The Prose Storyboard Language

A Tool for Annotating and Directing Movies

Remi Ronfard
INRIA/LJK, Grenoble, France
remi.ronfard@inria.fr

Vineet Gandhi
INRIA/LJK, Grenoble, France
vineet.gandhi@inria.fr

Laurent Boiron
INRIA/LJK, Grenoble, France
laurent.boiron@inria.fr

ABSTRACT

The prose storyboard language is a formal language for describing movies shot by shot, where each shot is described with a unique sentence. The language uses a simple syntax and limited vocabulary borrowed from working practices in traditional movie-making, and is intended to be readable both by machines and humans. The language is designed to serve as a high-level user interface for intelligent cinematography and editing systems.

Categories and Subject Descriptors

Applied computing [**Arts and humanities**]: Media arts

General Terms

Film Theory, Machinima, Context-Free Grammar, Timed Petri Net

Keywords

Virtual Cinematography

1. INTRODUCTION

In movie production, directors often use a semi-formal idiom of natural language to convey the shots they want to their cinematographer. Similarly, film scholars use a semi-formal idiom of natural language to describe the visual composition of shots in produced movies to their readers. In order to build intelligent and expressive virtual cinematography and editing systems, we believe the same kind of high-level descriptions need to be agreed upon. In this paper, we propose a formal language that can serve that role. Our primary goal in proposing this language is to build software cinematography agents that can take such formal descriptions as input, and produce fully realized shots as an output [7, 25]. A secondary goal is to perform in-depth studies of film style by automatically analyzing real movies and their scripts in terms of the proposed language [26, 24].

The prose storyboard language is a formal language for describing shots visually. We leave the description of the sound-track for future work. The prose storyboard language separately describes the spatial structure of individual movie frames (compositions) and their temporal structure (shots). Our language defines an infinite set of sentences (shot categories), which can describe an infinite set of shots.

In film analysis, there is a frequent confusion between shots and compositions. A *medium shot* describes a composition, not a shot. If the actor moves towards the camera in the same shot, the composition will change to a *close shot* and so on. Therefore, a general language for describing shots cannot be limited to describing compositions such as *medium shot* or *close shot* but should also describe screen events which change the composition during the shot.

Our language can be used indifferently to describe shots in pre-production (when the movie only exists in the screen-writer and director's minds), during production (when the camera records a continuous "shot" between the times when the director calls "camera" and "cut"), in post-production (when shots are cut and assembled by the film editor) or to describe existing movies. The description of an entire movie is an ordered list of sentences, one per shot. Exceptionally, a movie with a single shot, such as Rope by Alfred Hitchcock, can be described with a single, long sentence.

In this paper, we assume that all shot descriptions are manually created. We leave for future work the important issue of automatically generating prose storyboards from existing movies, where a number of existing techniques can be used [4, 32, 9, 12, 11]. We also leave for future work the difficult problems of automatically generating movies from their prose storyboards, where existing techniques in virtual camera control can be used [13, 6, 15, 8].

2. PRIOR ART

Our language is loosely based on existing practices in movie-making [31, 30] and previous research in the history of film style [3, 28]. Our language is also related to the common practice of the graphic storyboard. In a graphic storyboard, each composition is illustrated with a single drawing. The blocking of the camera and actors can be depicted with a conventional system of arrows within each frame, or with a separate set of floor plan views, or with titles between frames.

In our case, the transitions between compositions use a small vocabulary of screen events including camera actions (pan, dolly, crane, lock, continue) and actor actions (speak, react, move, cross, use, touch). Although the vocabulary could easily be extended, we voluntarily keep it small because our focus in this paper is restricted to the blocking of actors and cameras, not the high-level semantics of the narrative.

We borrow the term prose storyboard from Proferes [22] who used it as a technique decomposing a films script into a sequence of shots, expressed in natural language The name catches the intuition that the language should directly translate to images. In contrast to Proferes, our prose storyboard language is a formal language, with a well defined syntaxx and semantics, suitable for future work in intelligent cinematography and editing.

Our proposal is complementary to the Movie Script Markup Language (MSML) [23], which encodes the structure of a movie script. In MSML, a script is decomposed into dialogue and action blocks, but does not describe how each block is translated into shots. Our prose storyboard language can be used to describe the blocking of the shots in a movie in relation to an MSML-encoded movie script.

Our proposal is also related to the Declarative Camera Control Language (DCCL) which describes film idioms, not in terms of cameras in world coordinates but in terms of shots in screen coordinates [6]. The DCCL is compiled into a film tree, which contains all the possible editings of the input actions, where actions are represented as subject-verb-object triples. Our prose storyboard language can be used in coordination with such constructs to guide a more extensive set of shot categories, including complex and composite shots.

Our approach is also related to the work of Jhala and Young who used the movie Rope by Alfred Hitchcock to demonstrate how the story line and the director's goal should be represented to an automatic editing system [15]. They used Crossbow, a partial order causal link planner, to solve for the best editing, according to a variety of strategies, including maintaining tempo and depicting emotion. They demonstrated the capability of their solver to present the same sequence in different editing styles. But their approach does not attempt to describe the set of possible shots. Our prose storyboard language attempts to fill that gap.

Other previous work in virtual cinematography [29, 14, 10, 21, 16, 18] has been limited to simple shots with either a static camera or a single uniform camera movement. Our prose storyboard language is complementary to such previous work and can be used to define higher-level cinematic strategies, including arbitrarily complex combinations of camera and actor movements, for most existing virtual cinematography systems.

3. REQUIREMENTS

The prose storyboard language is designed to be expressive, i.e. it should describe arbitrarily complex shots, while at the same being compact and intuitive. Our approach has been to keep the description of simple shots as simple as possible, while at the same time allowing for more complex descriptions when needed. Thus, for example, we describe actors in

a composition from left to right, which is an economical and intuitive way of specifying relative actor positions in most cases. As a result, our prose storyboard language is very close to natural language (see Fig.6).

It should be easy to parse the language into a non ambiguous semantic representation that can be matched to video content, either for the purpose of describing existing content, or for generating novel content that matches the description. It should therefore be possible (at least in theory) to translate any sentence in the language into a sketch storyboard, then to a fully animated sequence.

It should also be possible (at least in theory) to translate existing video content into a prose storyboard. This puts another requirement on the language, that it should be possible to describe existing shots just by watching them. There should be no need for contextual information, except for place and character names. As a result, the prose storyboard language can also be used as a tool for annotating complete movies and for logging shots before post-production. Since the annotator has no access to the details of the shooting plan, even during post-production [19, 20], we must therefore make it possible to describe the shots in screen coordinates, without any reference to world coordinates.

4. SYNTAX AND SEMANTICS

The prose storyboard language is a context-free language, whose terminals include generic and specific terms. Generic terminals are used to describe the main categories of screen events including camera actions (pan, dolly, cut, dissolve, etc.) and actor actions (enter, exit, cross, move, speak, react, etc.). Specific terminals are the names of characters, places and objects that compose the image and play a part in the story. Non-terminals of the language include important categories of shots (simple, complex, composite), sub-shots and image compositions. The complete grammar for the language is presented in Fig.8 in extended BNF notation.

The semantics of the prose storyboard language is best described in terms of a Timed Petri Net (TPN) where actors and objects are represented as *places* describing the composition ; and screen events (such as cuts, pans and dollies) are represented as Petri net *transitions*. TPNs have been proposed for representing the temporal structure of movie scripts [23], character animation [17, 2], game authoring [1] and turn-taking in conversation [5]. Transitions in a TPN usually have non-zero duration. In some cases, transitions change the composition from an initial state to a final state. In those cases, the language does not attempt to describe the composition during the transition. In other cases, transitions maintain the composition while they are executed. Between transitions, the screen composition can be infered by inspecting the state of the TPN places (actors and objects) and their attributes between successive screen events. This TPN semantics is useful for generating prose storyboards from movies, and for generating movies from prose storyboards. This will be described in future work.

5. IMAGE COMPOSITION

Image composition is the way to organise visual elements in the motion picture frame in order to deliver a specific message to the audience. In our work, we propose a formal
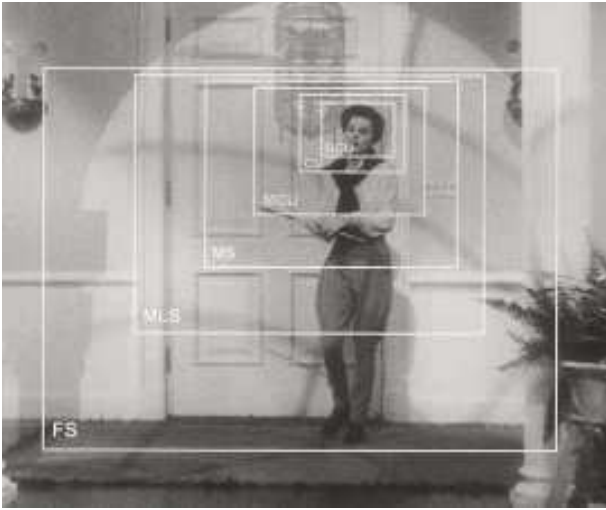
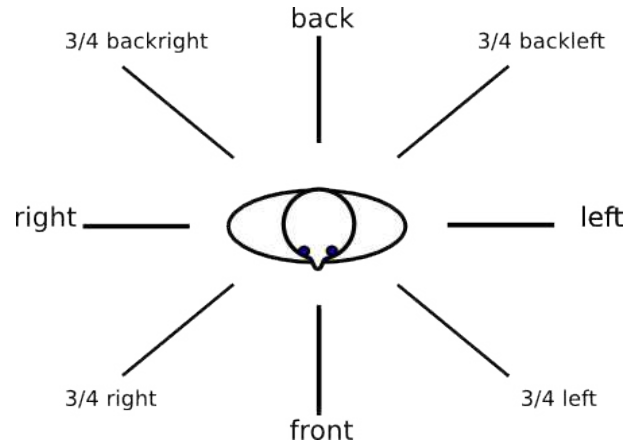**Figure 1: Shot sizes in the prose storyboard (reproduced from [27]).**

Figure 2: The profile angle of an actor defines his orientation relative to the camera. For example, an actor with a *left* profile angle is oriented with his left side facing the camera.

way to describe image composition in terms of the actors and objects present on the screen and the spatial and temporal relations between them.

Following Thomson and Bowen [31], we define a composition as the relative position and orientation of isual elements called *Subject*, in screen coordinates.

In the simple case of *flat staging*, all subjects are more or less in the same plane with the same size. Thus, we can describe this *flat composition* as follows:

```
<Size> on <Subject> [<Profile>][<Screen>]
{ and <Subject> [<Profile>][<Screen] }*
```

where *Size* is one of the classical shot size illustrated in Fig.1 and *Subject* is generally an actor name. The *Profile* and *Screen* terms describe respectively the orientation and the position of the subject in screen space. See full grammar in Fig.8 for more details.

In the case of *deep staging*, different subjects are seen at different sizes, in different planes. We therefore describe such shot with a *stacking* of flat compositions as follows:

```
<FlatComposition> { , <FlatComposition>}*
```

As a convention, we assume that the subjects are described from left to right. This means that the left-to-right ordering of actors and objects is part of the composition. Indeed, because the left-to-right ordering of subjects is so important in cinematography and film editing, we introduce a special keyword *cross* for the screen event of one actor crossing over or under another actor.

Shot sizes are used to describe the relative sizes of actors independently of the camera lens as illustrated in Fig.3.

The *Screen* term describe the subject position in screen coordinate. It allows to slightly modify the generic framing by shifting the subject position to the left or right corner as

shown in Fig.4. Thus, we can describe a more harmonious composition with respect to the head room and look room or the rules of thirds. We can also describe unconventional framing to create unbalanced artistic composition or to show other visual elements from the scene.

6. SHOT DESCRIPTIONS

Based on the taxonomy of shots proposed by Thomson and Bowen [31], our prose storyboarding language distinguishes three main categories of shots :

- A simple shot is taken with a camera that does not move or turn. If the composition changes during a simple shot, it can only be the effect of actors movement relative to the camera.

- A complex shot is taken with a camera that can pan, tilt and zoom from a fixed position. We introduce a single camera action (pan) to describe all such movements. Thus the camera can pan left and right, up and down (as in a tilt), and in and out (as in a zoom).

- A composite shot is taken with a moving camera. We introduce two camera actions (dolly and crane) to describe typical camera movements. Panning is of course allowed during dolly and crane movements.

In each of those three cases, we propose a simplified model of a shot, consisting in a sequence of compositions and screen events. Screen events can be actions of the camera relative to the actors, or actions of the actors relative to the camera. A shot is therefore a sequence of compositions and screen events. Screen events come in two main categories - those which change the composition (*events-to-composition*) and those which maintain the composition (*events-with-composition*).

In our model, we can be in only one of four different states:

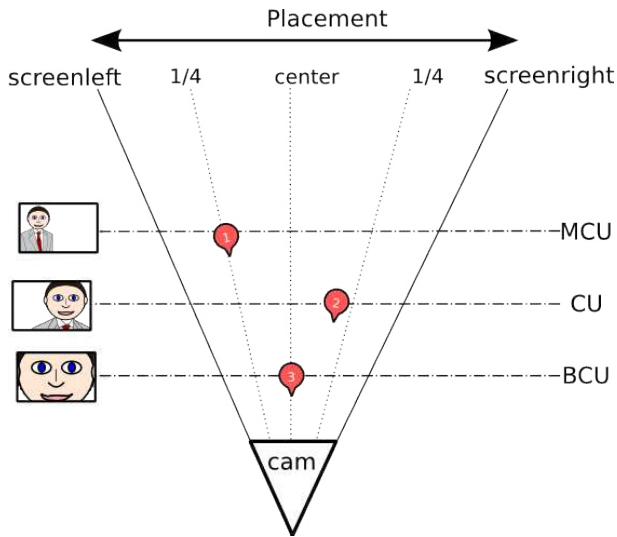1. Camera does not move and composition that does not change.

Figure 3: Shot size is a function of the distance between the camera and actors, as well as the camera focal length.
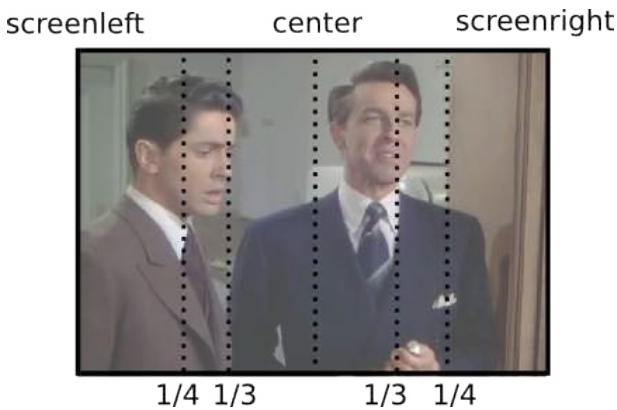

Figure 4: The horizontal placement of actors in a composition is expressed in screen coordinates.

2. Camera does not move and composition changes due to actor movements.

3. Camera moves and composition does not change

4. Camera moves and composition changes.

In case (1), the shot can be described with a single composition. In case (2), we introduce the special verb *lock* to indicate that the camera remains static while the actors move, leading to a new composition. In case (3), we use the constructions *pan with*, *dolly with* and *crane with* to indicate how the camera moves to maintain the composition. In case (4), we use the constructions *pan to*, *dolly to* and *crane to* to indicate how the cameras moves to change the composition, and we introduce a special verb *continue to* to indicate that the composition changes due to a combination of actor and camera movements. All three cases are illustrated in Fig.5, Fig.6 and Fig.7.
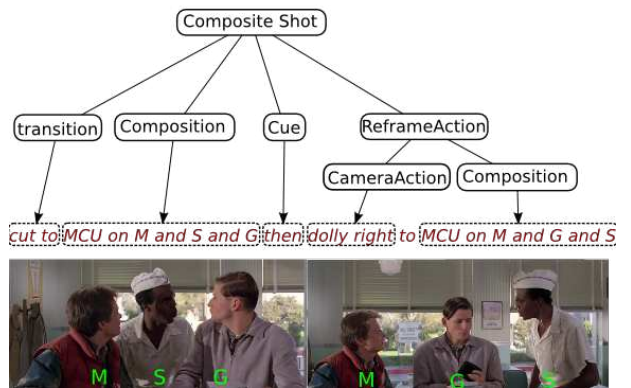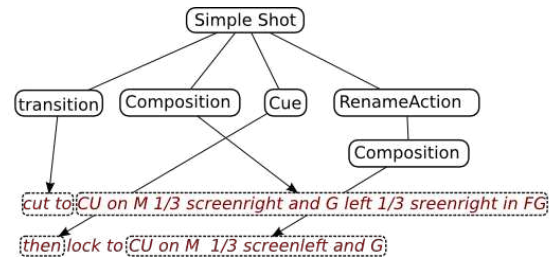

Figure 5: Two shot from the movie *Back to the future* with the description in *Prose Storyboard Language* (red) and their associated parse trees. Top : Simple Shot with a fixed camera. The composition changes due to actor movements. Bottom: Composite shot. The composition changes due to camera motion. In both examples the actors are named by the corresponding letters in green.

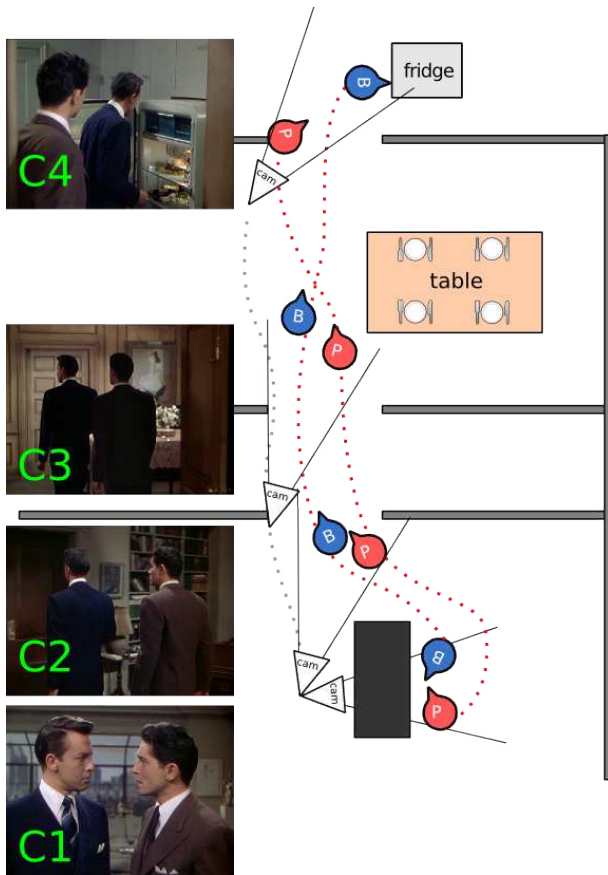Figure 6: Prose storyboard for two short sequences from the single-shot movie Rope.

Figure 7: Floor plan reconstruction for a short sequence of the single-shot movie Rope, corresponding to the prose storyboard in Fig.6

7. EXPERIMENTAL RESULTS

As a validation of the proposed language, we have manually annotated scenes from existing movies covering different styles and periods. In this paper, we illustrate the prose storyboard language on a particularly difficult example, a short sequence from the single-shot movie Rope by Alfred Hitchcock, where actor and camera movements interfere to produce a rich and dynamic visual composition. The prose storyboard for the example is presented in Fig.7 and Fig.6. Despite the complexity of the scene, the prose storyboard is quite readable and was relatively easy to generate.

This example also illustrates how default values are used to describe simple cases. For instance, default values for a two-shot are supplied by assuming that the two actors are both facing the camera and placed at one-third and two-third of the screen. Such default values are stored in a stylesheet, which can be used to accommodate different cinematographic styles, i.e. television vs. motion pictures, or musical vs. film noir.

8. CONCLUSION

We have presented a language for describing the spatial and temporal structure of movies with arbitrarily complex shots. The language can be extended in many ways, e.g. by taking into account lens choices, depth-of-field and lighting. Future work will be devoted to the dual problems of automatically generating movies from prose storyboards in machinima environments, and automatically describing shots in existing movies. We are also planning to extend our framework for the case of stereoscopic movies, where image composition needs to be extended to include the depth and disparity of subjects in the composition. We believe that the proposed language can be used to extend existing approaches in intelligent cinematography and editing towards more expressive strategies and idioms and bridge the gap between real and virtual movie-making.

9. REFERENCES

[1] D. Balas, C. Brom, A. Abonyi, and J. Gemrot. Hierarchical petri nets for story plots featuring virtual humans. In *AIIDE*, 2008.

[2] L. Blackwell, B. von Konsky, and M. Robey. Petri net script: a visual language for describing action, behaviour and plot. In *Australasian conference on Computer science*, ACSC '01, 2001.

[3] D. Bordwell. *On the History of Film Style*. Harvard University Press, 1998.

[4] M. Brand. The "inverse hollywood problem": From video to scripts and storyboards via causal analysis. In *AAAI/IAAI*, pages 132–137, 1997.

[5] C. Chao. Timing multimodal turn-taking for human-robot cooperation. In *Proceedings of the 14th ACM international conference on Multimodal interaction*, ICMI '12, pages 309–312, New York, NY, USA, 2012. ACM.

[6] D. B. Christianson, S. E. Anderson, L. wei He, D. H. Salesin, D. S. Weld, and M. F. Cohen. Declarative camera control for automatic cinematography. In *AAAI*, 1996.

[7] M. Christie, C. Lino, and R. Ronfard. Film Editing for Third Person Games and Machinima. In R. M. Y.

Arnav Jhala, editor, *Workshop on Intelligent Cinematography and Editing*, Raleigh, United States, May 2012. ACM.

[8] M. Christie and P. Olivier. Camera control for computer graphics. In *Eurographics State of the Art Reports*, Eurographics 2006. Blackwell, 2006.

[9] R. Dony, J. Mateer, and J. Robinson. Techniques for automated reverse storyboarding. *IEE Journal of Vision, Image and Signal Processing*, 152(4):425–436, 2005.

[10] D. Friedman and Y. A. Feldman. Automated cinematic reasoning about camera behavior. *Expert Syst. Appl.*, 30(4):694–704, May 2006.

[11] V. Gandhi and R. Ronfard. Detecting and naming actors in movies using generative appearance models. In *CVPR*, 2013.

[12] D. B. Goldman, B. Curless, D. Salesin, and S. M. Seitz. Schematic storyboarding for video visualization and editing. *ACM Trans. Graph.*, 25(3):862–871, 2006.

[13] L.-w. He, M. F. Cohen, and D. H. Salesin. The virtual cinematographer: a paradigm for automatic real-time camera control and directing. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, SIGGRAPH '96, pages 217–224, New York, NY, USA, 1996. ACM.

[14] A. Jhala and R. M. Young. A discourse planning approach to cinematic camera control for narratives in virtual environments. In *AAAI*, 2005.

[15] A. Jhala and R. M. Young. Representational requirements for a plan based approach to automated camera control. In *AIIDE'06*, pages 36–41, 2006.

[16] C. Lino, M. Chollet, M. Christie, and R. Ronfard. Computational model of film editing for interactive storytelling. In *ICIDS*, pages 305–308, 2011.

[17] L. P. MagalhŃes, A. B. Raposo, and I. L. Ricarte. Animation modeling with petri nets. *Computers and Graphics*, 22(6):735 – 743, 1998.

[18] D. Markowitz, J. T. K. Jr., A. Shoulson, and N. I. Badler. Intelligent camera control using behavior trees. In *MIG*, pages 156–167, 2011.

[19] W. Murch. *In the blink of an eye.* Silman-James Press, 1986.

[20] M. Ondaatje. *The Conversations: Walter Murch and the Art of Film Editing.* Random House, 2004.

[21] B. O'Neill, M. O. Riedl, and M. Nitsche. Towards intelligent authoring tools for machinima creation. In *CHI Extended Abstracts*, pages 4639–4644, 2009.

[22] N. Proferes. *Film Directing Fundamentals - See your film before shooting it.* Focal Press, 2008.

[23] D. V. Rijsselbergen, B. V. D. Keer, M. Verwaest, E. Mannens, and R. V. de Walle. Movie script markup language. In *ACM Symposium on Document Engineering*, pages 161–170, 2009.

[24] R. Ronfard. Reading movies: an integrated dvd player for browsing movies and their scripts. In *ACM Multimedia*, 2004.

[25] R. Ronfard. A Review of Film Editing Techniques for Digital Games. In R. M. Y. Arnav Jhala, editor, *Workshop on Intelligent Cinematography and Editing*, Raleigh, United States, May 2012. ACM.

[26] R. Ronfard and Thuong. A framework for aligning and indexing movies with their script. In *International Conference on Multimedia and Expo*, 2003.

[27] B. Salt. *Moving Into Pictures.* Starword, 2006.

[28] B. Salt. *Film Style and Technology: History and Analysis (3 ed.).* Starword, 2009.

[29] J. Shen, S. Miyazaki, T. Aoki, and H. Yasuda. Intelligent digital filmmaker dmp. In *ICCIMA*, 2003.

[30] R. Thompson and C. Bowen. *Grammar of the Edit.* Focal Press, 2009.

[31] R. Thompson and C. Bowen. *Grammar of the Shot.* Focal Press, 2009.

[32] E. Veneau, R. Ronfard, and P. Bouthemy. From video shot clustering to sequence segmentation. In *ICPR*, pages 4254–4257, 2000.

```
<Scene>  ::=  <Shot> *

<Cue>    ::=  At <timeref> | As <Actor> <Action> | then


<Shot>   ::=  [<transition >] to [<Camera>] <Composition> {<Fragment>}* |
              <transition > <Camera>


<Fragment>        ::=  <Cue> (<RenameAction> | <ReframeAction >)

<RenameAction>   ::=  (lock | continue) to <Composition>

<ReframeAction> ::=  <CameraAction> (to | with) <Composition>


<Composition>     ::=  [<angle >] <FlatComposition >{, <FlatComposition >}*

<FlatComposition >::=  <size>   on <Subject >[ <profile >][ <screen >]
        { and <Subject >[ <profile >][ <screen >][in (back | fore)ground]}*


<CameraAction >::=  [Speed]   pan [left | right | up | down]
                        | dolly [in | out | left | right ]
                        | crane [up | down]

<Speed>   ::=  slow | quick | following :(<Actor >|<Object >)

<Subject>  ::=  (<Actor >|<Object >)  |  (<Actor >|<Object >){, (<Actor >|<Object >)}+


<transition >   ::=  cut | dissolve | fade in

<angle >          ::=  (high | low) angle

<size >           ::=  ECU|BCU|CU|MCU|MS|MLS|FS|LS|ELS

<profile >        ::=  34 leftback | left | 34 left | front | 34 right | right | 34 leftback | back

<screen >         ::=  center | [(13 | 14)] screen (left | right )


<Action>  ::=  <Look>|<Move>|<Speak>|<Use>|<Cross>|<Touch>|<React>

<Look>    ::=  looks at <Subject>

<Move>    ::=  moves to <Screen >|<Place >|<Subject >

<Speak>   ::=  (speaks  |  says <string >) [ to <Subject >]

<Use>     ::=  uses <Object >

<Cross>   ::=  crosses [over | under] <Subject >

<Touch>   ::=  touches <Subject >

<React>   ::=  reacts to <Subject >

<Place>   ::=  from script

<Actor>   ::=  from script

<Object>  ::=  from script
```

Figure 8: EBNF grammar of the prose storyboard language.