Video Retrieval und Video Summarization

Maria Wagner
wagnerma@ifi.lmu.de

Universität München
Amalienstrasse 17, 80333 München, Germany

Zusammenfassung Diese Arbeit behandelt die Problematik des Video Retrieval, das heißt der inhaltsbasierten Suche nach Videos in Multimedia Datenbanken, und in diesem Zusammenhang auch die Video Summarization, der Zusammenfassung von Videos. Es werden alle Schritte betrachtet, die nötig sind, um Video Retrieval durchführen zu können. Anfangs wird die Analyse der Videodaten betrachtet, die stattfindet, wenn Videos das erste Mal in die Datenbank eingelesen werden. Diese beinhaltet die Analyse der Struktur des Videos, insbesondere die Szenenerkennung, sowie die Gewinnung von Low Level als auch semantischen Metadaten. Wenn das Video zusammen mit den Metadaten abgespeichert ist, kann eine Suchanfrage durchgeführt werden. Dabei werden verschiedene Möglichkeiten der Anfragestellung sowie Fusionsmöglichkeiten für Anfragen mit mehreren Beispielen dargestellt. Zur Darstellung der Ergebnisse einer Anfrage müssen die Videos in einer zusammengefassten Form vorliegen. Damit ist auch die Video Summarization ein wichtiges Thema im Zusammenhang mit dem Video Retrieval. Zur Illustration werden aktuelle Beispiele aus der Forschung dienen, wie das IBM-Projekt Marvel.

1 Einleitung

Digitales Video wird in vielen Bereichen immer wichtiger. Nicht nur große Medienagenturen oder Fernsehsender benötigen Video Retrieval Techniken, um ihre Daten zu verwalten. Sogar Endanwender sind davon betroffen. Sei es, um sich in einem Video on Demand Angebot zurechtzufinden oder eine Sendung auf dem digitalen Videorecorder zu finden. Durch die stetig steigende Anzahl an Videodaten wird es jedoch immer schwieriger, die großen Mengen an Daten zu verwalten. Deshalb werden leistungsfähige Video Retrieval Systeme benötigt, welche dem Nutzer erlauben, Videodaten effizient zu verwalten und wieder zu finden. Video Retrieval ist ein breites Forschungsgebiet, das sich aus vielen verschiedenen Teilbereichen zusammensetzt. Abbildung 1 zeigt verschiedene Stufen, die nötig sind, um Video Retrieval durchzuführen und die in dieser Arbeit behandelt werden.

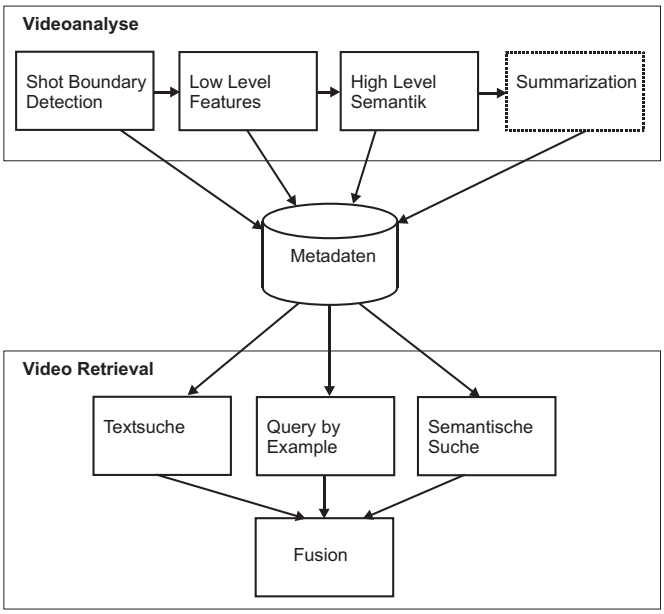

Abbildung 1. Betrachtete Teilaspekte des Video Retrieval

Wie zu erkennen ist, spielen die Metadaten dabei eine zentrale Rolle. Als Metadaten sind dabei alle Daten zu verstehen, die dazu dienen, das Video zu beschreiben. Dazu gehören die Strukturdaten, die aus der Shot Boundary Detection hervorgehen, die Low Level Features, die Semantik und auch die Summaries. Zur Beschreibung der Metadaten wird derzeit von den meisten Entwicklern der MPEG-7 Standard verwendet. Auf allen Stufen gibt es ganz unterschiedliche Ansatzpunkte, sie zu realisieren. Oftmals ist es so, dass sich ein Verfahren nur für eine bestimmte Kategorie von Videos eignet, zum Beispiel Sport oder Nachrichtensendungen. Daher sind auch viele Retrievalanwendungen auf eine bestimmte Kategorie spezialisiert. So ist zum Beispiel Informedia [17], [6] auf Nachrichtensendungen spezialisiert. Diese Arbeit soll einen möglichst umfassenden Blick auf die Thematik des Video Retrieval geben. Dabei sollen in den einzelnen Teilgebieten jeweils verschiedenen Lösungsmöglichkeiten vorgestellt werden.

2 Video Analyse

Damit ein Video in einer Datenbank gefunden werden kann, muss es erst auf seinen Inhalt hin analysiert werden. Dazu muss zuerst seine Struktur ausgelesen werden, um anschließend die relevanten Metadaten extrahieren zu können.

2.1 Strukturanalyse

Video ist ein strukturiertes Medium, worin Aktionen und Ereignisse in der Zeit eine Geschichte bilden. Ein Video muss daher als Dokument angesehen werden und nicht als eine unstrukturierte Sequenz von Frames. Bevor die Videodaten

in das VDBMS [1] integriert werden, muss es seiner Charakteristik entsprechend strukturiert werden. Videos können aus verschiedenen Streams zusammengesetzt sein, besitzen eine oder mehrere Audiospuren und gegebenenfalls Untertitel und sonstige Texteinblendungen. All diese Informationen sind bei der Strukturanalyse von Bedeutung. Dabei ist zu beachten, dass auch das Video selbst in Szenen und Kameraeinstellungen, so genannte Shots. Shots können dabei als syntaktische Einheiten, Szenen als semantische Einheiten angesehen werden. Diese Struktur ist in Abbildung 2 zu sehen. Im Rahmen dieser Arbeit sollen jedoch nur die Shots betrachtet werden.

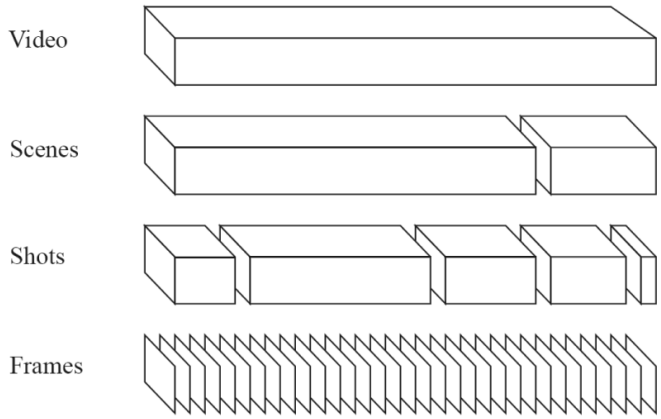

Abbildung 2. Struktur eines Videos [11]

Die Erkennung von Shots ist die Grundlage jeder Retrieval-Anwendung und Aufgabe der Shot Boundary Detection. Für die Shot Boundary Detection gibt es eine Vielzahl unterschiedlicher Algorithmen. Die prinzipielle Vorgehensweise bei Shot Boundary Detection ist nach [13] wie folgt:

1. Extrahiere ein oder mehrere Features aus jedem n-ten Frame des Videos
2. Berechne die Differenzen für aufeinander folgende Frames
3. Vergleiche die Differenzen mit einem vorgegebenen Grenzwert
4. Wird der Grenzwert überschritten, so wurde ein Schnitt festgestellt

Es gibt allerdings verschiedene Arten von Überblendungen. In [10] wird unterschieden zwischen *Hard Cuts*, *Fades* [2] und *Dissolves* [3]. Dementsprechend gibt es eine Vielzahl unterschiedlicher Herangehensweisen und Algorithmen für die Shot Boundary Detection, von denen einige im Folgenden beschrieben werden sollen. Tabelle 1 zeigt, welche Übergänge von welchem Algorithmus erkannt werden. Dabei sind die Hard Cuts leichter zu erkennen als die weichen Übergänge. Vor allem Dissolves sind schwer zu erkennen.

[1] VDBMS: Video Database Management System
[2] Fade: Überblendung vom bzw. ins Schwarze
[3] Dissolve: Nachfolgende Sequenz wird überlagernd eingeblendet

Feature/ Übergangstyp	**Hard Cuts**	**Fades**	**Dissolves**
Histogramm Differenzen	X		
Edge Change Ratio	X	X	X
Standardabweichung von Pixelintensitäten		X	
Kantenbasierter Kontrast			X

Tabelle 1. Shot Boundary Detection Verfahren [10]

Die Idee der Histogrammbasierten Algorithmen ist, dass sich die Farben bei Übergängen sehr schnell ändern, innerhalb eines Shots jedoch nur langsam. Mit dieser Methode können jedoch nur Hard Cuts erkannt werden. In [13] wird ein histogrammbasiertes Verfahren beschrieben, welches nur die Graustufen betrachtet. Die Differenz zwischen zwei aufeinander folgenden Frames wird wie folgt berechnet:

$$H_{G_{diff}}(n, n-1) = \sum_{i=0}^{255} \frac{(H_G(n)(i) - H_G(n-1)(i))^2}{Max(H_G(n)(i), H_G(n-1)(i))}$$

Dabei ist $H_G(n)(i)$ die Häufigkeit von Grauwert i Frame n.

Die Edge Change Ratio versucht, mit Hilfe von Änderungen in der Anzahl der Kantenpixel zwischen aufeinander folgenden Frames, Übergänge zu finden. Die Kanten müssen dabei erst durch einen Kantenerkennungsalgorithmus bestimmt werden. Die Edge Change Ratio ist wie folgt definiert: σ_n sei die Anzahl der Kantenpixel im Frame n, X_n^{in} und X_{n-1}^{out} die Anzahl der neuen Kantenpixel im Frame n, beziehungsweise der verschwindenden Kantenpixel im Frame n-1. Die Edge Change Ratio zwischen den Frames n-1 und n berechnet sich durch

$$ECR_n = max(X_n^{in}/\sigma_n, X_{n-1}^{out}/\sigma_n)$$

Wenn die ECR Werte zeitlich aufgetragen werden, lassen sich die Übergänge durch bestimmte Muster im ECR-Verlauf erkennen, wie sie in Abbildung 3 zu sehen sind. Theoretisch soll die ECR alle möglichen Übergänge erkennen können, jedoch hat sich bei praktischen Tests gezeigt, dass die Fehlerrate bei Fades und Dissolves viel zu hoch sind.

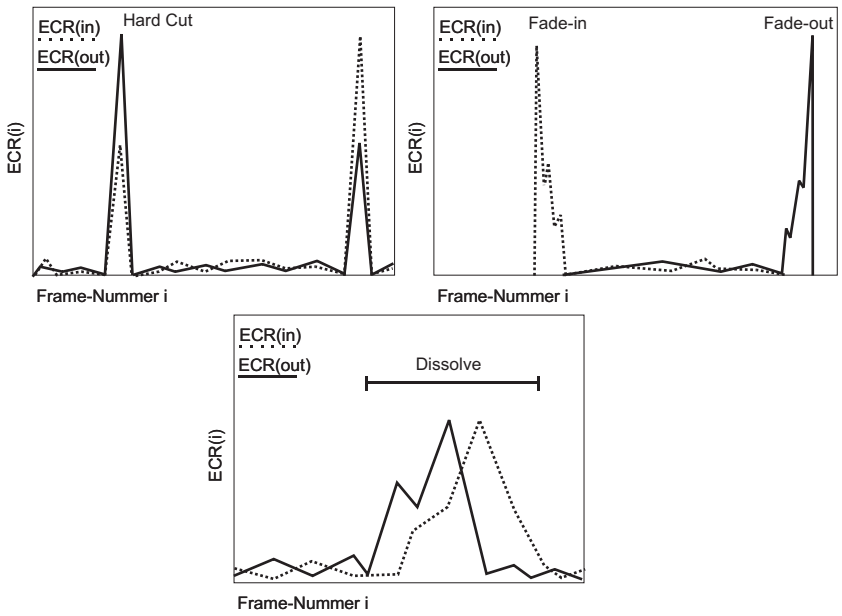

Abbildung 3. Muster bei der Edge Change Detection

Die Standardabweichung der Pixelintensitäten nutzt Kenntnisse der Videoproduktion, um Fades zu erkennen. Denn Fades werden normalerweise durch eine lineare Skalierung der Pixelintensitäten erzeugt. Diese Skalierung wird sichtbar, wenn man den zeitlichen Verlauf der Standardabweichung der Pixelintensitäten betrachtet. Ein solches Muster Zeigt Abbildung 4.

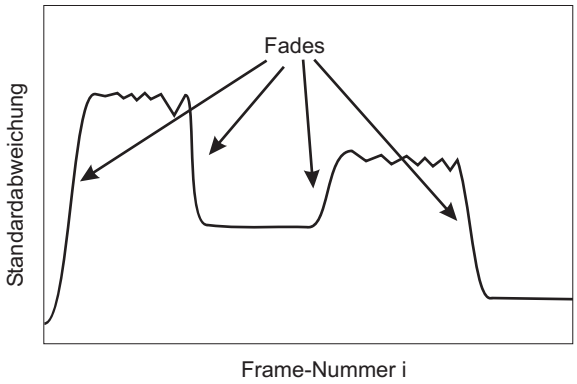

Abbildung 4. Standardabweichung der Pixelintensitäten

Dissolves sind eine Kombination aus einem Fade-out des vorangehenden Shot und einem Fade-in des neuen Shot. Während einem Dissolve nimmt sowohl Kontrast als auch Schärfe der Bilder ab. Die Idee der Kanten basierten Kontrast-Methode ist, die Verluste von Kontrast und Schärfe festzustellen und damit Dissolves zu erkennen. Problematisch hierbei ist jedoch, dass bei sehr schnellen

Bewegungen im Video die Kanten verwischt werden und dadurch ein falscher Dissolve erkannt werden kann.

Bei der Erstellung eines Videos im Editor ist klar, an welche Stellen Übergänge und Schnitte sind. Wenn die Shots bereits bei dieser Erstellung als Metadaten, zum Beispiel im MPEG-7-Format, gespeichert werden, kann die aufwändige Shot Boundary Detection entfallen. Nach der Shot Boundary Detection können mehrere Shots noch zu größeren Einheiten - Szenen oder Stories - zusammengesetzt werden.

2.2 Metadaten

Aus den Ergebnissen dieser Analyse können nun Metadaten für die Beschreibung des Videos abgeleitet werden. Da Metadaten sehr viel kleiner sind als die Originaldaten, ist es wesentlich effizienter, auf den Metadaten zu suchen. Gerade bei den riesigen Datenmengen, die ein Video beinhaltet sind Metadaten unerlässlich. Daher benötigen Video Retrieval Systeme Metadaten, um den Inhalt von Videodateien zu beschreiben. Die Metadaten werden meist auf einem separaten Metadatenserver abgespeichert. MPEG-7 wird dabei ein immer beliebteres Format für die Beschreibung von Metadaten. Wie bereits erwähnt, besteht ein Video aus einem Video-Signal, einer oder mehreren Tonspuren und textuellen Informationen. Dazu gehören Untertitel, Schlagzeilen in Nachrichten oder die Benennung von interviewten Personen. Bei der Betrachtung der Audioinformationen ist insbesondere die gesprochene Information wichtig, da sich hieraus viele wichtige Informationen ableiten lassen. Video beinhaltet also Bilder, Audio, Text und darüber hinaus Bewegung und Aktivität. Damit ist auch eine Vielzahl an Metadaten nötig, um ein Video zu beschreiben. Die Daten, mit denen man ein Video beschreiben kann, werden eingeteilt in Low Level Features und High Level Features, das heißt die Semantik des Videos.

Low-Level-Features Die meisten Eigenschaften, die aus einem Video gewonnen werden können sind Low Level Features. Sie werden direkt aus der digitalen Repräsentation des Videos gewonnen und haben meist wenig zu tun mit den semantischen Konzepten, wie ein Mensch das Video beschreiben würde. Dennoch sind sie von großer Bedeutung, da sich mit ihrer Hilfe semantische Konzepte ableiten lassen. Um die Gewinnung von Features zu verfeinern wird oft zusätzlich eine *Region Extraction* durchgeführt. Dabei werden Vordergrundbereiche von Hintergrundbereichen unterschieden. [17] Beispiele für Low Level Features, die von Videoanalysesystemen benutzt werden sind:

- Farben
- Texturen
- Formen
- Bewegungen von Objekten und der Kamera
- Gesichter
- Audiodeskriptoren, insbesondere für Spracherkennung

Für die automatische Gewinnung dieser Low Level Features gibt es bereits viele bewährte Methoden. Als Beispiele sollen im Folgenden die Gewinnung von Textinformationen mit Hilfe von Video OCR und Spracherkennung beschrieben werden.

Spracherkennung Nach [6] ist Spracherkennung eine große Unterstützung, um Informationen in einem Videoarchiv zu finden. Sie dient dazu, Sprache im Video in Text umzuwandeln. Da die Suche auf Texten schon weiter fortgeschritten ist als Video-Retrieval, kann dies die Effizienz immens steigern. Um dies durchzuführen muss zunächst die Audiospur vom Video getrennt werden [7]. Die Qualität von Spracherkennung hängt von der *word error rate* ab. Diese wiederum ist abhängig von der Art von Video, die untersucht wird. Bei einem einzelnen Studio-Sprecher liegt sie bei etwa 15 %. Sie steigt jedoch bei Außenaufnahmen. Bei Werbeeinblendungen mit viel Hintergrundgeräuschen und Musik liegt sie bereits bei 85 % [17]. Die besten Systeme sind hier in der Lage, die Werbung zu erkennen und nicht für die Spracherkennung zu verwenden.

Video OCR Eine weitere Methode, um Textinformationen aus einem Video zu extrahieren, ist die *Optical Character Recognition* (OCR). Diese Methode ist natürlich nur sinnvoll für Videos, in denen viele Texte zu sehen sind, wie zum Beispiel Nachrichtensendungen. Die Informationen in diesen Texten sind oftmals nicht in der Audiospur enthalten. Dazu zählen Namen von Personen und Orten oder Überschriften. Zunächst muss erkannt werden, wo im Video Texte zu sehen sind. Diese Regionen werden dann extrahiert und in eine schwarz-weiß Darstellung umgewandelt. Hierauf kann dann ein OCR-Verfahren angewandt werden. Die Texte werden in der Datenbank zusammen mit einem Zeitstempel abgelegt. Dies dient dazu um den Text mit dem Video zu synchronisieren. [17]

High-Level-Semantik Low Level Features sind für das Video Retrieval nicht ausreichend, da die Ähnlichkeit zwischen Videos oft auf einer höheren, semantischen Ebene liegt. Daher gibt es eine Lücke zwischen den Low-Level Feature Beschreibungen und der Beschreibung von semantischen Objekten, wie Ereignissen, Personen und Konzepten, die *Semantic Gap* [8]. Das Ziel ist es, dass das System die Ähnlichkeit zwischen Videos in der Art und Weise misst, wie es ein Mensch wahrnehmen würde. Ansätze zur Beschreibung und Gewinnung semantischer Informationen sind Ontologien und Maschinelles Lernen. Zur Modellierung der semantischen Informationen wird die Verwendung von Ontologien vorgeschlagen. Sie erlauben eine strukturierte semantische Annotation, indem Objekte und ihre Relationen untereinander beschrieben werden, zum Beispiel als gerichteter Graph [3]. Zur Beschreibung von Ontologien wird meist RDF (recource description framework) oder OWL (ontology web language), das auf RDF aufsetzt, verwendet. Aber auch MPEG-7 ermöglicht die Beschreibung von semantischen Einheiten und erlaubt die Definition von Relationen. Da mögliche semantischen Elemente in einem Video stark von der Art des Videos abhängen, schlägt [15] die Verwendung von Domänenontologien zur Beschreibung der Semantik eines Videos vor. Hierbei hängt die Auswahl an visuellen Features, die

betrachtet werden, von der Domäne ab, die betrachtet werden soll. In [15] wurde Billard als Wissensdomäne verwendet, in [3] Fußball. Voraussetzung für das Extrahieren semantischer Daten ist die Erkennung von Objekten und Bewegung dieser Objekte im Video, um sie mit den semantischen Konzepten der Ontologie assoziieren zu können. Für die Gewinnung der semantischen Daten gibt es verschiedene Möglichkeiten. Nach [3] ist die manuelle Annotation nach wie vor der verlässlichste Weg, um hochwertige semantische Deskriptoren zu erstellen. Der große Nachteil ist allerdings der hohe Zeitaufwand. Er ist etwa 10-mal so hoch wie die Dauer des Videos. Das heißt, um ein Video von einer Stunde zu annotieren, benötigt man ungefähr 10 Stunden [8]. Deshalb müssen die Annotationstools von hoher Qualität sein. Die derzeitige Forschung beschäftigt sich damit, semantische Konzepte automatisch zu erkennen und damit den Aufwand für die manuelle Annotation zu reduzieren. Dabei wir vor allem auf Methoden des Maschinellen Lernens gesetzt. Dazu zählen *Hidden Markov Modelle* und *Support Vector Machines*. [2] schlägt eine Kombination aus beiden Möglichkeiten vor. Hidden Markov Modelle dienen der Mustererkennung. Sie sind effektive Tools, um zeitliche Muster von Ereignissen zu beschreiben und werden schon länger für die Spracherkennung eingesetzt. Support Vector Machines dienen der binären Klassifizierung eines Problems und sind ebenfalls Instrumente des Maschinellen Lernens. Als Beispiel für ein System, welches Maschinelles Lernen einsetzt, soll hier das *Multimedia Analysis and Retrieval System* - MARVEL [8] von IBM dienen. Laut IBM soll es hiermit möglich sein, den Aufwand manueller Annotation auf 1 - 5 % zu senken. Das System besteht aus einer Multimedia-Analyse-Komponente und einer Multimedia Suchmaschine. Das Analyse-Tool benutzt Maschinelles Lernen, um semantische Konzepte von automatisch extrahiertem Audio, Sprache und visuellen Inhalten zu modellieren. Damit kann es neuen Videos automatisch Labels zuweisen, was den Aufwand für manuelle Annotation reduziert. Wie in Abbildung 5 zu erkennen ist, wird menschliche Interaktion nur für den Trainingsprozess benötigt. Wenn die Modelle dann erstellt und überprüft sind, können sie auf ungekennzeichnete Videos angewandt werden. Bei der Analyse werden zuerst textuelle Metadaten aus Spracherkennung und Video-OCR sowie Visuelle Features, wie Farben, Texturen und Formen extrahiert. Mit diesen Daten werden dann mit Hilfe der Modelle, die durch das Training entstanden sind, die semantischen Metadaten erstellt (vgl. Abbildung 6). Im Zuge der Videoanalyse wird meist auch die Summarization des Videos durchgeführt. Da die Summaries jedoch erst beim Retrieval und Browsen relevant werden, sind sie erst in Abschnitt 4 beschreiben.

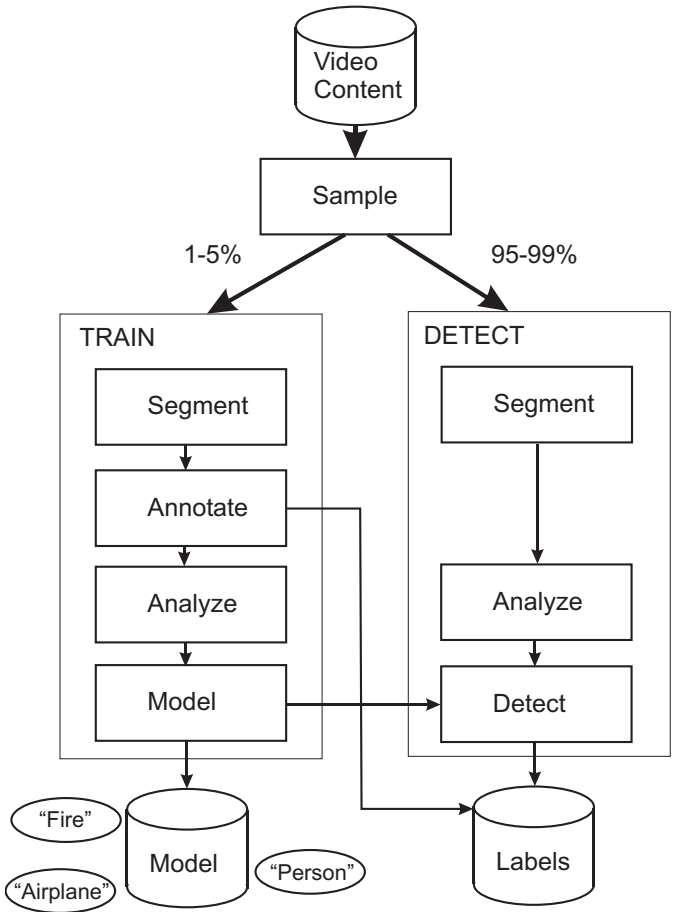

Abbildung 5. MARVEL-Analysesystem [8]

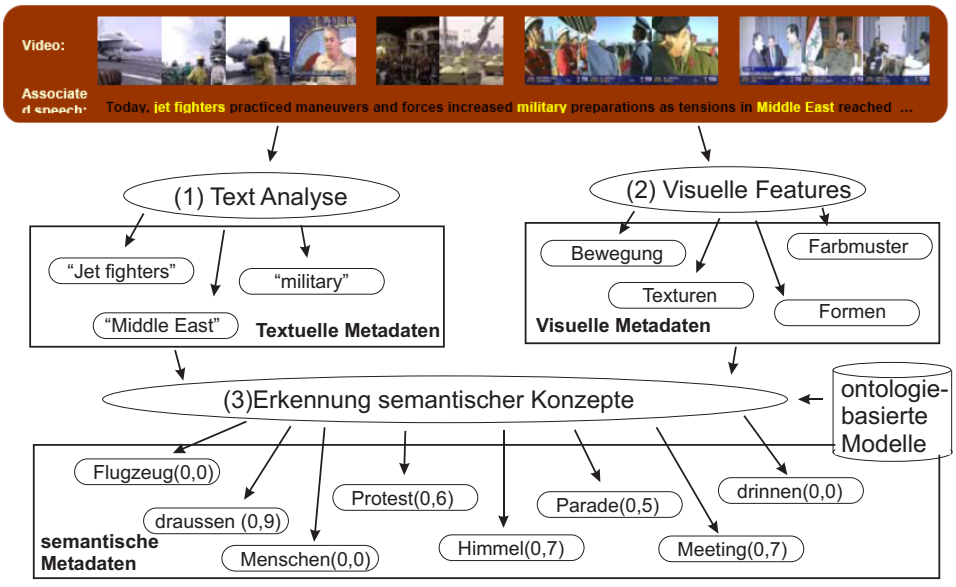

Abbildung 6. Extraktion semantischer Konzepte bei Marvel [14]

3 Anfrage und Suche

Wenn Video- und Metadaten in der Datenbank eingetragen sind, kann das eigentliche Retrieval, also das Stellen einer Suchanfrage an die Multimediadatenbank, durchgeführt werden. Dabei dienen die Beschreibungen der Shot Boundaries als Basiseinheiten für das Retrieval. Einfache Textanfragen reichen für das Suchen von Videodaten nicht aus. Da man im Video Bild-, Audio- und Textinformationen vorfindet, werden für das Video-Retrieval auch Techniken des Bild-, Audio- und Text-Retrieval benötigt. Video-Retrieval schließt also die Problematik aller Medienarten mit ein. Darüber hinaus können Suchanfragen als Kombination verschiedener Medienbeispiele gestellt werden. Was außerdem in keinem anderen Medium vorkommt sind einerseits Objekte, die sich über die Zeit mit einer Richtung und einer Geschwindigkeit über die Bildfläche bewegen können, andererseits die Bewegung der Kamera selbst. Darüber hinaus stellt sich auch hier das Problem der Suche nach semantischen Features. Eine Kombination aus verschiedenen Suchmethoden ist die beste Art für Video Retrieval. Hier stellt sich jedoch die Frage, wie diese Suchmethoden kombiniert werden können. Darauf soll in Kapitel 3.4 näher eingegangen werden. Zunächst sollen Möglichkeiten für das Stellen einer Anfrage an ein VDBMS vorgestellt werden.

3.1 Textuelle Daten

Retrievalsysteme, die Spracherkennung oder Video OCR bei der Videoanalyse einsetzen, wie zum Beispiel Marvel, geben dem Benutzer meist auch die Möglichkeit, auf diesen Daten eine Textbasierte Suche durchzuführen. Für die Suche nach textuellen Daten gibt es bereits sehr effiziente Systeme. Sie erwarten einen String als Anfrage, der in eine Anfragesequenz umgewandelt wird. Eine Möglichkeit für das Retrieval von textuellen Daten ist zum Beispiel das Okapi System. Dabei wird für jedes Dokument ein Relevanzwert berechnet. Jedes Dokument wird mit der Anfrage verglichen, deren Relevanzwert mit der Okapi-Formel berechnet wird [7]:

$$Sim(Q,D) = \sum_{qw \in Q} \left\{ \frac{tf(qw,D) \log\left(\frac{N - df(qw) + 0.5}{df(qw) + 0.5}\right)}{0.5 + 1.5 \frac{|D|}{avg_{dl}} + tf(qw,D)} \right\}$$

Dabei ist $tf(qw, D)$ die Termhäufigkeit von Wort qw im Dokument D, $df(qw)$Die Dokumenthäufigkeit [4] von Wort qw und avg_{dl} die durchschnittliche Dokumentlänge aller Dokument einer Sammlung.

3.2 Query by Example

Query by Example ist eine Suchmethode, die in den meisten Videodatenbanken Anwendung findet. Sie wird auch als contentbasiertes Retrieval [1] oder als featurebasierte Suche [8] bezeichnet. Das Prinzip von Query by Example ist eine

[4] Anzahl der Dokumente, in denen ein Term auftritt. Geht man von einer zufälligen Verteilung eines Worts in einem Korpus von Dokumenten aus, so werden durch den Übergang von der Häufigkeit zur Dokumenthäufigkeit die Häufigkeitsunterschiede besonders für häufige Terme verringert: Bei der Bestimmung der Dokumenthäufigkeit spielt es keine Rolle, ob ein Term oft in einem Dokument vorkommt oder nur einmal. vgl. [5]

Ähnlichkeitsmessung (Matching) zwischen einem Beispiel und den Inhalten der Datenbank mit Hilfe der extrahierten Deskriptoren. Als Beispiel-Content können Bilder, aber auch Videosegmente, dienen. Dabei kann der Benutzer entweder einen Shot oder einen Frame wählen, der schon in der Datenbank vorhanden ist, oder ein neues Video beziehungsweise Bild. Aus dem neuen Content müssen jedoch erst die Metadaten generiert werden, um es als Anfragebeispiel verwenden zu können. Sowohl [9] als auch [8] verwenden MPEG-7 Deskriptoren für das Matching. Abbildung 7 illustriert die Vorgehensweise bei Query by Example. Zunächst wird das Matching zwischen den Deskriptoren des Beispiel-Content

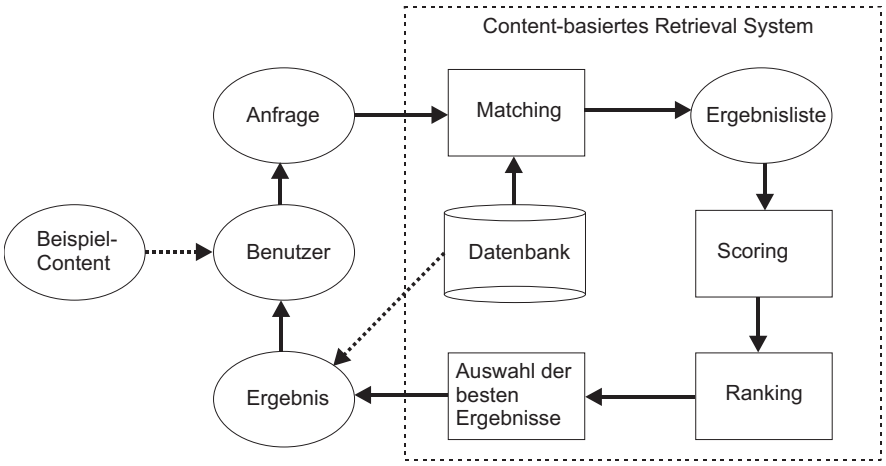

Abbildung 7. Retrieval Prozess [1]

und der Datenbankinhalte durchgeführt. Anschließend werden die Ergebnisse anhand ihrer Relevanz bewertet (Scoring) und geordnet (Ranking). Die besten Ergebnisse werden dem Benutzer, meist als Summary, präsentiert. In [1] wird die Ähnlichkeit durch die Feature-Ähnlichkeit bestimmt. Eine Möglichkeit ist das vektorbasierte Matching. Dabei kann die Ähnlichkeit durch die Distanz der Featurevektoren bestimmt werden, da ähnliche Vektoren im Vektorraum, der durch die Featurevektoren aufgespannt wird, nahe beieinander liegen. Folgende Formel berechnet die Distanz zweier Vektoren:

$$d_{q,t}^{r} = \sum_{m=0}^{M-1} |v_q[m] - v_t[m]|^r$$

Dabei sind v_q der Query-Vektor und v_t der Target-Vektor. Zur Messung dienen Minkowski Metriken[5], mit Werten r = 1 (Manhattandistanz) und r=2 (Euklidische Distanz) [1].

[5] Minkowski-Metrik: aus einer p-Norm abgeleitete Metrik. Eine p-Norm ist definiert durch

$$\|x\|_p := (\sum_{i=1}^{n} |x_i|^p)^{1/p}$$

mit einer reellen Zahl $p \geq 1$ und x_i der i-ten Komponente des Vektors x.

3.3 Semantische Suche

Sowohl [8] als auch [3] bieten dem Nutzer Möglichkeiten für eine semantische Suche. MARVEL lässt den Benutzer aus einem Lexikon von Schlüsselwörtern auswählen, die Teil des MPEG-7 Klassifikationsschemas sind. Dagegen stellt das in [3] vorgestellte System dem Benutzer ein grafisches Interface zur Verfügung, das es erlaubt, semantische Graphen zu darzustellen und zu editieren.

3.4 Fusionsmethoden

Die effizienteste Form des Video-Retrieval ist eine Kombination aus mehreren anderen Retrieval-Formen, wie sie oben beschreiben sind. Beispielsweise bietet das in [8] beschriebene System die Möglichkeit einer integrierte Suche nach Featuredeskriptoren, semantischen Konzepten und Text. In [12] wurden folgende Kombinationsmöglichkeiten untersucht:

- Kombination von unterschiedlichen Features eines Beispiels
- Kombination verschiedener visueller Beispiele, das heißt Videos und Bildern
- Kombination verschiedener visueller Beispiele und Text.

Es wurde unterschieden zwischen *Early Fusion*, wobei die Features vor dem Matching kombiniert werden und *Late Fusion*, wobei zuerst alle Features einzeln verglichen werden und anschließend die Ergebnisse zusammengeführt werden. Fusionsmethoden werden weiterhin in Rank-Basierte und Score-Basierte Methoden unterteilt. Rank-Basierte Methoden kombinieren verschiedene Suchresultate durch das Summieren der Rangpositionen der Dokumente auf den unterschiedlichen Ergebnislisten. Hier ist es auch möglich, die einzelnen Ergebnislisten unterschiedlich zu gewichten. Zu den Score-Basierten Methoden gehören CombSUM und CombMNZ Bei CombSUM werden die Scores der verschiedenen Retrievalmethoden für jedes Dokument aufsummiert. Bei CombMNZ werden zunächst die Scores von gekürzten Ergebnislisten aufsummiert, zum Beispiel den Top 1000 Ergebnissen. Anschließend wird der Durchschnittswert durch die Anzahl der Retrievalmethoden geteilt, die ihn zum Ergebnis hatten. Da die unterschiedlichen Retrievalmethoden, zum Beispiel Textsuche und Bildsuche, heterogene Ergebnisse liefern, müssen die Werte erst auf Werte zwischen 0 und 1 normalisiert werden. Ranks beziehungsweise Scores können folgendermaßen normalisiert werden:

$$norm\_rank_{shot} = \frac{N + 1 - rank_{shot}}{N}$$

Dabei ist N die Anzahl der Shots in der Ergebnisliste

$$norm\_score_{shot} = \frac{score_{shot} - score_{min}}{score_{max} - score_{min}}$$

Wobei $score_{min}$ und $score_{max}$ die geringste beziehungsweise höchste Bewertung in der Ergebnisliste sind. Abbildung 8 zeigt das Schema für Video Retrieval mittels Score-basierter Fusion.

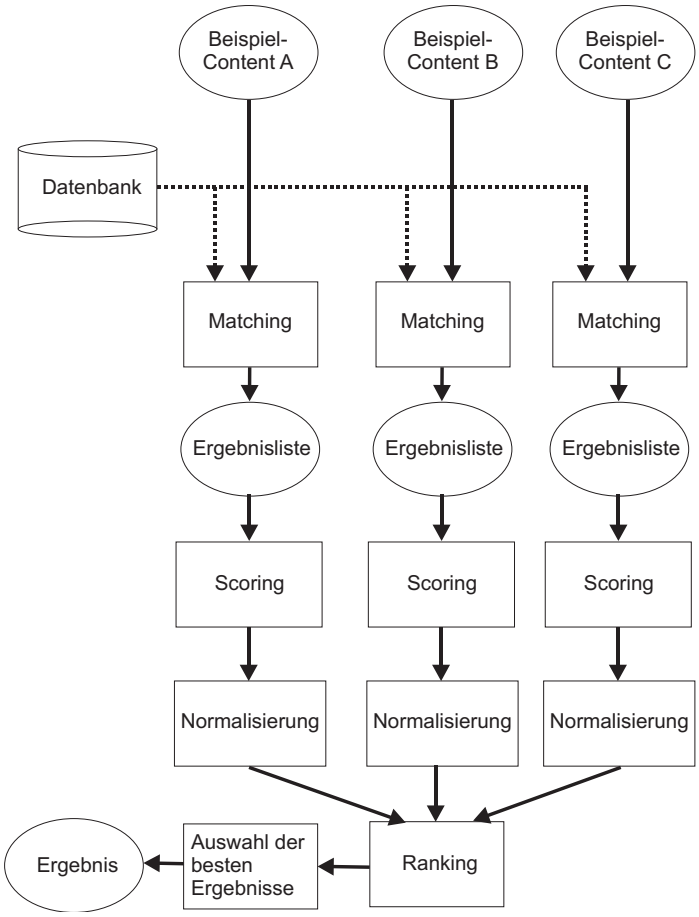

Abbildung 8. Score-basierte Fusion

4 Ergebnispräsentation: Video Summarization

Sind zur Suchanfrage passende Ergebnisse gefunden worden, besteht das Problem, wie diese dem Nutzer präsentiert werden sollen. Da Video ein Medium ist, das einem zeitlichen Ablauf unterliegt, ist es schwierig, seinen Inhalt so darzustellen, dass er auf einen Blick erfasst werden kann. Um dies zu erreichen, werden Zusammenfassungen benötigt, welche die wichtigsten beziehungsweise charakteristischsten Szenen zeigen. Die Möglichkeiten für eine derartigen Video Summarization sollen im folgenden Abschnitt behandelt werden.

4.1 Erzeugen von Video Summaries

Video Summaries werden meist im Zuge der Videoanalyse erzeugt und ebenfalls als Metadaten abgespeichert. Hier werden nun verschiedene Möglichkeiten für die Erzeugung von Videosummaries beschrieben.

Keyframes Soll ein Videosummary aus Keyframes erzeugt werden, müssen zunächst die relevanten Keyframes gefunden werden. Die einfachste Lösung ist es, den ersten Frame jedes Shots als Keyframe zu verwenden. Diese Methode funktioniert recht gut bei Shots mit wenig Bewegung. Bei Videos mit viel Bewegung und damit vielen Änderungen ist der erste Frame jedoch meist wenig repräsentativ. In [4] wird vorgeschlagen, die MPEG-7 Motion und Audio Deskriptoren zu verwenden, um den optimalen Keyframe eines Shots zu erhalten. Dabei werden Motion-Vektoren und damit die Intensität der Bewegungsaktivität benutzt, um die Änderungen zwischen aufeinander folgenden Frames zu messen. Damit wird angenommen, dass die kumulative Bewegungsintensität einen Anhaltspunkt für die kumulative Änderung im Content bietet. Abbildung 9 zeigt eine Strategie, einen einzelnen Keyframe mit Hilfe der Bewegungsintensität zu finden. Dabei wird angenommen, dass die beste Wahl für den ersten Keyframe derjenige Frame ist, bei dem die kumulative Bewegungsintensität halb so hoch ist wie ihr Maximalwert.

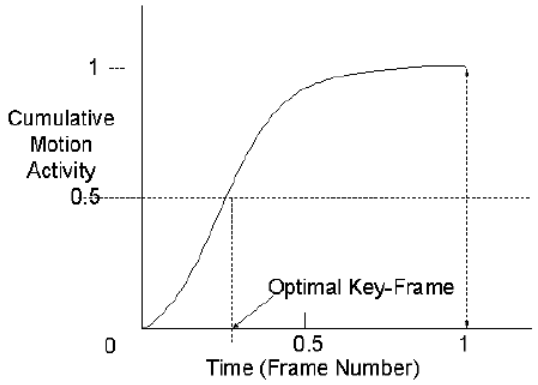

Abbildung 9. Auswahl eines Keyframes mit Hilfe der Bewegungsintensität [4]

Keyframes und Text

Video Eine andere Möglichkeit für die Zusammenfassung eines Videos ist, eine Kurzform des Videos zu erzeugen. Hier wäre die einfachste Möglichkeit, das Video selbst schneller abzuspielen. Der Nachteil daran ist jedoch, dass alle Teile des Videos gleich schnell beschleunigt werden. Dabei kann es passieren, dass Sequenzen mit viel Bewegung herausgeschnitten werden. Eine Verbesserung wäre es daher, die Sequenzen mit geringer Bewegungsintensität stärker zu beschleunigen als jene mit hoher Bewegungsintensität. [4] bezeichnet dies als angepasste Abspielgeschwindigkeit basierend auf der Bewegungsintensität oder *activity normalized playback*. Die Framerate kann dabei wie folgt angepasst werden:

Playback Framerate = (Original-Framerate) * (gewünschter Grad an Bewegungsintensität / Grad der Bewegungsintensität im Original)

Da eine Anpassung der Framerate sehr rechenintensiv ist, kann das gleiche Ergebnis auch durch Subsampling erzielt werden, wobei die Framerate konstant bleibt.

4.2 Präsentation von Video Summaries

Die Präsentation eines Summaries kann einfach durch eine Auflistung der entsprechenden Keyframes erfolgen. [16] beschreibt eine Möglichkeit, wie die semantische Bedeutung von Frames bei der Darstellung berücksichtigt werden kann. Dabei wurden den Videosegmenten *importance Scores* zugewiesen. Damit können Keyframes, die wichtigere Segmente repräsentieren, größer dargestellt werden als weniger wichtige. Die Summaries sollten in einem comicartigen Stil dargestellt werden. Dazu wurde ein Frame-Packing Algorithmus implementiert, der die Keyframes in mehrere Blöcke eingepasst, von denen jeder eine Zeile im Layout darstellt. Abbildung 10 zeigt ein Ergebnis des Algorithmus.

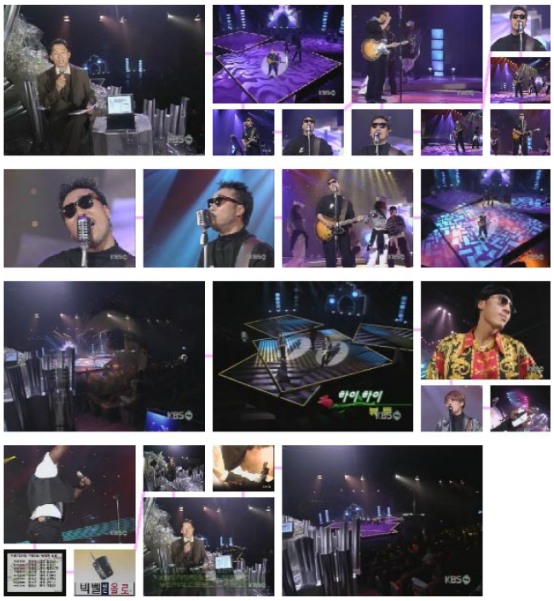

Abbildung 10. Präsentation eines Videosummaries im Comicstil [16]

5 Zusammenfassung

Video Retrieval ist ein Thema mit vielen Aspekten, denn Video ist ein strukturiertes Medium, welches nicht nur visuelle Inhalte, sondern auch Audio und Text beinhaltet. Daher muss das Video sowohl auf seine Struktur als auch auf die verschiedenen Metadaten hin untersucht werden, um es indexieren zu können. Die semantische Beschreibung ist dabei noch immer nicht vollständig gelöst und daher Gegenstand der derzeitigen Forschung. Das Ziel ist es, die semantische Lücke zwischen den automatisch zu extrahierenden Low Level Features und der menschlichen Interpretation zu schließen. Nach erfolgter Indexierung ist es möglich, das Video in der Datenbank zu suchen. Dazu wurden auch unterschiedliche Möglichkeiten beschrieben. Wenn Videos gefunden wurden, müssen sie als Summary angezeigt werden, um möglichst schnell einen Überblick über die Inhalte zu gewinnen.

Literatur

[1] Adams, B., Amir, A., Dorai, C., Ghosal, S., Iyengar, G., Jaimes, A., Lang, C., Lin, C., Natsev, A., Naphade, M., Neti, C., Nock, H.J., Permuter, H.H., Singh, R., Smith, J.R., Srinivasan, S., Treng, B.L., Ashwin, T.V., Zhang, D.: (IBM Research TREC-2002 Video Retrieval System) http://www.itl.nist.gov/iaui/894.02/projects/t2002v/results/notebook.papers/ibm.smith.pdf.

[2] Bae, T.M., Kim, C.S., Jin, S.H., Kim, K.H., Ro, Y.M.: Semantic Event Detection in Structured Video Using Hybrid HMM/SVM. In Leow, W.K., Lew, M., Chua, T.S., Ma, W.Y., Chaisorn, L., Bakker, E., eds.: Image and Video Retrieval. Springer-Verlag Berlin-Heidelberg (2005) 113–122

[3] Bailer, W., Mayer, H., Neuschmied, H., Haas, W., Lux, M., Klieber, W.: Content-based Video Retrieval and Summarization using MPEG-7. In: Proceedings of the SPIE. Volume V. (2003) 1–12 http://pamir.cs.deu.edu.tr/cse405/downloads/Semantic%20Video%20pdf/Content-based%20Video%20Retrieval%20and%20Summarization%20using%20MPEG-7_img1958.pdf.

[4] Divakaran, A., Peker, K.A., Radhakrishnan, R., Xiong, Z., Cabasson, R.: Video Summarization using MPEG-7 Motion Activity and Audio Descriptors. Technical report, MERL-A Mitsubishi Electric Research Laboratory (2003) http://www.merl.com/reports/docs/TR2003-34.pdf.

[5] Ferber, R.: Information retrieval. Web-Version (2003) http://information-retrieval.de/irb/ir.html.

[6] Hautmann, A.G.: Lessons for the Future from a Decade of Informedia Video Analysis Research. In Leow, W.K., Lew, M., Chua, T.S., Ma, W.Y., Chaisorn, L., Bakker, E., eds.: Image and Video Retrieval. Springer-Verlag Berlin-Heidelberg (2005) 1–10

[7] Hauptmann, A., Jin, R., Ng, T.D.: Video Retrieval using Speech and Image Information. In Yeung, M.M., Lienhart, R.W., Li, C.S., eds.: Storage and Retrieval for Media Databases 2003. (2003) 148–159 http://www.informedia.cs.cmu.edu/documents/ei03_haupt.pdf.

[8] IBM T. J. Watson Research Center: (MARVEL: Multimedia Analysis and Retrieval System) http://www.research.ibm.com/marvel/Marvel%20Whitepaper.pdf.

[9] Jung, B.H., Ha, M.H., Kim, K.S.: Video-based Retrieval System using MPEG-7 Metadata (2003) http://www.broadcastpapers.com/IBC2003papers/IBC03KBSVideoRet.pdf.

[10] Lienhart, R.: Comparison of Automatic Shot Boundary Detection Algorithms. In: Image and Video Processing VII 1999, Proceedings SPIE 3656-29, Bellingham, SPIE (1999) http://www.lienhart.de/spie99.pdf.

[11] Lienhart, R., Pfeiffer, S., Effelsberg, W.: Video Abstracting. Communications of the ACM **40**(12) (1997) 55–62 http://www.lienhart.de/cacm.pdf.

[12] Mc Donald, K., Smeaton, A.F.: A Comparison of Score, Rank and Propability-Based Fusion Methods for Video Shot Retrieval. In Leow, W.K., Lew, M., Chua, T.S., Ma, W.Y., Chaisorn, L., Bakker, E., eds.: Image and Video Retrieval. Springer-Verlag Berlin-Heidelberg (2005) 61–70

[13] Miene, A., Hermes, T., Ioannidis, G., Fathi, R., Herzog, O.: Automatic shot boundary detection and classification of indoor and outdoor scenes. TREC 2002 (2002) http://trec.nist.gov/pubs/trec11/papers/ubremen.miene.pdf.

[14] Smith, J.R., Naphade, N., Natsev, A., Tesic, J.: Multimedia Research Challenges for Industry. In Leow, W.K., Lew, M., Chua, T.S., Ma, W.Y., Chaisorn, L., Bakker, E., eds.: Image and Video Retrieval. Springer-Verlag Berlin-Heidelberg (2005) 28–37

[15] Song, D., Liu, H.T., Cho, M., Kim, H., Kim, P.: Domain Knowledge Ontology Building for Semantic Video Event Description. In Leow, W.K., Lew, M., Chua, T.S., Ma, W.Y., Chaisorn, L., Bakker, E., eds.: Image and Video Retrieval. Springer-Verlag Berlin-Heidelberg (2005) 267–275

[16] Uchihashi, S., Foote, J., Girgensohn, A., Boreczky, J.: Video Manga: Generating Semantically Meaningful Video Summaries. Proceedings ACM Multimedia (1999) 383–392 http://www.fxpal.com/publications/FXPAL-PR-99-136.pdf.

[17] Wactlar, H.D.: Informedia - Search and Summarization in the Video Medium. Proceedings of Imagina 2000 Conference, (2000) http://www.informedia.cs.cmu.edu/documents/imagina2000.pdf.