

D 5.2

CONSTRAINT-BASED IDIOM CONFIGURATION

Project Number	FP7-ICT-231824
Project Title	Integrating Research in Interactive Storytelling (NoE)
Deliverable Number	D5.2
Title of Deliverable	Constraint-based Idiom Configuration
Workpackage No. and Title	WP5 - Cinematography
Workpackage Leader	UNEW
Deliverable Nature	Software + Report
Dissemination Level	PU
Status	Final
Contractual Delivery Date	31 th December 2009
Actual Delivery Date	12 th January 2010
Author(s) / Contributor(s)	Guy Schofield (UNEW), Patrick Olivier (UNEW) Marc Christie (INRIA)
Number of Pages	28

Table of Contents

ABSTRACT	1
1. CONSTRAINT-BASED IDIOM CONFIGURATION	2
1.1 A DIRECTOR-VOLUME APPROACH TO CONSTRAINT-BASED IDIOMS.	2
1.1.1 <i>Introduction</i>	2
1.1.2 <i>The Elements of Automated Cinematography</i>	2
1.1.3 <i>Overview</i>	4
1.1.4 <i>Computing Director Volumes</i>	5
1.1.5 <i>Reasoning over Directors Volumes</i>	8
1.1.6 <i>Results and Discussion</i>	12
1.1.7 <i>Conclusion</i>	14
1.2 DESCRIPTION OF THE TOOL	14
1.2.1 <i>XML Parameterization</i>	14
1.2.2 <i>Idioms</i>	16
1.2.3 <i>Events</i>	17
2. STUDY OF 1984	18
2.1.1 <i>Analysis of 1984</i>	18
2.1.2 <i>Identifying Relevant Themes.</i>	19
2.1.3 <i>Software Implementation</i>	21
2.1.4 <i>XML Parameterization</i>	22
3. REFERENCES	25



Abstract

In this deliverable we first describe our Director Volume approach to the problem of constraint-based idiom configuration. The approach we propose is an automated system that constructs a movie from a sequence of low-level narrative elements (e.g. character actions and motions, and object motions). The system computes appropriate viewpoints on these elements, and performs cuts following user-defined cinematic conventions and styles. This represents a novel and expressive approach that couples camera planning and editing, which stands in contrast to existing techniques that are mostly procedural and do not encounter for dynamic visibility in editing.

We then present in detail the configuration files related to the software deliverable, that define the geometry of the environment, the set of actions, and the possible mappings between idioms and actions.

Finally, we report the detailed analysis of the 1984 canteen scene with specific reference to its narrative outcomes and the motivations of and relationships between its characters including aspects of dominance, isolation and affinity. We detail cinematographic techniques which can be used to elucidate these factors and describe their implementation as constraints. Finally, we describe the construction of a 3D CGI test-bed in which these constraints have been applied in a number of configurations in order to manipulate the narrative outcomes of the scene.



1. Constraint-based Idiom Configuration

1.1 A Director-Volume approach to constraint-based idioms.

1.1.1 Introduction

The provision of fully automated camera control for interactive environments, including viewpoint computation, viewpoint planning and editing, is a complex problem that raises three important issues.

1. Such a system needs to be underpinned by a narrative model that both structures the way the actions are organized, and allows us to reason as to their relative importance and presentation. Following Young's bipartite model [You07], interactive narratives can be considered to have two levels: (1) the story, the chronologic account of events over time; and (2) the discourse, the way in which events are organized and presented to the spectator.
2. How can we encode cinematic conventions and relate these to the narrative goals of the application? In relation to camera control, these conventions are often expressed as a set of idioms, that is, camera configurations that can be assembled to bring a story to the screen in a way that allows the viewer to understand the spatial relations/context of each scene, and the sequence of actions occurring within it. For example, such idioms would describe conventional camera placements and movements for variety of settings involving two or more characters in dialogue. An automated camera control framework must be capable of characterizing these declaratively through the use of constraints and objectives that can be solved in real-time.
3. How to develop efficient solution mechanisms which must address the innate complexity of well understood problems such as visibility determination and path planning? For example, the simultaneous computation of the visibility of multiple targets in a dynamic environment requires a pragmatic solution which will provide some guarantee as to the efficacy of results. Facilitating jump-cut continuity editing, where a user's view switches between two different viewpoints, requires the ability to reason more than just on local visibility of the current camera. Similarly, selecting appropriate tracking shots involves planning a camera's motion in a dynamic environment whilst simultaneously taking account of the visibility of scene elements from viewpoints along the spatio-temporal path.

In the context of the IRIS project, we propose a fully automated system that allows the construction of a cinematically expressive movie from a specified sequence of low-level narrative elements (e.g. character actions and motions, and object motions). Our system computes appropriate viewpoints on these narrative elements, by selecting specific regions of referred to as Director Volumes that capture both visibility and shot information pertaining to the scene. We then rely on filtering-based techniques to reason over Director Volumes and choose appropriate candidate shots. Cinematographic rules are expressed as **constraints** (applied on with constraint-based filtering processes) and cinematographic style is expressed as **preferences** to choose where to position the camera, how to plan paths between viewpoints, and when to perform cuts. Our system constitutes a novel and significant step towards the expressive coupling of camera planning and editing in 3D graphical environments.

1.1.2 The Elements of Automated Cinematography

Camera control in computer graphics has received significant attention in recent years.



Contributions span a range of sub-problems, from viewpoint computation and specification of shot properties, to camera motion control and the planning of visually felicitous paths. To contextualize our formulation of automated cinematography we considered a number of significant contributions across the broad spectrum of problems to be solved, from viewpoint computation and camera motion planning, to editing:

A. Viewpoint Computation

The ability to compute viewpoints with specifiable properties is of particular value in a range of applications, including visual servoing, medical and scientific visualization, and image-based rendering. For such applications, viewpoints typically maximize the visibility of salient features and highlight the spatial relations between objects with a view to enhancing a viewer's perception and understanding of a scene. Vasquez et al. [VFSH03] described the problem as maximizing the amount of information about a scene for a viewer. Drawing on Information Theory they relied on the notion of viewpoint entropy to compute a minimal set of N good views of a 3D environment (i.e. a set that maximizes viewpoint entropy).

Others contributions have considered the problem of viewpoint selection as one of information composition. Bares et al. [BK01] automatically generated camera shots using the composition heuristics of expert photographers. Camera constraints are specified using storyboard frames to indicate how a desired shot should appear. The 3D environment is then analyzed to determine camera parameter settings leading to a shot closely matching the specification. Christie et al. [CN05] proposed a partitioning of space, into semantic volumes, which capture the regions around scene elements for which a camera will have a consistent set of visual properties (i.e. qualitatively equivalent shots). However, in general approaches to viewpoint computation approach the problem without considering the temporal evolution of a scene.

B. Motion Planning

Existing methods for camera motion planning augment classical path planning approaches with features such as path continuity and visibility along the path. Planning techniques can be divided into two main classes: local approaches, that constrain the search to a reduced set of candidate solutions, and global approaches, which require the construction of, and search over, an a priori representation of the whole environment. In terms of local approaches, techniques based on potential fields have been proposed to plan camera's position in dynamic environments [Bec02]. Image-based visual servoing has also been used to compute camera DOFs based on constraints expressed in the camera's field of view [CM01]. To provide a broader knowledge of the environment, Li et Cheng proposed to use a lazy PRM attached to the object of interest (e.g. a virtual character) to reactively plan suitable camera paths [LC08]. Although such local approaches are responsive to local changes and dynamic elements of the scene (i.e. occluders) they typically fail to produce camera paths with properties that account for the global properties of a scene. Integrating global knowledge of the environment significantly improves the quality of the path but generally relies on an offline analysis of static environments. For instance, [NO03] proposed pre-computed PRM constructed over a complete environment in the automatic generation of camera movements. Within static environments, [AVF04] combined global planning, a search through a discretization (cells) of an environment, with a secondary local search of each cell. In order to adapt to dynamic aspects, Oskam et al. [OSTG09] recently presented a single target tracking real-time system that uses a global visibility-aware roadmap together with an estimate of the pair-wise visibility between portions of the scene, and which adapts at run-time to moving occluders. Such global planning methods enable large-scale camera transitions and can be combined with local search to follow a target and avoid obstructed viewpoints.

C. Editing

Incorporating the editing process (where and when to cut between viewpoints) in computer



graphics applications has been relatively under-addressed [HCS96] [CAwH_96] [CLDM03] [FF04]. Typically this is realized as set of idioms (series of reference viewpoints linked to a given configuration of actors) using a procedural language (usually a finite state machine): when a given action occurs, apply the associated viewpoint transition. Such approaches fail to recognize the importance of narrative discourse level [You07]. That is, story elements (plot and character) are defined in terms of plans that drive the dynamics of a virtual environment, and discourse elements (the narrative's communicative actions) are defined in terms of discourse plans whose communicative goals include conveying the story world plan's structure. Integration of these narrative aspects has only recently been considered [ER07] [Jha09]. However, the realization has again been procedural in character, a script (i.e. a series of actions) is taken as input and an automatic visualization of the scenario w.r.t. procedural cinematic rules is produced.

The procedural application of idioms often leads to highly restrictive editing rules and cannot furnish the expressive range of real world cinematography. Instead, series of idioms are deterministically applied to a story, and the idiom application architecture (e.g. finite state machines) makes idiom composition impossible when two actions are unfolding in parallel. Crucially, such approaches do not address low-level issues, such as visibility, that must be simultaneously accounted for. Indeed, the fact that no representation of an environment takes account of the exact visibility in the whole environment is a major issue in automated camera control. Therefore, automated cinematography must incorporate a high-level editing model, in which rules (to specify spatio-temporal context) and idiom preferences (i.e. style) are non-deterministically applied, and incorporate an integrated facility to reason as to visibility in camera movement and cut planning.

1.1.3 Overview

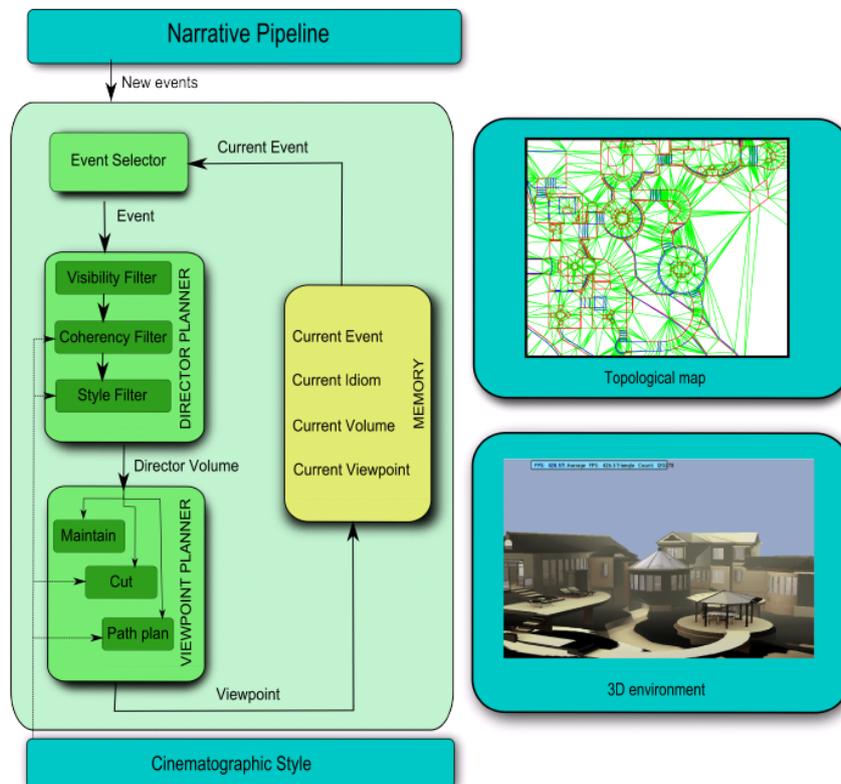


Figure 1: Overview of the camera control process: building and reasoning over occlusion-free Director Volumes.



We propose a new approach to virtual camera control and editing that directly addresses the visibility of key subjects for static occluders, and provides an expressive means to reason over idioms, shots and shot transitions. An overview of this process is given (fig. 1).

Our system takes as input a 3D environment together with a set of *narrative events* that describe the actions unfolding in the environment. Each narrative event is associated to set of viewpoints (Director Volumes) that convey the event according to established cinematographic conventions (i.e. idioms). At runtime, a viewpoint planner first selects the most relevant event and chooses the best idiom to convey this event by filtering the Director Volumes according to (i) the visibility of key-subjects, (i) the coherency in the sequence of viewpoints, (iii) and a characterization of style. A transition planner then determines appropriate viewpoint transitions, either by using a jump cut or a path-planning process between viewpoints.

1.1.4 Computing Director Volumes

As already described in section 2, real-time visibility computation is a prohibitively expensive process. We address this constraint by reducing the practical complexity through two assumptions as to the geometric properties of the environment. Firstly, we estimate the visibility of a target subjects (actor) using a 2D convex hull of the projection of its geometry onto the floor. Secondly, we restrict the setting to static 3D indoor environments which we represent as a 2D cell-and-portal structure [TS91]. Cells represent the rooms and portals represent doorways and connections between cells. Hard visibility constraints exist in relation to extremities of the portals (e.g. walls or doors). Each boundary between a wall and a portal supports a stabbing line that represents a separation between visibility and occlusion. By assuming that the indoor environment resides on plane (i.e. distinct floors) we extract a 2.5D topological representation. All geometric elements in the scene above a specified height (here 1.5m) are treated as occluders.

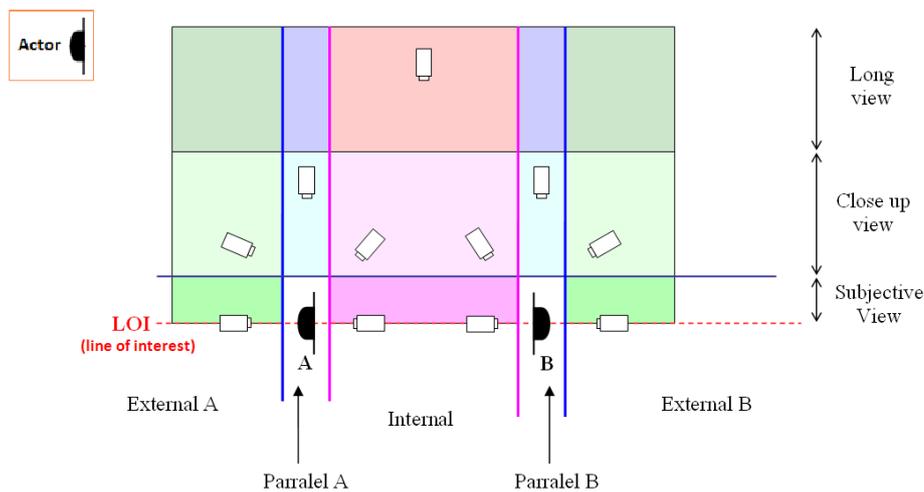


Figure 2: Example of semantic volumes for a key configuration of actors (adapted from [Ari76]).

A. Computing Semantic Volumes

Visual composition refers to the process of positioning the camera to furnish an image with the desired set of cinematographic properties. Given this set of properties, one can therefore gather, in the space of viewpoints, all satisfactory shots. Following [CN05], we refer to these volumes as **Semantic Volumes** (i.e. that aggregate viewpoints that give rise to the qualitatively similar visual properties).



Following this idea of viewpoint partitions, we have fully specified sets of shots for key configurations of actors, which we use to characterize our Semantic Volumes (see Figure 2). For example, in the case of a dialogue between two actors, a semi-space defined one side of the line of interest (LOI) captures the range of positions in which we can formulate over-the-shoulder shots of the two actors. In order to represent such volumes, we rely on a BSP data-structure. This structure is dynamically and procedurally processed in relation to the actors configurations and efficiently characterizes and partitions the environment into sets of viewpoints.

Some regions of space cannot be selected as candidates since the actors are fully or partially occluded. In order to efficiently compute the visibility information that we then store in the regions, a topological level of representation of the geometry is employed.

B. A Topological Representation of the Environment

Topological representations of an environment are spatial structures that support fast visibility computation and path planning around (typically static) obstacles. For the class of environments we consider these should be capable of addressing scene elements such a large vertical occluders, such as walls and pillars, and be capable of addressing configuration that constrain visibility (such as bottlenecks in the environment).

We use TopoPlan (Topological Planner) [Lam09] to preprocess the static environment geometry. TopoPlan analyses the scene geometry and generates a topological map as a triangular cell and portal decomposition based on a 2D constrained Delaunay triangulation.

The resulting representation contains pertinent information as to large vertical occluders and visibility bottlenecks, and allows us to rapidly compute the potential visibility of an object by taking into account of these large occluders.

C. Computing Visibility Volumes

Our process to compute the visibility of the actors uses a dynamic analysis of the potential visibility based on the precomputed static 2D topology and the configuration of the actors

Using the static representation

Generating good shots of static and moving actors requires us to maximize shot properties that relate to them and in particular to their visibility. Whilst full visibility and complete occlusion is trivial to compute, partial visibility, which constitutes a large proportion of the shots of an actor, is more difficult to efficiently calculate. We use an original visibility analysis method, based on the precomputed topology, to create a visibility cartography for an actor.

Numerous visibility culling methods already exist to establish fully or partially occluded polygons [CF92] [BHS98] but generally such techniques neither helps us identify fully occluded volumes nor, in the case of a partially occluded polygon, provide it with a visibility degree estimation. Consequently, we instead base our visibility analysis on the use of stabbing lines to make a dynamic categorization of the visibility volume of an actor, by which we can also estimate a partial visibility degree. Where there is no occluder between a viewpoint and the actor (for example, any viewpoint within the same topological cell), the actor will always be fully visible, we therefore only focus on viewpoints that are located in other topological cells.

From a viewpoint p , two stabbing lines are defined such that each line is tangential to, but on opposite sides, of the convex hull of the actor (fig. 3) and are referred to as extremal points. We can rapidly compute the extremal points, in linear time, by observing that a point e on the convex hull is extremal if and only if the other points immediately preceding and following e are located on the same side of a stabbing line passing through e .

To determine the visibility of an actor a with a static occluder, where point p is an extremity of this occluder, the stabbing line associated to an extremal point e of the actor separates the



region where e is visible and the region where e is occluded. Since the actor is fully included between stabbing lines, visibility categorization of the whole actor is then processed by combining visibility categorization of each stabbing line. Three regions are then obtained: a region where the actor is fully occluded, a region where the actor is fully visible, and an intermediate region where the actor is partially visible. From a viewpoint v , the 2D visibility of an actor corresponds to a segment s linking its extreme points. We thus propose to evaluate the visibility degree of the actor by computing the visible portion of s . By drawing a line through v and p , the degree of visibility of the actor can be estimated from the angle subtended.

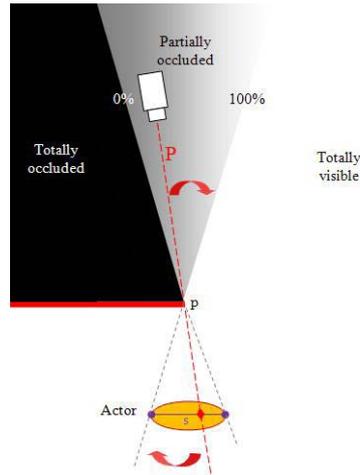


Figure 3: From a viewpoint, two stabbing lines are defined with respect to an actor. By combining both stabbing lines carried by an extremity p of an occluder, a categorization into a full occlusion region, a partial visibility region and a full visibility region of the actor is obtained. 2D visibility of an actor corresponds to a segment s linking its extreme points. From a viewpoint included in the region of partial visibility, the visible portion of the actor is evaluated by using relative angle between the plane P (passing by the viewpoint and the extreme point p of the occluder) and the stabbing lines of the actor. The visibility degree is then simply evaluated.

Integration to topological analysis

In reality the representation of the environment is based on the topological analysis and is a set of convex topological cells and portals. A portal comprises two points carried by one or more walls. Thus each point of a portal defines two stabbing lines for an actor, providing a subdivision of an adjacent topological cell into visibility cells. Moreover, since topological cells are convex, visibility cells will also be convex. The environment visibility characterization (Figure 4(a)) proceeds as: (1) the decomposition of topological cells into visibility cells by using stabbing lines; and, (2) the recursive propagation of stabbing lines to adjacent cells.

Dynamic visibility analysis structure

In this context, a BSP characterization is used for each topological cell. A node represents a stabbing line, and a leaf corresponds to a visibility cell of the actor. This structure provides: (1) an on-the-fly cell subdivision; and, (2) a reduction of the search complexity when characterizing a given viewpoint in the topological cell.

The new camera position is finally chosen among a set of candidates in appropriate semantic volumes

D. Director Volumes

In the previous sections, we have computed volumes corresponding to semantic information,



and volumes corresponding to visibility information on an actor. Combining visibility volumes and semantic volumes, by using the compatibility between both BSP structures, leads to BSP partitioning into semantic volumes augmented with visibility information on actors. We propose to intersect visibility volumes relative to each actor, obtaining a multiple visibility volume partition (fig. 4(b)). The representation of semantic information as a BSP now allows to easily intersect it with visibility information, and to obtain a camera director partition of the whole environment (fig. 4(c)).

This information composition now provides a good basis to reason on camera director volumes (semantic volumes augmented with visibility information) to either perform a jump, move in the environment, or maintain the camera position. In the following we will present our editing method, providing the camera paths and jumps given as input to the dynamic resolution of non-occluded camera positions.

1.1.5 Reasoning over Directors Volumes

In the application contexts we envisage in the IRIS project, our automated cinematography module is therefore required to (1) maintain rhetorical consistency with past shots; and (2) interactively interrupt the current shot to handle new elements as they become more relevant. Unlike most previous approaches [HCS96,ER07,Jha09] we enforce shot coherency rather than simply react to new narrative elements.

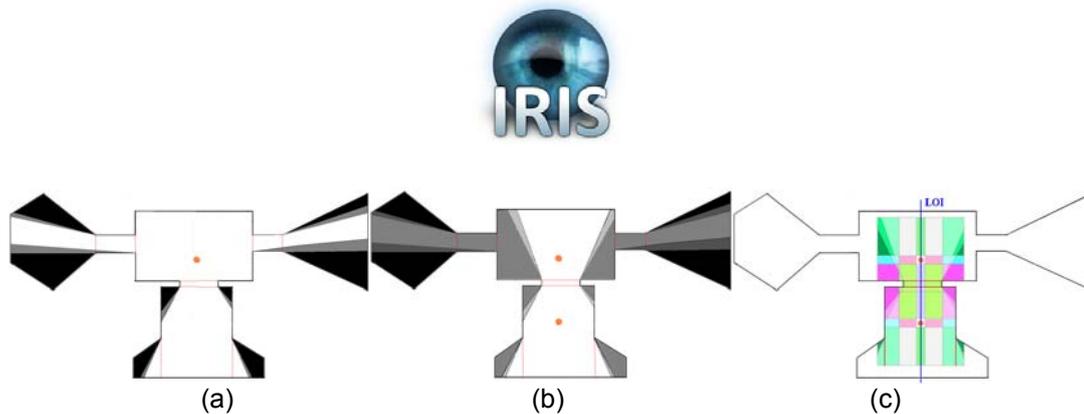


Figure 4: (a) Partitioning into visibility cells. Regions are divided into three groups: full visibility cells (white), full occlusion cells (black), and partial visibility cells (grey level) (b) Partitioning into multiple visibility cells (Two actors)(c) Semantic volumes linked to a configuration with two actors, and merged with the visibility information. Each semantic volume is colored, and the visibility degree is represented by the coloration intensity of the cell.

A. Narrative elements in Virtual Cinematography

The central notion we build our approach upon is **narrative element**, and its coherent translation into cinematographic idioms. A narrative element is a component of the discourse and conveys relevant information or an action from the story. Each narrative element has a specific purpose and its conveyance leads to changes in a viewer's cognitive or emotional state. Narrative elements vary in accordance with the nature of the story, ranging from prototypical actions (a character stands up, walks, talks to another character) to more subtle notions such as the display of a relationship between characters (e.g. show dominance, conflict or isolation). As narrative elements are indexed in time, we refer to them as events. The Narrative Pipeline (Figure 1) therefore provides these events into the cinematography module.

Our editing model is built upon the notions of idioms, coherency and style.

Idioms An idiom is a sequence of reference shots that convey a narrative element. In contrast to approaches that consider only a single solution viewpoint [HCS96, CAwH_96], each shot is specified in terms of Semantic Volumes that relate to key subjects.

Coherency Coherency affords compliance with cinematography conventions that require maintenance of the spatiotemporal context when the camera changes viewpoint. Conventions are well established in relation to the actors' motions, spatial locations, the line-of-interest and actions. Though some conventions are implicitly encoded within idioms, it is necessary to maintain coherency when the camera changes from one event to another, when visibility fails in the middle of an idiom, or in case of interleaved idioms (parallel editing).

Style Style is defined as a number of preferences as to: (1) shot pacing (which is the rate at which cuts are performed); (2) the camera speed (which favors either static or dynamic shots); (3) the minimum actor visibility in a shot (the level at which a character is considered occluded); and (4) order in which idioms should be considered where more than one applies.

The selection and sequencing of shots is fundamental to a viewer's understanding of the spatial, temporal and causal context. The appropriate choice of the next shot therefore needs to comply with a number of general rules, according to which we remove incoherent Director Volumes when switching viewpoints:

- Action continuity: between two shots, coherency must be maintained in the apparent speed and/or direction of motion.
- Screen direction: motion within the frame (up, down, left or right) must be maintained.



- Matching spatial relations: key spatial relations pertinent to the current action must be maintain, for example, in a dialogue if a character looks right towards another (off screen) character in one shot, this second actor should look towards left in the subsequent shot.

Following these rules, our editing process proceeds through a sequence of filters over the available director volumes, incrementally removing inappropriate volumes (constraint-based filtering):

- Filtering according to the idioms that relate to that event according to: (1) visibility, captured in Director Volumes (i.e. suppresses idioms whose semantic volumes are occluded); (2) style, through the application of preferences between idioms; and (3) the current viewpoint, each shot has a minimum/maximum duration;
- Filtering according to coherency (i.e. relative position in relation to the previous semantic volume). For example, jump cuts must be between viewpoints that subtend an angle of at least 30 degree on the subject.

We then propose a tree exploration of idioms to plan the following camera positions (Fig. 5) and similar to a tree of feasible [CAwH_96]:

- Every event is defined by a set of idioms that represent it;
- Every idiom is defined as a series of viewpoints (i.e. a series of semantic volumes);
- Transitions between semantic volumes may be discrete or continuous.

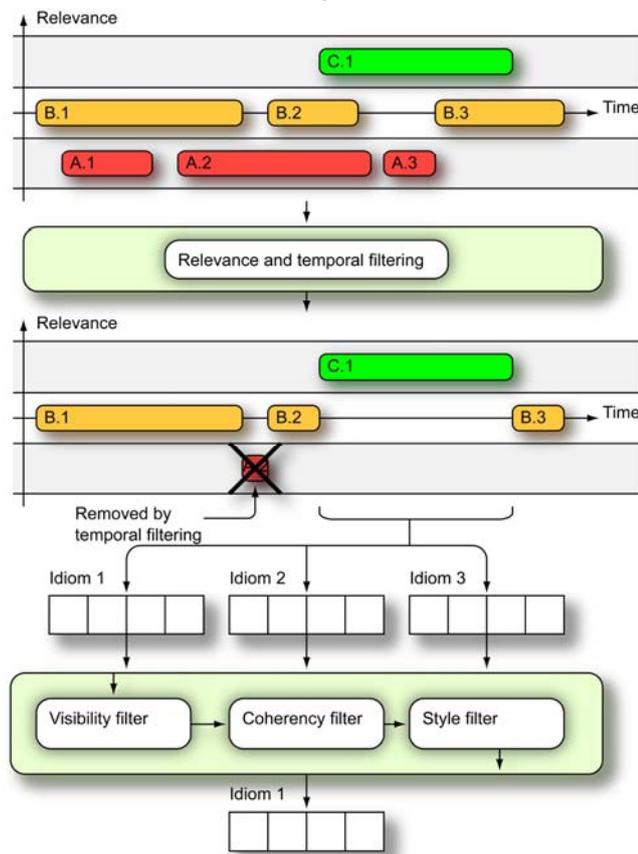


Figure 5: From the story (narrative elements), a tree of possible idioms is extracted and processed to plan the sequence of camera viewpoints



B. Selecting Transitions

Once the target Director Volumes is chosen, a decision has to be made as to the nature of the transition to perform between viewpoints, either a jump or continuous cut. Continuous transitions are motivated by either style or necessity. These transitions strongly relate to the pacing of a movie (the rate at which cuts are performed). Furthermore continuous transitions allow the director to break continuity rules, for example constraints relating to the line-of-interest, line-of-motion or line-of-action. Thus in cases where no Director Volumes are applicable after filtering on the grounds of coherency, the camera moves continuously to any visible Director Volume.

C. Building a Planning Graph

To perform continuous interpolations between Director Volumes, we rely on the cell-and-portal connectivity graph provided by the topological representation and augmented by the visibility and semantic partitions. The cell-and-portal decomposition offers (by construction) a connectivity graph that avoids obstacles. Cells in the graph are then further split according to the visibility BSP partitions and the semantic BSP partitions relating to the source and target Director Volumes (see fig. 6). A planning graph is then constructed based on this connectivity graph by: (1) locally sampling the portal segments to generate waypoint nodes; and (2) creating edges that connect all pairs of waypoints nodes together representing the transitions between waypoints. The task of planning a path thus reduced to multi-criteria optimization process (visibility of actors, semantic properties, path length). For ensure that the camera negotiates the cells appropriately, we define a cost that is expressed as a weighted sum of the path length and visibility of actors along the path. We associate different fixed costs to a fully visible cell and to a fully occluded cell to ensure termination. The cost c of a path t composed by s segments is defined as

$$c(t) = \sum_{s \in t} \Phi(s) \Delta(s)$$

where $\Phi(s)$ is for visibility degree along segment s , with $\Phi(s) = 1$ when fully visible, and $\Phi(s) = T$, T constant, when fully occluded. $\Delta(s) = |s|$ is the path length. The search within the graph is based on a Dijkstra process, to the difference that the target may be specified as multiple (and disjoint) cells. The process is suboptimal in such that exploration stops with the first hit on a target cell.

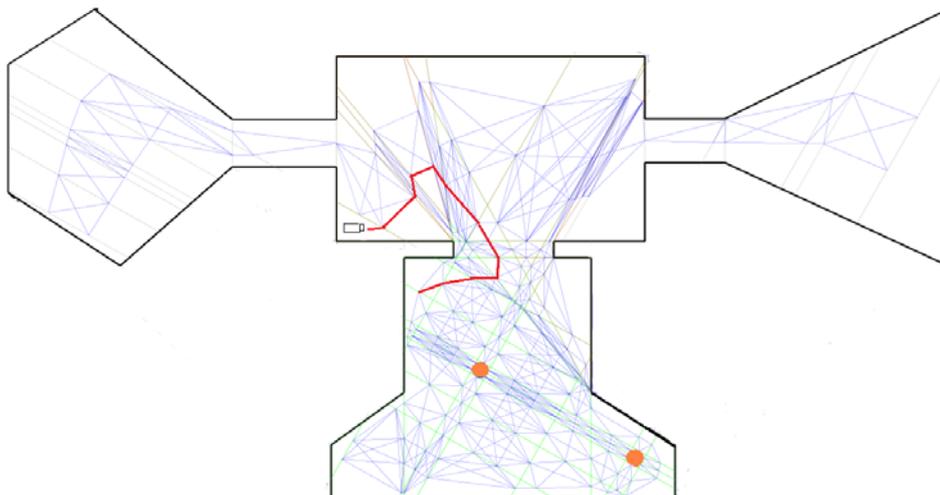


Figure 6: We build upon the planning graph to compute path requests between Director Volumes, with respect to the visibility of key subjects. Each cell is characterized by its



degree of visibility and its distance between the source and target Director Volumes. The connectivity graph and paths are recomputed as the scene evolves. Here the graph based on the Visibility and Semantic Volumes is colored blue, and the result of a camera path request is colored red.

1.1.6 Results and Discussion

We present two examples that illustrate the key features of our automated cinematography system.

Our first example comprises two characters, A and B with a doorway. The narrative elements are typical character interactions for which idioms are well defined and overlap in time to illustrate the parallel editing feature of related actions:

- Event 1: **A moves towards B** can be captured using a sequence of three shots: an internal or parallel shot of character A; an apex shot of characters A and B; and an internal or parallel shot on character B.
- Event 2: **A crosses a doorway** can be captured using a sequence of two shots: a side shot, 3/4 rear, or rear shot; and a front, 3/4 front or side shot.

As the first event is triggered, the specified idiom is selected and the Director Volumes are filtered by merging Visibility and Semantic Volumes (here there is no coherence to maintain with previous shots). A viewpoint is then selected in one of the Director Volumes (1.1) which in turn will constrain the choices on the next shots (1.2 and 1.3) due to the line-of-interest coherency requirement. It is possible that the second event may be ignored, depending on the temporal overlap between idioms and the minimum duration of shots. However, once the new event is selected motion coherency is enforced due to the left-to-right motion of the subject on the screen, which filters the available volumes for shot 2.1, and forces the selection of a new viewpoint. The shots are displayed in Figure 7.

The second scene takes place in an environment with multiple pillars. Two characters A and B are engaged in a discussion while walking in the environment. The only narrative actions are the dialogue utterances. This example displays demonstrates how our system overcomes some challenging problems. Firstly, around $t = 1$ s: utterances of characters A and B are very short. The system therefore ignores some of the events or postpones their shooting to avoid overly reactive cutting between shots. Secondly, around time $t = 5$ s, no apex shots are available to frame both characters. The filtering system therefore selects the next appropriate idiom in the list (an over-the-shoulder shot). Events and shots are displayed in figure 8.

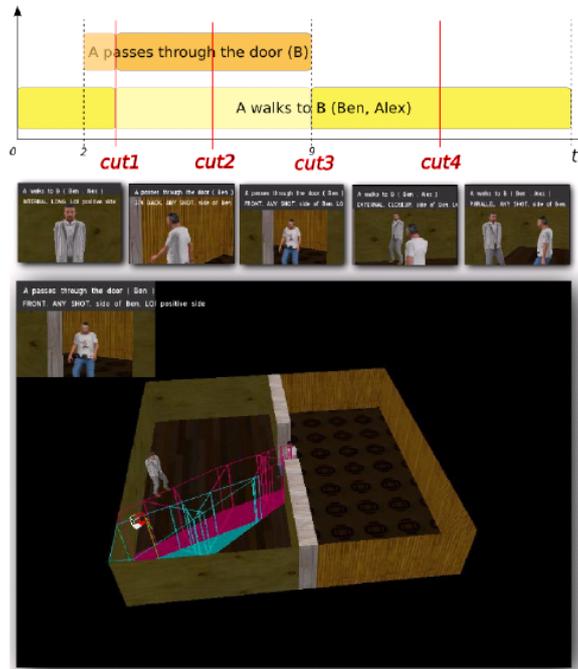


Figure 7: Conveying two related actions in parallel while maintaining coherency over the line of action and the line of interest.

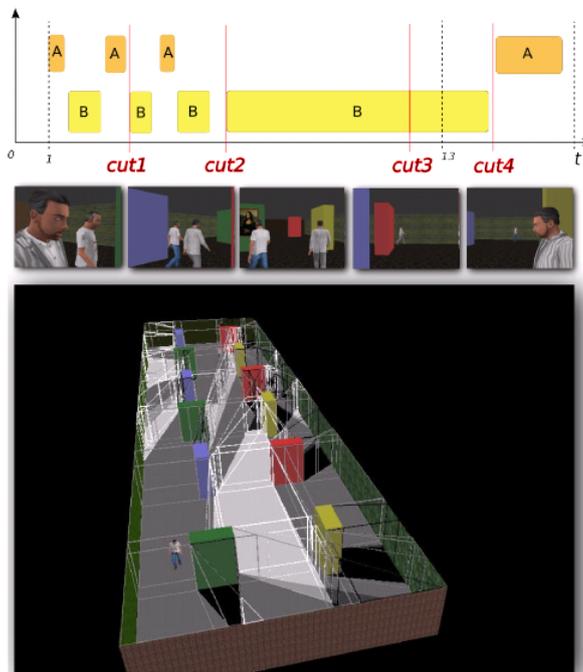


Figure 8: Following two actors in a dialogue scene. Camera avoids reactive switches with short utterances and uses one character shots when apex shots are not available.



1.1.7 Conclusion

In summary, we have presented a real-time cinematography system for virtual environments relies on an encoding of cinematic idioms and coherency rules to produce appropriate edits and camera paths from a set of narrative actions. Our method relies on a spatial partitioning providing a characterization into visibility cells (fully visible, partially visible, or fully occluded) and characteristic viewpoints (the Semantic Volumes). We then use these cells to reason at a symbolic level on how shot transitions should be performed, by relying on a constraint-based filtering process encoding style and cinematic conventions.

1.2 Description of the tool

The system input is based on an XML parameterization, that allows to describe the environment geometry, the path of each actor, the narrative elements (the events and the associated idioms), together with a cinematographic style that defines some degrees of variation in the application of the editing process (pacing, dynamicity of the camera, idiom preferences).

1.2.1 XML Parameterization

```
<Configuration>
  <Environment camera="passage_porte_01"
  actor="passage_porte_01"
  mesh="passage_porte_01_ville" />
  <Idioms file="idioms.xml" />
  <Events file="passage_porte/events.xml" />
  <!-- Characters -->
  <Character name="Alex"
  scale="0.3" radius="0.05" nbPoints="5"
  mesh="homme2.mesh" nodes="homme2Nodes.txt" attitude="marche3.s4d" periodic="false" >
    <Position x="0.83" y="0" z="-2.75" />
    <Position x="0.90" y="0" z="-2.6" />
  </Character>
  <Character name="Ben"
  mesh="homme1.mesh"
  scale="0.3" radius="0.04" nbPoints="5"
  nodes="homme1Nodes.txt" attitude="marche3.s4d" periodic="false" >
    <Position x="0.83" y="0" z="-0.45" />
    <Position x="1.83" y="0" z="-1.40" />
    <Position x="1.40" y="0" z="-1.42" />
    <Position x="1." y="0" z="-1.45" />
    <Position x="0.83" y="0" z="-2.4" />
  </Character>
  <!-- Cinematography System -->
  <ControlledCamera speed="0.00" />
  <PathPlanning visibility_weight = "1e3" />
  <Visibility min="0.5" max="1" />
  <Shot min="2" max="4" />
</Configuration>
```

- **Environment:** describes the topological representation of the environments related to the camera and the characters (Golaem::Path env3d file format), together with the geometric representation of the environment (Ogre mesh file format).



- **Idioms:** specifies the name of the XML file where the **idioms** are defined.
- **Events:** specifies the name of the XML file where the **events** are defined.
- **Character:** describes the parameters pertained to a character: (i) the 3D geometry (Ogre mesh file format), (ii) the animation (Golaem::Motion s4d file format), (iii) the abstraction of its 2D geometry (a circle approximated by a set of regularly sampled vertices), and (iv) its path (as a list of waypoints).
- **ControlledCamera:** specifies the maximum speed (in meters per second) of the camera (a proxy for dynamicity).
- **PathPlanning:** specifies the maximum value of the cost functions for arcs in the roadmap when planning paths (a mean to control the nature of paths)
- **Visibility:** specifies the minimum/maximum visibility degree of a character when planning camera paths and cuts.
- **Shot:** specifies the camera pacing, by constraining the minimum/maximum shot duration.



1.2.2 Idioms

```
<Rules>

  <!-- Actions -->

  <Action name="A walks to B" >
    <Idiom name="Aw2B-01" />
    <Idiom name="Aw2B-02" />
  </Action>

  <Action name="A passes through the door" >
    <Idiom name="Apttd-01" />
    <Idiom name="Apttd-02" />
  </Action>

  <!-- Idioms -->

  <!-- A walks to B -->

  <Idiom name="Aw2B-01">
    <Viewpoint type="EXTERNAL" shot="CLOSEUP" actor="0" />
    <Viewpoint type="INTERNAL" shot="LONG" actor="0" />
    <Viewpoint type="EXTERNAL" shot="CLOSEUP" actor="1" />
  </Idiom>

  <Idiom name="Aw2B-02">
    <Viewpoint type="INTERNAL" shot="LONG" actor="ANY" />
    <Viewpoint type="EXTERNAL" shot="CLOSEUP" actor="0" />
    <Viewpoint type="EXTERNAL" shot="CLOSEUP" actor="1" />
  </Idiom>

  <!-- A passes through the door -->

  <Idiom name="Apttd-01">
    <Viewpoint type="BACK" actor="0" shot="LONG" />
    <Viewpoint type="3/4 FRONT" actor="0" shot="CLOSEUP" />
  </Idiom>

  <Idiom name="Apttd-02">
    <Viewpoint type="3/4 BACK" actor="0" shot="LONG" />
    <Viewpoint type="3/4 FRONT" actor="0" shot="CLOSEUP" />
  </Idiom>

</Rules>
```

- **Action:** describe a type of action (or narrative element) by a list of idioms that can convey it, ordered by priority. The name of the action type is built such that each capital letter (A, B, C, etc.) refers to one character.
- **Idiom:** describes an idiom by specifying a series of characteristic viewpoints represented by the Semantic Volumes defined in Figure 2.



1.2.3 Events

```
<Events>  
  
<Action name="A walks to B" begin="0" end="10" relevance="1">  
  <Character name="Ben" />  
  <Character name="Alex" />  
</Action>  
  
<Action name="A passes through the door" begin="2" end="4.5" relevance="2">  
  <Character name="Ben" />  
</Action>  
  
</Events>
```

- **Action:** specifies the set of actions (or narrative elements) that occur in the environment. Each narrative element is defined by its duration, relevance (the higher the more important) and the characters involved in the actions. In the name of the action, by norm, A represents the first character, B represents the second one.



2. Study of 1984

An important objective of WP5 is to *allow the evaluation of competing cinematic idioms using more intensive constraint-based formalisms*. By the application of appropriate cinematic idioms taken from a variety of sources including Arijon, Katz and Mascelli and including manipulation of lighting configurations, the camera can take an active part in the storytelling process.

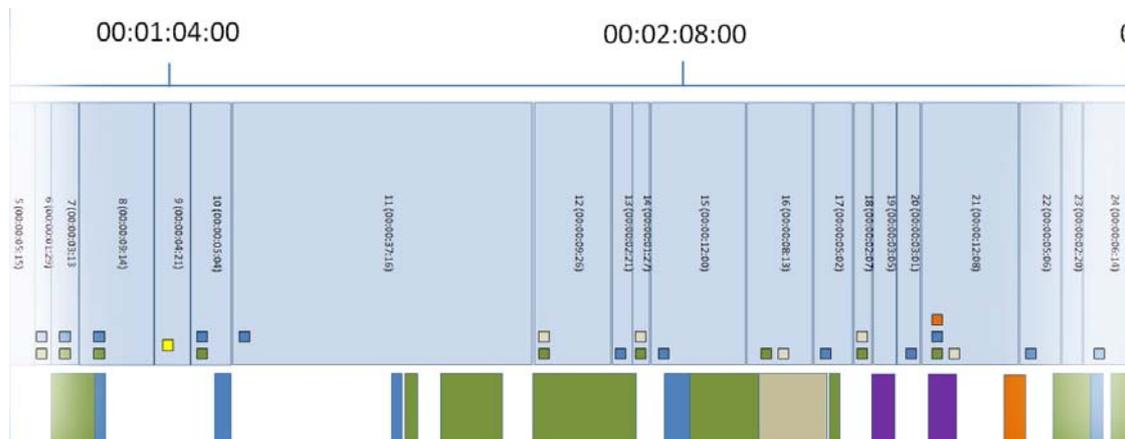
2.1.1 Analysis of 1984

As detailed in D5.1, the Canteen Scene in Michael Radford's 1984 was analysed with reference as to how cinematography was used not only to make clear the spatial configuration of the scene but to expose narrative factors such as relationships between characters. The main reason for selecting this particular scene was the comparative lack of clear and significant actions on the part of the characters (the scene ostensibly comprises a conversation where the characters eat, drink and chat about their day to day lives). Nevertheless, over the 4:12 minutes of the scene, the viewer learns important facts about the characters, the world they inhabit and their relationships to each other. Much of this narrative information is realised through the use of established cinematographic techniques.

Initially, the scene was broken down shot-by-shot using video-editing software. The length of each cut, the relationship of framing to dialogue and action content, editing and lighting configurations were all noted. Next an exhaustive list of individual actions performed by characters within the scene was drawn up, completely independent of cinematographic considerations.



Fig 9. Section of the complete list of actions performed by characters during the canteen scene



Key: Smith X, Syme X, Parsons X, Telescreen X, OP Member X

Fig 10. Part of the shot-length diagram of the canteen scene showing relationship of dialogue to editing.

The key objectives in this analysis were twofold. The first task was to identify themes which were crucial to the development of the story, such as the relationship of individual characters to each other and their roles in the world described by the narrative. The second was to note how cinematographic techniques were employed to develop these themes, with attention given to factors such as camera placement (including the employment of commonly used idioms), framing, editing, and lighting.

2.1.2 Identifying Relevant Themes.

As a useful step in developing a constraint-based approach to cinematography, certain qualities important to the viewer's understanding of the dramatic outcome of the scene were selected. Isolation, dominance of one character over another and characters' affinity for each other were selected as factors which could be manipulated to affect a viewer's perception of inter-character relationships in the scene, substantially changing the dramatic outcomes of a section of the narrative independently of dialogue and other character actions.

In addition to these factors, constraints were developed which would allow the cinematography to support the development of a particular character as the protagonist of the story. From research undertaken prior to D5.1 it was noted that in addition to standard cinematographic conventions used to establish spatial configuration of a scene and establish a legible temporal flow, the following rules could be used to develop these qualities:-

A. Isolation

One of the central themes of the film is Smith's disillusionment with and consequent isolation from the party. Throughout the film this is related partly through dialogue but is reinforced by the way Radford places and frames him in certain scenes. In the canteen scene he is arranged literally 'back to the wall', in a corner seat, alone in the frame, with only Big Brother's image on the telescreen behind him. His companions by comparison are always pictured surrounded by other party members. Techniques which could be used to reinforce isolation include:



- Pairing a shot in which a character is alone (spatially isolated) with a shot in which multiple characters are present.
- Keeping the camera on the character for longer than usual (e.g. while another character is speaking).
- Framing a character so that he/she appears smaller than his counterparts
- Framing a character so that he/she is lower in the frame than other characters.

B. *Dominance*

Domination of a particular conversation by one character lends weight to the character's words, increasing the perceived importance of their part of the dialogue. In the canteen scene Syme dominates the conversation, reinforcing the idea that the party's views override other considerations in the society depicted in the film. (N.B. Dominance here refers to assuming a position of superiority over other characters in specific dialogues rather than in the narrative as a whole). Techniques used to expose this include:

- Using interior shots for the dominant character and exterior shots for the dominated.
- Placing the dominant character higher in the frame than the dominated. Standing characters usually dominate seated characters.
- Making the dominant character closer to the camera (and therefore larger in the frame).
- Angling the camera more favorably towards the dominant character.
- Increasing the light level on the dominant character.
- Slightly raising the volume of the dominant character's dialogue.

C. *Affinity*

Establishing affinity between characters, for example in identifying a character's friends versus his enemies is often vital to the viewer's understanding of the characters' motivations. Techniques used to establish affinity between characters include:

- Symmetrical composition of shots imply affinity between characters.
- Similarity in composition of shots (in terms of background, light level etc.) suggest affinity.
- Maintaining a consistent level eye-line between characters.
- Similar volume levels during dialogue imply affinity between characters.
- External Shots and especially Apex shots also imply a connection between characters.

D. *Determining the Protagonist*

In the canteen scene it can be clearly seen that although Smith is clearly the hero of the film, he contributes little in terms of dialogue or decisive actions. It is left to the cinematography to identify him to the viewer.

- Coverage of a character engaged in actions not central to the plot.
- Extending the amount of coverage (in both length and number of shots) of a particular character in relation to others.



- Increasing in particular the number of reaction shots on a particular character.
- Framing a particular character to appear larger than others.

2.1.3 Software Implementation

In developing a prototype with which to demonstrate and experiment with these rules, a real-time 3D model of the scene was built using 3D Studio Max. Built from careful study of Radford's scene, this model mimics the configuration of the space, is textured and lit in a similar way and includes characters representing Smith, Syme, Parsons, Julia and the extras in the scene. Dialogue tracks for each character were composed using audio from the film and stock sources and the characters were animated to follow the motions of the actors.



Fig 11. 3D Version of the Canteen Scene running in Ogre

Running in the Ogre open source 3D engine, camera placement and editing is parameterised in XML allowing the setup of constraints prior to runtime which select between a number of predetermined camera positions, using them to construct alternative versions of the scene. Alternative configurations presented in the prototype version include:

A. *Making Syme seem isolated instead of Smith*

- Change the position of the cameras so that Syme is spatially isolated instead of Smith.
- Extend the length of shots focussing on Syme. Keep the camera on Syme while he is eating, drinking, or watching the screen.



B. Making Syme dominant in the conversation.

- Use an interior shot for Syme and exterior shot for Smith.
- Bring the camera closer to Syme (making him larger in the frame) than Smith.
- Angle the camera more directly on Syme.
- Increase lighting on Syme in relation to the other characters
- Raise the volume of Syme's voice slightly over Smith.
- Keep the camera slightly longer on Syme.

C. Show affinity between Smith and Syme

- Show Syme when he is talking, rather than Smith, or show both of them.
- When the OP member intrudes, include a view of both of them looking at him, rather than just Syme.
- Use Apex or External Shots.
- Remove the shots of Julia.
- Show both of them when they look at the screen and listen.

D. Making Syme the hero instead of Smith.

- Do not show Smith pouring and drinking gin at the beginning. Instead show Syme.
- Include shots which imply that Syme dominates Smith.
- Raise the volume of Syme's dialogue slightly over that of Smith.

2.1.4 XML Parameterization

Below is an example of the XML schema used for specifying constraints for Dominance during the canteen scene.

```
<ShotStyles>
  <ShotStyle name="1-Person">
    <Characters>
      <Character type="Main">
        <ShotParameters>
          <Size minHeight="0.6" maxHeight="0.8" calibration="10" />
          <Position minX="-1" maxX="1" minY="-1" maxY="1" calibration="10" />
          <Orientation min="10" max="20" calibration="10" />
        </ShotParameters>
      </Character>
    </Characters>
    <Camera pitch="10" elevation="10" fieldOfView="45"></Camera>
  </ShotStyle>
  <ShotStyle name="2-Person">
```



```
<Characters>
  <Character type="Dominant">
    <ShotParameters>
      <Size minWidth="0.7" maxWidth="0.8" minHeight="0.8" maxHeight="1"
calibration="10" />
      <Position minX="-1" maxX="1" minY="-1" maxY="1" calibration="10" />
      <Orientation min="180" max="170" />
    </ShotParameters>
  </Character>
  <Character type="Dominated">
    <ShotParameters>
      <Size minWidth="0.2" maxWidth="0.4" minHeight="0.4" maxHeight="0.6"
calibration="10" />
      <Position minX="" maxX="" minY="" maxY="" calibration="" />
      <Orientation min="10" max="20" calibration="10" />
    </ShotParameters>
  </Character>
</Characters>
<Camera pitch="30" elevation="30" fieldOfView="45"></Camera>
</ShotStyle>
<ShotStyle name="Any">
  <Characters />
  <Camera pitch="10" elevation="10" fieldOfView="45"></Camera>
</ShotStyle>
</ShotStyles>
<Shots>
  <Shot name="SmithPoursGin" style="1-Person" startTime="0" duration="4">
    <Characters>
      <Character type="Main" nodeName="SmithHead" />
    </Characters>
  </Shot>
  <Shot name="SymeWatchesScreen" style="1-Person" startTime="4" duration="8">
    <Characters>
      <Character type="Main" nodeName="SymeHead" />
    </Characters>
  </Shot>
</Shots>
```

Shot Parameters include x and y positions in the frame, camera pitch and field of view settings. These parameters allow the specification of a character's size in the frame. Different framings can be specified depending on the desired effect.

Character type allows idioms to be selected based upon whether the character is to be considered for a protagonist, dominant or dominated character.

Shot Style allows specification of a particular idiom.



Shot Name, Start Time and **Duration** allow identification of the shot and the length of the cut.

NodeName specifies which character to apply shot parameters to.



Fig 12 Example shots composed by the system.



3. References

- [Ari76] ARIJON D.: Grammar of the Film Language. Hastings House Publishers, 1976.
- [AVF04] ANDÚJAR C. G., VÁZQUEZ P. P. A., FAIRÉN M. G.: Way-finder: Guided tours through complex walkthrough models. Proceedings of the Eurographics 2004 Conference (EG 2004) 23, 3 (2004).
- [Bec02] BECKHAUS S.: Dynamic Potential Fields for Guided Exploration in Virtual Environments. PhD thesis, Fakultät für Informatik, University of Magdeburg, 2002.
- [BHS98] BITTNER J., HAVRAN V., SLAVIK P.: Hierarchical visibility culling with occlusion trees. Computer Graphics International Conference 0 (1998).
- [BK01] BARES W., KIM B.: Generating virtual camera compositions. In Proc. of the 6th international conference on Intelligent user interfaces (IUI 01) (2001), ACM Press.
- [CAWH_96] CHRISTIANSON D. B., ANDERSON S. E., WEI HE L., SALESIN D. H., WELD D. S., COHEN M. F.: Declarative Camera Control for Automatic Cinematography. In Proc. of the 13th National Conference on Artificial Intelligence and the 8th Innovative Applications of Artificial Intelligence Conference (1996), AAAI Press / MIT Press.
- [CF92] CHIN N., FEINER S.: Fast object-precision shadow generation for area light sources using bsp trees. In SIGGRAPH '92: Proc. of the 1992 symposium on Interactive 3D graphics (1992), ACM.
- [CLDM03] COURTY N., LAMARCHE F., DONIKIAN S., MARCHAND E.: A cinematography system for virtual storytelling. In Proc. of the International Conference on Virtual Storytelling 2003 (November 2003), vol. 2897.
- [CM01] COURTY N., MARCHAND E.: Computer animation: A new application for image-based visual servoing. In Proc. of International Conference on Robotics and Automation, (ICRA 2001) (2001).
- [CN05] CHRISTIE M., NORMAND J.-M.: A semantic space partitioning approach to virtual camera control. In Proc. of the Eurographics Conference (EG 2005) (2005), vol. 24, Computer Graphics Forum.
- [CON08] CHRISTIE M., OLIVIER P., NORMAND J.-M.: Occlusion-free Camera Control. Research Report RR6640, INRIA, 2008.
- [ER07] ELSON D. K., RIEDL M. O.: A lightweight intelligent virtual cinematography system for machinima production. In Proc. of the 3rd Conf. on AI for Interactive Entertainment (2007).
- [FF04] FRIEDMAN D. A., FELDMAN Y. A.: Knowledge-Based Cinematography and Its Applications. In Proc. Of the 16th European Conference on Artificial Intelligence (ECAI 2004) (2004), IOS Press.
- [HCS96] HE L., COHEN M. F., SALESIN D. H.: The virtual cinematographer: A paradigm for automatic realtime camera control and directing. In SIGGRAPH (Aug. 1996), ACM Computer Graphics.
- [Jha09] JHALA A.: Cinematic Discourse Generation. PhD thesis, Faculty of North Carolina State University, 2009.
- [Lam09] LAMARCHE F.: Topoplan: a topological path planner for real time human navigation under floor and ceiling constraints. Computer Graphics Forum 28, 2 (2009).
- [LC08] LI T.-Y., CHENG C.-C.: Real-time camera planning for navigation in virtual environments. In SG '08: Proc. of the 9th international symposium on Smart Graphics (2008), Springer-Verlag.
- [NAT90] NAYLOR B., AMANATIDES J., THIBAUT W.: Merging bsp trees yields polyhedral set operations. In SIGGRAPH '90: Proc. of the 17th annual conference on Computer graphics and interactive techniques (1990), ACM.



[NO03] NIEUWENHUISEN D., OVERMARS M. H.: Motion Planning for Camera Movements in Virtual Environments. Tech. Rep. UU-CS-2003-004, Institute of Information and Computing Sciences, Utrecht University, 2003.

[OSTG09] OSKAM T., SUMNER R. W., THUREY N., GROSS M.: Visibility transition planning for dynamic camera control. In SCA '09: Proc. of the 2009 ACM SIGGRAPH/Eurographics symposium on Computer animation (2009), Eurographics Association.

[TS91] TELLER S. J., SÉQUIN C. H.: Visibility preprocessing for interactive walkthroughs. In SIGGRAPH '91: Proc. of the 18th annual conference on Computer graphics and interactive techniques (1991), ACM.

[VFSH03] VÁZQUEZ P.-P., FEIXAS M., SBERT M., HEIDRICH W.: Automatic view selection using viewpoint entropy and its application to image-based modelling. Computer Graphics Forum 22, 4 (2003).

[You07] YOUNG R. M.: Story and discourse: A bipartite model of narrative generation in virtual worlds. Interaction Studies 8, 2 (2007).