Generating Story Variants with Constrained Video Recombination*

Alberto Piacenza, Fabrizio Guerrini, Nicola Adami, Riccardo Leonardi
Department of Information Engineering, University of Brescia, Italy
{firstname.lastname}@ing.unibs.it

Jonathan Teutenberg, Julie Porteous, Marc Cavazza
School Of Computing, Teesside University, Middlesbrough, UK
{J.Teutenberg,J.Porteous, M.O.Cavazza}@tees.ac.uk

ABSTRACT

We present a novel approach to the automatic generation of filmic variants within an implemented Video-Based Storytelling (VBS) system that successfully integrates video segmentation with stochastically controlled re-ordering techniques and narrative generation via AI planning. We have introduced flexibility into the video recombination process by sequencing video shots in a way that maintains local video consistency and this is combined with exploitation of shot polysemy to enable shot reuse in a range of valid semantic contexts. Results of evaluations on output narratives using a shared set of video data show consistency in terms of local video sequences and global causality with no loss of generative power.

Categories and Subject Descriptors

I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding—*Video Analysis*

General Terms

Algorithms

Keywords

Interactive Storytelling, Logical Story Unit, Markov Chains, Narrative Modeling

1. INTRODUCTION

The original idea behind interactive cinema, since Činčera's *Kinoautomat*, was to allow user preferences to modify the unfolding of the narrative, whilst preserving the overall story world. However, the "branching narrative" approach to interactive films has eventually been abandoned due to the exponential cost attached to the shooting of filmic material for all alternative scenes, and the adverse effects on spectators' filmic experience of mandatory interaction at branching points [13]. Another limitation of branching video applied to cinema is that it depends largely on local decision points and is unable to make use of global filmic properties, falling short of core narrative principles which, since Aristotle, have emphasised the overall shape of dramatic action.

This double limitation of video content generation explains why most recent research in Interactive Storytelling (IS) has taken place using computer graphics [21] and more recently computer games engines [24], which provide a visual medium facilitating the dynamic generation of visual content, e.g. through built-in mechanisms for keyframed animation generating motion sequences that can still be split and recombined through scripting and other mechanisms. These systems have increasingly adopted the visual codes of film itself, through real-time camera control and simplified forms of real-time editing. Yet, despite recent progress in graphics rendering and wide-scale acceptance of 3D animation in film, the visual quality of video is still far superior to that of real-time generated graphics.

Current interest in storytelling applied to video is dominated by the storyfication or emergent storytelling paradigm [27]. Storyfication, which stems from a different perspective on narrative than the one traditionally associated to film, is about using temporal and semantic relationships to attribute meaning to a sequence of events, e.g. forming a life narrative from personal videos. Conversely, the challenge of Video-Based Storytelling (VBS) for films that we are addressing consists in generating alternative stories from the same baseline content whilst preserving its dramatic nature and global narrative properties, by transposing recent advances of automatic narrative generation back to the field of video. These correspond to different variants of a given film, each exhibiting different narrative properties, but each constituting a proper drama, albeit with a different course of action. This achievement would constitute an important step towards interactive films, but it makes it necessary to leverage the combinatorial properties of individual video segments beyond the simple reordering of short sequences.

Narrative generation techniques developed in IS support generation using global dramatic properties and contextual phenomena. This would make it possible to capitalise on the "Kuleshov effect" which explains how identical shots[1] featuring characters receive a different interpretation by the viewers depending on their context. However, this requires a semantics of individual video units compatible with the

*Area chair: Susanne Boll

MM'11, November 28–December 1, 2011, Scottsdale, Arizona, USA.

[1]Shots are sequences of continuous still images (frames) as filmed through a single, uninterrupted camera take [12].

global logic of AI-based narrative generation. Another important aspect for content recombination is to be able to derive flexible units of content from the baseline video material. This also overcomes the need to manually tag fixed individual units of content, which requires significant effort and may not be compatible with dynamic changes in context that are at the heart of the narrative variant philosophy.

Interactive Narratives rely on the ability to dynamically generate the sequence of narrative actions rather than following pre-defined branching points. Narrative generation thus plays a central role, since it supports the propagation of changes introduced by the user: either initial preferences that will trigger a wholly different story variant, or real-time interventions, which will alter the course of an unfolding story, leading to a "recomputation" of future story actions. Techniques underlying narrative generation, such as planning, operate by maintaining causal consistency (narrative actions being formalised through pre- and post-conditions) over the entire narrative, from the point in time where new information is provided until the state defined as the story end. Narrative generation can thus be applied at any point in time, and the frequency at which new information can be taken into account defines the sampling rate of interaction [24]. However, in the case of our first VBS prototype, there is still an additional overhead in the amount of processing required to automatically refine individual video units to be used in the subsequent generation of the remainder of the narrative. For that reason, we have focussed our examples on narrative generation, with user interaction limited to an initial parameterisation of story properties that will determine the overall nature of the story variant. Our underlying planner has previously demonstrated its potential for anytime interactivity when used with 3D graphics content [24].

Our technical solution, which we present in this paper, features in a small-scale (by filmic standards), yet fully-implemented and functional VBS system which we use as an illustration throughout the paper. The system uses Michael Radford's screen adaptation of Shakespeare's play *The Merchant of Venice* [25] as its baseline video. The paper is organised as follows: in Section 2 we provide the motivation for this work while relating to previously proposed IS systems. Section 3 introduces the key elements of our system's architecture. In Sections 4 and 5 the video processing technology needed to achieve VBS is described while narrative generation is discussed in Section 6. Finally, Section 7 presents early evaluation results, and we conclude by discussing future work and current limitations.

2. RELATED WORK AND MOTIVATION

The development of video-based storytelling is an active area of Multimedia systems research, and several prototypes based on the storyfication paradigm [27] have been introduced by Brooks [6], Aguierre Smith et al [28, 29], Cesar et al [8], Bocconi et al [4], Zsombori [32] and Shen et al [27]. These systems organise user-contributed content [8] into a narrative [27] or support the generation of personalised documentaries [6, 4, 32]; most of them aim at organising video content into a narrative format, rather than modifying a pre-existing narrative structure. One common feature of these systems is their commitment to semantic categorisation of basic video units, which is consistent with a bottom-up approach, aiming at producing semantic consistency from local principles. These semantic categories are often generated through manual tagging, which assumes, but is also consistent with, a static unit of content.

These early works adopt a discourse oriented perspective (see the importance of shots as units in [14], or rhetorical relations in [3]), without considering the global properties of the plot. Zsombori et al [32] developed authoring and delivery tools for interactive television as part of the NM2 project. Their approach relied on improvements to branching narrative techniques and didn't exploit AI techniques such as planning, hence their system is unable to reason about global narrative properties and maintain global causal consistency (a fact acknowledged by the authors [32]). Other approaches such as IDC [26] and AUTEUR [22] reason at the level of individual actions to output a restricted set of short videos and have made reference to some planning concepts, although mostly restricted to action description. Jung et al [18] considered the narrative properties of video content and emphasised the locality of dramatic actions but also recognised the need for editorial relations to support the narrative structure. It could be said that these systems included the elementary components of planning actions, but without connecting them to modern planning algorithms which operate on global rather than local constraints.

The challenge we are addressing is to reconcile this narrative approach with the visual quality that only video can provide. Our VBS system has different narrative objectives than the systems discussed above and hence is closer in its philosophy to those IS systems that use (top-down) plan-based narrative generation with 3D graphics. In IS, state-of-the-art plan-based narrative generators such as [24] are capable of generating complex narratives containing 40+ narrative actions and working consistently on the global aspects of an output narrative, which we see as a key factor in the induction of a narrative experience for authored media.

Our earlier approach [23] attempted to directly combine state-of-the-art plan-based narrative generation with video content. It used somewhat naive mappings between planning actions and video segments with the consequence that the output was restricted to variants that were simple rearrangements of the original input story, without fundamentally changing the original semantics of the actions. In other words, there was a mismatch between our semantic/action level and our narrative level: the two limitations we faced were i) the static nature of our basic video units and ii) the lack of sophistication of our semantic representation.

In this paper we shall describe how these limitations have been addressed as part of our new prototype. Our solution involved the development of a shared semantic representation that facilitated the conceptual integration of video processing and narrative generation based on AI planning techniques for story consistency. For instance, the automatic categorisation of emotional aspects of video can provide some form of baseline semantic analysis which is both relevant to narrative aspects and also to how users tend to consume filmic media [19]. This approach enables the generation of planning compatible actions as units of recombination in a way that was previously not possible, thus enabling the construction of completely new filmic variants.

3. SYSTEM OVERVIEW

An overview of the architecture of our system is shown in Fig. 1. The output is a system generated filmic variant of the baseline input movie or potentially a completely new

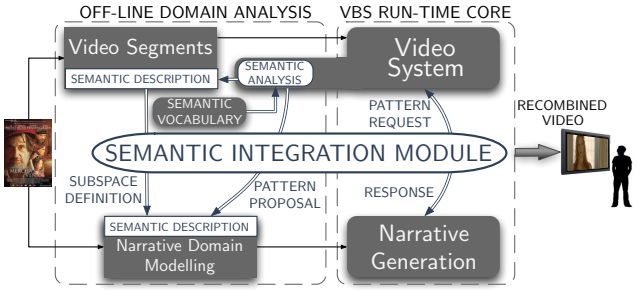


Figure 1: Architecture of the VBS system: a baseline video is analysed and a semantic subspace is created within the semantic integration module and then shared with the domain model; the integration module handles communication between the video system and the narrative generation modules to automatically produce novel filmic variants through video recombination.

story except for the locations and the actors involved (their role and interaction may be completely different from those of the original). The heart of the system is a video integration module which acts as the nexus between the analysis of the input video and the generation of narrative variants for output. The integration of video analysis and narrative generation is achieved via the use of a shared semantic representation which enables communication between the content automatically identified by the video system and the model of the narrative domain used for narrative generation.

During an initial phase of domain analysis, low-level features of the input baseline video are processed to determine basic contextual information (such as characters and locations) and subdivided into shots which form the building blocks for recombined video generation. Also during the domain analysis phase, a representation of the narrative domain is created which includes specification of key narrative actions to be defined for subsequent narrative generation. Then, at the end of the domain analysis phase, the identified shots and narrative actions are mapped to points in the shared semantic representation. These points are defined in terms of semantic features such as the presence of different characters and their mood and provide a common vocabulary for communication between the system components.

In the VBS core, the narrative generation module outputs story variants in response to user input. These variants (sequences of narrative actions) must be constructed in a way that ensures the generation of a consisent plot or preserves the consistency of a modified one. This can't be done without video analysis level reasoning about local causality and consistency and hence our approach integrates it with a joint though separate high level reasoning about global narrative properties. Each action in an output narrative can map to multiple points in the shared semantic subspace (see Section 4), so the video presentation of individual narrative actions isn't limited to any single segment of the baseline video (as it is with the naive mapping used in [23]); instead, appropriate video subparts for a narrative action can be selected from any video segments which map to relevant points for that action in the semantic subspace, while preserving fundamental consistency in the content. This is a powerful result since it provides a complete decoupling of the narrative model and the baseline video content which is only described by its semantics. Importantly, this enables the presentation of completely novel filmic variants.

Another innovative feature of the system is that the output of video processing is able to place constraints on the narrative generation. In particular, the list of semantic descriptors identified in the baseline video are used to create a fully shared semantic subspace, so the planner can avoid the actions mapping to semantics not present in the video.

The VBS system supports two different modalities for the creation of the recombined video, each reflecting the direction of the flow of semantic information in the core. The narrative-driven setup is done online and consists of the video system answering specific requests for desired content from the narrative side expressed using the semantic vocabulary. In the video system-driven setup, the latter tries to identify possible narrative actions by mixing video segments according to internal models that can be derived from the integration of consistent pieces of content (in terms of locations, involved characters, and similar models of interactions). The video clips and the associated semantic information are then proposed to the narrative generator, which will add them to its domain model if they are deemed adequate.

4. VIDEO MODELING

The baseline video is initially described in terms of the semantic vocabulary and modeled to allow effective content recombination. Subsection 4.1 details the automatic algorithms for preliminary temporal segmentation of the original movie into logical basic segments – the building blocks for recombined video. Subsection 4.2 introduces the intermediate level semantic vocabulary that is used to communicate between the narrative domain and the video system. Semantic modeling of video content, necessary for the video recombination process, is detailed in Subsection 4.3.

4.1 Video Segmentation

From a high level point of view, the movie can be separated into a succession of semantically consistent segments, also called *Logical Story Units* (or LSUs) [30], each one conveying a concept of the story being narrated. Alternatively, at the most basic level, a video can be divided into *shots*. Each LSU is therefore a sequence of temporally adjacent shots conveying a common concept in the context of the story. Both shots and LSUs represent fundamental entities with which the video system constructs new video sequences.

The video system first extracts the shots from the film by applying a shot boundaries detector [5] [20]. The algorithm used in this system is based on the classical twin comparison method, employed on statistical color intensity distributions of adjacent frames: an abrupt variation in the color distribution is interpreted as a shot boundary. Particular types of shot boundaries, e.g. dissolves, are also taken into account through a suitable dissolve model [1].

LSUs are then formed starting from the shot segmentation. In [30], a video is represented by a Scene Transition Graph (STG). The nodes of the graph are clusters of visually similar and temporally close shots and the edges represent the shot transitions. It is also shown that the STG can be decomposed into separated cyclic subgraphs through the removal of so-called cut-edges and that each subgraph identifies a LSU. This process is shown in Fig. 2.

In the current implementation, visual clustering is per-

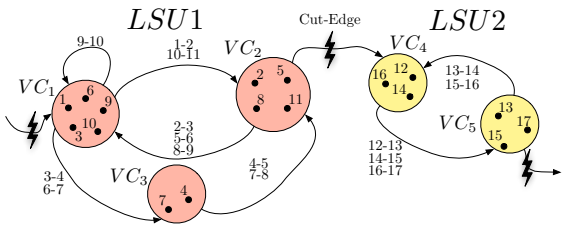


Figure 2: LSU segmentation using visual clusters of shots and temporal transitions. The clusters, VC_1–VC_5, are obtained through hierarchical clustering; points inside clusters represent individual shots. The numbers on the links refer to shot transitions.

formed by extracting a codebook of visual words by dividing shot keyframes in square blocks and then running a Tree-Structured Vector Quantization algorithm [16] to LUV color space values of the blocks. The codebook size is determined by controlling the distortion on the reconstructed keyframe. Then, a shot similarity measure is defined by averaging the distortion increase caused by representing each shot using the codebook of the other. Last, the final shot visual clusters are obtained through hierarchical clustering as in [2].

4.2 Intermediate Semantic Representation

The video system annotates the extracted shots using several semantic *tags* that describe intermediate level concepts specified in an agreed vocabulary. The tags allow the definition of a *semantic subspace* in which video content is described. In particular, each shot is described by a *semantic point* of the subspace, that is an instance of the tags describing the semantic point abstraction. We used the following tags:

- **Characters:** A list of characters present in the shot, specified by an anonymous *name* tag, e.g. A, B, $\emptyset$ or {A,B}, and the prevalent *mood* of the individual characters, taking one of three possible values: positive, negative or neutral.
- **Field:** A ternary value indicating the *field* of vision of the camera: close-up, medium or wide.
- **Environment:** Three binary values for the general shot environment (when applicable): *time* (day or night), *location* (indoor or outdoor), and *crowd* presence (present or not).

The selection of the tags is important both for the video synthesis and for the narrative generation. The former must have sufficient information on the shots to recombine them without losing consistency. In turn, narrative generation must be able to describe the actions from a more abstract level, but, at the same time, the tags can't delve into too much detail to avoid complicating the mapping between narrative actions and semantic description. In the light of these considerations, it is clear that, aside from the characters present in the shot, their prevalent mood is also necessary to accurately convey the high level meaning of the narrative action, e.g. positive mood for merry actions. All other tags are necessary for the coherence of the recombined content, by allowing mixing of shots that are set in the same context.

4.3 Semantic Modeling of Scenes

As illustrated in Fig. 2, each LSU can be modeled as a completely connected STG, where the nodes represent clusters of visually similar shots. In order to describe the LSU using the semantic vocabulary, the shots belonging to the LSU must be re-clustered, using their semantic description. The initial nodes in the STG may be split in subgraphs. The resulting new graph transforms the initial LSU to a Markov chain, that is referred to as a *Semantic Story Unit* (SSU). In an SSU, instead of using the visual clusters described in Subsection 4.1, we introduce the concept of *semantic clusters*, that are built grouping the shot of the given LSU according to their semantic tag values. The SSU entity introduced in this work joins the temporal LSU concept with the semantic description of the shots to obtain a model that captures the structural semantic behavior of the scenes.

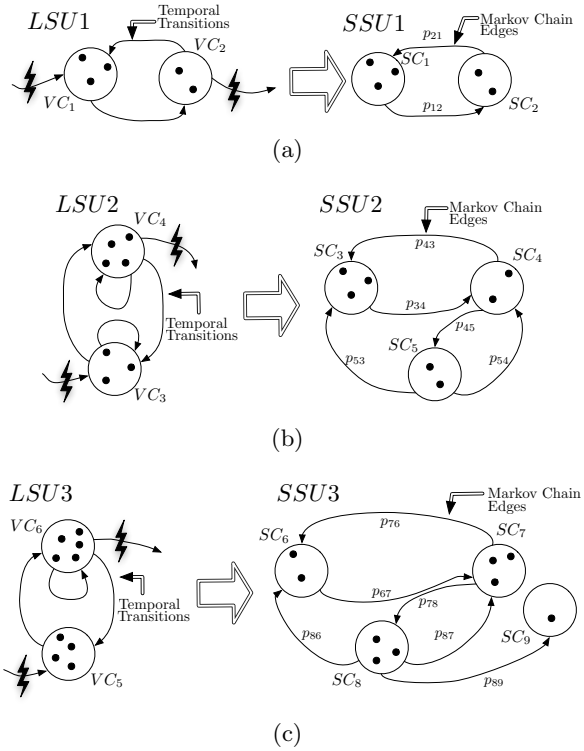


Figure 3: SSU generation in various VC-SC correspondence cases: (a) the visual clusters and the semantic clusters are perfectly matched, (b) one of the visual cluster has spawned two different semantic clusters, (c) additional cut-edge added.

The construction of an SSU is depicted in Fig. 3. In case (a), the visual clusters (VC) and the semantic clusters (SC) are perfectly matched. In case (b), instead, one visual cluster contains shots with different semantics and hence has spawned two different SCs. As a consequence of the spawning, the cyclic properties of the original LSU may be lost; in particular the chain could contain sink nodes, hence introducing additional cut-edges. An example of this situation is illustrated in case (c).

The Markov chain that constitutes the SSU possesses a transition probability P. The transition probabilities between the nodes are evaluated using the number of actual temporal shot transitions. In general, p_{ij} is computed by dividing the number of temporal transitions that go from shots of the cluster SC_i to shots of the clusters SC_j, divided by the number of shots in the cluster SC_i. In Fig. 3 the

auto-transitions associated with the probabilities p_{ii} have been omitted for clarity. For example, assuming that the visual clusters of $LSU1$ in Fig. 2 are perfectly matched to the semantic clusters of the corresponding SSU, as in case (a) of Fig. 3, the transition probability matrix would be:

$$P = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix} = \begin{bmatrix} 0.2 & 0.4 & 0.4 \\ 0.75 & 0 & 0.25 \\ 0 & 1 & 0 \end{bmatrix}$$

For sink nodes, such as SC_9 in case (c) of Fig. 3, a unitary probability is assigned to their auto-transition.

5. VIDEO RECOMBINATION

This section details the cooperation workflow between the narrative construction and video system modules: the information exchange taking place between the two subsystems and the technologies for video recombination.

5.1 Semantic Information Exchange

After semantic analysis of the baseline video the *semantic subspace* is defined by organizing the available resources for use by the system, namely, the semantic points.

Part of the domain model is a manually constructed mapping between high-level actions and so-called *semantic patterns*. The patterns are a list of semantic points that need to be present in the considered action for correctly conveying the high level meaning, e.g., a semantic pattern of a dialogue could be constituted by two points with character A and two points with the same environmental tags and character B. With the available semantic points known, the model can be updated to avoid those not present in the original content.

Examples of the semantic information exchanged in the modalities described in Section 3 are depicted in Fig. 4. Part (a) shows the video-driven set up where, aside from the semantic subspace definition, the video system also provides a list of semantic patterns generated via the *semantic patterns proposal* of Subsection 5.2. This video-driven setup is motivated by its detailed knowledge of the movie structure, i.e. it can put together novel semantic patterns using the conceptual information. Moreover, the visual content for these narrative actions is of high quality by construction.

In the narrative-driven setup (part (b) of the figure), a *semantic pattern request* is sent to the video system as it chooses the next narrative action. The pattern requested is obtained through the mapping between the narrative actions and the semantic vocabulary, with suitable parameters for characters and location pertinent to the action.

When the video system is able to satisfy the request, the narrative generator is notified to provide the subtitles to add to the shots. Since the narrative generator is constrained by the semantic subspace definition, the request can in principle be satisfied. However, in certain cases the video system can fail (see Subsection 5.2), which forces the story generator to rewind its engine and to find an alternative narrative action.

5.2 Semantic Clusters Recombination

This subsection formalizes the SSU-based innovative techniques for the manipulation of Markov chains (SSU transition graphs) for the generation of the recombined video, namely the substitution and deletion of semantic clusters belonging to a given SSU and pairwise Semantic Story Units fusion respectively.

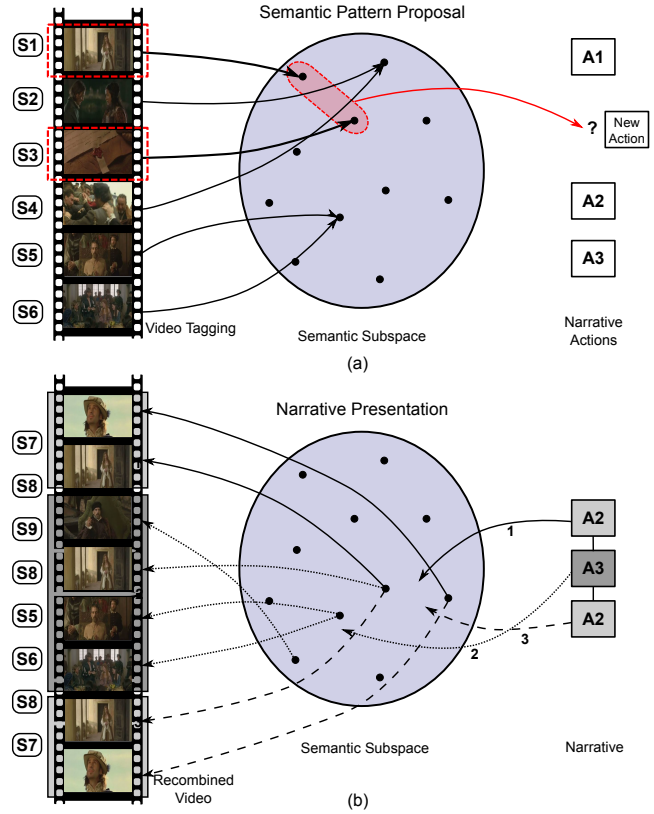


Figure 4: Mapping between video and narrative using the shared semantic subspace: each action is mapped to a semantic pattern and the patterns are depicted as a shaded region in the subspace. In (a) the video system proposes semantic patterns as possible actions and the resulting recombined video is assessed and in (b) the narrative generator requests recombined video with given semantic patterns associated with a narrative composed by three actions. The dashed arrows (label 3) denote a successive request of the same pattern as that of label 1.

5.2.1 *Semantic Cluster Substitution and Deletion*

When the video system receives a semantic pattern request, it generates a shot sequence after having constructed a suitable SSU. To properly answer to a pattern request, the ideal situation is to have an SSU composed by semantic clusters which have a one-to-one correspondence with the needed semantic points. In that case, the system would perform a random walk across the Markov chain model of the SSU. From any given node, a shot is chosen with the only constraint being to preserve causality of visual information.

Auto-transitions in the SSU are not desirable since shots described by the same semantic points played back to back introduce visual flicker. Hence, an operator $D[\cdot]$ puts the diagonal values of P to 0 and uniformly redistributes the probability among the other columns. Obviously, the redistribution of some probability perturbs the transition matrix.

If no perfectly matching SSU is found, the video system tries to generate a new target SSU that satisfies the pattern, starting from so-called candidate SSUs. A given SSU is a potential candidate to be modified into the target SSU if

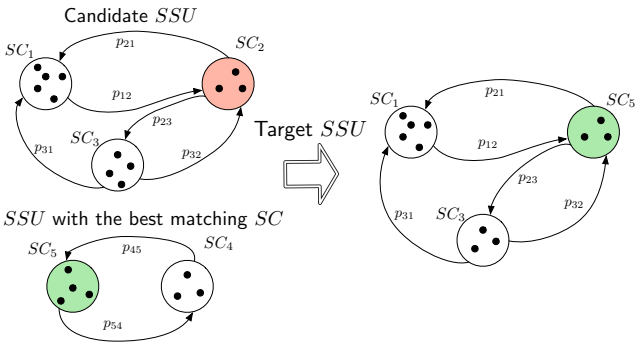


Figure 5: Semantic cluster substitution: some shots in SC_5 (green) are substituted into SC_2 (red).

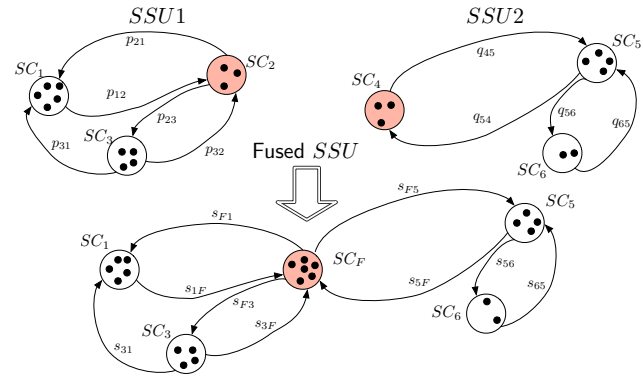


Figure 6: SSU fusion process: the fused SSU has aggregated SC_2 from $SSU1$ and SC_4 from $SS2$, that represent the same semantic point, into the fused cluster SC_F. Auto-transitions are omitted.

both these conditions are satisfied: the candidate SSU has at least a matching semantic point with the target SSU and it has no less semantic clusters than the target SSU so as not to generate artificial cluster transitions. To avoid the complete loss of the existing logical structure, only two operations are allowed: substitution and deletion of semantic clusters. The rationale under this process is to minimize the changes to the underlying structure of the candidate SSU inherited by the original video content. When semantic criteria are not applicable, low level visual distortion (the shot similarity measure introduced in Section 4.1) is used instead.

The algorithm for choosing the best candidate SSU is based on satisfying as many semantic points as possible, starting from the most numerous one in terms of requested points. The tie-breaking criteria among candidates is the fewest number of shots that are unrelated to the desired semantic clusters. At the end of this process, the semantic clusters of the best candidate SSU with unrelated semantic points must be substituted and those in excess must be deleted to adhere to the semantic pattern.

First, let us concentrate on cluster substitution. Fig. 5 depicts such a situation. The candidate SSU has SC_1 and SC_3 satisfying the semantic pattern, but SC_2 does not and need to be substituted. Our objective is to preserve the number of shots in SC_2 so that the transition matrix is not perturbed. Then, we try to find a matching semantic cluster in some other SSU that satisfies the requested semantic point and that has sufficient shots: in the example, SC_5 is subsampled and put in place of SC_2. If more than one cluster fits, then the closer one in terms of average shot similarity is selected. Alternatively, if no matching semantic cluster has enough shots, the visually closer clusters are exhausted in sequence until the number of needed shots is satisfied. In the case that more semantic clusters need to be substituted, the process is repeated sequentially.

Last, excess semantic clusters may need to be deleted. This operation is more penalizing because it is bound to perturb the transition matrix, since some cluster transitions disappear and hence this is equivalent to reducing the number of dimensions of P. Moreover, some diagonal value may acquire non-zero values, that is some auto-transition may also appear in the target SSU. Again, $D[\cdot]$ needs to be applied before performing the random walk for shot selection.

The video system should compute a cost to highlight how much of the original SSU structure has been lost to answer the request; if this exceeds a threshold the video system responds with failure, indicating that the request can't be satisfied. In this work, the cost associated with the construction of the target SSU is $C = w_c \cdot n_c + w_d \cdot n_d + w_a \cdot n_a$, where n_c is the number of substituted shots, n_d is the number of deleted shots, n_a is the number of shots auto-transitions before the application of $D[\cdot]$ and w_c, w_d and w_a are the associated weights ($w_d > w_a > w_c$ for the reasons above).

5.2.2 *Semantic Story Units Fusion*

In the video system driven setup, the video system tries to generate new possible narrative actions by fusing SSUs and obtain new semantic patterns. Such a new semantic pattern is proposed for evaluation thanks to the generation of an example video clip. If it is possible to attach a high level meaning to the clip then the semantic pattern and the narrative actions describing it are added to the set of narrative actions. Obviously, single, non-trivial SSUs are first proposed because they are obtained directly from the corresponding LSUs, which likely reflect narrative actions of the original plot. In addition, the video system tries to combine different SSUs to obtain a sequence of shots that may convey some complex meaning. For example, if two SSUs associated to dialogues are mixed, the fused SSU could possibly represent a new dialogue between multiple characters.

Obviously, it is necessary to fuse only SSUs that have a minimum degree of coherence. In this work, we exhaustively try fusing only pairs of SSUs that have at least a matching semantic cluster. Fig. 6 illustrates the case where two semantic clusters, SC_2 and SC_4, share the same semantic point and as such it is decided to merge them into the fused SSU SC_F, while trying to inherit as much structure as possible from the individual participating SSUs.

Formally, starting from the $SSU1$ with transition matrix P and $SSU2$ with transition matrix Q, we construct a fused SSU with transition matrix S. We will use Fig. 6 as a working example. The concerned transition matrices are:

$$P=\begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix}, Q=\begin{bmatrix} q_{44} & q_{45} & q_{46} \\ q_{54} & q_{55} & q_{56} \\ q_{64} & q_{65} & q_{66} \end{bmatrix}, S=\begin{bmatrix} s_{FF} & s_{F1} & s_{F3} & s_{F5} & s_{F6} \\ s_{1F} & s_{11} & s_{13} & s_{15} & s_{16} \\ s_{3F} & s_{31} & s_{33} & s_{35} & s_{36} \\ s_{5F} & s_{51} & s_{53} & s_{55} & s_{56} \\ s_{6F} & s_{61} & s_{63} & s_{65} & s_{66} \end{bmatrix}$$

To preserve the original structures, all the transitions be-

tween the clusters inherited by the originals should be preserved (in our case, between SC_1 and SC_3 and between SC_5 and SC_6). The transitions to and from the fused cluster should be weighted by the numbers of shots in the original clusters (defined as n_2 and n_4, respectively), as this reflects the relative importance of the individual component clusters. The most practical solution is therefore to extend the transition matrices P and Q, rearranging them in the same row order as S and identifying the individual semantic clusters that will be fused with the final fused clusters, as in:

$$P^* = \begin{bmatrix} p_{FF} & p_{F1} & p_{F3} & 0 & 0 \\ p_{1F} & p_{11} & p_{13} & 0 & 0 \\ p_{3F} & p_{31} & p_{33} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \; Q^* = \begin{bmatrix} q_{FF} & 0 & 0 & q_{F5} & q_{F6} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ q_{5F} & 0 & 0 & q_{55} & q_{56} \\ q_{6F} & 0 & 0 & q_{65} & q_{66} \end{bmatrix}$$

where F in P^* identifies with SC_2 in P and F in Q^* identifies with SC_4 in Q. Then, the fused SSU transition matrix S is constructed by concatenating the rows of S_F, S_P and S_Q, where S_P and S_Q are the rows of P^* and Q^* respectively corresponding to the non-fused SCs and S_F are the rows corresponding to the fused clusters (in this case a single row), obtained by computing:

$$S_F = \frac{n_2}{n_2 + n_4} P_F^* + \frac{n_4}{n_2 + n_4} Q_F^*$$

where again n_2 and n_4 are the shots number in the original SC_2 and SC_4 clusters being fused. With this formulation, no new transitions between semantic clusters are introduced. The resulting matrix is therefore:

$$S = \begin{bmatrix} & & S_F & & \\ p_{1F} & p_{11} & p_{13} & 0 & 0 \\ p_{3F} & p_{31} & p_{33} & 0 & 0 \\ q_{5F} & 0 & 0 & q_{55} & q_{56} \\ q_{6F} & 0 & 0 & q_{65} & q_{66} \end{bmatrix}$$

6. NARRATIVE INTEGRATION

Each story variant proposes a different sequence of narrative actions. We have described in previous sections how action variants could be generated from common video footage using semantic representations. To assemble a consistent story, it is necessary to preserve the global consistency of actions: this is achieved using AI Planning techniques, which can handle the combinatorics of individual actions whilst preserving the overall story logic as previously described [31]. For the narrative generation, we use a forward-chaining state-based planner (bottom half of Fig. 1).

In this section we consider the key enablers of narrative integration in the system: the narrative domain actions and mappings from them to semantic patterns; and the use of the shared semantic space in constraining narrative generation (further details of the planner can be found in [23]).

6.1 Mapping Narrative Actions to Semantics

The domain model used by our planner corresponds to narrative states and is formalised using the PDDL language [15]. The states refer to categories of actions and character attributes, both generic and specific to the baseline plot. Narrative actions contain variables that are instantiated at run time with the names of specific characters, objects, locations corresponding to video semantics.

Also specified during the domain analysis are mappings between the narrative actions and points in the shared semantic subspace. As with the narrative actions these mappings from actions to semantics contain variables which are ground at run time by assigning values to character, object, and location variables. Action and semantics share the same variable assignments. The mapping used for action translation in our implementation results in semantic sequences of between 2 and 4 semantic points. As we show in the evaluation, this is sufficient for representing what are for the most part binary character interactions, though systems with larger-scale actions may require longer sequences with a greater authorial overhead in encoding the mapping.

6.2 Constrained Narrative Generation

In storytelling systems that feature computer generated presentation, every potential narrative action is guaranteed to have a valid presentation. With a shift to presentation based on recombination of pre-recorded video data, this guarantee no longer holds. Our narrative planning procedure therefore must be capable of adapting the generation process to avoid areas of the narrative space for which video data is not available. These adaptations are two-fold: static modifications that can be applied to any narrative generated for the given domain and video data; and dynamic modifications to recover from unexpected presentation failure.

In Subsection 4.2 we described the process of automatically extracting the parameters for our semantic abstraction from a set of video data. As VBS systems do not permit on-line dynamic content generation the semantic subspace available for presentation is determined prior to narrative generation. Static analysis can then ensure the planner constructs narratives using only those actions that map to sequences of semantics that are a subset of those available. This is achieved by performing a filtering procedure as the variables in each narrative action are ground (by substitution with specific character names and so on), which applies the action-semantics mapping and accepts only those actions that map to semantics appearing in the video, avoiding those for which no representative shot exists in the video data.

This static filtering process guarantees the exclusion of single actions which cannot be presented given the available video data, however it may not exclude all necessary actions. For example, when the requested semantics map to a logical structure that is radically different from SSUs in the original video, this request may be declined. Thus the planner must be able to recover from failure during plan construction.

Failure occurs at one of two points depending on the execution process. If all semantic requests are processed prior to presentation, then the unpresentable action can be removed from the domain and a new plan constructed. However, to enable reactive narrative generation, planning and execution must be interleaved [7] so when failure is detected presentation of earlier actions may already have begun. This is the situation assumed in our approach which achieves online failure recovery by removing the failed action from the domain and re-planning from the previous accepted action. This results in slightly longer narratives as there are fewer decision points at which the narrative can 'work around' a failure. The effect of this in practice is explored in Section 7.

7. EXPERIMENTAL RESULTS

Experiments were conducted to evaluate the filmic variants generated by our VBS system in terms of: narrative variant

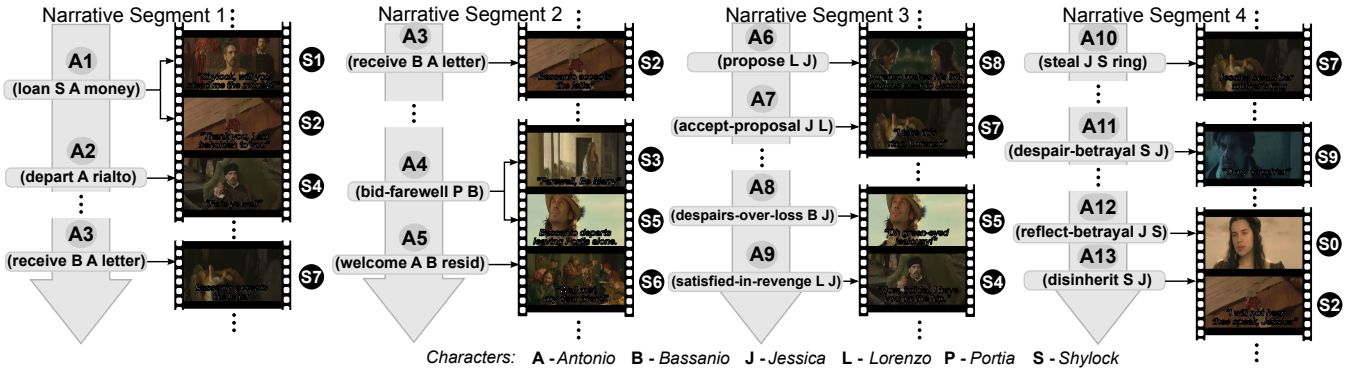


Figure 7: Sample VBS system output: shown running downwards are excerpts of four narratives generated using our *Merchant of Venice* domain model; each narrative shows a number of actions alongside a selection of video shots chosen by the system for the visual presentation of that action (see text for further detail).

generation; consistency and comprehensibility; and visual quality. The results of this evaluation are discussed below.

7.1 Generated Filmic Variants

Fig. 7 shows excerpts from filmic variants generated by our VBS system. Each narrative includes selected actions along with system chosen video shots for visual presentation. The narratives highlight a number of important points:

• The generative possibilities afforded by our plan-based approach result in a large space of potential narratives and the figure shows four very different narratives from this space: narrative 1 contains elements of the original play, a loan is established (A1) and a letter received telling of the perilous circumstances of the benefactor (A3); narrative 2 tells of a profligate character deserting his wife (A2) and later receiving the indulgent welcome of a good friend (A5); narrative 3 tells of romance (A6, A7), lost love (A8) and revenge (A9); and narrative 4 tells a very different tale of familial betrayal (A11), despair (A12) and disinheritance (A13).

• Individual narrative actions can appear in different narratives with different visual presentations (shots) used in each. For example, action A3 is in narratives 1 and 2 but its presentation differs: shot S7 in narrative 1 and S2 in narrative 2. This is possible because different shots are described by the same semantic point. Importantly, this flexibility allows the VBS system to exploit large video corpora enabling multiple presentations on subsequent runs or similar narratives.

• A single shot can be used in multiple semantic contexts because the same semantic pattern can be part of the mapping to diverse narrative actions, e.g., shot S2 appears in both narrative 1 and 2 in totally different contexts.

7.2 Generative Power

The power of a narrative generator can be expressed in terms of expected narrative path length n, and the number of choices at each narrative step (i.e. branching factor) b. The size of the space of narratives that can be generated from a given initial state is: b^n. Here the branching factor is described in terms of the number of characters in the narrative c, and the average number of actions each character has available to them a, giving $b = c \cdot a$.

As outlined in Section 6.2, a pattern request can fail and so the average number of actions a can be reduced. In this case, if p is the fraction of actions removed, then the size of the narrative space becomes $((1-p)(c \cdot a))^n$. However, since removing actions forces the narrative to circumnavigate affected regions of the previous path, n is expected to increase when actions are removed and hence narrative possibility is not compromised.

To evaluate how these changes affect the generative power in practice, we have calculated the branching factor for our *Merchant of Venice* domain and narrative lengths from our three example initial states. Fig. 8 shows the effect on branching factor and length that randomly removing increasing number of actions from the domain has. The error bars show one standard deviation based on 9 repeats over 3 narrative initial states. The high variance is to be expected, as narrative spaces are well known to be non-uniform [9].

The three narratives averaged a branching factor of 8 when all actions were available. In most cases this was around 4 actions for each of two on-stage characters. As actions are removed the branching factor follows the expected downward trend decreasing linearly with number of actions. Interestingly, the increase in length as convoluted narratives are required to work around unpresentable actions more than compensates for the decrease in branching factor. The rapid rise in average narrative length n seen in Fig. 8 means that the overall space of narratives the planner draws from actually increases in size.

When using the entire footage of the original *Merchant of Venice* film to source video data there are still 6% of the actions in our example domain with no corresponding semantics. This appears near the left of Fig. 8 where the branching factor has not yet reduced, but produces narratives that are around 3 actions longer. The greater possibilities enabled by these additional actions in the narratives more than makes up for the slight decrease in options for character actions. In fact, the narrative space is on average 11 times larger than that being explored when all actions are available.

It should also be noted that when more than 15% of actions were removed prior to narrative generation, runs occurred in which no valid narrative could be found. This is due to the planner encountering dead-ends or low branching factor regions in the narrative space. The rate of failure began at 18% (when 85% of actions remain), and rose as high as 90% (only 60% of all actions remain). So as a rule of thumb it appears that sufficient video is required to cover at least 85% of a domain for successful narrative generation.

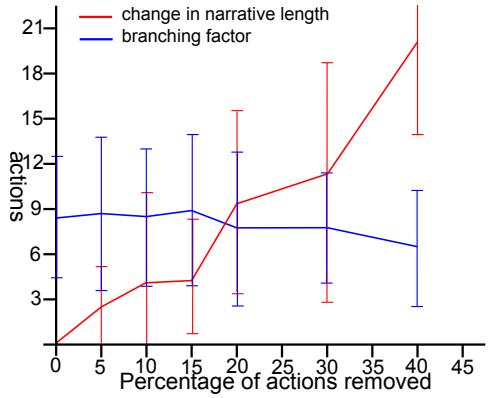


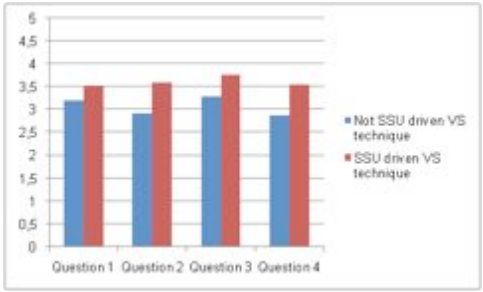
Figure 8: The change in generative power after constraining available video, measured in branching factor size and increase in narrative length.

7.3 Narrative Comprehensibility

We also explored how comprehensible the narratives output by our VBS system were to users. The QUEST model [17] represents narratives as a conceptual graph that provides measures that are able to rate the relative quality of comprehension questions. Asking users to assign goodness of answer (GOA) values to question-answer pairs and assessing their correlation with QUEST-predicted quality has proven a useful technique for measuring presentation's effect on comprehension in IS [10]. An additional benefit of this approach over free-form questionnaires is that it eliminates the need for qualitative assessment of user responses.

Correlation between the QUEST model's expected quality of question-answer pairs and user ratings would provide strong evidence that our VBS system produces easily comprehended narratives. To determine if this is so, we created a narrative and video sequence for each of our three example initial states. Four questions and four answers were randomly selected from the QUEST model of each of our three narratives. This gave 16 question-answer pairs for each narrative, which were presented to 10 participants for a total of 480 evaluations. Participants were asked to watch the video for a narrative and rate the goodness of each answer for each question with a value from 1 (very bad) to 5 (very good).

User responses were compared against measures of reachability and arc distance in the QUEST graph for each narrative. We set expected values for the GOA with 5 (very good) for those with arc distance 1, 4 for those with arc distance 2, and so on, with 1 (very bad) expected for question-answer pairs that are unreachable in the QUEST graph. The mean difference between these expected values and those of the participants was 1.07 – significantly lower than the 1.6 mean that would result from random selection. This was significant with $p < 0.01$ by a two-tailed single sample T-test. Furthermore, the correlation between user GOA and the arc distance measure was 0.49 by Pearson product-moment coefficient, which can be interpreted as somewhere between a medium and large correlation [11] (see page 116). Given that no normalisation between participants' results was performed and that the relationship between our arc distance measure and GOA is not necessarily linear, this level of correlation is strong evidence that the video-based presentation of stories has not compromised comprehensibility.



	SSU driven		*Not SSU driven*	
	Mean	Conf. Int.	Mean	Conf. Int.
Question 1	3.19	0.34	2.88	0.31
Question 2	3.23	0.35	2.50	0.41
Question 3	3.50	0.27	3.00	0.29
Question 4	3.23	0.31	2.57	0.31

Figure 9: Visual Quality Test Results: users were questioned about shot adequacy, coherency, transitions and enjoyability of videos generated with and without our SSU techniques (see text for detail).

7.4 Visual Quality

We also ran subjective tests on the quality of video content by generating recombined video clips, of about 4 minutes, relating to two alternative plots. For comparison, the same plots were used to generate video that didn't exploit the SSU techniques from Section 5.2; instead, output video was formed by taking shots satisfying the semantic patterns guaranteeing only causality of the shots in the same narrative action. These 4 videos were shown in random order to users, who were asked the following questions for each video:

- *Shot duration adequacy*: if the pace of the shots seems right to the user, not too frenetic nor too slow.
- *Shot content coherency*: if the shots visual content is consistent with the subtitles meaning, conveying the narration.
- *Actions Transition*: if the transition between consecutive narrative actions is smooth or it appears artificial.
- *Overall enjoyability*: if the recombined video is pleasant, with emphasis on perception rather than understanding.

Again, the answers were integer grades ranging from 1 (low quality) to 5 (high quality). Fig. 9 reports the Mean Opinion Score (MOS) of the answers, along with the 95% confidence interval. From the grades given to the content quality provided by the VBS system, it can be concluded that the users were generally satisfied with the experience, although there is still room for improvement. Also, Fig. 9 highlights that the shot recombination technology benefits from the SSUs underlying structure inherited by the LSUs.

8. CONCLUSIONS

The paper overviews a complete implementation of an innovative VBS system that fully integrates AI planning with a sophisticated video analysis and recombination process. The integration has been enabled through the definition of a common semantic vocabulary that permits cooperation between planning and video processing. This also extends the flexibility of the planning by removing the limits imposed by using only actions originally scripted in the baseline video. Moreover, the intermediate level semantics also guarantee

that the actual video content played back is more closely tailored to the intended high level meaning of the narrative actions. The system showcases novel techniques for manipulating the Markov models representing semantic story units.

Our experimental results show that the alternative narratives generated through scripted video recombination achieve promising grades from user tests and highlight the importance of the SSU-based technology presented here in exploiting the underlying logical structure of the original content.

9. ACKNOWLEDGMENTS

This work has been funded (in part) by the European Commission under grant agreement IRIS (FP7-ICT-231824).

10. REFERENCES

[1] N. Adami and R. Leonardi. Identification of editing effect in image sequences by statistical modeling. In *Proc. Picture Coding Symposium (PCS)*, pages 157–160, Portland, OR, USA, 1999.

[2] S. Benini, P. Migliorati, and R. Leonardi. Hierarchical structuring of video previews by leading cluster analysis. *Signal, Image and Video Processing*, 4(4):435–450, 2010.

[3] S. Bocconi, F. Nack, and L. Hardman. Using Rhetorical Annotations for Generating Video Documentaries. In *IEEE Int. Conf. on Multimedia and Expo (ICME)*, pages 1070–1073, Netherlands, 2005.

[4] S. Bocconi, F. Nack, and L. Hardman. Automatic generation of matter-of-opinion video documentaries. *Journal of Web Semantics*, 6(2):139–150, 2008.

[5] J. Boreczky and L. Rowe. Comparison of video shot boundary detection techniques. *Electronic Imaging*, 5(2):122–128, 1996.

[6] K. M. Brooks. Do Story Agents Use Rocking Chairs? The Theory and Implementation of One Model for Computational Narrative. In *Proc. ACM Multimedia Conf.*, pages 317–328, New York, NY, USA, 1996.

[7] M. Cavazza, F. Charles, and S. J. Mead. Character-based interactive storytelling. *IEEE Intelligent Systems*, 17(4):17–24, 2002.

[8] P. Cesar, D. Bulterman, and L. Soares. Human-centered television: directions in interactive television research. *ACM Trans. on Multimedia Computing, Communication and Applications*, 4(4), 2008.

[9] Y.-G. Cheong and R. M. Young. Narrative generation for suspense: Modelling and evaluation. In *Proc. 1st Joint Int. Conf. on Interactive Digital Storytelling (ICIDS)*, pages 144–155, 2008.

[10] D. Christian and M. Young. Comparing cognitive and computational models of narrative structure. In *Proc. of the National Conf. of the American Association for AI (AAAI)*, pages 385–390, San Jose, CA, USA, 2004.

[11] J. Cohen. *Statistical power analysis for the behavioral sciences*. Academic Press, New York, 1988.

[12] C. Cotsaces, N. Nikolaidis, and I. Pitas. Video shot detection and condensed representation. *IEEE Signal Processing Magazine*, 23(2):28–37, 2006.

[13] C. Crawford. *Chris Crawford on Game Design*. New Riders Publishing, 2003.

[14] G. Davenport, T. Smith, and N. Pincever. Cinematic Primitives for Multimedia. *IEEE Computer Graphics and Applications*, 11(4):67–74, 1991.

[15] A. Gerevini and D. Long. Technical report, 2005. http://www.cs.yale.edu/homes/dvm/papers/pddl-bnf.pdf.

[16] A. Gersho and R. M. Gray. *Vector Quantization and Signal Compression*. Kluwer Academic, 1992.

[17] A. C. Graesser and D. Hemphill. Question answering in the context of scientific mechanisms. *Journal of Memory and Language*, 30(2):186–209, 1991.

[18] B. Jung, J. Song, and Y. Lee. A narrative-based abstraction framework for story-oriented video. *ACM Trans. on Multimedia Computing, Communication and Applications*, 3(2), 2007.

[19] C. Lanz, S. Nowak, and U. Kühhirt. Determination of categories for tagging and automated classification of film scenes. In *Proc. 8th European Conf. on Interactive TV (EUROITV)*, pages 297–300, Finland, 2010.

[20] G. Lupatini, C. Saraceno, and R. Leonardi. Scene break detection: A comparison. In *Proc. Workshop on Research Issues In Data Engineering (RIDE)*, pages 34–41, Orlando, FL, USA, 1998.

[21] M. Mateas and A. Stern. Structuring Content in the Façade Interactive Drama Architecture. In *Proc. 1st Conf on AI and Interactive Digital Entertainment (AIIDE)*, Marina Del Rey, USA, 2005.

[22] F. Nack. *AUTEUR: The Application of Video Semantics and Theme Representation for Automated Film Editing*. PhD thesis, 1996.

[23] J. Porteous, S. Benini, L. Canini, F. Charles, M. Cavazza, and R. Leonardi. Interactive Storytelling via Video Content Recombination. In *Proc. ACM Multimedia Conf.*, pages 1715–1718, Italy, 2010.

[24] J. Porteous, M. Cavazza, and F. Charles. Applying Planning to Interactive Storytelling: Narrative Control using State Constraints. *ACM TIST*, 1(2):1–21, 2010.

[25] M. Radford. MGM Home Ent. (Europe) Ltd., 2004. The Merchant of Venice (film adaptation).

[26] W. Sack and M. Davis. IDIC: Assembling Video Sequences from Story Plans and Content Annotations. In *Proc. ICMCS*, pages 14–19, Boston, USA, 1994.

[27] E. Shen, H. Lieberman, and G. Davenport. What's Next?: Emergent Storytelling from Video Collection. In *Proc. 27th Int. Conf. on Human Factors in Computing Systems (CHI)*, pages 809–818, USA, 2009.

[28] T. Smith and G. Davenport. The Stratification System. A Design Environment for Random Access Video. In *3rd Int. Workshop on Network and Operating System Support for Digital Audio and Video, (NOSSDAV)*, pages 250–261, California, USA, 1992.

[29] T. Smith and N. Pincever. Parsing Movies In Context. In *Proc. Summer Usenix Conf.*, Nashville, USA, 1991.

[30] M. M. Yeung and B.-L. Yeo. Time-constrained clustering for segmentation of video into story units. In *Proc. Int. Conf. on Pattern Recognition (ICPR)*, volume 3, pages 375–380, Wien, Austria, 1996.

[31] R. Young. Creating Interactive Narrative Structures: The Potential for AI Approaches. In *Notes of the AAAI Spring Symposia on AI and Interactive Entertainment*, pages 81–82, Stanford, CA, 2000.

[32] V. Zsombori, M. Ursu, J. Wyver, D. Williams, and I. Kegel. ShapeShifting Documentary: A Golden Age. In *Proc. 6th European Conf. on Interactive TV (EUROITV)*, pages 40–50, 2008.