MPEG-7

- Motivation
- Anwendungsgebiete
- Content Description

- Document Description Language (DDL)
- Description Schemes (DS)
- Descriptions (D)

Beispiele:
- Audio: Beschreibung von Melodien
- Audio: Beschreibung und Vergleich von Klangfarben
- Video: Erkennung von Szenenwechseln (Schnitte)
- Video: automatische low-level Szenenbeschreibung

(www.cselt.it/mpeg/)

Literatur

MPEG Homepage, www.cselt.it/mpeg/
MPEG-7 Homepage: www.mpeg-7.com/

diverse Konferenz/Workshop-Beiträge und Tutorials auf obigem Server, u.a.:
E.J.Delp, Image and video databases: who cars?, MPEG7 IMA tutorial, 2001
P. Salembier, Status of MPEG-7, IBC 2000, Amsterdam

zum Vergleich: inhaltsbasierte Bildsuche (Gnu Image finding tool):
viper.unige.ch/ www.gnu.org/software/gift/
"Suchen ohne Worte", c't 15/2001, 162ff

MPEG-7: Motivation

- A/V-Material zunehmend digital verfügbar
- weltweiter Zugriff via Internet / Datenbanken / Tauschbörsen

wie sucht und findet man Multimedia-Daten?

- effiziente Algorithmen für Volltextsuche bekannt (inverted tables)
- z.B. Internet-Suchmaschinen (Google, Altavista)
- oder (manuelle) Klassifizierung (Yahoo)

- aber: bisher keine entsprechenden Algorithmen für A/V

=> Annotation von A/V-Daten mit Textbeschreibungen

- Metabeschreibung (data about data)
- nicht unbedingt in den A/V-Daten selbst enthalten
- unabhängig vom Format der A/V-Daten

(Herre)

MPEG-7: Ziele

MPEG-Standard zur
- Beschreibung des Inhalts audio-visueller Information

- zur schnellen Suche und Identifikation von Inhalten
- Beschreibung diverser Aspekte der Medien:
- "low-level, structure, semantic, models, collections, creation, ..."

- für eine Vielfalt von Anwendungen
- unabhängig von Datenformat der Medien selbst
- auch zur Beschreibung von analogem Material

- für Audio, Sprache, Bilder, Video, 3D-Graphik, ...
- Szenenbeschreibung der Kombination mehrerer Medien

(Salembier)

MPEG-7: Anwendungen . . .

vielfältige Anwendungen denkbar:

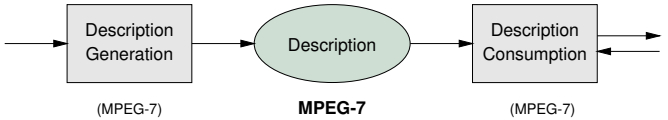
- Organisation und Suche in AV-Datenbanken (Bilder, Video, Radio, ...)
- Programmauswahl bei Rundfunk / Fernsehen
- Überwachung (z.B. Stauwarnungen, Maschinensteuerung, ...)
- Luftbildauswertung (z.B. Kartographie, Ökologie, Exploration)
- Journalismus (z.B. Suche nach Personen und Ereignissen)
- E-commerce, Teleshopping (z.B. Suche nach bestimmten Stoffen)
- Personalisierte News-Services (z.B. im push-services im Internet)
- Unterhaltung (z.B. Suche nach einem Karaoke-Stück)
- Kultur (z.B. Museen)
- Ausbildung, Multimedia, ...
- uva.

MPEG-7: Anfragen . . .

Beispiel für mögliche Anfragen an MPEG-7 Beschreibungen:

- Text-basierte Suche, z.B. nach Schlüsselwörtern:
- z.B. alle Filme, deren Beschreibung das Wort "MPEG" enthält
- Semantische Beschreibungen
- Suche nach ähnlichen Bildern
- z.B. ausgehend von einer Vorlage des Anwenders
- Suche nach Musikstücken
- z.B. ausgehend von der Melodie oder einem Rhythmus
- Suche nach "low-level" Eigenschaften
- z.B. alle Filme mit charakterischen Objektbewegungen (Trajektorien)

MPEG-7: Leistungsumfang



MPEG-7:

- Definition der Beschreibungen
- inklusive des zugehörigen Datenformats

nicht standardisiert (vorgesehen für späteren Wettbewerb):

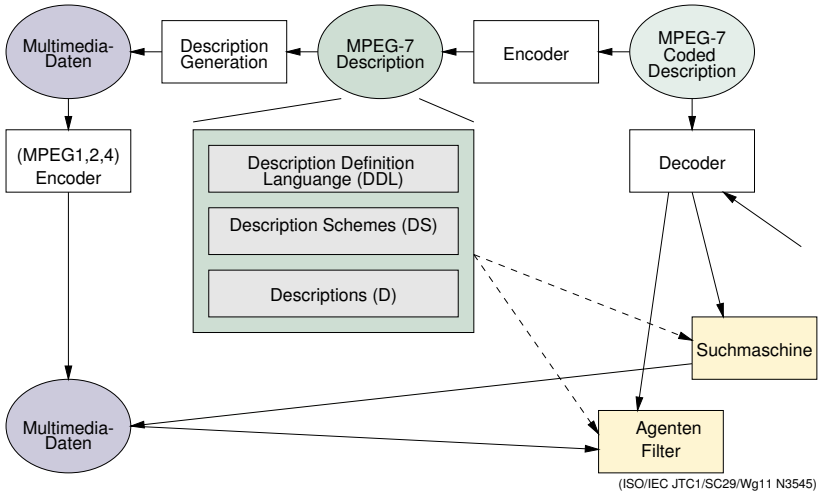
- die Erzeugung der Beschreibungen
  Merkmals-Extraktion, Indizierung, Annotation, Authoring, ...
- das Auswerten der Beschreibungen
  Suche, Browser, Filter, ...

MPEG-7: Teile

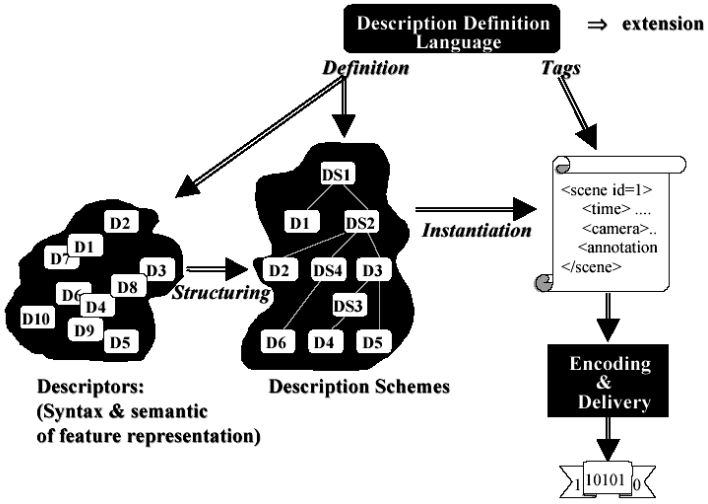
MPEG-7 = ISO/IEC 15938

1) Systems
2) Description Definition Language (DDL)
3) Visual
4) Audio
5) Multimedia Description Schemes (DS)
6) Reference Software

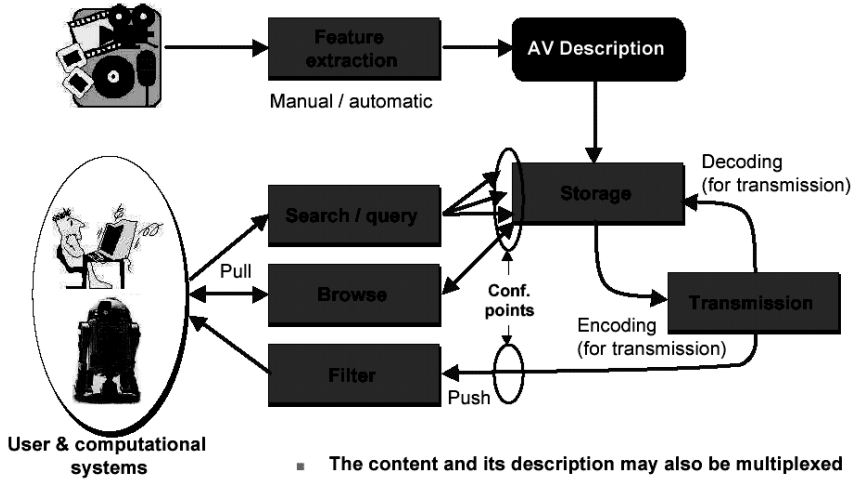
MPEG-7: Blockdiagramm



MPEG-7: D, DS, DDL, Kodierung



MPEG-7: Informationsflüsse



- **The content and its description may also be multiplexed**

MPEG-7: DDL

"Description Definition Language":

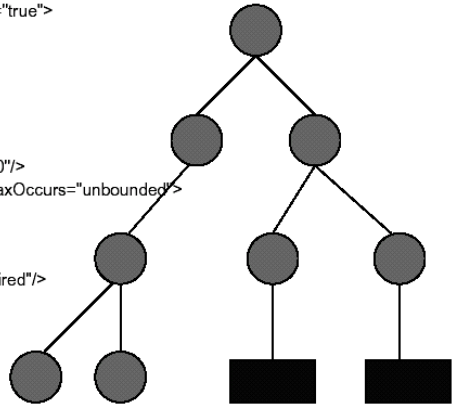
- Data — z.B. MPEG-4 Video, CDDA, Word-Dokument
- Descriptor — Beschreibung eines Merkmals
- Description Schema — Struktur/Semantik von Descriptors

- basiert auf XML
  - einfache Datentypen, Elemente
  - Vererbung, abstrakte Datentypen

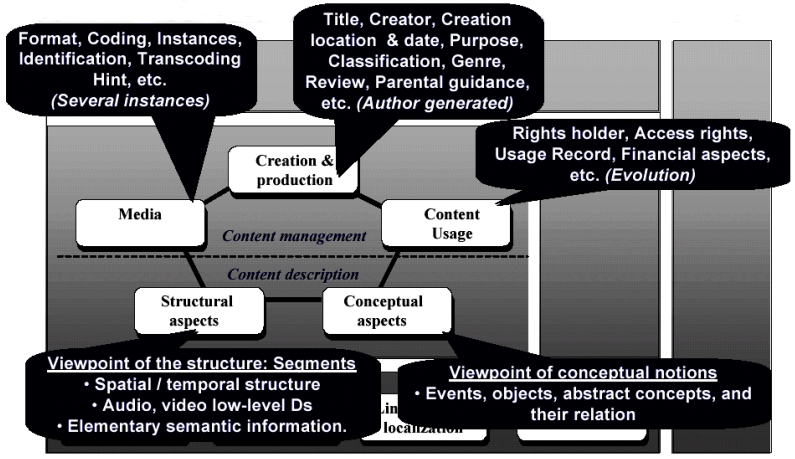
- Erweiterungen durch MPEG-7:
  - Array- und Matrix-Datentypen
  - Datentypen für MimeType, CountryCode, RegionCode, usw.
  - typisierte Referenzen

MPEG-7: DDL Beispiel

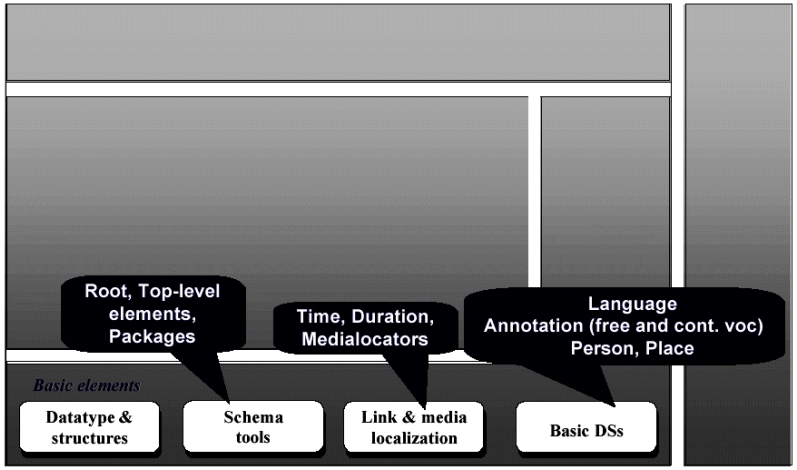
```
<complexType name="AudioLLDScalarType" abstract="true">
 <complexContent>
  <extension base="mpeg7:AudioDType">
   <choice>
    <element name="SegmentSummary">
     <complexType>
      <sequence>
       <element name="Mean" type="float" minOccurs="0"/>
       <element name="OtherMethod" minOccurs="0" maxOccurs="unbounded">
        <complexType>
         <simpleContent>
          <extension base="float">
           <attribute name="label" type="string" use="required"/>
          </extension>
         </simpleContent>
        </complexType>
       </element>
      </sequence>
     </complexType>
    </element>
```



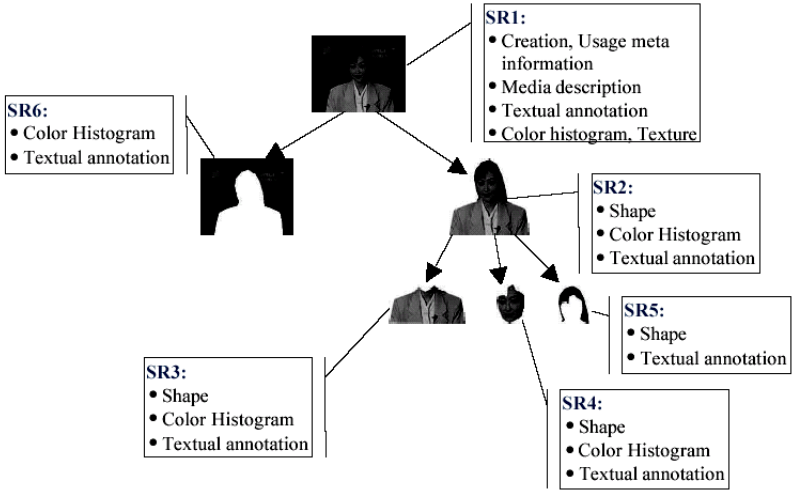
MPEG-7: Content Management



MPEG-7: DDL Basic Elements

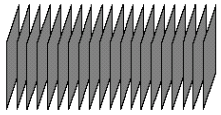


MPEG-7: Segment Tree



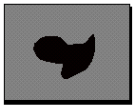
MPEG-7: low-level AV Descriptors

Video segments



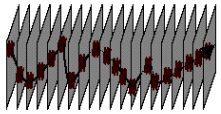
- Color
- Camera motion
- Motion activity
- Mosaic

Still regions



- Color
- Shape
- Position
- Texture

Moving regions

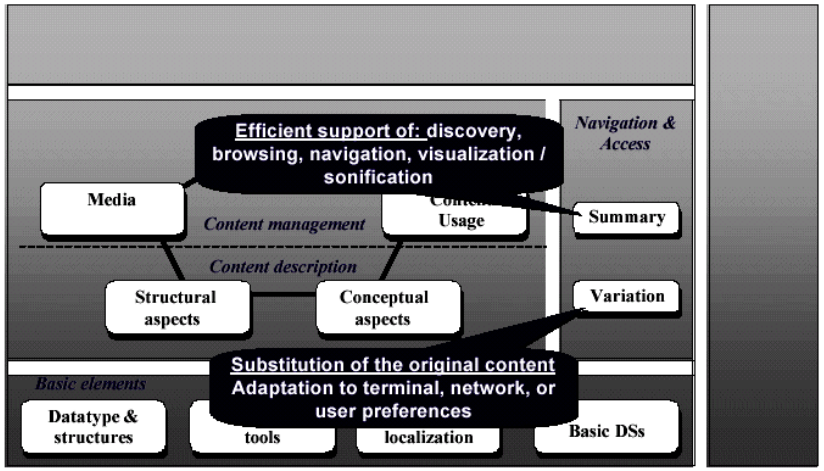


- Color
- Motion trajectory
- Parametric motion
- Spatio-temporal shape

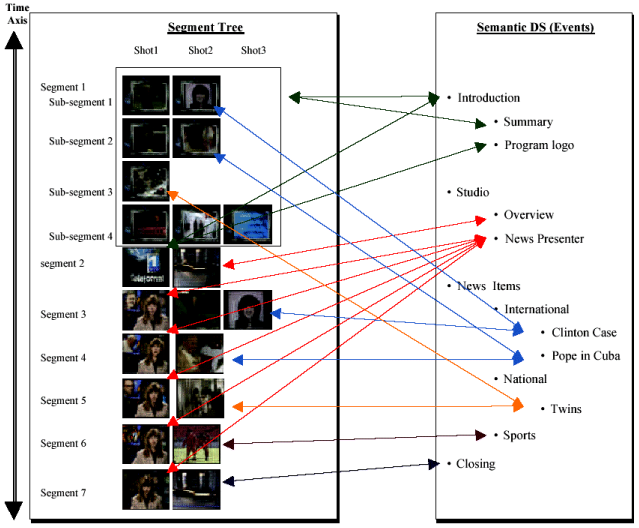
Audio segments

- Spoken content
- Spectral characterization
- Music: timbre, melody

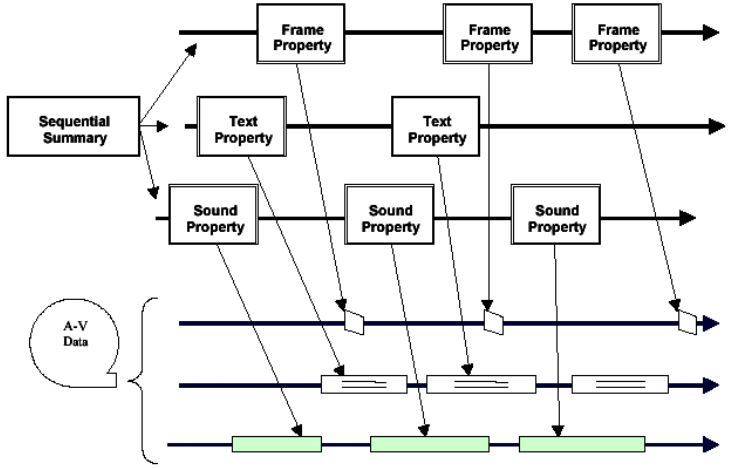
MPEG-7: Navigation



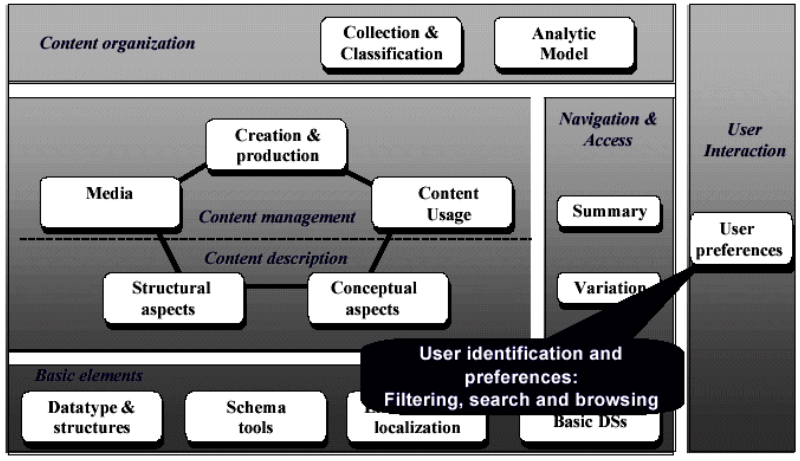
MPEG-7: Events



MPEG-7: Sequential Summary



MPEG-7: User Interaction



MPEG-7: Audio

einige aktuell untersuchte Anwendungen:

- Beschreibung von Sprache, Spracherkennung
- Framework zur autom. Erkennung von Audiodaten:
  - Klangfarben, Instrumentfamilien
  - Melodiebeschreibung und -erkennung
  - robuste Identifizierung von Musikstücken
- low-level Merkmale;
  - Wellenformen, Spektrum, Cepstrum
  - Signalparameter, Grundfrequenz, Obertöne
  - Klangfarben, Hüllkurven
  - usw.

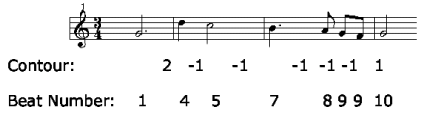
MPEG-7: Melody Description

- Menschen erinnern Musik oft über Melodien
- Melodien als Vorlage zur Suche nach Musikstücken
- aber Vorsingen über Mikrophon sehr fehleranfällig:
- falsche Tonart, anderes Tempo, veränderter Rhythmus, ...
- Erinnerung nur unvollständig, usw.

=> kompakte und robuste Repräsentation?

- Folge von relativen Tonhöhen ("pitches"):
  - invariant gegen Transponieren und Klangfarbe
  - 5-stufige Werteskala für Tonhöhendifferenz: (-2, -1, 0, +1, +2)
  - robust gegen ungenaues Vorsingen und die meisten Fehler
- zusätzlich Abspeichern eines (quantisierten) Rhythmus
- Erzeugen der Beschreibung z.B.aus MIDI-Dateien

MPEG-7: Beispiel "Moon River"




```
<!-- MelodyContourDS description of “Moon River”   -->
<!-- (7 intervals = 8 notes total)                  -->
<Contour>
  <ContourData>2 -1 -1 -1 -1 -1 1</ContourData>
</Contour>
<!-- Meter of melody                                -->
<Meter>
  <Numerator>3</Numerator>
  <Denominator>4</Denominator>
</Meter>
<!-- Beat positions of notes                        -->
<!-- (8 notes = 1 more than number of intervals)    -->
<Beat>
  <BeatData>1 4 5 7 8 9 9 10</BeatData>
</Beat>
```

MPEG-7: Audio Matching

- inhaltsbasierte Erkennung von Audiodaten
- durch robusten Vergleich mit Referenzdaten

- Robustheit notwendig: Erkennung trotz Anwendung von:
  - lineare Filter (Lautstärke, Filter, Equalizer, ...)
  - nicht-lineare Filter (Kompression, MP3-Kodierung, ...)
  - geschnittenen Daten

Anwendungen:

- gezielte Suche nach bestimmten Musikstücken
- Suche nach ähnlichen Stücken (z.B. E-Commerce)
- "Audio Fingerprinting"
- z.B. zur Überwachung von Verwertungsrechten
- automatische Zuordnung von Metadaten (wie CDDB, ID3v2)

MPEG-7: Audio Matching

Erkennungsrate des Algorithmus? Demo:

- Testdatenbank mit 15.000 Stücken (Pop/Rock, je 30 sec.)

Originalsignal:	100.0%
Ausschnitt (20 sec.):	99.9%
MP3 (96 kb/s stereo):	100.0%
MP3 & Ausschnitt:	99.7%
Lautsprecher/Mikrophon:	99.3%

- Signaturen insgesamt 15 MByte (1KB/Stück/30 sec.)
- sehr schnelle Erkennung (0.25 sec auf P3-500 / 80x Echtzeit)
- aber: Skalierbarkeit? Verhalten bei sehr ähnlichem Material?

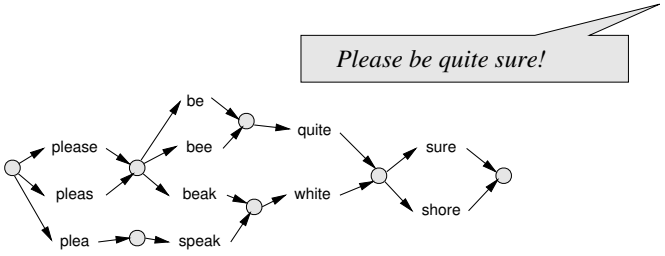
MPEG-7: Audio Matching

aber wie?

- AudioSpectrumFlatness() Descriptor

- beschreibt Spektrum des Audiosignals
- in mehreren Frequenzbändern (z.B. tonal - noise)
- robust gegenüber fast allen Filteroperationen
- sehr kompakt kodierbar, z.B. 4 Werte/s mit 8 bit/Wert
- (aber Binärformat noch nicht in MPEG-7 spezifiziert)

MPEG-7: Speech Description



- Spracherkennung meistens mit HMM (hidden markov models)
- Eingabedaten sehr oft mehrdeutig

MPEG-7 speech description:

- Speicherung der "lattices" anstelle des erkannten Textes
- erlaubt spätere Auswahl der richtigen Deutung

Bild-/Videodatenbanken: das Problem

extreme Datenmengen von Videodatenbanken:

- unkomprimiertes Video kaum handhabbar
- aber welches Kompressionsverfahren ist geeignet?
- Bsp (MPEG-2, 6Mb/s): 90.000 Bilder/h, 3 GB/h
- Archiv eines Senders: 68 GB/day, 24.800 GB/year, 788M frames/year

- ideale Datenbank sammelt viele Sender...
- zum Vergleich: Google derzeit 3G Webseiten (à 100 KB ?!)

- Verwaltung der Daten? Suche in komprimierten Daten möglich?
- Klassifikation der Daten? automatisch oder manuell ...
- Formulierung von Anfragen? Anfragesprachen?
- Browsing der Datenbank? Index, Zusammenfassungen, ...
- Auslieferung der Daten: I/O-Bandbreite, Streaming, QoS?

TV/Video: Marktbedeutung

Prozentsatz der US-Haushalte mit

- mindestens einem Fernseher: 98 %
- zwei Fernsehern: 34%
- drei oder mehr Fernsehern: 40 %
- mindestens einem Videorekorder 84 %

- "the average American watches 3hrs 35mins of TV each day"

Zahlen für Europa / Deutschland ?!

- kein Wunder, dass die GEZ mir nicht glaubt :-)

(Delp, IMA, www.oc-profam-net.org/media/tv_statistics.htm)

Interesse an interaktivem TV . . .

- how appealing is interactive TV?

very appealing	14%
somewhat appealing	34%
not very appealing	21%
not at all appealing	29%
don't know / not sure	2%

Ursache / Probleme ?

- reine Konsumhaltung: "Fernseher leergucken"
- Potential wird nicht erkannt, vgl. single- vs. multiplayer Games
- ???

(Angus Reid Group, Red Herring, August 2000, ot of 1000 Americans)

What do users want?

time-shifting programs	47%
video conferencing	36%
video on demand	35%
getting many more channels	33%
being able to control camera angles	30%
using TV to surf the web	24%
using TV to write and receive email	24%
play games with groups of people who have iTV	14%
shopping over TV	12%

- und was wollen die Anbieter / Sender ?!

(ebenda)

Bild-/Videodatenbanken: Wozu?

drei Anwendungs-Szenarien:

- Heimanwender-Datenbank
- Video-on-Demand
- Digitale Bibliotheken

- weitere?

Heim-Datenbanken . . .

Szenario:

- billige Digitalkameras und Videokameras
- jeder hat seinen PC, seine Webseite, seine Kameras
- Erwartung: in 10 Jahren über 90% aller Bilder und Videos digital

=> Markt für "Heim"-Bild- und Videodatenbanken !?

- Suche nach den Hochzeitsfotos / der Einschulung / usw.
- Aufbau von Bildserien / Geschichten ("wie die Kinder wachsen")

Problem:

- mehr als 60 Mrd. Fotos pro Jahr aufgenommen ...
- ... aber jedes Foto weniger als 1 Mal angeschaut

=> Sammlung im Schuhkarton reicht auch in Zukunft aus
=> vermutlich keine Marktbedeutung

Video-on-demand . . .

Szenario:

- Anwender wollen gezielt nach (Unterhaltungs-) filmen suchen
- Videodatenbank erlaubt die effizienten Suche
- personalisierte Informationen / Präferenzen
- Datenbank zugänglich via WWW oder das DVB- / Kabelnetz

Problem:

- Durchschnittsanwender wählen nach einfachen Kriterien:
  Film-Kategorien / Schauspieler / Filmkritiken / Mundpropaganda / ...

- das sind alles Text-Informationen
- keine komplexen Suchfunktionen notwendig

=> sondern nur eine gute Programmübersicht / -zeitschrift

digitale Bibliotheken . . .

Szenario:

- vernetzte Datenbanken für Schule / Ausbildung
- natürlich auch für (kommerzielle) Recherchen
- erst sekundär auch zur Unterhaltung
- Datenbank wird von Profis (nicht Heimanwendern) verwaltet / gepflegt

  Beispiel: Datenbank mit allen Bundesliga-Spielen:

  - alle Anwender: Wiederholung interessanter Szenen
  - Reporter: Recherche / Vorbereitung von Reports
  - Talentscout: Suche nach Talenten
  - Fan: "zeig mir das letzte Tor von St. Pauli"
  - usw.

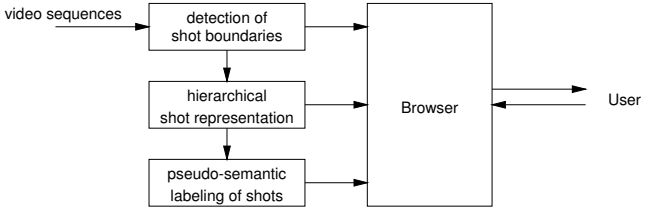
=> dieses Szenario könnte (sollte) funktionieren

Was ist der "Inhalt" eines Films?

- **“Play it again Sam!”**
- **Man facing woman**
- **Casablanca**
- **Ingrid Bergman**
- **Humphrey Bogart**
- **Famous movies**
- **Close-up shot**
- **“Not Sports”**

Content is dependent on the particular user group querying the system

ViBE: Videodatenbank

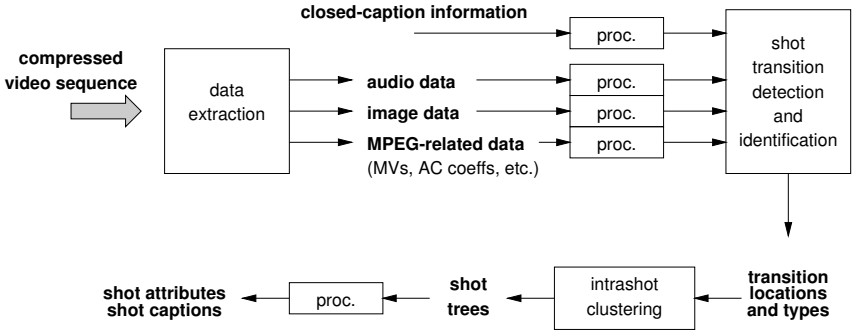


vier Grundfunktionen:

- Erkennung von Szenenwechseln, Erkennung von Szenen
- hierarchische Repräsentation von einzelnen Shots
- pseudo-semantische Benennung von Shots
- interaktives Browsen der Datenbank mit "relevance feedback"
- Framework mit Option zur Integration weiterer Komponenten

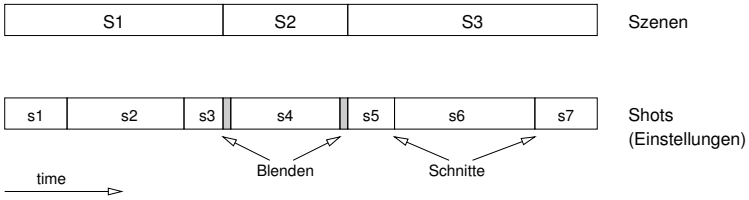
(www.ima.umn.edu/~dep, www.ece.purdue.edu/~ace)

Video-Analyse: Beispiel



- automatisches Erzeugen von Szenenbeschreibungen
- direkt aus den (komprimierten) Eingabedaten

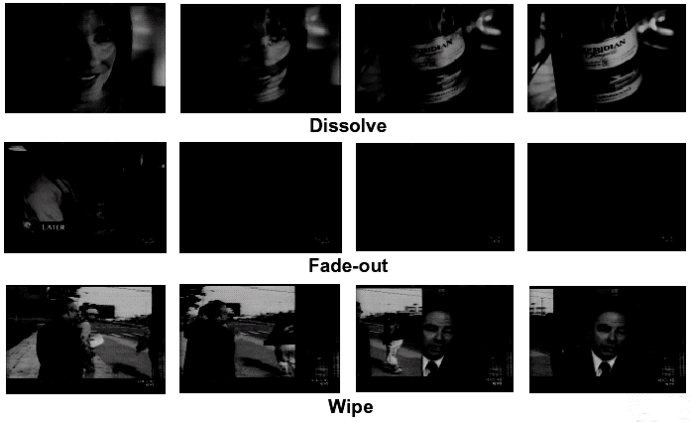
ViBE: temporale Segmentierung



- automatische Auftrennung eines Films in zusammengehörige Szenen
- Zuordnung aufgrund inhaltlicher oder visueller Merkmale

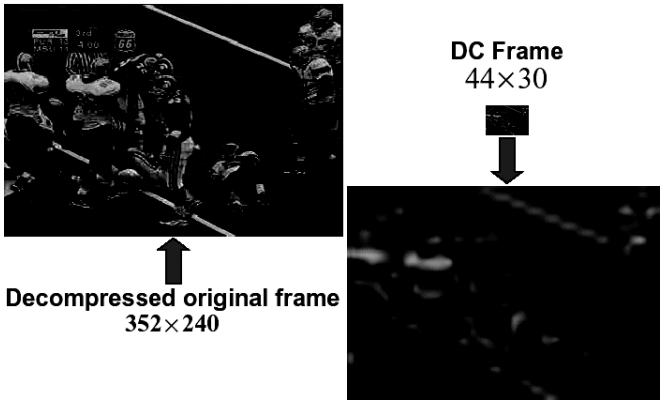
- erfordert die Erkennung von Szenenwechseln (shot boundaries)
- und möglichst auch die Art der Szenenwechsel

ViBE: Szenenwechsel



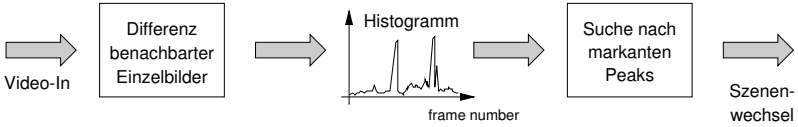
- harte Schnitte, Überblenden, Ausblenden, Wischblenden, usw.
- Übergänge oft typisch für bestimmte Inhalte / Genres / usw.

ViBE: DC-Frames



- Berechnung verwendet nur die (MPEG-) DC-Koeffizienten
- dramatisch reduzierte Datenmenge für die Erkennung
- aber Auflösung evtl. zu gering (Details stecken in den AC-Koeffizienten)

ViBE: Erkennung von Schnitten



diverse Algorithmen vorgeschlagen:

- pixelbasierte Differenz aufeinanderfolgender Einzelbilder
- Grauwert- oder Farbhistogramme
- Kantenerkennung
- Auswertung der "Motionvectors" aus der Bewegungskompensation
- Modellbasierte Verfahren
- Klassifikation der Bildinhalte
- zeitbasierte Verfahren, Adaption an "typische" Szenenlänge

ViBE: Testdatensatz

	frames	cuts	dissolves	fades	others
soap opera	67582	337	2	0	0
talk show	107150	331	108	1	6
sports	78051	173	45	0	29
news	58219	297	7	0	6
movies	54160	262	15	6	1
cspan	90269	95	19	0	0
TOTAL	**455431**	**1495**	**196**	**7**	**42**

Testdaten mit Videosequenzen, insg. 10 Stunden Video:

- 6 unterschiedliche Genres
- jeweils MPEG-1, 1.5 Mb/s, CIF 352x240
- aus Fernsehaufnahmen (Werbung herausgeschnitten)

ViBE: Performance mit den Testdaten

	Tree Classifier			Sliding Window			Simp. Thresholding		
	Detect	FA	MC	Detect	FA	MC	Detect	FA	MC
soap	0.941	13.3	0	0.916	99	0	0.852	24	0
talk	0.942	32.3	7.5	0.950	45	1	0.968	171	15
sports	0.939	82.5	34.8	0.785	59	1	0.925	251	73
news	0.958	38.0	0.75	0.886	61	0	0.926	212	1
movies	0.821	43.3	2	0.856	25	0	0.816	25	3
cspan	0.915	54.3	8.5	0.994	40	0	0.943	3	20

Fairly constant performance across video genres

- drei verschiedene Algorithmen untersucht
- Tree-Classifier erreicht fast gleichmässige Erkennungsrate
- kein Verfahren ist für alle Fälle optimal

ViBE: "pseudo semantic labeling"

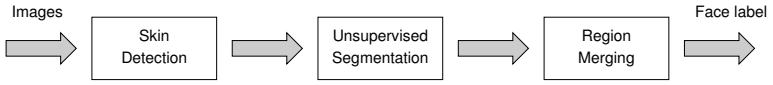
- automatische Klassifikation / Annotation von Szenen
- aufgrund von "mid-" und "low-level" Merkmalen
- insbesondere ohne Bild-"verstehen"

- möglichst gute Korrelation mit "high-level" Beschreibung (Semantik)
- möglichst einfache Berechnung - z.B. ohne Dekompression der Videos

ausgewählte Beispiel-Merkmale:

- "Head-Shoulders" (Sprecher in der Szene - oder nicht?)
- Innen- / Außenszene
- Actionszene (viel Bewegung)
- künstliche / natürliche Umgebung

ViBE: "head shoulders label"

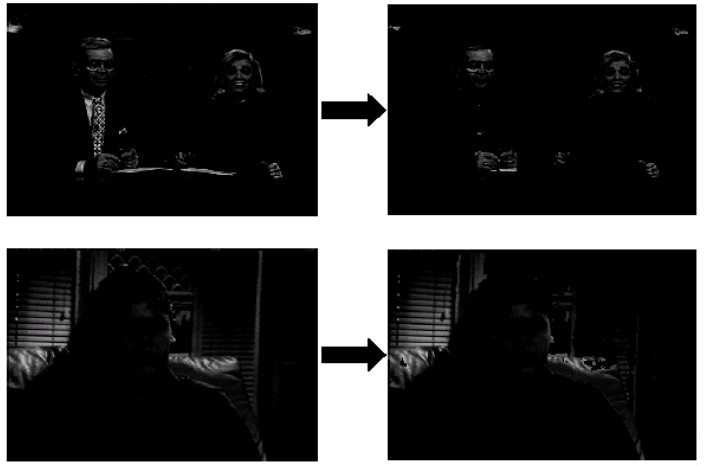


gibt es eine sprechende Person in der Szene ?

- Suche nach "Haut" in den einzelnen Videoframes . . .

- Auswertung von Helligkeit und Chrominanz
- liefert Kandidaten für skin / no-skin Bereiche
- anschließend Segmentierug und Zusammenfassung von Bereichen
- zusätzliche Auswertung von Textur und Bewegungsinformation

ViBE: "skin detection"



- als "skin" erkannte Bereiche nach der Segmentierung

ViBE: "face recognition"

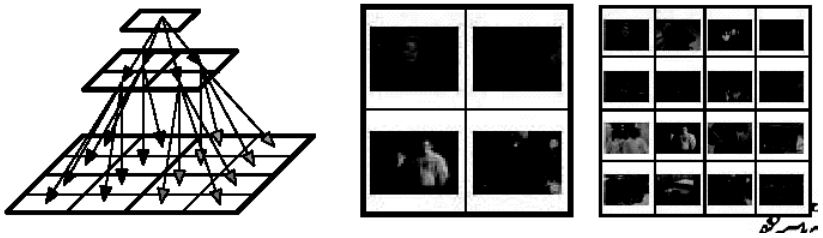
Face Recognition Results

	Shots	Faces	Detect (%)	FA(%)	Correct(%)
news1	231	76	73.7	16.1	80.5
news2	72	29	93.1	23.3	83.3
news3	78	33	87.9	15.6	85.9
news4	103	42	90.5	13.1	88.3
news5	188	51	76.5	13.9	83.5
movie	142	92	84.8	28.0	80.3
drama	100	90	94.4	20.0	93.0
total	**914**	**413**	**85.2**	**17.0**	**84.0**

- noch verbesserungsfähig . . .

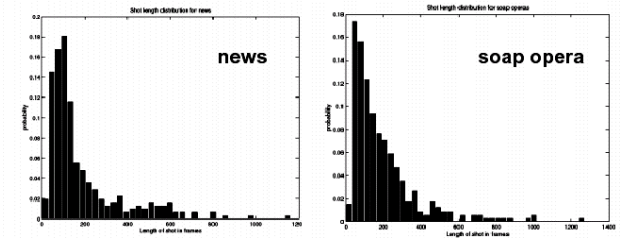
ViBE: hierarchische Organisation

- **Organize database in a pyramid structure**
  - **Top level of pyramid represents global variations**
  - **Bottom level of pyramid represents individual images**
- **Spatial arrangement makes similar images neighbors**
- **Embedded hierarchical tree structure**



ViBE: "shot length distribution"

- **Shot boundaries are "man-made" according to editing rules**
- **Shot length is an indication of editing pattern**
- **Shot length distributions for different genres are different**



- Erkennung des Genres aus dem Histogramm der Szenenwechsel

ViBE: Browser und Navigation