VIDEO SKIMS: TAXONOMIES AND AN OPTIMAL GENERATION FRAMEWORK

Hari Sundaram *Shih-Fu Chang*
Dept. Of Electrical Engineering, Columbia University,
New York, New York 10027.
Email: {sundaram, sfchang}@ee.columbia.edu

ABSTRACT

This paper presents a new conceptual framework for summarization that considers the relationship between entities, device properties and user information needs. We summarize using a skim — an audio-visual clip that is a drastically condensed version of the original video. An entity is defined to be a sequence of elements that are related to each other by a certain property. In this paper we discuss the causes, the different entity types and also present a skim taxonomy. Each entity is associated with a utility. The skim is generated by a constrained utility maximization over those entity-utilities that satisfy the user information needs as well as the device rendering capabilities. We construct an optimal skim within this framework that retains a particular subset of entities. These entities have been chosen since they can be automatically computed in a robust manner. The user studies show that the optimal skims perform well in a statistically significant sense, at compression rates as high as 90%.

1. INTRODUCTION

In this paper we present a new conceptual formulation for the problem of skim generation. Then we show application of this framework for a specific type of audio-visual skim. This conceptual formulation is needed for several reasons. While there has been much prior work in both image and video based summarization schemes [2][6][7][12], they all focus very narrowly on creating summaries that preserve the semantics of the original. However, this summary generation is static, in the sense that neither the user information needs nor are the abilities of the device are taken into account when generating the summary. Our framework considers the audio-visual skim generation in terms of the following: (a) entities defined on audio-visual data, at different levels of abstraction, (b) utilities assigned to each entity, (c) the user information needs, (d) the device on which the skim is to be rendered and (e) a skim generation mechanism that maximizes the utility of the entities considered.

We define an entity to be a sequence of events related to each other via a certain property. Entities can be caused by static (physical events, content producer) and dynamic (prior knowledge and user expectations) factors. Entities can be of different types: (a) temporal, (b) syntactical, (c) semantic and due to the (d) conventions of the domain. Each entity is associated with a different utility function, that is altered depending on whether the entity is dropped or transformed.

A skim is short video clip containing the entities that satisfy the user's information needs (or tasks) as well the capabilities of the device on which it is being rendered. In this paper we present a taxonomy of skims: (a) semantic, (b) affect, (c) events and (d) information centric. These skims differ on their computability and in their usefulness to the viewer. In this work, we have focused on information centric skims since they are computable.

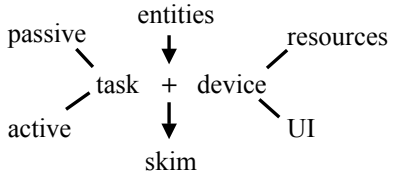

Figure 1: a skim depends on the user's information needs and the device.

We detect the basic video entity and then detect the syntactical entities; we then show how we can impose duration constraints on these entities stemming from visual constraints and syntax analysis. We detect the basic audio entities (segments and significant phrases) using SVM classifiers. Finally we construct synchronous entities via tied multimedia segments. The information-centric skims are generated in utility maximization framework, that maximizes the utility of the entities requested by the user. The user studies indicate that the optimal skims are better in a statistically significant sense at high compression rates (80% ~ 90%).

The rest of this paper is organized as follows. We begin by defining entities and discuss in some detail the causes and the different types of entities. Then, we present a skim taxonomy and in section 3, we present a taxonomy of skims and in section 4, we present our specific skim generation mechanism. Finally we summarize of our work a outline directions of future work.

2. WHAT IS AN ENTITY?

An *entity* is a sequence of elements that are related to each other via a certain property. For example, a shot is a video entity since it is a sequence of video-frames that is consistent with respect to color / motion. Entities exist at many levels — shots, syntactical elements, semantic elements as well as due to the content producer. This definition is easily extended in a hierarchical fashion (e.g. a dialog sequence is an entity, whose constituents are shots that share a topological property — adjacent shots differ, while every second shot is alike.). We now discuss the causes of relationships between modalities and also discuss a few of the different types of relations.

2.1 Causes

Entities defined via relations on elements across different modalities can occur due to at least four different factors: (a) the relationship is caused by a physical event (e.g. the firing of a gun). (b) due to the creator of the content (c) the prior knowledge of the viewer of the physical events shown as well as the conventions of the domain and (d) the expectations of the

viewer. We term the first two relations to be *static* and the last two as *dynamic*. This is because the relations defined in the first two points deal with relationships that have been inserted statically by the content producer. However, in the last two cases, regardless of the static relations in the video, the viewer's prior knowledge and expectations change the relations (or create new ones) amongst the entities.

2.2 Types

In this section, we discuss some of the different types of relations that exist within and across modalities.

Temporal: These relations (which can occur within and across modalities) can either be (a) synchronous (e.g. lip movement accompanied by speech) or (b) causal (e.g. sounds of a doorbell followed by a visual of the door opening).

Syntactical: A topological relationship between elements — this can be amongst and within modalities. Examples of topological relations include the dialog, footsteps etc.

Semantic: These are high-level relations that exist due to the content producer and due to the prior knowledge of the viewer.

Film conventions: There are relations that exist in film that arise out of the director's desire to create a certain affect (i.e. emotional response). We list a few of them below:

1. Graphic: In this form, the director will ensure that objects appear in certain locations of the video, in a periodic manner (e.g. location of the principals in a dialog).
2. Rhythmic: The director can induce a certain rhythm to the shot sequence by carefully selecting the durations of the constituent shots.
3. Spatial: In the absence of an establishing shot, the audience will spatially connect the shots in the sequence (i.e. the audience will assume that the shots are in the same physical location [1]).

Note that different domains of production will contain different rules of syntax. For example, in soccer, the content producer will return to a zoom-out view (thus showing a global game state), after showing close ups and medium-shots.

3. SKIMS AND ENTITIES

In this section, we first discuss factors affecting skims and then we shall formulate the skim in terms of the entities found in the data, the user needs and the device on which the skim is to be rendered.

3.1 Factors that affect skims

There are at least two factors that affect the skim generation algorithm — the task of the user and the device constraints. We divide up tasks into two broad categories: active and passive tasks. A task is defined to be active when the user requires certain information to be present in the final summary (e.g. "find me all videos that contain Colin Powell."). In a passive task, the user does not have anything specific in mind, and is more interested in consuming the information. Examples include previews in a set-top box environment, browsing in a video digital library. The device on which the skim is to be rendered affects the skim in at least two ways: the nature of the user interface and the device constraints. The UI can be complex (e.g. the PC), medium (e.g. a palm pilot) and simple (e.g. a cell phone). The UI affects the resolution of the skim, and also influences the kinds of tasks that the user has in mind (e.g. it is difficult to input a query on a cell phone). The computational resources available on the device — cpu speed, memory, bandwidth, availability of a video rendering device. They all affect the skim in the following ways: the resolution of the skim, as well as the decision to include video in the skim.

3.2 Entities, tasks and the device

A skim can be viewed as a time compressed audio-visual clip that contains those entities (from the original video) that best matches the information needs (or tasks) of the user and the capabilities of the device on which it is being rendered (see figure 1). For example, consider the problem of generating a summary for a baseball game that contains MPEG-7 metadata. Then, independent of the viewer, the video data contains entities that arise due to physical events, the semantics of the game as well as due to the content producer.

Now, let us assume that we have a viewer who is interested in watching a summary of a Yankees game highlighting the performance of the short-stop Derek Jeter. Then, the summary must contain all the entities relevant to the user's information needs — (a) events that change the game score (b) the entities that include this player (e.g. hitting, running around bases, fielding the ball and throwing someone out). Now, if the device in question cannot render video (but only audio and still images), then many of the entities that the user has requested, cannot be fulfilled (e.g. causal entities such as hit followed by speech).

3.3 Taxonomy: Skims come in different flavors

In this section we attempt to identify some of the different skim forms and for each skim type, we also discuss its computability and usefulness to the viewer.

Semantic: Here, we attempt to preserve the key semantics in the data. These could be specified by the user (in the form of a query — "What did George Bush do today?"), or the content producer, who may specify (via MPEG-7 metatags) the content to be retained in the skim. This skim type is the closest to what the user wants, but in general, it is very difficult to compute the entities required to satisfy the user needs. However, if the content producer inserts MPEG-7 metatags, generating the skim may be feasible.

Affect: In this form, the user is interested in a skim that retains the "mood" or the affect [1] generated by the content producer. As an example, the heightened pace (arising from the sequence of fast cuts) during an action sequence can be maintained by preserving the film rhythm. This is useful for example, in creating movie previews. The viewer may be interested in knowing if the film is exciting, sad etc. While there has been some work done in computing affects [1], this is a challenging area of research.

Events: An event refers to the change of state (or the change in property) of an entity. This skim will contain a subset of all the events in the domain. For example, the user could specify that in the skim of a particular soccer game, only those events pertaining to a particular soccer player be retained. This is particularly useful in constrained domains (e.g. baseball videos). In constrained domains, it will be possible to construct high-

performance event detectors. Note that while events can have specific semantics associated with them, a semantic skim attempts to answer higher level questions. The answers will comprise as events well as non-events.

Information centric: This skim will attempt to parse the discourse structure of the speech in the video, and determine the most significant audio segments using prosody analysis. While this skim type can be computed automatically [11], detecting significant phrases in noisy environments is still very challenging task. The assumption that speech conveys the maximum information, may not be true — the important segments could lie in other forms such as music or the environmental sounds.

3.4 Entities and utilities

The two basic entities that we will deal with are the video shot and the audio segment. We shall also attempt to preserve those entities related to syntactical elements (e.g. the dialog, the progressive phrase [10]). Note that by preserving the dialog syntactical element, we also implicitly preserve graphic and spatial entities. We shall explicitly attempt to preserve the film rhythm (this is a form of affect) and we shall also deal with an entity defined by two overlapping audio segments. Utilities are determined by the nature of the entities, the user tasks, and the devices. Such utilities will be affected (scaled) when the associated entity is altered, dropped, trimmed, or transformed. The scaling function can be established based on a user study or via an empirical analysis of the entities.

4. SKIM GENERATION

In this section, we discuss the skim generation procedure in terms of an utility maximization framework. We begin by discussing the specific goals of this work; then we shall present an overview of our work on visual analysis and audio analysis [10][11]. This analysis helps us determine the utility function corresponding to the basic entities (i.e. video shots and audio segments). Finally, we present our optimization framework and experiments.

4.1 Goals

The goal of this work is the automatic generation of audio-visual skims for *passive* tasks, that summarize the video. This work focuses on creating information centric skims. This skim type has been chosen since it is easily computable. We make the following assumptions:

1. We do *not* know the semantics of the original.
2. The data is not a raw stream (e.g. home videos), but is the result of an editing process (e.g. films, news).

Since we work on passive tasks, the information needs of the user are a priori unknown. A decision to detect certain set of predefined events will induce a bias in the skim, thereby conflicting with the assumption that the user needs are unknown.

4.2 Visual complexity and film syntax

We assume that the basic video entity (i.e. the video shot) has already been detected via a shot detection algorithm. The entities corresponding to the elements of syntax are easily determined via a robust detection algorithm [9]. Then, given these two entities, there are constraints on these entities that arise from the following two questions:

1. What is the minimum duration required to comprehend a shot?
2. How does the syntactical structure of the video affect our comprehension of the video?

In order to answer these two questions we do the following: (a) compute a measure of visual complexity of a shot and then relate it to comprehension time via a psychological experiment. (b) use film theory to detect the interesting structures and the minimum number of structural elements to understand each structural entity. The details can be found in [10].

4.3 Audio analysis

In this section, we show how we compute the two audio entities: the basic audio segment and segment beginnings (SBEG's) [5]. We build a tree-structured classifier to classify each frame (100ms) into four generic classes: silence, clean speech, noisy speech and music / environmental sounds. Silence frames are first separated from the rest of the audio stream using an adaptive threshold on the energy. We train two SVM classifiers which are then used in cascade: the remaining frames are separated into speech vs. non-speech (music or environmental sounds); and the speech class is further classified as clean and noisy speech. We then apply a modified Viterbi decoding algorithm [11] to smooth the sequence of frame labels.

SBEG's are important as they serve as the introduction of new topic in the discourse. There has been much work in the computational linguistics community [5] to determine the acoustic correlates of the prosody in speech. Typically, SBEG's have a preceding pause that is significantly longer than for other phrases, higher initial pitch values (mean, variance), and smaller pauses that end the phrase than for other phrases [5]. Once we've extracted the acoustic features per candidate phrase, the phrase is then classified using a SVM classifier [11].

4.4 Synchronous entities

Synchronous entities are constructed using the idea of tied multimedia segments. A multimedia segment is said to be fully *tied* if the corresponding audio and video segments begin and end synchronously, and in addition are *uncompressed.* Tied multimedia segments are associated those speech segments that contain significant phrases. Entities with only one end synchronous are deemed as partially synchronous entities.

4.5 The optimization framework

We use a constrained utility framework to create audio-visual skims [11]. There are three key components to our framework: (a) a utility model for video shots and audio segments (b) constraints stemming from audio-visual minimum duration considerations and from visual syntax constraints and (c) a constraint relaxation strategy to ensure a feasible solution. We shall only summarize our work here, the details can be found in [11].

4.5.1 Utility functions

In order to determine the skim duration, we need to measure the comprehensibility of a video shot and a audio segment as a function of its duration. The non-negative utility function of a video shot $S(t, c)$, where t is the duration of the shot and c is its complexity, is modeled to be a bounded, differentiable, separable, concave function:

$$S(t,c) = \beta c(1-c) \bullet (1 - \exp(-\alpha t)). \quad <1>$$

The utility function for the sequence of shots is the sum of the utilities of the individual shots:

$$U_v(\vec{t}_v, \vec{c}, \phi_v) = \frac{1}{N_{\phi,v}} \left(\sum_{i:\phi_v(i)=1} S(t_{i,v}, c_i) - \sum_{j:\phi_v(j)=0} P(t_{p,j}) \right) \quad <2>$$

where, $\vec{t}_v : t_0, t_1 \ldots t_N$ and $\vec{c} : c_0, c_1 \ldots c_N$. represent the durations and complexities of the shot sequence and where $P(t_{p,j})$ represents a negative shot dropping utility. The utility function for an audio segment is described in a similar fashion [11].

4.5.2 The film rhythm entity penalty function

We assign a negative utility (i.e. a positive penalty) to the "film rhythm." This function computes the penalty associated with deviation from the original affect entity. We define the rhythm penalty function as follows:

$$R(\vec{t}, \vec{t}_o, \phi) = \sum_{i:\phi(i)=1} f_{o,i} \ln\left(\frac{f_{o,i}}{f_i}\right), \quad f_{o,i} = \frac{t_{o,i}}{\sum_{i:\phi(i)=1} t_{o,i}}, f_i = \frac{t_i}{\sum_{i:\phi(i)=1} t_i}. \quad <3>$$

where R is the penalty function, and where, t_i is the duration of the i^{th} shot in the current sequence, while $t_{o,i}$ is the duration of the i^{th} shot in the original sequence. The ratios are recalculated with respect to only those shots that are not dropped, since the rhythm will change when we drop the shots. We discuss the negative utility associated with the audio slack entity in [11]. We do not explicitly model the utility of the elements of syntax or the entities due to synchronous elements. The utilities are *implicitly* maximized by the search strategy for the optimal skim. This strategy is biased in favor of retaining as many elements of syntax as possible, as well as for maximizing the number of synchronous entities in the skim.

4.5.3 The search strategy

We focus on the generation of passive information centric summaries that have maximum coherence. Since we deem the speech segments to contain the maximum information, we achieve this goal by biasing the audio utility functions in favor of the clean speech class. In order to ensure that the skim appears coherent, we do two things: (a) ensure that the principles of visual syntax are not violated and (b) have maximal number of synchronous entities. These entities ensure synchrony between the audio and the video segments. The target skim duration is met by successively relaxing the synchronous constraints. Relaxing the synchronization constraints has two effects: (a) the corresponding audio and video segments are no longer synchronized (b) they can be compressed and if necessary dropped. The details of the search strategy and the mathematical framework can be found in [11].

4.6 Experimental results

The scenes used for creating the skims were from three films: *Blade Runner (bla), Bombay (bom), Farewell my Concubine (far).* We conducted experiments with three different skim generation mechanisms: the algorithm presented in this paper, a proportional skim and a semi-optimal skim with proportional video and the optimal audio segments from our algorithm. We created three skims (one from each algorithm) at each of the three different compression rates (90%, 80%, and 50%), thus creating nine skims. We conducted a pilot user study with twelve graduate students and the test results [11] indicate the optimal skim is perceptually better in a statistically significant sense at the higher rates (90%, 80%).

5. CONCLUSIONS

In this paper, we have discussed a new conceptual framework that examines the roles of entities in the data, the user task, the device and the UI in the process of skim generation. We defined an entity to be a sequence of elements sharing a particular property. We then looked at the different entity types and their causes. Each entity was associated with a utility. We constructed a skim taxonomy and discussed the computability and usefulness of each skim type. Finally, we generated a specific skim using a subset of the entity types discussed in this paper, using a constrained utility maximization approach, to demonstrate the applicability of the approach.

We believe that the conceptual framework discussed here is powerful and easily adapts to different skim generation requirements. However, there are many interesting avenues of research. Firstly, given a particular domain, we need a principled methodology to determine the utility of the various entities and how they relate to each other. Another issue is that of cross-modal entity creation to efficiently summarize the content. For example, if the user is interested in the location of the video, by creating a text overlay that indicates the location, we will be able to reduce the duration of the entities that *show* the location.

6. REFERENCES

[1] B. Adams et. al. *Automated Film Rhythm Extraction for Scene Analysis,* Proc. ICME 2001, Aug. 2001, Japan.

[2] M.G. Christel et. al *Evolving Video Skims into Useful Multimedia Abstractions,* ACM CHI '98, pp. 171-78, Los Angeles, CA, Apr. 1998.

[3] T.M. Cover, J.A. Thomas *Elements of Information Theory,* 1991, John Wiley and Sons.

[4] J. Feldman *Minimization of Boolean complexity in human concept learning,* Nature, pp. 630-633, vol. 407, Oct. 2000.

[5] J. Hirschberg, B. Groz *Some Intonational Characteristics of Discourse Structure,* Proc. ICSLP 1992.

[6] L. He et. al. *Auto-Summarization of Audio-Video Presentations,* ACM MM '99, Orlando FL, Nov. 1999.

[7] T. S-Mahmood, D. Ponceleon, *Learning video browsing behavior and its application in the generation of video previews,* Proc. ACM Multimedia 2001, pp. 119 - 128, Ottawa, Canada, Oct. 2001.

[8] S. Sharff *The Elements of Cinema: Towards a Theory of Cinesthetic Impact,* 1982, Columbia University Press.

[9] H. Sundaram, Shih-Fu Chang *Determining Computable Scenes in Films and their Structures using Audio-Visual Memory Models,* ACM Multimedia 2000, pp. 95-104, Los Angeles, CA, Nov. 2000.

[10] H. Sundaram, Shih-Fu Chang, *Constrained Utility Maximization for generating Visual Skims,* IEEE Workshop on Content-based Access of Image and Video Libraries (CBAIVL-2001) Dec. 2001 Kauai, HI USA.

[11] H. Sundaram L. Xie Shih-Fu Chang *A framework work audio-visual skim generation.* Tech. Rep. # 2002-14, Columbia University, April 2002.

[12] S. Uchihashi et. al. *Video Manga: Generating Semantically Meaningful Video Summaries* Proc. ACM Multimedia '99, pp. 383-92, Orlando FL, Nov. 1999.