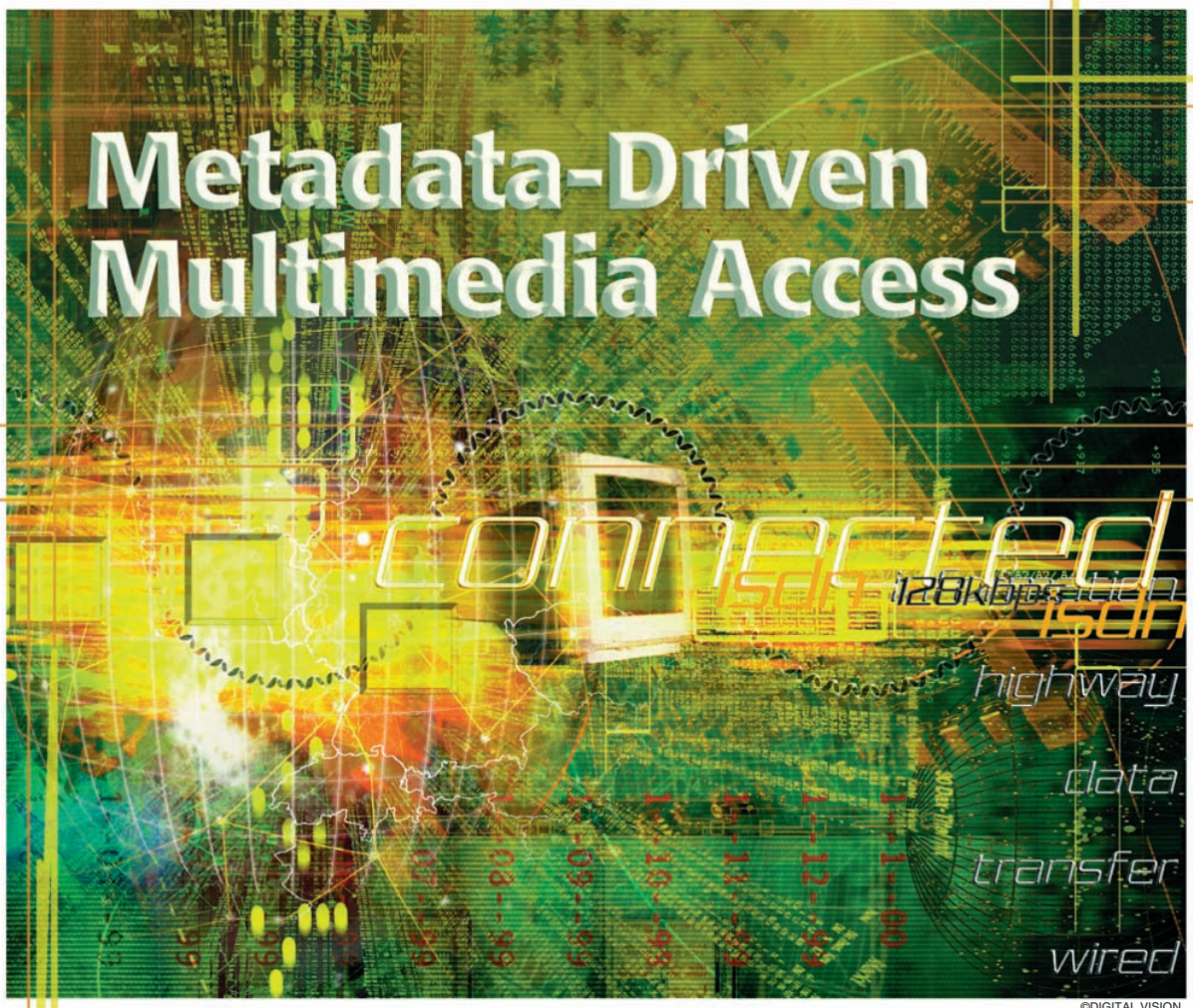
Metadata-Driven Multimedia Access



©DIGITAL VISION

Peter van Beek, John R. Smith, Touradj Ebrahimi, Teruhiko Suzuki, and Joel Askelof

With the growing ubiquity and mobility of multimedia-enabled devices, universal multimedia access (UMA) is emerging as one of the important components for the next generation of multimedia applications. The basic concept underlying UMA is universal or seamless access to multimedia content, by automatic selection and adaptation of content based on the user's environment [1]. Methods in this context may include selection among different pieces of content or among different variations of a single piece of content. Methods for adaptation include rate reduction, adaptive spatial and temporal sampling, quality reduction, summarization, personalization, and reediting of the multimedia content. The different relevant parameters in the user's environment include device capabilities, available bandwidth, user preferences, usage context, as well as spatial and temporal awareness.

UMA is partially addressed by scalable or layered encoding, progressive data representation, and object- or scene-based coding (such as in MPEG-4 [2], [3]) that inherently provide different embedded quality levels of the same content. From the network perspective, UMA involves important concepts related to the growing variety of communication channels, dynamic bandwidth variation, and perceptual quality of service (QoS). UMA also involves different preferences of a user (recipients of the content) or a content publisher in choosing the form, the quality, or the personalization of the content. UMA promises an integration of these different perspectives into a new class of content adaptive applications that could allow users to access multimedia content without concern for specific coding formats, terminal capabilities, or network conditions.

Several methods for UMA are enabled or supported by the use of metadata. We will take a closer look at the use of

Authorized licensed use limited to: Florida Atlantic University. Downloaded on May 08,2025 at 22:15:44 UTC from IEEE Xplore. Restrictions apply.
1053-5888/03/$17.00©2003IEEE

Amazon/DivX
Exhibit 1017
Page 1

metadata in UMA; in particular, we discuss the MPEG-7 metadata standard, recently finalized by the MPEG committee. MPEG-7, formally named Multimedia Content Description Interface, provides standardized tools for describing multimedia content [4], [5]. Thus, while earlier MPEG standards, such as MPEG-1, MPEG-2, and MPEG-4, specify standard syntax to encode the content itself, MPEG-7 instead specifies a syntax for encoding metadata associated to this content. The MPEG-7 framework is based on the eXtensible Markup Language (XML) and the XML Schema language [6]. MPEG-7 has standardized a comprehensive set of description tools, i.e., descriptors (Ds) and description schemes (DSs) to exchange information about the content (e.g., creation date, title, author, genre) and information present in the content (low-level audiovisual (AV) features such as color, texture, shapes, timbre, and tempo; mid-level AV features such as spatio-temporal segmentation; and high-level features such as content semantics). Ds and DSs are defined as schemas using a textual description definition language, largely equivalent to XML Schema. MPEG-7 also provides system tools for transport and storage of metadata fragments, including a generic compression format for binary encoding of XML data. MPEG-7 descriptions in textual format are simply XML instance documents that conform to the syntactic rules expressed by Ds and DSs. Note that MPEG-7 standardizes neither the extraction nor the usage of descriptions.

In particular, MPEG-7 supports UMA by providing a wide variety of tools for describing the segmentation, transcoding hints, variations, and summaries of multimedia content. MPEG-7 also provides tools for describing user preferences and usage history. In this article, we will discuss methods that support UMA and the tools provided by MPEG-7 to achieve this. We also briefly discuss the inclusion of metadata in JPEG 2000 encoded images. We present these methods in the typical order that they may be used in an actual application (although, of course, these methods can be used individually or in any other way an application requires). Therefore, we first discuss the (personalized) selection of desired content from all available content, followed by the organization of related variations of a single piece of content. Then, we discuss segmentation and summarization of AV content, and finally, transcoding of AV content.

Personalized Selection of Multimedia Content

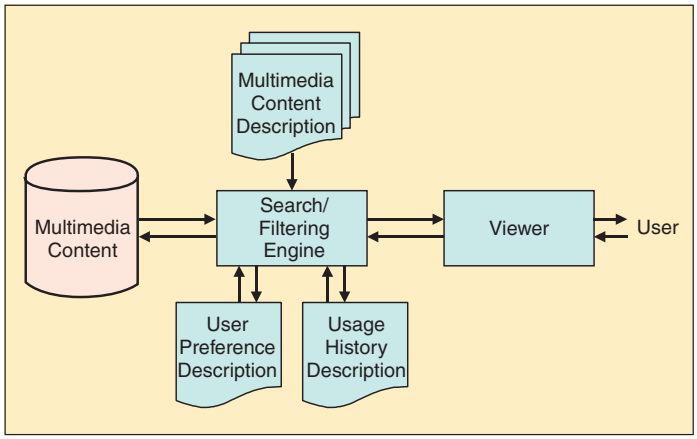
Naturally, discovering and selecting a particular piece of content among those offered by providers are among the first steps in accessing it. This may range from selecting among broadcast TV channels and programs using a paper or electronic TV program guide, all the way to selecting and downloading songs from an Internet digital music service. To provide effective access to growing amounts of multimedia content, it is important to consider the user's preferences, for example, by including a search engine, filter engine, and/or recommendation engine in the delivery system. Capturing a representation of the user's preferences relating to multimedia content in a user profile provides various benefits. Multimedia content consumers will be able to capture their preferred content and employ software agents to automatically figure out their personal tastes and discover, select, and recommend new multimedia content. A standardized format enables users to enter or update their preferences using one device and then import them into multiple other devices for instant customization. Users can carry a representation of their preferences in a secure smart card or other type of removable storage.

Personalization Approaches

Typically, one or more of the following types of information is available to a system that provides automatic filtering or recommendation services, as shown in Figure 1:

- descriptions of the multimedia content
- descriptions of users' preferences related to multimedia content, i.e., user profiles
- descriptions of users' content usage history, i.e., logs of past usage.

Also, the system may use some type of user identifiers and some demographic information. Descriptions of a content item, in terms of various attributes such as author, title, genre, parental rating, language, and keywords, to name a few, enable users to query the system and to search for desired content. This is a basic search engine or information retrieval paradigm, supported strongly by standards such as MPEG-7. However, this approach is mainly useful for very narrow and ad hoc queries. A second paradigm, called information filtering, utilizes a user profile to capture long-term preferences, often in terms of the same types of attributes that describe the content itself. Such a profile can be matched against content descriptions to automatically filter out undesirable content items



1. Overview of an interactive multimedia delivery system for personalization and customization.

Authorized licensed use limited to: Florida Atlantic University. Downloaded on May 08,2025 at 22:15:44 UTC from IEEE Xplore. Restrictions apply.

Amazon/DivX
Exhibit 1017
Page 2

MPEG-7 supports information filtering and collaborative filtering through tools for describing user preferences and usage history.

or to recommend desired content. User profiles can be constructed manually (by explicit input) or can be inferred automatically (based on implicit input) by utilizing a usage history, e.g., based on information about which content items the user viewed, listened to, or recorded. The latter can be performed periodically to adapt to changing user tastes. Explicit approaches may include having the users to set up their profile in advance or may use ratings the user(s) provided for particular content items. The filtering or recommendation approach is useful in a situation where there is a single user or only a few users of a device and when all information about users' preferences are required to remain in the hands of the users. A third approach, called collaborative filtering, applies to communities of users that share their explicit opinions or ratings of content items. In a collaborative filtering system, recommendations for a particular user are based on the correlation between that user and a set of other users and information about how each user rated particular content items. Such ratings, again, may be seen conceptually as part of the user's usage history. Pure collaborative filtering techniques do not require any metadata or content descriptions; however, they require content items to be rated by several users before they can be recommended to anyone. Information retrieval and filtering techniques rely on the availability of content descriptions (metadata) and hence are sometimes called content based in literature. Hybrid approaches proposed more recently build on the strengths of both information filtering and collaborative filtering and may make use of both metadata and user ratings.

The above approaches are discussed in more detail in [7] and [8], which also provide an overview of prior work in literature, including hybrid approaches. Discussions of the privacy implications of such personalization systems are included in [7]. Novel algorithms for automatically determining a user's profile from his/her content usage history and for automatically filtering content according to the user's profile are presented in [9]. The profiling and filtering agents proposed in [9] support generation and utilization of MPEG-7 user preferences and usage history descriptions. The bootstrapping problem of content filtering or recommendation engines (i.e., how to quickly capture a representation of a user's preferences without a complicated dialog or lengthy learning period) is addressed in [10].

Personalization Tools in MPEG-7

MPEG-7 supports information filtering and collaborative filtering through tools for describing user preferences and usage history (see [4] and [5]).

The UsageHistory DS consists of lists of actions performed by the user over one or more nonoverlapping observation periods. Each action list is action-type specific, i.e., a single list contains actions of a certain type (such as "PlayStream," "PlayRecording," or "Record") only. A variety of actions that are recognized and tracked have been defined in an extensible dictionary (called a classification scheme in MPEG-7). Associated with each user action are the time and duration of the action, an identifier of the multimedia content for which the action took place, and optional referencing elements that point to related links or resources about the action. The most important pieces of information are the content identifiers, which provide links to the descriptions of the corresponding content items. This approach eliminates the need to store duplicate content descriptions and facilitates updates. The time information associated with each user action can be specified in terms of the general time, which denotes the time of occurrence in coordinated universal time (UTC); media time, which denotes the time information that is encoded with a piece of multimedia content; or both. This provides accurate timing information for actions such as "FastForward" or "Rewind."

The UserPreferences DS can be used to capture a variety of individual preference attributes. Preference attributes related to the creation of content include: title (e.g., a favorite TV series), creators (e.g., favorite actor, songwriter), keywords (e.g., news topics), location, and time period (e.g., recorded in Spain or during the 1950s). Preference attributes related to classification of the content include: country and date of release, languages used, production format (e.g., daily news versus documentary), and genre (e.g., science fiction versus western). Preference attributes related to the dissemination of content include: delivery type (e.g., terrestrial broadcast, DVD-ROM), source (e.g., broadcast TV channel or web service), date and time of availability, disseminator (e.g., publisher or broadcaster), and media format (e.g., video-coding format, aspect ratio). Another set of preference attributes relates to efficient navigation and summarization.

Individual preference attributes can be grouped to express various combinations of preferences, and attributes can be grouped hierarchically such that certain preferences are conditioned on other preferences being satisfied. The UserPreferences DS enables users to specify preferences that apply only in a specific context, in terms of date, time, and location. Many of the individual attributes listed here have an associated numerical weight that allows users to specify the relative importance of their preferences with respect to each other and to express negative preferences and dislikes. The UserPreferences DS also enables users to indicate whether their preferences or parts of their preferences should be kept private or not. Fi-

Authorized licensed use limited to: Florida Atlantic University. Downloaded on May 08,2025 at 22:15:44 UTC from IEEE Xplore. Restrictions apply.

Amazon/DivX
Exhibit 1017
Page 3

nally, the DS enables users to indicate whether the automatic update of their usage preferences description, e.g., by a software agent, should be permitted or not. A very simple example user preference description in MPEG-7 format expressing a favorite actor and director is shown in Figure 2.

Application Scenarios

Collaborative filtering systems are currently prevalent on the World Wide Web, where the recommendation engine resides in a server that provides a particular service, e.g., online shopping and product review sites. Such engines analyze the actions and requests performed by communities of users of that service and may rely on explicit user ratings. Information filtering systems are prevalent in advanced digital TV applications, where a recommendation engine is often part of the set-top box middleware or personal video recorder (PVR) software. Advanced digital TV is an important application area for the MPEG-7 standard and the tools discussed above. Content descriptions can be used to populate electronic program guides (EPGs) used to select TV programs for viewing and recording. Moreover, the techniques discussed above can be used to guide the user in finding desired TV programs among the many available channels more efficiently.

Multimedia Content Variations

Given a particular piece of content, such as a song or television program, several alternative versions or variations may exist, for instance in several coding formats or at several bit rates. In general, these variations can be derived from the multimedia content by applying well-known methods for editing, extraction, summarization, or translation or can simply represent alternative versions of the multimedia data. In UMA applications, the variations can be selected and delivered as replacement, if necessary, to adapt to client terminal capabilities, such as display size, processing power, local storage, data format compatibility, network conditions, or user preferences. For example, Internet content providers often provide multiple variations of the same content, each tuned to the client's access bandwidth limitations and the user's audio/video format preference. Selection among these variations can be automated using metadata describing each variation as well as the client's capabilities [1], [11].

MPEG-7 Variation Tools

In MPEG-7, the concept of variations is defined as an alternative version of multimedia content, which may be derived through transcoding, summarization, translation, reduction, or other types of processing. The quality of the variation compared to the original (source content) is given by a variation fidelity value. The priority of a variation relative to other variations is indicated by a variation priority attribute. The variation relationship attribute specifies the type of association or relationship of the variation AV program with the source AV program. This attribute may indicate general types of processing of the content such as revision by editing or post-processing, substitution of (parts of) the content, or data compression. Processing types usually associated with transcoding are reduction of the bit rate, reduction of the color detail (bit depth), reduction of the spatial or temporal resolution, reduction of the perceived quality, and changes of the coding format. Other processing types that can be indicated are summarization (presenting the important information of the original content in a compact form), abstraction (abstracts are often authored separately, unlike summaries), extraction (e.g., voice excerpts from audio content, or objects and events from video content), and modality translation (e.g., text-to-speech conversion, speech recognition, video-to-image or video mosaicing, text extraction from images).

Example of a Variation Set

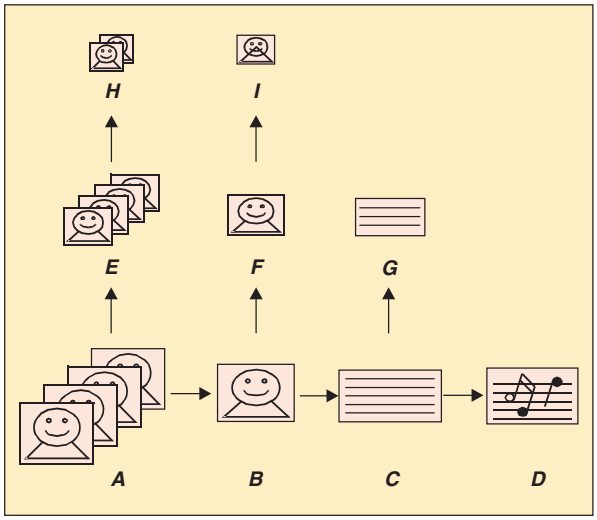
Figure 3 depicts an example of an MPEG-7 variation set containing a set of variations of a source video. The example depicts the source video (A) and shows eight variations: two variations are video content (E and H), three variations are images (B, F, and I), two variations are text (C and G), and one variation is audio (D). The content modality varies from left to right, while the fidelity of the variations varies from top to bottom. In this example, the variations are derived from the source video. For example, variation E (video) is derived from the source video A via spatial reduction and compression. Variation B (image) may be derived from the source video A by Extraction.

```
<Mpeg7>
    <Description xsi:type="UserDescriptionType">
        <UserPreferences>
            <FilteringAndSearchPreferences>
                <CreationPreferences>
                    <Creator preferenceValue="30">
                        <Role href="urn:mpeg:mpeg7:cs:RoleCS:ACTOR"/>
                        <Agent xsi:type="PersonType">
                            <Name>
                                <GivenName>Tom</GivenName>
                                <FamilyName>Hanks</FamilyName>
                            </Name>
                        </Agent>
                    </Creator>
                    <Creator preferenceValue="50">
                        <Role href="urn:mpeg:mpeg7:cs:RoleCS:DIRECTOR"/>
                        <Agent xsi:type="PersonType">
                            <Name>
                                <GivenName>Clint</GivenName>
                                <FamilyName>Eastwood</FamilyName>
                            </Name>
                        </Agent>
                    </Creator>
                </CreationPreferences>
            </FilteringAndSearchPreferences>
        </UserPreferences>
    </Description>
</Mpeg7>
```

▲ *2. Example MPEG-7 user preferences description in XML format.*

Authorized licensed use limited to: Florida Atlantic University. Downloaded on May 08,2025 at 22:15:44 UTC from IEEE Xplore. Restrictions apply.

Amazon/DivX
Exhibit 1017
Page 4



▲ 3. Illustration of different variations of a source video (A). The eight variations (B, …, I) have different fidelities and modalities (video, image, text, and audio) with respect to the source content.

Summarization of Audiovisual Content

In this section, we discuss segmentation and summarization of AV content in more detail. Segmentation of audio, video, and images refers to the partitioning of the data in space and/or time to produce coherent elementary components called segments. The most common types of segments either correspond to temporal intervals of audio or video or to spatial regions of a still image. Summarization of audio and video usually refers to the generation of a compact presentation of the essential information in the content. Such a presentation may contain audio components (segments), visual components (segments), textual components, or a combination of all three.

In the context of UMA, summarization of AV content has been motivated primarily by the need to reduce information overload on the human consumer, to enable rapid access to important information in the AV content, or to enable the viewer to consume more content in a given time. AV summaries are useful in many applications, such as entertainment (e.g., sports highlights), informational (e.g., a digest of broadcast news), and educational or training (e.g., a synopsis of presentations or lectures). A summary may be used instead of the original content to quickly gain an understanding of the most important parts. A summary can also be used as a preview to help in deciding whether to consume the entire content (e.g., movie trailers).

We refer to summary descriptions or summary metadata as the information defining a summary, such as identifiers, locators, and time stamps. The combination of summary descriptions and the original AV content allows construction and presentation of an AV summary. Segment or summary descriptions can be used to navigate the content or to enable selective recording of broadcast video streams. Highly entertaining or informative video clips could be "bookmarked" for later review and exchanged with friends and family. In the broadcast video application, summaries support smart fast-forwarding or quick catch-up of missed television segments.

Automatic Summarization Techniques

Summary descriptions of AV data could be generated manually; however, this is often too time consuming and costly. Work towards automatic summarization algorithms dates back approximately ten years. Almost all work on summarization reported in literature employs an analysis-synthesis approach. The AV data is first analyzed to extract elementary components, such as shots or key frames, and to assess their relative importance. Then, the most important components are selected, possibly condensed further, and organized to generate a summary. A summary is called generic if its author determined the important components, while a summary is called query based if it is based on specific input from a user, i.e., the summary is adapted to a given query and possibly personalized to the user's preferences.

Overviews of techniques used to analyze multimedia content are presented in [12] and [13]. Early research focused on automatic shot boundary detection and key frame extraction, using low-level features such as motion and color. More advanced work focused on detection and classification of specific high-level events in particular domains, such as segmentation of stories in news broadcasts [14] or, more recently, detection of plays in sports video programs [15]. Other work includes analysis of documentaries [16] and home movies [17]. More recently, summarization techniques have also been used to support content-based adaptive streaming of video over packet-based or wireless networks [18]. Finally, video mosaicing is a summarization technique that produces panoramic scene views by warping and aligning video frames [19].

Shot Boundary Detection and Key Frame Extraction

Video summarization is a growing field of research and development in video analysis and representation. Here, we provide an overview of techniques for shot boundary detection and key frame extraction, which are important components in many video summarization methods. In such methods, the video is first segmented along the time axis into basic units called shots. A shot is defined as a sequence of frames captured by one camera in a single continuous action in time and space. This temporal segmentation corresponds to the detection of the shot boundaries. In a typical video, different types of shot boundaries can exist. A cut is an abrupt shot change that occurs between two consecutive frames. A fade-in starts with a black frame; then, gradually the image of the next shot appears, until this shot is shown at full strength. A

Authorized licensed use limited to: Florida Atlantic University. Downloaded on May 08,2025 at 22:15:44 UTC from IEEE Xplore. Restrictions apply.

Amazon/DivX
Exhibit 1017
Page 5

fade-out is the opposite of a fade-in. A dissolve consists of the superimposition of a fade-out over a fade-in. Fade-in, fade-out, and dissolve are also referred to as gradual transitions. Once the shot boundaries are detected, the salient content of each shot is represented in terms of a small number of frames, called key frames. Color is one of the most common visual primitives used for shot boundary detection. This is based on the hypothesis that two consecutive frames from different shots are unlikely to have similar colors. Zhang et al. [20] used histograms as descriptors for color. After defining a histogram distance, if the so-called distance between the color histograms of two consecutive frames was higher than a threshold, then a cut was detected. Based on the same principle, but using a more sophisticated approach, gradual shot boundaries could also be detected. Zabih et al. [21] proposed to use edges as visual primitives for temporal video segmentation. In a first stage, edge detection is performed on two consecutive frames. A dissimilarity measure is then applied to detect cuts and gradual changes based on the fraction of edge-pixels that appear or disappear between two consecutive frames. Bouthemy et al. [22] proposed to use the iteratively reweighted least squares (IRLS) technique to estimate efficiently the dominant motion prior to shot boundary detection. This robust estimation technique allows for detection of points that belong to the portion of the image undergoing the dominant motion (inliers). If a cut occurs between two consecutive frames, the number of inliers is close to zero. On the opposite, if consecutive frames are within the same shot, then the number of inliers is nearly constant. Vasconcelos and Lippman [23] used the Bayes rule to derive a log-likelihood test for shot boundary detection. An activity measure is defined as the residual error, after alignment of two consecutive frames. This is accomplished by estimating the dominant motion. Abdeljaoued et al. [24] proposed a shot boundary detection based on feature point extraction and tracking and computation of an activity measure. They then defined a model-based classification in order to identify various types of shot boundaries.

Many criteria can be used to extract key frames from a shot. For instance the first or middle frame in a shot can be chosen as a key frame. Although simple, this technique is not always efficient. Key frame extraction can also be obtained by comparing the current frame of a shot to the last selected key frame by using a color histogram-based distance. If this distance is higher than a threshold then the current frame is selected as a new key frame. Otherwise the next frame is processed. This process can be iterated until the end of the shot. Thus any significant action in the shot is represented by a key frame. Wolf [25] proposed a motion-based approach for key frame extraction. First the optical flow for each frame is determined, and a motion metric based on optical flow is computed. Then, by analyzing the motion metric as function of time, key frames are selected at the minima of motion. This is based on the assumption that in a good key frame, the camera usually stops on a new position or the characters hold gestures to emphasize a given action.

UMA promises an integration of different perspectives into a new class of content adaptive applications that could allow users to access multimedia content without concern for specific coding formats, terminal capabilities, or network conditions.

MPEG-7 Summary Descriptions

An MPEG-7 summary description defines the summary content, how it relates to the original content, and how an actual summary of the original AV content can be composed from these and presented to the user (see [1] and [5]). The MPEG-7 summary description tools are based on the following key concepts.

▲ *Original (source) content*: the original content that is being summarized. There may be one or more source content items being used in a single summary (i.e., MPEG-7 supports both single-document and multidocument summarization). The original content can be identified using unique identifiers and can be located using media locators (e.g., URL and/or time stamps).

▲ *Summary content*: parts or components of the source content that are used to create a summary. Examples of such components are key video segments, key audio segments, and key frames. These components are located using timestamp elements defined with respect to the time-base of the original content. Note that a summary may also use alternative or external content that is not physically part of the source content to represent parts of the original. An example of the latter may be the use of an iconic image to abstractly represent an important event in a video segment.

▲ *Summary*: an abstract or extract of the source content that conveys the essential information of the original. A summary is a composite of the summary content that can be presented to the user. For example, a summary may simply be a concatenation of video segments taken directly from the original, where segments are specified by their start time and duration. A summary (or its parts) may be presented at increased or decreased speeds relative to the source.

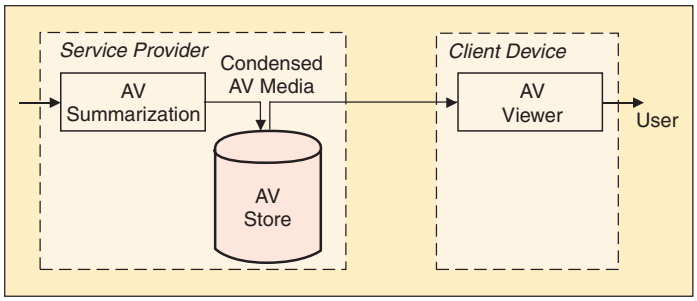
Within MPEG-7, two description schemes have been developed in the area of summarization: the HierarchicalSummary and SequentialSummary DSs. The HierarchicalSummary DS is used to specify summaries of time- varying AV data that support sequential and hierar-

Authorized licensed use limited to: Florida Atlantic University. Downloaded on May 08,2025 at 22:15:44 UTC from IEEE Xplore. Restrictions apply.

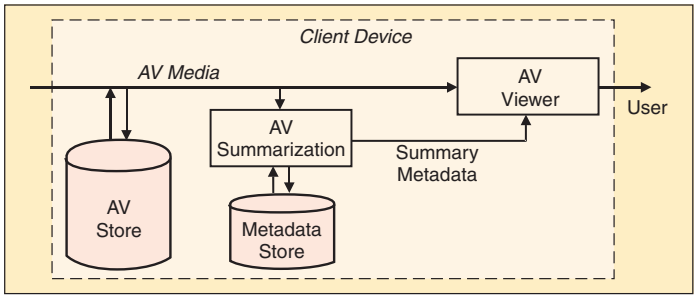
Amazon/DivX
Exhibit 1017
Page 6

chical navigation, for example, highlight video summaries of various duration or summaries containing key events of various importance. These summaries are composed of contiguous audio, video, or AV segments, as well as their key frames and key sounds. The SequentialSummary DS is used to specify summaries of time-varying AV data that support sequential navigation, for example, a variable-speed fast-forward of video or an audio slideshow. These summaries are composed of images or video frames and audio clips that can be synchronized and presented to the user at varying speeds.

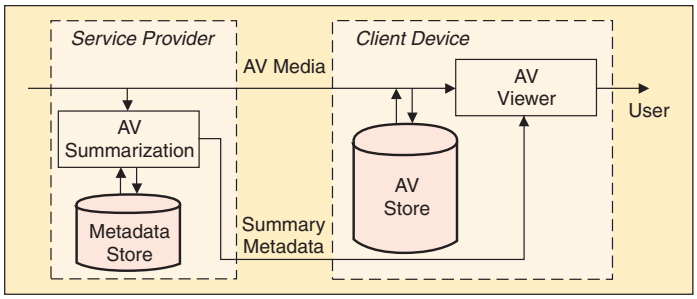
Furthermore, MPEG-7 defines a set of tools to describe partitions and decompositions of image, video, and audio signals in space, time, and frequency, which can be used to support progressive retrieval of the data. Examples of such partitions are spatial regions-of-interest of a still image and frequency subbands of a video signal. Examples of decompositions are spatial quad-trees of an image, frequency subband or wavelet decompositions, or graphs that organize partitions resulting from simultaneously decomposing an audio or video signal in time, space, and/or frequency.



▲ *4. Summarization by a service or a content provider, where the summarized AV content is delivered to the client device.*



▲ *5. Summarization in the client device.*



▲ *6. Summarization by a service or a content provider, where summary metadata is delivered to the client device.*

Note that the MPEG-7 standard leaves considerable room for applications to render or to present a given summary in a variety of ways. For example, a set of key frames could be shown simultaneously according to some spatial layout or could be shown sequentially in some user-defined tempo.

Application Scenarios

A standardized format, such as that provided by MPEG-7, enables delivery of AV summaries to consumers, via broadcast, Internet, or wireless personal communications services, as well as storage on removable or packaged media. There is currently a growing trend towards on-demand television in both cable and satellite TV systems. Video-on-demand (VOD) services are being introduced, which offer an increasing amount of digital video content to TV consumers. Also, there is a growing adoption of advanced TV devices such as set-top boxes, home gateways, and PVRs/DVRs (personal/digital video recorders) with increasingly large amounts of video storage capabilities, similarly used for on-demand or time-shifted TV viewing. Several scenarios can be envisioned for introducing summarization into such TV applications, as illustrated in Figures 4-6.

The first scenario simply extends an existing VOD service, where the VOD service provider performs the AV summarization, edits the AV data, and makes a condensed version of the content available through its service (see Figure 4). An example could be summaries of sports game videos; viewers may not be interested in watching a recording of the entire game after it has been broadcast live yet may be interested in watching the highlights only. This scenario does not require any infrastructure upgrades of the VOD system, since the condensed content is made available through the same mechanism as the regular content.

A second scenario is where the summarization engine is added to a client device with video storage capabilities such as a PVR (see Figure 5). In this case, the engine would be invoked after the AV data has been recorded and would subsequently generate metadata describing summaries (in terms of the time codes of important segments etc.) used by the viewer to display the appropriate parts of the original content. Advantages of this approach are that the source content is still available for viewing if the user wishes so. In addition, such summaries can be highly personalized. However, the summarization engine likely adds to the complexity and therefore the cost of the client device, which often have limited memory and processing capabilities.

The third scenario moves this additional complexity back into the server, while retaining the

Authorized licensed use limited to: Florida Atlantic University. Downloaded on May 08,2025 at 22:15:44 UTC from IEEE Xplore. Restrictions apply.

Amazon/DivX
Exhibit 1017
Page 7

flexibility of the second scenario (see Figure 6). In this case, the service provider performs summarization and makes the resulting summary metadata available in parallel with the original AV content. This content can be recorded by the client device as is the case currently with PVR or similar devices and is available to the viewer in its entirety; however, the summarization metadata can be utilized at will to consume the content in a personalized and rapid manner. This third scenario requires delivery of summary metadata to the consumer in a standardized format such as MPEG-7. This metadata can be delivered alongside the content, i.e., multiplexed together with MPEG-2 or MPEG-4 audio and video streams, or separately, e.g., simply as a standalone XML file.

MPEG-7 supports UMA by providing a wide variety of tools for describing the segmentation, transcoding hints, variations, and summaries of multimedia content.

Transcoding of Audiovisual Content

UMA is one of the most important concepts supporting content distribution over networks with varying characteristics or involving client terminals with varying capabilities [1], [11], [26]. A UMA system may employ transcoding to adapt an image, audio, or video signal to the varying network characteristics or client terminal capabilities. A transcoder is an advantageous solution to avoid storage of the same content in several formats at the server and to precisely meet constraints on the bit rate that were not known a priori. Several transcoding methods have been proposed for reducing bit rate, as well as temporal and spatial resolutions, and for converting a coded bit stream into different formats [27], [28]. For audio and video applications, a low-delay, low-complexity, and subjective quality preserving transcoding method is required. As the transcoder often operates in a real time environment, its computational cost is an important issue, especially for the low-cost home server market. The transcoder implementation efficiency becomes crucial in the case when the server has to deliver video contents at different bit rates to more than one client at the same time, requiring several transcoding instances operating in real time. Therefore, the complexity of a transcoder should be as low as possible while maintaining picture quality as high as possible. To facilitate transcoding in the face of such requirements, so-called transcoding hints can be generated in advance and stored together with (or separately from) the compressed AV content. A system model of a transcoder using transcoding hints metadata is depicted in Figure 7.

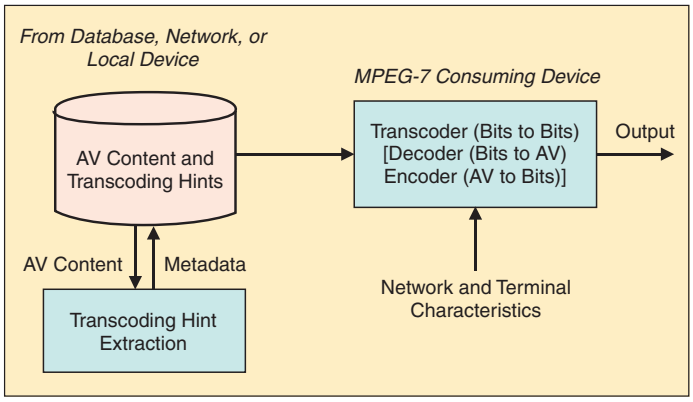
Transcoding Hints in MPEG-7

In MPEG-7, the concept of transcoding hints relates to metadata that indicates how multimedia content can be adapted or transcoded. The MPEG-7 MediaTranscodingHint DS can be used to guide the transcoding of multimedia, and it supports object-, segment-, and region-based description of the importance, priority, and value of content, as well as a description of transcoding behavior based on transcoding utility functions and network scaling profiles. In addition, the MediaFormat DS supports description of image and video frame sizes (width and height), frame rates of video, file sizes for image, video and audio download, storage formats and MIME-types, and many other attributes.

The MPEG-7 media transcoding hints allow content servers, proxies, or gateways to adapt image, audio, and video content to different network conditions, user and publisher preferences, and capabilities of terminal devices with limited communication, processing, storage, and display capabilities. Transcoding hints can be used for complexity reduction as well as for quality improvement in the transcoding process [29]-[32].

Among the MPEG-7 transcoding hints that we describe are the motion hint, the difficulty hint, the importance hint, the shape hint and the spatial resolution hint.

The motion hints describe the motion range, the motion uncompensability, and the motion intensity. These hints can be used for a number of tasks, including anchor frame selection, coding mode decisions, frame-rate and bit-rate control, as well as bit-rate allocation among several video objects for object-based MPEG-4 transcoding. These hints, especially the motion range hint, also enable computational complexity reduction of the transcoding process. The motion range hint helps a transcoder in setting an appropriate motion vector search range during motion estimation [32]. This is useful in particular when transcoding a video source from an intraframe-only coding format to an interframe (motion compensated) coding format.



▲ *7. Generic illustration of a transcoding application scenario.*

Authorized licensed use limited to: Florida Atlantic University. Downloaded on May 08,2025 at 22:15:44 UTC from IEEE Xplore. Restrictions apply.

Amazon/DivX
Exhibit 1017
Page 8

The difficulty hint describes the encoding complexity of segments within a video sequence (or regions within an image). This hint can be used for improved bit rate control and bit rate conversion from constant bit rate (CBR) to variable bit rate (VBR). The bit allocation and the selection of quantizer parameters depend on the rate-distortion characteristic [33], [34]. Based on these models, the difficulty hint is used to calculate the bit budget. The hint is a weight, which indicates the relative encoding complexity of each segment. It is normalized within the content. The encoder can assign more bits to a difficult scene and remove bits from a relatively easy scene. Since mainly the pictures with low quality are improved, the overall subjective quality is significantly improved. To encode efficiently in VBR mode, the encoder has to know the encoding difficulty of the whole sequence in advance.

The importance hint specifies the relative semantic importance of video segments within a video sequence, or of regions or objects within an image, and can be used by a rate control mechanism to improve subjective quality. The encoder can allocate more bits to important parts of the content. Using the difficulty hint and importance hint together, a video transcoder can control bit rate and quality efficiently. The importance hint can be used to annotate different image regions with their importance. This information can then be used to transcode an image for adaptive delivery according to constraints of client devices and bandwidth limitations. For example, text regions and face regions can be compressed with a lower compression ratio (higher quality) than the remaining regions. The less important parts of the image can be blurred and compressed with a higher ratio to reduce the overall bit rate of the compressed image or video. The MPEG-7 importance hint information has advantages over methods for automatic extraction of regions from images in the transcoder in that the importance hints can be provided by the content authors or publishers, providing them with greater control over the adaptation and delivery of content. The importance hint takes values from 0 to 1, where 0 indicates the lowest importance and 1 the highest.

The shape hint specifies the amount of change in an object shape boundary over time and is used to overcome the composition problem when encoding or transcoding multiple video objects with different frame rates. For instance, when video objects are converted into different temporal resolutions, holes, which are uncovered areas in which no pixels are defined, could appear in the composited scene due to the movement of one object, without the update of adjacent or overlapping objects. The transcoder examines shape hints to determine if composition problems will occur at the reconstructed scene if various temporal changes occur after transcoding. In this case, the temporal rates for each object can be computed with the assistance of additional transcoding hints or content characteristics [35].

The spatial resolution hint specifies the maximum allowable spatial resolution reduction factor for perceptibility. The spatial resolution hint takes values from 0 to 1, where 0.5 indicates that the resolution can be reduced by half, and 1.0 indicates the resolution cannot be reduced.

Note that each transcoding hint may be associated with segments of video data, i.e., temporal intervals or spatial regions, which are described by other metadata.

Transcoding of JPEG 2000 Images

The new still image compression standard JPEG 2000 [36], [37] has been developed with the intention of facilitating access of digital still images with various devices and through limited bandwidth channels, making it simple to extract different versions of an image from one and the same compressed image file. JPEG 2000 includes such techniques as tiling, wavelet decomposition, bitplane encoding, and interleaving to form a scalable code stream. Depending on the progression order of the code stream, one can access, for example, a low-quality version (quality progressive) or a low-resolution version (resolution progressive) of the image simply by decoding the first part of the compressed bit stream and ignoring the remainder. It is also simple to parse the code stream and retrieve only the data pertaining to a particular region-of-interest (ROI) of the image or to retrieve the image in a progression order different from that used when encoding the image. Furthermore, JPEG 2000 Part 9 will standardize the protocol necessary for a client to request particular regions of an image at a particular resolution from a server. If the client is unaware of the image content, it may be useful to include metadata in the image file to tell a server or a gateway what version of the image to deliver to a limited client. JPEG 2000 Part 2 [38] specifies a region-of-interest description box, which contains the number of ROIs in the image, whether or not each is a codestream ROI (coded with higher quality and/or placed in the beginning of the codestream), their relative importance, their shape (rectangular or elliptical), and their size. JPEG 2000 Part 2 also provides a wide range of other metadata.

Image Transcoding Optimization

Given that multiple ROIs in an image can be annotated using (MPEG-7 or JPEG 2000) metadata, the transcoding of the image needs to consider the individual importance and spatial resolution hint for each region. Overall, this can be seen as an optimization problem in which the image needs to be manipulated, for instance through cropping and rescaling, to produce an output image that maximizes the overall content value given the constraints on its delivery. This optimization problem can be expressed as follows. The device imposes a constraint on the size of the image (i.e., size of the screen). The transcoding engine seeks to maximize the benefit derived from the content value of the transcoded image. The goal,

Authorized licensed use limited to: Florida Atlantic University. Downloaded on May 08,2025 at 22:15:44 UTC from IEEE Xplore. Restrictions apply.

Amazon/DivX
Exhibit 1017
Page 9

thus, is to maximize the total content value given the constraints. Following the particular structure provided by the MPEG-7 transcoding hints, we consider that each region R_i has an importance I_i, where $0 \le I_i \le 1$, and spatial resolution hint S_i, where $0 \le S_i \le 1$. We consider also that each region has a content value score V_i after rescaling the image globally using rescaling factor L, where $L \ge 0$. The content value score V_i indicates the value of transcoded region R_i and is a function of its importance I_i, spatial resolution hint S_i, and the rescaling factor L as follows:

$$V_i = \begin{Bmatrix} I_i \text{ if } 1 \le L \\ LI_i / S_i \text{ if } 0 \le L < 1 \\ 0 \quad \text{otherwise} \end{Bmatrix}.$$

Then, the problem can be stated as follows: select a subset of regions R_i and a rescaling factor L such that the overall value $\sum_s V_s$ of the subset is maximized, while the minimum bounding rectangle that encapsulates the selected rescaled regions fits the device's screen size.

One way to solve this problem is by using exhaustive search over all possible combinations of the rescaled regions. In this case, for each unique combination of regions, the image is cropped to include only those selected regions, and the cropped image is then rescaled to fit the device's screen size. This candidate solution is then evaluated in terms of its content value given the rescaling and selection of regions. Finally, the combination of regions with maximal value is selected as the optimal transcoding of the image. The complexity of this approach is usually acceptable considering that each image will typically only have a handful of annotated regions.

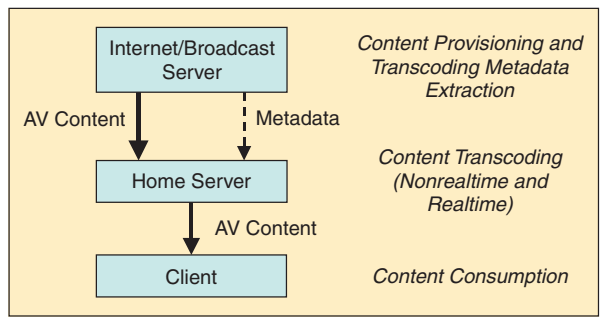
Application Scenarios

In this section, a few distinct application scenarios that could benefit by using MPEG-7 and JPEG 2000 transcoding hints are summarized. Each of these applications differs in the way the transcoding hints may be extracted, and in their usage.

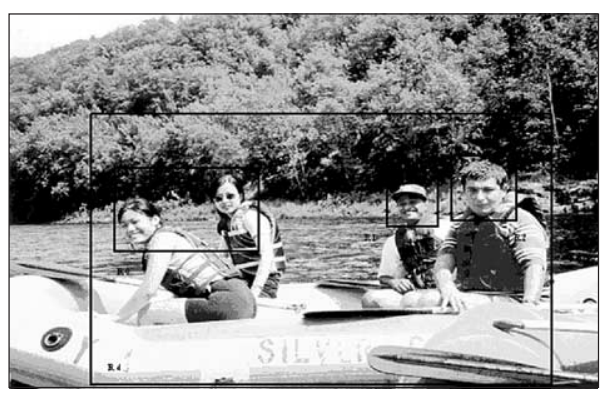
An intuitive application is the server and gateway application. Figure 7 illustrates the generic system structure. In the case of content being stored on a server, the metadata extraction is performed at the server, where real-time constraints are relaxed. The content is transmitted through the network with extracted MPEG-7 transcoding hints. The server/gateway converts the media format using the associated hints. Since the transcoding hints have been generated offline, the server/gateway can implement the transcoding efficiently, therefore minimizing the amount of delays. No complexity is added in the client device (e.g., a mobile phone). As a variation of this application scenario, Figure 8 illustrates a situation with two servers, a broadcast/Internet server, and a home server. In the example shown, the broadcast/Internet server distributes content, such as MPEG-2 encoded video, with associated transcoding hints to a local home server. The home server transcodes the bit stream using transcoding hints. In some situations, non-real time operation is allowed. The home server enables a wide range of interconnections and may be connected to the client through a wireless link (e.g., Bluetooth), by cable link (e.g., IEEE 1394, or USB), or by packaged media, like flash memory or magneto/optical disks.

Another application is scalable video content playback. In this application, content is received at the client, but the terminal does not have the resources to decode and/or display the full bit stream in terms of memory, screen resolution, or processing power. Therefore, the transcoding hints are used to adaptively decode the bit stream, e.g., by dropping frames/objects or by decoding only the important parts of a scene, as specified by the received transcoding hints.

Another application example is a client-server scenario where the client requests an image from the server at a resolution that is small relative to the original image size. If the server has access to metadata indicating one or more ROI(s) in the image being more important than the rest of the image, the server could choose to send only the important region(s) of the image at the desired resolution, rather than globally rescaling the image. This is illustrated in the following example using the image shown in Figure 9. In this example, we consider four different display devices: PC, TV browser, handheld computer, and personal digital assistant (PDA), or mobile phone. Figure 10 shows the results without using transcoding hints. In this



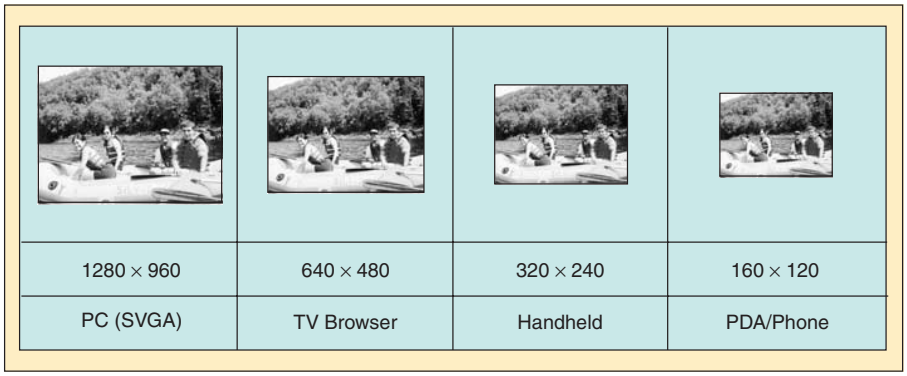
▲ *8. Two-server application scenario.*



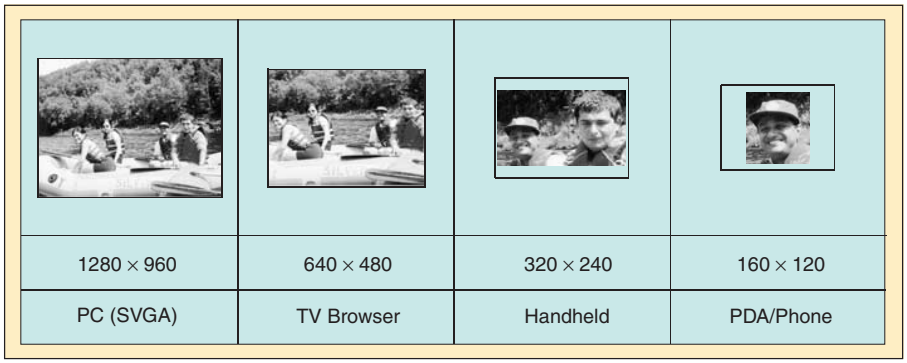
▲ *9. Image with four ROIs annotated using MPEG-7 transcoding hints.*

Authorized licensed use limited to: Florida Atlantic University. Downloaded on May 08,2025 at 22:15:44 UTC from IEEE Xplore. Restrictions apply.

Amazon/DivX
Exhibit 1017
Page 10



▲ *10. Example of transcoded image output by globally rescaling the image to fit the screen.*



▲ *11. Example of transcoded image output using the MPEG-7 transcoding hints for image region importance and minimum allowable spatial resolution. The transcoding uses a combination of cropping and rescaling to maximize the overall content value.*

case, the original image is adapted to fit each screen size by globally rescaling the image. Clearly, the image details are lost when the size of the output image is small. For example, it is difficult to discern any details from the image displayed on the PDA screen (far right). A similar loss of details occurs for all regions, including important regions, resulting in a lower overall content value for a given display size. Figure 11 shows the results using metadata describing the ROIs and transcoding hints. In this case, the original image is adapted to each screen size by using a combination of cropping and rescaling, fitting a set of selected image regions to the screen. The advantage of this type of adaptation is that the important image details are preserved when the size of the output image is small. For example, it is possible to discern important details from the image displayed on the handheld and PDA screens. The result is an adaptive loss for different regions, resulting in a higher overall content value of the image for a given display size.

If the image data itself were encoded using JPEG 2000, there are several ways in which the transcoding could be done. Fully transcoding the image into an image containing only the ROI(s) by decoding, cropping, and reencoding would be inefficient. If the image was tiled, the server could send only those tiles containing the ROI(s); otherwise it could transcode the image into a tiled image with one or more of the tiles covering the ROI(s). Again, the latter method requires full transcoding. Also, using very small tiles decreases the compression efficiency of JPEG 2000. Another approach would be to parse the code stream and only send packets of data containing information about the ROI(s). Part 9 of the JPEG 2000 standard will contain the protocol necessary to achieve this without full transcoding, requiring the server only to remove the irrelevant packets and to add the description of the remaining content. In another scenario, the client may want the image at full resolution but wishes to receive only a specified number of bytes. In this case one could use information about ROIs and their importance to allocate more bits to tiles or data packets pertaining to the ROI(s). It is also possible, based on JPEG 2000 Part 1, to encode the image with better quality for the ROI(s) and to put the information pertaining to the ROI(s) first in the codestream [39]-[41].

Concluding Remarks

In this article, we have highlighted a number of methods used to achieve UMA and discussed the types of metadata that support this. This overview is bound to be incomplete, especially since research in this area is ongoing, spurred on by the development of standards like MPEG-4, MPEG-7, MPEG-21, and JPEG 2000. Existing and new techniques in the UMA area can perhaps be classified along a number of axes, such as the following.

▲ Does the technique provide a method for selection or one for adaptation? For example, video transcoding is mostly a pure adaptation technique, while summarization and scalable coding can perhaps be seen as either selection (from parts of the content) or adaptation.

▲ Is the technique content based (manipulates the data based on semantic features, e.g., the importance of certain objects in an image, or of particular words in a speech signal) or noncontent based (manipulates the data based on features that can be detected immediately, e.g., the

Authorized licensed use limited to: Florida Atlantic University. Downloaded on May 08,2025 at 22:15:44 UTC from IEEE Xplore. Restrictions apply.

Amazon/DivX
Exhibit 1017
Page 11

coding format, or the frame rate)? Video transcoding is usually noncontent based, while summarization is usually content based. Recently, combinations of these two approaches have started appearing in literature.

▲ Is the technique standards based? This can be a significant factor in influencing whether a technique will be used in practical applications.

▲ Is the technique suitable for real-time applications? Naturally, real-time operation presents additional constraints as well as challenges. However, real-time techniques offer a desirable feature in many applications.

▲ What is the nature of the media data, i.e., is it audio, video, images, or a combination? Of these, video data is the most demanding in terms of delay and bandwidth or storage requirements.

▲ What is the location where the technique is to be applied primarily, i.e., is it in the server, in the client terminal, or somewhere in the network?

▲ Is the application push (e.g., broadcast TV) or pull (e.g., World Wide Web) oriented?

In general, the use of the various metadata tools to achieve UMA in real applications depends on many factors and constraints. Furthermore, a number of new metadata standards are being developed that may play a significant role in industrial applications. Among the bodies developing metadata specifications supporting UMA are the W3C, the TV-Anytime Forum, and MPEG. The TV-Anytime Forum [42] is targeting AV services based on high volume digital storage in consumer platforms such as PVRs. Its metadata specification includes tools for summarization and personalization. MPEG is currently developing a broader specification in the area of description of usage context as part of the MPEG-21 standard [43]. Finally, it should be noted that further standardization to support delivery of metadata is currently ongoing in various international standards groups such as MPEG and the TV-Anytime Forum.

Peter van Beek received the M.Sc.Eng. and Ph.D. degrees in electrical engineering from the Delft University of Technology, The Netherlands, in 1990 and 1995, respectively. From 1996 to 1998, he was a research associate with the Department of Electrical Engineering at the University of Rochester, Rochester, New York. In 1998, he joined Sharp Laboratories of America, Camas, Washington, where he is currently a principal researcher. He has contributed to the development of the MPEG-4 and MPEG-7 standards, and was active in the TV-Anytime Forum. He was co-editor of the Multimedia Description Schemes part of MPEG-7. His research interests are in the areas of image and video processing, multimedia compression, multimedia management, and networked video.

John R. Smith is manager of the Pervasive Media Management Group at IBM T.J. Watson Research Center, where he leads a research team exploring techniques for multimedia content management. He is currently chair of the MPEG Multimedia Description Schemes (MDS) group and serves as co-project editor for MPEG-7 Multimedia Description Schemes. He received his M.Phil. and Ph.D. degrees in electrical engineering from Columbia University in 1994 and 1997, respectively. His research interests include multimedia databases, multimedia content analysis, compression, indexing, and retrieval. He is an adjunct professor at Columbia University and a member of IEEE.

Touradj Ebrahimi received M.Sc. and Ph.D. degrees in electrical engineering from the Swiss Federal Institute of Technology, Lausanne (EPFL), in 1989 and 1992, respectively. In 1993, he was a research engineer at the Corporate Research Laboratories of Sony Corporation, Tokyo. In 1994, he was as a researcher at AT&T Bell Laboratories. He is currently a titular professor at the Signal Processing Institute of the School of Engineering at EPFL, where he is involved in research and teaching for multimedia information processing and coding. In 2002, he founded Emitall, a research and development company in electronic media innovations. He was the recipient of the IEEE and Swiss national ASE award in 1989 and the winner of the first prize for the best paper appearing in *IEEE Transactions on Consumer Electronics* in 2001. In 2001 and 2002, he received two ISO awards for contributions to MPEG-4 and JPEG 2000 standards. He is author or co-author of over 100 scientific publications and holds a dozen patents.

Teruhiko Suzuki received his B.S. and M.S. degrees in physics in 1990 and 1992, respectively, from Tokyo Institute of Technology. He is currently an assistant manager of Digital System Development Division, Silicon & Software Architecture Center at Sony Corporation. From 1999 to 2000, he was also a visiting researcher at Visual Computing Laboratory at University of California at San Diego. His research interests include image/video compression, image/video processing, and multimedia contents delivery. He participated to MPEG standardization from 1995 and contributed to the standardization of MPEG-2, MPEG-4 and MPEG-7.

Joel Askelof joined Ericsson Research in 1998. He works in still-image compression, universal multimedia access, and augmented reality. He has been involved in the standardization of JPEG2000 since 1998, both as co-editor of JPEG2000 part 5 (reference software) and currently as the head of the Swedish delegation. He holds an M.S. in engineering physics from Uppsala University.

References

[1] R. Mohan, J.R. Smith, and C-S. Li, "Adapting multimedia Internet content for universal access," *IEEE Trans. Multimedia,* vol. 1, pp. 104-114, Mar. 1999.

[2] *Information Technology—Coding of Audio-Visual Objects—Part 2: Visual*, ISO/IEC 14496-2, 1999.

Authorized licensed use limited to: Florida Atlantic University. Downloaded on May 08,2025 at 22:15:44 UTC from IEEE Xplore. Restrictions apply.

Amazon/DivX
Exhibit 1017
Page 12

[3] F. Pereira and T. Ebrahimi, Eds., *The MPEG-4 Book*. Englewood Cliffs, NJ: Prentice-Hall (IMSC Press Series), 2002.

[4] *Information Technology—Multimedia Content Description Interface—Part 5: Multimedia Description Schemes,* ISO/IEC 15938-5, 2002.

[5] B.S. Manjunath, P. Salembier, and T. Sikora, Eds., *Introduction to MPEG-7: Multimedia Content Description Interface*. Chichester, West Sussex, U.K.: Wiley, 2002.

[6] *World Wide Web Consortium (W3C), Extensible Markup Language (XML)*. Available: http://www.w3c.org/XML/

[7] *Special Issue on Personalization and Privacy, IEEE Internet Comput.,* vol. 5, pp. 29-62, Nov./Dec. 2001.

[8] N. Good, J.B. Schafer, J.A. Konstan, A. Borchers, B. Sarwar, J. Herlocker, and J. Riedl, "Combining collaborative filtering with personal agents for better recommendations," in *Proc. Conf. Am. Assoc. Artificial Intelligence (AAAI-99),* Orlando, FL, July 1999, pp. 439-446.

[9] A.M. Ferman, J.H. Errico, P. van Beek, and M.I. Sezan, "Content-based filtering and personalization using structured metadata," in *Proc. 2nd ACM/IEEE-CS Joint Conf. Digital Libraries (JCDL 2002),* Portland, OR, July 2002, p. 393.

[10] K. Kurapati and S. Gutta, "Instant personalization via clustering TV viewing patterns," in *Proc. IASTED Int. Conf. Artificial Intelligence & Soft Computing (ASC 2002),* Banff, Canada, July 2002

[11] J.R. Smith, R. Mohan, and C-S. Li, "Scalable multimedia delivery for pervasive computing," in *Proc. ACM Multimedia,* Orlando, FL, Nov. 1999, pp. 131-140.

[12] Y. Wang, Z. Liu, and J.-C. Huang, "Multimedia content analysis," *IEEE Signal Processing Mag.,* vol. 17, pp. 12-36, Nov. 2000.

[13] N. Dimitrova, H. Zhang, B. Shahraray, I. Sezan, T. Huang, and A. Zakhor, "Applications of video content analysis and retrieval," *IEEE Multimedia,* vol. 9, pp. 42-55, July-Sept. 2002.

[14] M.T. Maybury and A.E. Merlino, "Multimedia summaries of broadcast news," in *Proc. 1997 IASTED Int. Conf. Intelligent Information Syst.,* Grand Bahama Island, Bahamas, 1997, pp. 442-449.

[15] B. Li, J. Errico, H. Pan, and I. Sezan, "Bridging the semantic gap in sports," in *Proc. SPIE Conf. Storage and Retrieval for Media Databases 2003,* Santa Clara, CA, Jan. 2003.

[16] M.G. Christel, M.A. Smith, C.R. Taylor, and D.B. Winkler, "Evolving video skims into useful multimedia abstractions," in *Proc. ACM Computer-Human Interface Conf. 1998,* Los Angeles, CA, pp. 117-178.

[17] R. Lienhart, "Dynamic video summarization of home video," in *Proc. SPIE, vol. 3972, Storage and Retrieval for Media Databases 2000,* San Jose, CA, 2000, pp. 378-391.

[18] S.-F. Chang, D. Zhong, and R. Kumar, "Real-time content-based adaptive streaming of sports videos," in *Proc. IEEE Workshop Content-Based Access to Image and Video Libraries,* Maui, HI, Dec. 2001, pp. 139-146.

[19] H. Wallin, C. Christopoulos, and F. Furesjö, "Robust parametric motion estimation for image mosaicing in the MPEG-7 standard," in *Proc. IEEE Int. Conf. Image Processing (ICIP 2001),* Thessaloniki, Greece, Oct. 7-10, 2001, pp. 961-964.

[20] H.-J. Zhang, J. Wu, D. Zhong, and S.W. Smoliar, "An integrated system for content based video retrieval and browsing," *Pattern Recognit.,* vol. 30, no. 4, pp. 643–658, 1997.

[21] R. Zabih, J. Miller, and K. Mai, "Feature-based algorithms for detecting and classifying scene breaks," in *Proc. ACM Multimedia,* San Francisco, CA, Nov. 1993, pp. 189-200.

[22] P. Bouthemy, M. Gelgon, and F. Ganansia, "A unified approach to shot change detection and camera motion characterization," *IEEE Trans. Circuits Syst. Video Technol.,* vol. 9, pp. 1030–1044, Oct. 1999.

[23] N. Vasconcelos and A. Lippman, "A Bayesian video modeling framework for shot segmentation and content characterization," in *Proc. IEEE Workshop Content-Based Access of Image and Video Libraries*, San Juan, Puerto Rico, 1997, pp. 59–66.

[24] Y. Abdeljaoued, T. Ebrahimi, C. Christopoulos, and I. Mas Ivars, "A new algorithm for shot boundary detection," in *Proc. 10th European Signal Processing Conf. (EUSIPCO 2000),* vol. I, Tampere, Finland, Sept. 5-8, 2000, pp. 151-154.

[25] W. Wolf, "Key frame selection by motion analysis," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing,* vol. 2, Atlanta, GA, 1996, pp. 1228–1231.

[26] J.R. Smith, "Universal multimedia access," in *Proc. SPIE Multimedia Systems and Applications IV*, vol. 4209, Nov. 2000.

[27] A. Vetro, C. Christopoulos, and H. Sun, "Video transcoding architectures and techniques: An Overview," *IEEE Signal Processing Mag.,* vol. 20, pp. 8-29, Mar. 2003.

[28] T. Shanableh and M. Ghanbari, "Heterogeneous video transcoding into lower spatio-temporal resolutions and different encoding formats," *IEEE Trans. Multimedia*, vol. 2, pp. 101-110, June 2000.

[29] J.R. Smith and V.R. Chillakuru, "An application-based perspective on Universal Multimedia Access using MPEG-7," in *Proc. SPIE Multimedia Systems and Applications V*, vol. 4518, Aug. 2001, pp. 74-83.

[30] P.M. Kuhn and T. Suzuki, "MPEG-7 metadata for video transcoding: Motion and difficulty hints," in *Proc. SPIE, Storage and Retrieval for Media Database 2001*, vol. 4315, 2001, pp. 352-361.

[31] T. Suzuki and P.M. Kuhn, "MPEG-7 metadata for segment based video coding," in *Proc. Picture Coding Symp. 2001,* Seoul, Korea, pp. 25-28.

[32] P.M. Kuhn, T. Suzuki, and A. Vetro, "MPEG-7 transcoding hints for reduced complexity and improved quality," in *Proc. Int. Packet Video Workshop*, Kyongju, Korea, 2001, pp. 276-285.

[33] T. Chiang and Y.-Q. Zhang, "A new rate control scheme using quadratic rate-distortion modeling," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 7, pp. 246-250, Feb. 1997.

[34] A. Vetro, H. Sun, and Y. Wang, "MPEG-4 rate control for multiple video objects," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 9, pp. 186-199, Feb. 1999.

[35] A. Vetro, H. Sun, and Y. Wang, "Object based transcoding for scalable quality of service," in *Proc. IEEE Int. Symp. Circuit Syst.,* Geneva, Switzerland, vol. 4, May 2000, pp. 17-20.

[36] *Information Technology—JPEG 2000 Image Coding System,* ISO/IEC International Standard 15444-1, ITU Recommendation T.800, 2000.

[37] C. Christopoulos, A.N. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: An overview," *IEEE Trans. Consumer Electron.*, vol. 46, no. 4, pp. 1103-1127, 2000.

[38] *Information Technology—JPEG 2000 Image Coding System: Part 2 Extensions*, ISO/IEC Final Draft International Standard 15444-2, ITU Recommendation T.801, Aug. 2001.

[39] C. Christopoulos, J. Askelof, and M. Larsson, "Efficient methods for encoding regions of interest in the upcoming JPEG2000 still image coding standard," *IEEE Signal Processing Lett.*, vol. 7, pp. 247-249, Sept. 2000.

[40] J. Askelof, M.L. Carlander, and C. Christopoulos, "Region of interest coding in JPEG 2000," *Signal Process. Image Commun.,* vol. 17, pp. 105-111, 2002.

[41] R. Grosbois, D. Santa-Cruz, and T. Ebrahimi, "New approach to JPEG 2000 compliant region of interest coding," in *Proc. SPIE, Applications of Digital Image Processing XXIV*, vol. 4472, Dec. 2001, pp. 95-104.

[42] *TV-Anytime Forum*. Available: http://www.tv-anytime.org

[43] J. Bormans, J. Gelissen, and A. Perkis, "MPEG-21: The 21st century multimedia framework," *IEEE Signal Processing Mag.*, vol. 20, pp. 53-62, Mar. 2003.

Authorized licensed use limited to: Florida Atlantic University. Downloaded on May 08,2025 at 22:15:44 UTC from IEEE Xplore. Restrictions apply.

Amazon/DivX
Exhibit 1017
Page 13