

In der Clipdatabase sind zugriffsrelevante und statistische Daten abgelegt, die während der Erzeugung des Clips gelesen wurden. Somit sind Abfragen über die Anzahl der Bilder, ihrer Art der Codierung usw. möglich. Ebenso ist ein schnellerer Zugriff möglich, wenn die Position der einzelnen Bilder bekannt ist.

Der Clip-Player stellt den Inhalt des Clips auf Anforderung dar. Er muß effizient implementiert sein und die Echtzeitproblematik berücksichtigen. Der Clip-Decomposer leistet den eigentlichen Schnitt und stellt auf Anforderung die gewünschten Bilder zusammen mit Informationen über Codierungsart und Referenzierungen zur Verfügung.

Alle Einheiten werden über den ControllerPort mit den Anforderungen des Editors versorgt und geben falls gefordert ihre Daten über diesen Controllerport an den Editor zurück. Der Controllerport stellt die Kommunikationseinheit mit dem Editor dar und definiert deshalb die Schnittstelle des Clips als Objekt zur Außenwelt der übrigen Objekte.

5.6.3. DER DESTINATION-CLIP

Der Destination-Clip bildet das Ergebnis des Editiervorganges. Er ist ein Äquivalent zum Source-Clip, nur daß hier auch Videodaten geschrieben werden.

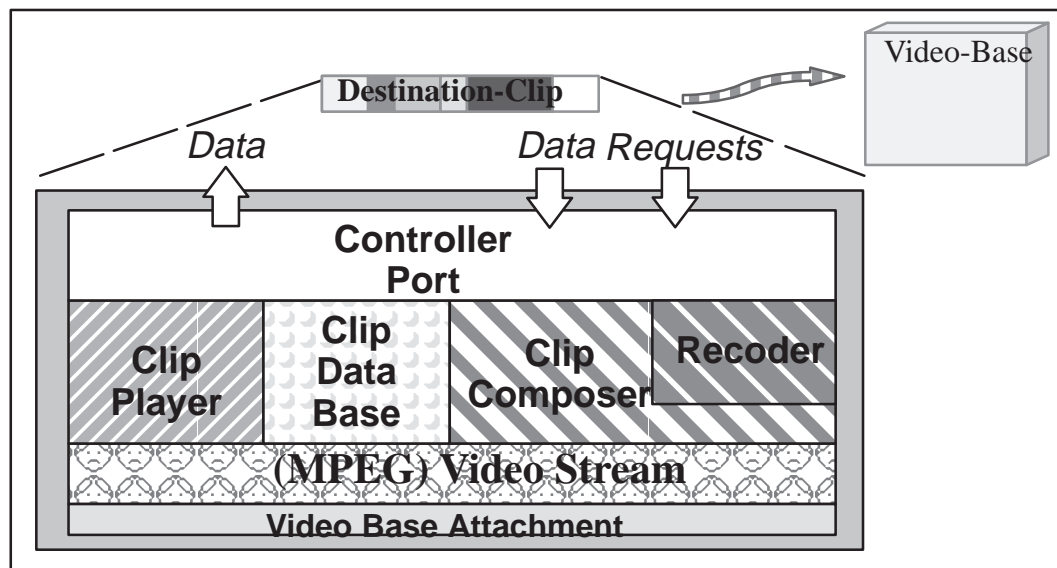


Abb. 5.6.c Der Aufbau des Destination-Clips

Die Videobase-Attachment-Einheit regelt die Erzeugung eines leeren Videostreams in der Videobase sowie das Schreiben von Videodaten in die Videobase.

Die Clip-Database wird während des Editierens aufgebaut und sieht eine dynamische Datenverwaltung vor, um sich den Änderungen, die sich durch das Entstehen des Videoclips ergeben, anpassen zu können.

Der Clip-Player des Destination-Clips entspricht exakt dem Clip-Player des Source-clips.

Im Unterschied zum Source-Clip enthält der Destination-Clip statt eines Decomposers nun einen Composer. Er baut den neuen Clip aus den ihm gelieferten Teilen auf. Bei der Verwendung von MPEG kann es vorkommen, daß einige Bilder umcodiert werden müssen, da sie in der DCPM-Codierung auf Referenzbilder verweisen, die in dem neuen Clip nicht vorhanden sind (vgl. Kap. 5.4.6.). Der Destination-Clip erhält durch den Editor die hierzu nötigen Informationen (Parameter (vgl. Kap. 5.4.7.) und Referenzbilder) und nimmt dann evtl. die Umcodierung der von Cut-In und Cut-Out betroffenen Interbilder vor. Dies geschieht im Recoder des Destinations-Clip-Composers. Die Umcodierung erfolgt nicht direkt im Source-Clip, um die in Kap. 5.4.5. und Kap. 5.4.6. beschriebenen Optimierungen der Umcodierung sowie die Parameteranpassung innerhalb der Zielsequenz durchführen zu können.

Der Controllport stellt wieder die Schnittstelle zur Außenwelt dar. Er kann außer Kommandos auch (Video-)Daten vom Editor empfangen. Nach der Beendigung des Editierens müssen die Videodaten wieder in die Videobase zurückgeschoben werden.

5.6.4. DER VIDEO-EDITOR

Der Videoeditor ist das zentrale Objekt des Systems. Er verwendet mehrere Source-Clips, um daraus ein oder mehrere verschiedene Destination-Clips zu erstellen.

Er stellt zunächst die initiale Systemschnittstelle zum Benutzer dar. Jede Teileinheit des Editors trägt einen Teil zu dieser Benutzerschnittstelle bei.

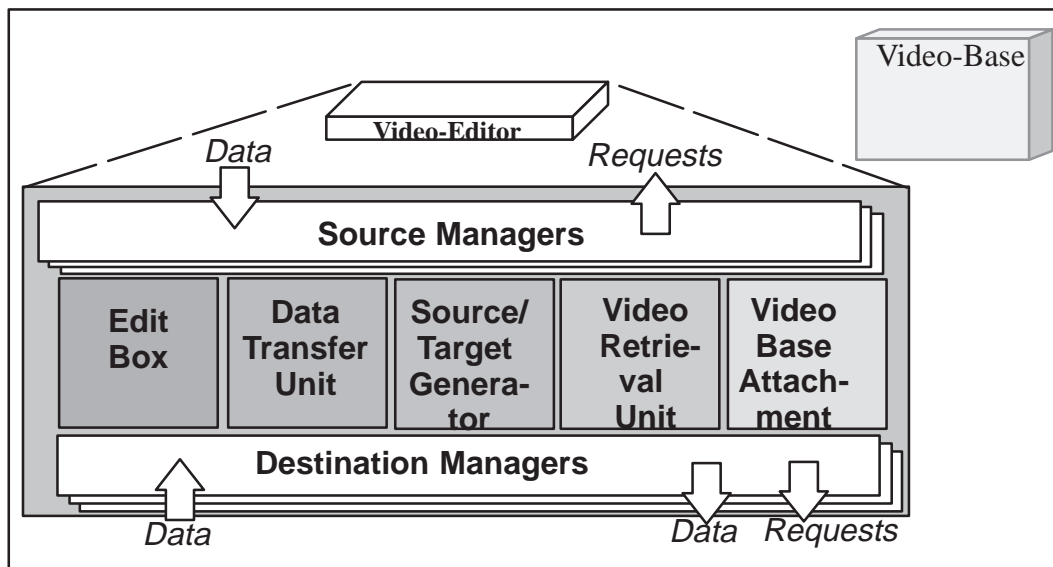


Abb. 5.6.d Der interne Aufbau des Editors

Der Source/Target-Generator dient dazu, die Source- und Destination-Clips zu erzeugen (instanzieren). Dazu kann der Benutzer mit Hilfe der Video-Retrieval-Ein-

heit Source-Videos aus der Video-Base auswählen und in je einem Source-Clip verfügbar machen. Zum Abfragen der Video-Base und dem Erzeugen und Schreiben neuer Videos bedient sich die Video-Retrieval-Einheit der Video-Base-Attachment-Einheit. Diese ist ein austauschbares Modul, daß je nach Typ der vorhandenen Video-Base (DBMS/Files) mehr oder weniger komplex strukturiert ist. Pro Source-Clip existiert ein Source-Manager, der den Datenaustausch und die Steuerung über dessen Contoller-Port durchführt.

Die Daten, welche die Source-Clips an den Editor übertragen, setzen sich zusammen aus:

- Antworten auf statistische Anfragen an die Clip-Database
- Videosequenzen (Cut-In \Rightarrow Cut-Out + Referenzbilder)
- Zur Umcodierung nötige Parameter (z.B. Quantisierungsmatrizen)
- Timecode (im Falle der Synchronisation verschiedener Clips auf diesen Clip als Master)

Diese Daten werden durch die Data-Transfer-Einheit an die entsprechenden Empfänger übertragen bzw. dem Benutzer präsentiert.

Die Editbox steuert die eigentlichen Schnittfunktionen und realisiert somit das Anordnen der verschiedenen Source-Clips in einem (oder mehreren) Destination-Clips. Sie übernimmt die visuelle Repräsentation des Insert- und des Assemble-Cut (vgl. Kap. 5.4.5.3. ff) sowie die die Umsetzung der Preview-, Aufzeichnungs- und Review-Befehle (vgl. Kap. 3.3.1.) des Benutzers.

Die verschiedenen Source- und Destination-Clips sowie der Editor und die Video-Base können in getrennten Prozessen oder sogar auf verschiedenen Rechnern laufen. Die Teileinheiten der einzelnen Komponenten können ebenso parallelisiert und verteilt ausgeführt werden. Die Umstetzung in eine C++-Klassenhierarchie erfolgt direkt auf der Basis der Teileinheiten.

6. AUSBLICK

Das in dieser Arbeit entwickelte Konzept für ein MPEG-Schnittsystem bietet noch eine Reihe von Entwicklungsmöglichkeiten, die in diesem Kapitel kurz aufgezählt werden sollen.

Graphische Benutzerschnittstelle

Auf Basis der Implementierung der in dieser Arbeit vorgestellten technischen Grundlagen muß eine intuitive, leicht zu erlernende Benutzerschnittstelle entworfen werden. Dazu eignet sich eine graphische Darstellung von Bearbeitungsmetaphern aus der Videoediting-Welt am besten.

MPEG-2

Einer der nächsten Schritte in der Weiterentwicklung des DiVidEd-Systems sollte in der Adaption auf die Verwendung der MPEG-2 Norm bestehen. MPEG-2 bietet, ersten Entwürfen zufolge, eine höhere Datenrate, also geringere Kompression bei besserer Qualität, sowie die Codierung im Zeilensprungverfahren aufgezeichneter Bilder. Eine Umsetzung der in dieser Arbeit entwickelten Methoden zum Schnitt eines MPEG-1 Videos sollte ohne Probleme möglich sein, da beide Normen sehr ähnlich definiert sind.

Weitere Videocodierungen

Zusätzlich zu MPEG sollte das System in Zukunft auch andere Videocodierungen wie z.B. M-JPEG, AVI, DVI, Quicktime usw. bearbeiten und evtl. auch mischen können.

Audio

Die Bearbeitung der Audio-Streams muß noch in das System mit einbezogen werden. Zusätzlich zu dem in konventionellen Schnittsystemen üblichem „Audio follows Video“, das den Tonschnitt einfach dem Videoschnitt unterordnet, sind in einem digitalen Schnittsystem auch alle anderen Möglichkeiten des Tonschnittes, der Tonmischung und sogar der Tonkanalauswahl gegeben.

DVE – Digitale Video Effekte

In das Schnittsystem sollte ein integriertes Videoeffektgerät mit der Bildmischfunktion gekoppelt werden, ähnlich wie dies bei konventionellen Systemen auch schon der Fall ist. Dieses Videoeffektgerät kann dann speziell auf die Eigenschaften der Videocodierungen eingehen.

Digitalisierung analoger Quellvideos

Die Gewinnung der MPEG-Sourcen wurde innerhalb der Arbeit nicht behandelt. Dennoch kann man sie in das System integrieren, was z.B. dann sinnvoll sein kann, wenn es im Offline-Schnitt von analogen Videosourcen verwendet werden soll.

Integriertes Editieren des Drehbuches

Neben dem Editieren des Videos und des Tones dazu stellt auch das Editieren und Aktualhalten des Drehbuches bzw. der Shotliste eine anspruchsvolle Aufgabe dar, die zudem eng mit dem Editieren des Endvideos verknüpft ist. F. Coppolla hat dazu die Idee eines integrierten Editiersystemes entwickelt, welches Editierentscheidungen in einem der drei Bereiche Video, Ton oder Script automatisch auf die anderen Bereiche überträgt (vgl. [Copp88]). Dies ist nur mit einem digitalen System zu erreichen.

GLOSSAR

Bild

Ein Bild ist ein einzelnes Bild aus einer Bildfolge. Es besteht aus einer Anzahl von Zeilen * einer Anzahl von Bildpunkten je Zeile; z.B. beim PAL TV-Standard: 625 Zeilen á 720 Punkte

Bildfolge

Als Bildfolge wird eine „natürliche“ Folge von Einzelbildern bezeichnet, wie sie etwa von einer Film- oder Videokamera erzeugt würde, d.h., aufeinander folgende Bilder sind untereinander inhaltlich stark korreliert.

Bildmischer

Analoges oder digitales Gerät zum Mischen und Überblenden von Videosignalen verschiedener Quellen

Bildschirmflimmern

Obwohl das Auge schon bei 18 Bildern/s eine flüssige Bewegung wahrnimmt, ist bei dieser Bildwiederholrate noch deutlich das Wechseln des Bildes (besonders an großen einfarbigen Flächen) als Flimmern sichtbar. Die Grenze zur Flimmerfreiheit ist stark vom Betrachter abhängig und liegt bei mindestens 42 Bildwechseln pro Sekunde. Im PAL TV-Standard werden mit Hilfe des Zeilensprungverfahrens 50 (Halb-) Bilder pro Sekunde gezeigt.

Bildwiederholrate

Die Bildwiederholrate definiert die Anzahl der Bilder, die pro Sekunde gezeigt werden. Um eine flüssige Bewegung zu erzeugen sind mindestens 18 Bilder/s nötig. Der PAL TV-Standard verwendet eine Bildwiederholrate von 25 Bildern/s.

CCD

Charge Coupled Device

Lichtempfindlicher Sensor auf Siliziumbasis, der eine zeilenweise Abtastung eines Bildes ermöglicht. CCD-Chips werden in Kameras zur Umwandlung des Lichtes in elektrische Signale eingestezt.

CCIR

Comité Consultatif International des Radio Communications,

Internationaler Beraterausschuß für das Funkwesen der Internationalen Fernmeldeunion (ITU), einer Unterorganisation der UNO mit Sitz in Genf.

CoDec

Abkürzung für *Coder* und *Decoder*.

Component Video Signale

Hier werden die Signale der einzelnen Videocomponenten Luminanz, 2x Chrominanz getrennt übertragen. Damit ist pro Komponente eine größere Bandbreite verfügbar als beim Compositesignal, also eine bessere Qualität möglich.

Composite Video Signal

Ein Videoübertragungssignal, das die Informationen von Luminanz und Chrominanz trägt. Das FBAS Signal enthält zu dem noch zusätzlich die Synchronisationssignale. Der Vorteil dieser Übertragungsmethode liegt darin, daß nur ein Übertragungskanal benötigt wird.

Cut-In

Die Stelle eines Videos (das Bild), an dem ein Schnitt durchgeführt wird. Dieses Bild ist dann das erste Bild einer Teilsequenz von Bildern, die an ein anderes Video angefügt werden sollen.

Cut-Out

Die Stelle eines Videos (das Bild), an dem ein Schnitt durchgeführt wird. Dieses Bild ist dann das letzte Bild einer Teilsequenz von Bildern, die an ein anderes Video angefügt werden sollen. Ebenso beschreibt der Cut-Out der Zielsequenz das letzte Bild der alten Szene.

DCT

Diskrete Cosinus Transformation;
verwandt mit der diskreten Fourier-Transformation.
Sie transformiert die Pixelwerte der in MPEG definierten 8x8 Pixelblocks Frequenzkoeffizienten des Ortsfrequenzraumes. Dadurch geht keinerlei Information über den Bildinhalt verloren (abgesehen von dem diskreten Charakter der Operation), d.h., nach der Rücktransformation dieser Koeffizienten in Pixelwerte ist das originale Bild wieder hergestellt.

EBU

European Broadcast Union,
Europäische Rundfunkunion

Edit Decision List (EDL)

Diese Schnittliste enthält alle Daten, die nötig sind, um einen Schnitt durchführen zu können. Es sind dies:

Cutin- und Cutouttimecodes bzw. Länge der Quellszene, Cutin- oder Cutouttimecode auf dem Zielmaterial, Umschnittart, Spezialeffekte, Angaben zum Tonschnitt

Gerätespezifisch können auch noch weitere Informationen hier vermerkt werden.

Die EDL ist heute oft in elektronischer Form gespeichert, und kann somit direkt von einem automatischen Schnittsystem zur Ausführung der Schnittfolge verwendet werden.

Editieren, linear

Beim linearen Editieren hat man nur sequentiellen Zugriff auf das Bildmaterial, speziell der Zielbildfolge. Dies ist besonders bei einer Speicherung auf Magnetbändern (Video) der Fall. Man muß also die Zielsequenz in der gewünschten Reihenfolge aus den Quellsequenzen aufbauen und dazu immer die Quellbänder spulen. Auch die Verwendung digitaler Speicherformen wie MPEG kann einen sequentiellen Zugriff erzwingen, da hier Bilder evtl. nicht unabhängig von ihren Vorgängern dargestellt werden können.

Editieren, nichtlinear

Lineares Editieren entspricht etwa dem Editieren eines Textes, bei dem alter Text immer nur überschrieben, nie aber neuer Text eingefügt werden kann. Daher muß zur Realisierung eines Einfügens aller nachfolgender Text neu geschrieben werden. Nichtlineares Editieren erfordert die Möglichkeit, wahlfrei auf die Quell- und Zielbildfolgen zugreifen zu können, so daß erstens kein Spulen des Quellmaterials notwendig ist und zweitens die Zielsequenz in beliebiger Reihenfolge aufgebaut und evtl. auch wieder umsortiert werden kann. Es ist jederzeit an jeder beliebigen Stelle der Zielsequenz möglich, eine neue Szene einzufügen oder rauszuschneiden, ohne daß andere, schon früher aufgezeichnete Sequenzteile deshalb erneut aufgezeichnet werden müßten.

FBAS

Siehe Composite Video Signal.

FCC

Federal Communications Commission der USA

Halbbild:

Ein Halbbild enthält die Hälfte der Zeilen des Vollbildes. Dabei wird jede zweite Zeile für ein Halbbild verwendet. Pro Vollbild existiert also je ein „gerades“ und ein „ungerades“ Halbbild, je nachdem, ob die geraden oder ungeraden Zeilen verwendet werden. Halbbilder werden im Zeilensprungverfahren verwendet.

HouseSync Signal

Das Housesync Signal gibt an alle Komponenten eines Videostudios einen einheitlichen, zentralgesteuerten Takt ab, nach dem die Geräte dann ihren Bildaufbau, bzw. ihre Bildverarbeitung ausrichten. Nur so ist z.B. das Mischen der Signale zweier unterschiedlicher Signalquellen möglich.

Inter-Codierung

MPEG-Bildcodierungstyp, der die Information anderer Bilder des Streams zur Decodierung benötigt. Inter-Bilder referenzieren andere

Inter- oder Intra-Bilder, statt die Information direkt zu übertragen, Damit ergibt sich eine Einsparung in der Datenmenge, die zur Beschreibung dieser Bilder notwendig ist. Bilder dieses Typs werden als P- (Referenzen nur in einer Richtung) oder B-Bilder (bidirektionale Referenzen) bezeichnet.

Internet

Internationales Computerdatennetzwerk

Intra-Codierung

MPEG-Bildcodierungstyp, der die Decodierung des intracodierten Bildes ohne Information über andere Bilder des Streams erlaubt. Intra-Bilder enthalten damit alle Informationen, die zu ihrer Decodierung notwendig sind. Bilder dieses Typs werden als I-Bilder bezeichnet.

MAZ, Magnetische Aufzeichnung

Aufzeichnung des Video-, Audio- und der Kontrollsignale auf Magnetband. Die hohen Bandbreiten des Videosignals machen eine Aufzeichnung im Schrägspurverfahren nötig, d.h., die Signale werden nicht longitudinal auf einer Spur des Bandes, sondern in mehreren schräg auf dem Material liegenden Spuren geschrieben. Die MAZ ermöglicht und erfordert den elektronischen Schnitt.

NTSC

US-Fernsehstandard mit 30 Bildern/sec bei einer Bildauflösung von 525 Zeilen á 720 Punkte

Offline Editing

Beim Offline Editing entsteht als Ergebnis eine EDL. Es wird also keine Aufzeichnung durchgeführt. Es können allerdings Zuspielderäte zur Betrachtung des Quellmaterials eingesetzt werden.

Online Editing

Ziel des als Online Editing bezeichneten Vorgehens ist es, ein fertiges Video zu erstellen. Daher werden alle Player und Recorder direkt vom Schnittsystem angesteuert und alle Schnitte auch direkt ausgeführt.

PAL

Europäischer Fernsehstandard mit 25 Bildern/sec bei einer Bildauflösung von 625 Zeilen á 720 Punkte.

Quantisierung

Quantisierung ist eine Methode der Datendiskretisierung und kann ebenfalls zur Datenreduktion bereits diskretisierter Daten angewendet werden. MPEG nutzt die Quantisierung, um die Werte der DCT-Koeffizienten, die im Bereich von 0 bis 2^{11} liegen können, auf wenige Werte in demselben Wertebereich zu reduzieren. Legt man z.B. als Quantisierungsstufe 64 fest, ergeben sich im genannten Wertebereich 32 verschie-

dene mögliche Werte: 0, 64, 128, 192, ..., 2048, die sich dann mit 5 Bits codieren lassen.

Referenzauflösung

Mit Referenzauflösung oder „Auflösung der Referenzen“ wird in dieser Arbeit das Umcodieren von intercodierten Bildern bezeichnet. Es hat den Zweck, Referenzen eines Typs (vorwärts oder rückwärts) aufzulösen und das Bild so zu codieren, daß Referenzen dieses Typs nicht mehr vorkommen.

Referenzbild

Als Referenzbild wird in dieser Arbeit ein Bild bezeichnet, das von einem anderen, intercodierten Bild referenziert wird. Es kann sich dabei also nur um ein I- oder P-Bild handeln.

Referenzvektor

Als Referenzvektor wird in dieser Arbeit der Motionvector der MPEG-Norm bezeichnet. Er definiert die Richtung und Entfernung des Bildbereiches des Referenzbildes, den ein Macroblock des referenzierenden Bildes referenziert. Der Referenzvektor kann in Ganz- oder in Halbpixelauflösung gegeben sein. Dies wird pro Bild festgelegt.

Rückwärtsreferenz

Als Rückwärtsreferenz wird in MPEG eine Referenz auf ein zeitlich gesehen zukünftiges Bild bezeichnet. Die Existenz einer solchen Referenzierungsmöglichkeit macht eine Umsortierung der Bildfolge zur Erhaltung der Decodierbarkeit notwendig. Das Referenzbild muß, obwohl es erst nach dem referenzierenden Bild dargestellt wird, schon vor ihm übertragen und decodiert werden.

Schnitt, elektronischer

Das Kopieren einer MAZ-Aufzeichnung auf eine andere. Dazu ist ein hoher steuerungstechnischer Aufwand zur Synchronisation der beteiligten Geräte erforderlich.

Schnittgenauigkeit

Die Schnittgenauigkeit gibt an, wie genau ein Gerät / ein System in der Lage ist, ein Bild in einer Bildfolge anzusteuern, um an dieser Stelle einen Schnitt durchzuführen, d.h. ab hier eine neue Bildfolge zu beginnen. Wünschenswert ist hier natürlich eine Genauigkeit von einem Bild, manchmal sind gerätebedingt aber auch nur Genauigkeiten von 2, 4, oder mehr Bildern erreichbar. Also beginnt die neue Bildfolge nach Durchführung des Schnittes irgendwo innerhalb der Toleranz von 2, 4, oder mehr Bildern nach dem angegebenen Bild.

Schnittliste

Siehe EDL

SMPTE

Society of Motion Picture and Television Engineers,
eines der US-Normungsinstitute und Entwickler des Timecodes

Videocontroller

Zentrales Gerät des Videostudios, dient als Fernsteuerung der Recoder und Player sowie der Bild- und Tonmischpultes im Online-Schnitt und als automatische Schnittsteuereinheit bei der Ausführung einer im Offline-Schnitt entstandenen EDL.

Timecode

Der Timecode ist die eindeutige Adresse eines Bildes in der Bildfolge, die zusammen mit der Bildinformation aufgezeichnet wird. Er setzt sich zusammen aus Stunde, Minute, Sekunde und Bildnummer. Somit impliziert er eine streng lineare Anordnung der Bildfolgen, was aber ähnlich der Speicheradressen einer Rechenanlage nicht auch automatisch einen linearen Zugriff erzwingt.

Die Codierung des Timecodes ist herstellerspezifisch. Die Aufzeichnung kann auf zwei verschiedene Weisen erfolgen:

Als VITC in den Bildzeilen, die in der vertikalen Austastlücke des Bildes übertragen werden, oder

als LTC, der in einer der Tonspuren untergebracht ist.

Vorwärtsreferenz

Als Vorwärtsreferenz wird in MPEG eine Referenz auf ein zeitlich gesehen weiter zurück liegendes Bild bezeichnet. Das Referenzbild liegt also *vor* dem referenzierenden Bild. P-Bilder können nur vorwärts referenzieren. B-Bilder können auch vorwärts referenzieren (zusätzlich zu Rückwärtsreferenzen).

WWW

World Wide Web

Eine vom Kernforschungsinstitut in Cern/CH angeregte Initiative zur Nutzung des Internet, die die Referenzierung und den Zugriff auf beliebige elektronisch gespeicherte Dokumente ermöglicht. Diese Dokumente sind in Form eines SGML Dokumenttyps beschrieben und lassen daher auch eine visuell ansprechende Darstellung zu.

Wysiwyg

Abkürzung von engl.: „What you see is what you get“.

Der optische Eindruck der Darstellung auf dem Bildschirm sollte der des Endproduktes (Ausdruck, Video, ...) möglichst ähnlich sein.

YCrCb

Digitale Entsprechung des YUV-Signales, definiert in der CCIR 601

YUV

Video Bild Komponenten Signal, definiert in der CCIR 624-4, das die Bildinformation in drei verschiedene Kanäle aufteilt. Der Y-Kanal über-

trägt die Luminanzinformation, also das Grauwertbild. Die U- und V-Kanäle übertragen die Chrominanz, also die Farbinformation, die zusammen mit Y das vollständige Farbbild ergibt.

Zeilensprungverfahren

Im Zeilensprungverfahren wird jedes Bild durch zwei Halbbilder dargestellt. Es wird also erst das gerade und dann das ungerade Halbbild angezeigt. Dieses Verfahren ist nötig, um ein Bildschirmflimmern zu verhindern, das bei Bildwiederholraten von unter ca. 42 Hz deutlich sichtbar würde.

LITERATURVERWEISE

- [Adob93] Adobe:
„Adobe Premiere 3.0: Handbuch“
Adobe Systems Inc., Mountainview, Calif., 1993
- [Avid] Autor unbekannt
„AVID/1 Media Composer: Operators Manual“
Avidtechnology Inc., Burlington, MA
- [Behr93a] Behr, Bernd:
„Fremdes Terrain“
c't, 7/93, p 130
- [Behr93b] Behr, Brend; Bertuch, Manfred:
„Maus statt Schneidetisch“
c't, 7/93, pp 124–128
- [Bert93a] Bertuch, Manfred:
„Living Bits on Disk“
c't, 7/93, pp 118–122
- [Bert93b] Bertuch, Manfred:
„PCs zu Videorecordern“
c't, 7/93, pp 132–136
- [Bore91] Borenstein, Nathaniel S.:
„Multimedia Electronic Mail: Will a Dream Become a Reality?“
Com. ACM, 4/91 (34/4), pp 117–119
- [CCIR601] CCIR:
„Encoding parameters of digital Television for Studios“
CCIR Recommendations, Recommendation 601, 1982
- [CCIR624] CCIR:
„Characteristics of systems for monochrome and colour television“
CCIR, Recommendation 624–4, Genf
- [Cher93] Cherfaoui, M.; Joly, P.:
„A survey of automatical tools for the content analysis of digital Video“
ftp.irit.fr/private/svideo.ps, 1993

- [Cock91] Cockroft, Greg; Hourvitz, Leo:
„NEXTStep: Putting JPEG to Multiple Uses”
Com. ACM, 4/91 (34/4), p 45
- [Copp88] Coppolla, Francis Ford:
„Electronic Cinema and Personal Computers”
Keynote Address, Apple Multimedia Developers Conference, San Jose, Calif., 1988
- [Dave89] Davenport, Glorianna; Mackay, Wendy E.:
„Virtual Video Editing in Interactive Multimedia Applications”
Communications of the ACM 7/1989
- [Dave94] Davenport, Glorianna; Elliot, Eddie:
„Videostreamer”
Proc. CHI 94: Human Factors in Computer Systems Conference, Bosten, MA, Apr. 1994, pp 65–66
- [DCT90] IEEE:
„Specification for the implementation of 8x8 inverse cosine transform”
Draft Standard P1180/D2, 1990
- [Denn74] Dennewitz, Rolf D.:
„Die Loewe Story”
Loewe AG, 1974
- [Enc88] Encarnação, J.; Straßer, W.:
„Computer Graphics”
3. Auflage, Oldenburg, 1988
- [Fara92] Farag, W.; Remley, F.:
„Digitized Image Data Compression and Transfer”
CITI Technical Report 92–5, University of Michigan, 1992
- [Fast93] Autor unbekannt
„VideoMashine: Manual”
Fast Electronics Inc, Natick, MA, 1993
- [FAX80] CCITT:
„Standardization of group 3 facsimile apparaturs for document transmissions”
CCITT, Facsimile VII.2, Recommendation T4, 1980
- [Fels93a] Felsenberg, Alexander:
„Einsatz des non-linearen Computer-Editings in der Filmproduktion”
Professional Production 1/1993, pp 15–22, Verlag Gerhard Spiels, Kottgeisering
- [Fels93b] Felsenberg, Alexander:
„High Resolution-Editing Cinesite in Burbank”
Professional Production 6/1993, pp 38–39, Verlag Gerhard Spiels, Kottgeisering

- [Film94] Autor unbekannt
„Frauen führen heimlich die Regie“
Film & Tv Kameramann 2/43 (1994), pp 56–57, Verlag J. Weber, München
- [Foley90] Foley; van Dam; et. al.:
„Computer Graphics – Principles and Practice“
2nd Edition, Addison Wesley, 1990
- [Fox91] Fox, Edward A.:
„Standards and the Emergence of Digital Multimedia Systems“
Com. ACM, 4/91 (34/4), pp 26–29
- [Gall91] Le Gall, Didier:
„MPEG: A Video Compression Standard for Multimedia Applications“
Com. ACM, 4/91 (34/4), pp 46–58
- [Gebh94] Gebhard, Christine; Voigt–Müller, Gerhard:
„Zwischen den Welten“
Film & Tv Kameramann 3/43 (1994), pp 38–47, Verlag J. Weber, München
- [Hard93] Hardman, L.; Rossum, D.v.; Bulterman, D.:
„Structured Multimedia Authoring“
Technical Report CS–R9304, Universitet Amsterdam, 1993
- [Harn91] Harney, Kevin; Keith, Mike; Lavelle, Gary; Ryan, Lawrence D.; Stark, Daniel J.:
„The i750 Video Processor: A Total Multimedia Solution“
Com. ACM, 4/91 (34/4), pp 64–80
- [Heise83] Heise, W.; Quattrochi, P.:
„Informations– und Codierungstheorie“
Springer 1983
- [Hert90] Hertwich, Ralf Guido:
„Audio and Video in Distributed Computer Systems: Why and How?“
TR–90–026, 7/1990, International Computer Science Institute, Berkeley, Calif.
- [Hilg93] Hilgfort, Ulrich:
„TV–Studio im Zweierkomplement“
c‘t, 7/93, pp 138–139
- [Hoffe91] Hoffert, Eric M.; Gretsche, Greg:
„The Digital News System at EDUCOM: A Convergence of Interactive Computing, Newspapers, TV and High–Speed–Networks“
Com. ACM, 4/91 (34/4), pp 113–116
- [Hoffm92] Hoffmann, Georg Rainer:
„Imaging: Bildverarbeitung & Kommunikation“
Script zur Vorlesung, TH Darmstadt, WS 1991/92
-

- [Huff53] Huffman, David A.:
„A Method for the Construction of Minimum–Redundancy Codes”
Proc. IRE, 7/53, pp 1098–1101
- [Hung93] Hung, Andy C.:
„PVRG–MPEG CODEC 1.1”
Online Manual, 6/93
- [HYTM91] ISO/IEC:
„Information technology – Hypermedia/Time–based Structuring Language (Hytime)”
ISO/IEC, Committee Draft International Standard 10743, 4/91, Genf
- [JPEG92] International Standard 10918:
„Digital Compression and Coding of Continuous Still Images”
ISO/IEC JTC1 International Standard 10918, 92
- [Kröm91] Krömker, Detlef:
„GDV III: Animation und Simulation”
Script zur Vorlesung, TH Darmstadt, WS 1990/91
- [Krug93a] Krug, Hermann:
„Desktop Video mit AVID Media Suite Pro”
Professional Production 5/1993, pp 42–43, Verlag Gerhard Spiels, Kottgeisering
- [Krug93b] Krug, Hermann:
„Kreatives Editing mit Henry”
Professional Production 4/1993, pp 16–20, Verlag Gerhard Spiels, Kottgeisering
- [Krug93c] Krug, Hermann:
„TV–Symposium: Die digitalen Offline–Systeme streben zielsicher den Online–Einsatz an”
Professional Production 8/1993, pp 18–20, Verlag Gerhard Spiels, Kottgeisering
- [Krug93d] Krug, Hermann:
„TV–Symposium: Equipment für Production und Postproduction”
Professional Production 8/1993, p 21, Verlag Gerhard Spiels, Kottgeisering
- [Krug93e] Krug, Hermann:
„TV–Symposium: Diskussionen über den Einsatz der Datenreduktion bei der Bandaufzeichnung”
Professional Production 8/1993, pp 14–17, Verlag Gerhard Spiels, Kottgeisering
- [Krug93f] Krug, Hermann:
„TV–Symposium: Digitale Bearbeitung von Film”
Professional Production 8/1993, pp 11–12, Verlag Gerhard Spiels, Kottgeisering
- [Krug93g] Krug, Hermann; Letocha, Thomas:
„Desktop–Video: Das Postproduction Studio auf dem Schreibtisch”
Professional Production 1/1993, pp 23–27, Verlag Gerhard Spiels, Kottgeisering

- [Krug94] Krug, Hermann:
„NAB'94: Die digitale Revolution geht unaufhaltsam weiter”
Professional Production 5/1994, pp 10–19, Verlag Gerhard Speihs, Kottgeisering
- [Lege84] Leger, A.:
„Implementations of fast discrete cosine transform for full color video services and terminals”
Proc. IEEE, Global Telecommunications Conf., 1984
- [Lieb91] Liebhold, Michael; Hoffert, Eric M.:
„Toward an Open Environment for Digital Video”
Com. ACM, 4/91 (34/4), pp 103–112
- [Liou91] Liou, Ming:
„Overview of the px64 kbit/s Video Coding Standard”
Com. ACM, 4/91 (34/4), pp 59–63
- [Lipp91] Lippman, Andrew:
„Feature Sets of Interactive Images”
Com. ACM, 4/91 (34/4), pp 92–102
- [Litt93] Little, T.D.C.; et al.:
„A Digital On–Demand Service Supporting Content–Based Queries”
Proc. ACM Multimedia, 8/93, pp 427–435
- [MPEG93] International Standard ISO/IEC 11172:
„Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s”
Part1: Systems
Part2: Video
Part3: Audio
ISO/IEC MPEG 11172, First Edition 8/93
- [Müll92] Müller, A.:
„Der elektronische Schnitt”
Heiko Sven Hausemann Filmproduktions– und Verlags–GmbH, Hamburg, 1992
- [Münz93] Münzer, Wulf, Dr.:
„Digitales Fernsehen und digitale Heimaufzeichnung”
Professional Production 4/1993, pp 39–41, Verlag Gerhard Speihs, Kottgeisering
- [Neid93] Neidecker–Lutz, B.; Ulichney, R.:
„Software Motion Picture”
Digital Technical Journal Vol 5, No 2, 1993
- [NJW93] Autor unbekannt
„Zum Gerichtsurteil des VG Oldenburg, NJW 1991, 2921”
Neue Juristische Wochenschrift 1993, 551
-

- [Ohan93] Ohnaian, Thomas A.:
„Grundfragen zur Bildspeicherung bei nonlinearen Schnittsystemen“
Professional Production 7/1993, pp 27–28, Verlag Gerhard Spiels, Kottgeisering
- [Panc92] Pancha, Pramod; Zarki, Magda El:
„A Study of MPEG Coding for Variable Bit Rate Video Transmission“
Technical Report VPTL 92–01, UPenn–DEE, 4/92
- [Panc93] Pancha, Pramod; Afzal, Hassan; Ho, Grace; Urabe, Toshiyuki:
„MPEGTool: An X Window Based MPEG Encoder and Statistics Tool“
Proc. ACM Multimedia, 8/93
- [Poon94] Poon, Alex; Weber, Karon:
„Marquee: A Tool for Realtime Video Logging“
Proc. CHI 94: Human Factors in Computer Systems Conference, Boston, MA, Apr. 1994, pp 58–65
- [Pott94] Potthoff, Martin:
„Edit–Transfer: Schnittlisten Verwaltung für Apple Macintosh“
Film & Tv Kameramann 4/43 (1994), pp 104–110, Verlag J. Weber, München
- [PP93a] Autor unbekannt
„Digitale Komponentenaufzeichnung“
Professional Production 6/1993, pp 28–29, Verlag Gerhard Spiels, Kottgeisering
- [PP93b] Autor unbekannt
„Digitale Systeme mit Datenreduktion zur Bildverarbeitung“
Professional Production 6/1993, pp 30–33, Verlag Gerhard Spiels, Kottgeisering
- [PP93c] Autor unbekannt
„Digitales System ohne Datenreduktion zur Bildverarbeitung“
Professional Production 6/1993, pp 34–36, Verlag Gerhard Spiels, Kottgeisering
- [PP93d] Autor unbekannt
„Neues Desktop–Video–System Media 100 für die non–lineare Videoproduktion“
Professional Production 12/1993, p 6, Verlag Gerhard Spiels, Kottgeisering
- [PP94] Autor unbekannt
„Fernseh–Zukunft: Digital Video Broadcasting“
Professional Production 4/1994, p 6, Verlag Gerhard Spiels, Kottgeisering
- [Prit80] Pritchard, D.:
„Worldwide color television standards – similarities and differences“
SMPTE, 1980
- [Rana91] Ranagan, P. Venkat; Vin, Harrick M.:
„Designing File Systems for Digital Video and Audio“
Proc. 13‘th ACM symposium on Operating Systems Principles (SOSP‘91), Monterey, Operating Systems Review, 10/91 (Vol.25/No.5), pp 69–79
-

- [Roed91] Roedler, D.:
„Telekommunikationsdienste und -netze“
Script zur Vorlesung, TH Darmstadt / Digital Equipment, 1991
- [Rowe92] Rowe, Lawrence A.; Smith, Brian:
„A Continuous Media Player“
Proc. 3rd Intl. Workshop on Network and Operating System Support for Digital Audio and Video, San Diego 11/92
- [Rowe93a] Rowe, Lawrence A.; Smith, Brian; Patel, Ketan:
„Performance of a Software MPEG Video Decoder“
Proc. ACM Multimedia Conference, 8/93
- [Rowe93b] Rowe, Lawrence A.; Smith, Brian C.:
„A New Family of Algorithms for Manipulating Compressed Images“
IEEE Computer Graphics and Applications, 7/93 (No.5/Vol.13), pp 34–42
- [Schu72] Schulz, Werner:
„Das kleine Videopraktikum“
Franzis-Verlag, München, 1972
- [Sijs91] Sijstermans, Frans; Van der Meer, Jan:
„CD-I Full-Motion Video Encoding on a Parallel Computer“
Com. ACM, 4/91 (34/4), pp 81–91
- [Spie93] Spiehs, Gerhard:
„Marktbefragung: Post-Production Studios“
Professional Production 6/1993, pp 11–22, Verlag Gerhard Spiehs, Kottgeisering
- [Stro92] Stroustrup, Bjarne:
„Die C++ Programmiersprache“
Addison Wesley, München, 2.Auflage 1992
- [Süll93] Süllow, Klaus:
„Technologie zur spontanen Nutzung von digitalem Video“
Der GMD-Spiegel, 2/93, pp 71–73
- [TC86] IEC:
„Time and control code for video tape recorders“
IEC Standard Publication 461, 2nd Edition, 1986
- [Tsich87] Tsichritzis, D.; et al:
„Muse: A Multi-Media Filing System“
IEEE Software, 6/87 (4(2)) pp 4–15
- [Turn72] Tuner, I.:
„Video recording – Its Application to Information Retrieval“
Report No. 73/14, Institution of Electrical Engineers, London 1972

- [Vett84] Vetterli, M.:
„Multidimensional subbandcoding: Some theory and algorithms”
Signal Proceedings 6, 1984
- [Voig94] Voigt–Müller, Gerhard:
„Neues vom nonlinearen Editing”
Film & Tv Kameramann 4/43 (1994), pp 18–35, Verlag J. Weber, München
- [Wall91] Wallace, Gregory K.:
„The JPEG Still Picture Compression Standard”
Com. ACM, 4/91 (34/4), pp 30–44
- [Wolf93] Wolf, Fritz:
„Konstanzer Workshop über die Digitale Fernsehzeitung”
Professional Production, 12/93, pp 40–43, Verlag Gerhard Spiels, Kottgeisering