Film Directing for Computer Games and Animation

Rémi Ronfard

EUROGRAPHICS 2021 / N. Mitra and I. Viola
(Guest Editors)

Volume 40 (2021), Number 2

Film Directing for Computer Games and Animation

Rémi Ronfard

Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK & Ecole des Arts Décoratifs, EnsadLab, Paris, France

Abstract

Over the last forty years, researchers in computer graphics have proposed a large variety of theoretical models and computer implementations of a virtual film director, capable of creating movies from minimal input such as a screenplay or storyboard. The underlying film directing techniques are also in high demand to assist and automate the generation of movies in computer games and animation. The goal of this survey is to characterize the spectrum of applications that require film directing, to present a historical and up-to-date summary of research in algorithmic film directing, and to identify promising avenues and hot topics for future research.

CCS Concepts

• ***Computing methodologies*** → ***Animation; Scene understanding;*** • ***Applied computing*** → ***Performing arts;***

1. Introduction

This state of the art report surveys a long line of research in automatic film directing for computer games and animation. The notion of an automatic film director can be traced back to a conversation between Francois Truffaut and Alfred Hitchcock in 1966 where Hitchcock joked that he was dreaming of "a machine in which he (would) insert the screenplay at one end and the film would emerge at the other end" [Tru85]. In a keynote talk at the third Eurographics Workshop on Intelligent Cinematography and Editing in 2014 [RBJ14], Mark Riedl reiterated a similar vision when he proposed the grand challenge of automated filmmaking [Rie14] to researchers in computer graphics and animation. In this survey, our goal is to review previous work in automated film directing, to assess their contributions and limitations, and to propose new directions for future work.

A previous state of the art report [CON08] has investigated automatic camera control and virtual cinematography, leaving aside some important concepts in film directing such as decoupage, mise-en-en-scène and montage. In this survey, we would like to focus on those three inter-related aspects of film directing, with a careful review of thirty important papers covering forty years of research.

While it is traditional to separate cinematography from film editing when reviewing work in film directing, as in the Eurographics Workshop on Intelligent Cinematography and Editing series [RBJ14, RCB15, RCG16, BGGR17, WSJ18, CWLG20], this in fact raises difficulties. One important role of a (virtual) film director is to choose which camera angles need to be shot in the first place, a role which is not part of cinematography or film editing.

In this survey, we instead choose to decompose the role of the (virtual) film director into three mains tasks, called *decoupage*, *mise-en-scène* and *montage*. All three terms are borrowed from the French and commonly used in film studies [Bar20, Kes20, For20]. Broadly speaking, the role of a film director is to translate a story into a movie, and this can be decomposed into three different tasks. Decoupage is the choice of camera shots which need to be produced [Bar20]. Mise-en-scène consists in "staging events for the camera" [Kes20] to produces the chosen camera shots, which includes actor direction or character animation, cinematography and lighting. Montage is the ordering and length of shots used in the final movie [For20]. It is useful to make the distinction here between "camera shots" and "movie shots", since the camera shots planned during decoupage and created during mise-en-scène can be edited, trimmed and even reframed during montage before they become movie shots.

Montage and decoupage are the two faces of film editing, and their roles are complementary. In classical cinema, decoupage is performed in pre-production (before shooting) and montage is performed in post-production (after shooting). In computer graphics and animation, decoupage is often presented in the form of a storyboard, i.e. a graphic representation of the chosen camera shots. Each panel in the storyboard is used as a blueprint for creating the corresponding camera shot. The camera shots are then trimmed, reordered and assembled into movie shots during montage. In interactive games and animation, the situation is quite different because events may be staged for the camera in real time using automated mise-en scene. As a result, both decoupage and montage also need to be recomputed and coordinated in real time. The three tasks must ideally take place simultaneously, while the game is playing, which raises additional issues, not correctly dealt with by the traditional categories of cinematography and film editing.

The paper is organized as follows. Section 2 introduces the basic

concepts of decoupage and their relation to computer games and animation. Section 3 similarly introduces and illustrates the fundamentals of mise-en-scène theory. Section 4 reviews important concepts in montage theory and provides an in-depth discussion of the "rules of editing" in the context of computer games and animation. Section 5 proposes a taxonomy of film directing techniques, which are divided into procedural, declarative, optimization and learning methods. We review thirty important papers covering forty years of research and discuss how they solve (or not) the three inter-related problems of automatic decoupage, mise-en-scène and montage. Finally, Section 6 presents several current and future application domains for automated film directing and Section 7 discusses open issues and new directions for future work.

2. Decoupage

This section covers the task of selecting which camera angles will be useful to present the action taking place in the story world. We use the French term of decoupage to describe this important step in the film directing workflow, although other authors use different terms (camera planning, storyboarding, production planning, previzualization, etc). Useful theoretical references can be found in film directing books by Steven Katz [Kat91], Steven Sharff [Sha82] and Nicholas Proferes [Pro08]. Another useful resource is the book by Richard Pepperman [Pep05] which contains many movie scenes broken down into shots and provides a good account of the process of decoupage as performed by several well known film directors.

All methods covered in this survey need to solve the problem one way or another, since it decides which shots will need to be produced (mise-en-scène) and edited (montage). In some case, this is left to a human operator. In other cases, the choice is left open by computing a large number of shots and deciding later which ones are really needed, in a generate-and-test approach. Other methods make decisions on the decoupage and the montage of the movie as a single step, i.e. choose a linear sequence of shots before shooting it. This mimics the workflow of cartoon animation production, where the decoupage and the montage are decided together during the storyboarding stage. It should be noticed here that all those different approaches to decoupage are equally valid, depending on the requirements of the application. The requirements for directing a video game in real time, or creating a machinima movie in a game engine, or creating a cinematic replay of a gaming sessions are very different. And the requirements for directing an immersive reality experience in real time are different from all of the above cases. But in each case, a decoupage needs to be established, i.e. the action in the story world needs to be segmented into story units, and a finite number of shots needs to be chosen to cover each story unit.

Decoupage is probably the most overlooked concept in film directing, especially from a computer graphics perspective. We emphasize its importance because we feel it is a key element in shaping directing styles. Given the same story, different directors will likely make very different shot choices, and those decisions will affect the look and feel of their movies in recognizable and meaningful ways. Film directing techniques covered in this survey need to make similar decisions and those choices will similarly affect the look and feel of the generated movies.

While decoupage is an important step in many papers reviewed in this survey, we have found only one paper entirely dedicated to the decoupage problem. Wu et al. have proposed a language of film editing patterns that can be mined from real movies and applied to novel situations [WPRC18] to provide a suitable decoupage. This looks like a promising direction for future research.

3. Mise-en-scène

This section covers the task of generating the camera shots decided in the decoupage, which involves the staging of the action in front of the camera. Mise-en-scène is a vast topic in computer graphics, which includes the positioning (blocking) of the cameras and characters within the scene, the lighting of the scene, the animation of the characters, and the animation of the camera. In many applications, the placement and animation of the characters is given and the role of mise-en-scène is limited to the placement and animation of the camera, i.e. cinematography. In real-time games, non-player characters can also be placed and animated in real-time and become an integral part of the mise-en-scène. In both cases, mise-en-scène is an intermediate stage between decoupage and montage, and plays a central role in film directing.

One fundamental part of cinematography, as outlined in Maschielli's 5*C*'s of cinematography [Mas65] is to provide shots that can easily be edited together in montage. In the early days of cinema, the interplay between cinematography and editing was a matter of trial and error. As noted by Barry Salt [Sal03], it took several years before cinematographers and editors understood the "exit left enter right" editing rule. Before that, the rule was usually obeyed because it appeared to work better in most cases. But the "wrong" solution was still used from time to time. When it finally became clear what the "right" solution was, cinematographers stopped shooting the alternate solution because they knew it was useless. After more than a century of cinema, good professional cinematographers have thus "internalized" the rules of montage in such a way that they can avoid shots that will not cut together.

In games, we are probably still at an earlier stage because it is not yet quite clear how the rules of montage should translate for an interactive game, which is a very different situation from a movie.

In computer graphics, the camera is controlled by animators. A good professional animator should have a similar sense of which shots will cut together in montage. When this is not the case, the editor is left with fewer or no options. As a result, the scene may have to be shot again from another angle. This is usually not a problem because it is easy (and cheap) to do so. When implementing automated systems, it is important to take the rules of montage into account in the mise-en-scène. Otherwise, a lot of effort will be wasted on attempting to edit shots that "do not cut together". This will be examined in depth in Section 4.

In traditional mise-en-scène, decoupage and montage can be taken into account by following one of several working practises. We mention three of them.

Cutting in the head means that the director has already decided a very precise shot by shot decoupage of the intended movie, usually in the form of a storyboard. In that case, the mise-en-scène follows the storyboard as a blueprint for shooting each action

or *beat* in the screenplay from a single viewpoint. Textbooks in film-making warn against the dangers of the method because it cannot recover easily from errors in planning. This approach is very suitable for real-time applications. It consists in planning the montage first, resulting in a shot list that can then be rendered *exactly as planned* following the timeline of the final movie. One drawback of that approach is that the animation itself cannot always be predicted in all its actual details. As a result, it may be difficult to plan exactly *when to cut* from shot to shot.

Three-take technique A common variant of "cutting in the head" consists in shooting a little more of the action from each planned camera position. As a result, each action is shot from three camera positions - one according to the decoupage, one from the immediately previous viewpoint and one from the next viewpoint. This has the advantage that the exact cutting point can be resolved at a later stage during montage.

Master-shot technique Another common practice consists in planning all the camera works for shooting the scene in one continuous take - the "master shot" - and then adding shots of various sizes to show the details of the action in various sizes (close-ups and medium shots). Montage can then more carefully prepared by ensuring that all those shots will cut nicely with the master shot, resulting in a typical sequence of "Master-Closeup-Master-Closeup", etc.

Note that those techniques are very useful in practice because they are more general than "film idioms" where the camera positions are prescribed once and for all.

4. Montage

This section covers the task of editing and assembling available camera shots into a sequence of consecutive movie shots.

Here scenes are described in terms of actions and communicative goals that must be translated into successive shots. Cutting between cameras adds considerable freedom in the focalization and order of presentation of the visual material. Cutting between cameras also introduces constraints. We review the most important constraints and corresponding rules (180 degree rule, 60 degree rule) and explain how they can be expressed and solved algorithmically. Then, we review the principles that can be used to evaluate the quality of a shot sequences and the algorithmic strategies that can be used to solve for the best sequence.

4.1. Editing rules and constraints

It is important to understand the motivation between the so-called "rules of editing". Most of them are in fact constraints. What that means is that it may not be possible to cut from any two arbitrary cameras because some transitions may provoke *false inferences* [Bra92, Smi05, Gd07, Cut14, Tan18]. For a cut between two shots to work, it is fundamental that it does not break the logic of human perception and narrative understanding.

Psychologists d'Yewalle and Vanderbeeken offer a useful classification of editing errors [Gd07]. Editing errors of the "first order" are small displacements of the camera or image size, disturbing the perception of apparent movement and leading to the impression of jumping. Editing errors of the "second order" are violations of the spatial-cognitive representation of the 3-D scene. One example is the 180-rule violation, where the camera crosses the line between two actors and as a result the actors appear to swap positions. Another example is the motion continuity violation, when the camera crosses the line of an actor's movement and as a result the actor appears to change directions. Editing errors of the "third-order" are when successive shots have too little in common to be integrated into a single chronological sequence of events.

An important part of automated movie editing consists in preventing editing errors of all orders. But that is of course not the entire story because there are still infinitely many "correct" camera pairs that can be cut together at any given time. A second part of automated editing is therefore to evaluate *when* to cut to *which* shot. The classical Hollywood concept of editing [Mas65] recommends that successive shots should minimize perceptually disruptive transitions. The modern viewpoint [God56] stresses the consistency of the narrative structure which overrule disturbing transitions, as attention will primarily be directed to grasping the succession of significant events in the story. A good computational theory of film editing should probably stand in the middleground between those two viewpoints. On the one hand, it is difficult to get a good model of "perceptually disruptive transitions". At best, a computational model may be expected to avoid the most obvious mistakes, still leaving a large number of possibilities. On the other hand, the narrative structure of an animated scene may not always be easily uncovered, again leaving multiple choices.

Few editors have written about their art with more depth than Walter Murch [Mur86]. In his book, he introduces a Rule of Six with six layers of increasing complexity and importance in the choice of how and when to cut between shots:

Three-dimensional space of action. Respect of 3-D continuity in the real world: where people are in the room and their relations to each other (accounts for only 4 % of what makes a good cut)

Two-dimensional space of screen. Respect of 2D continuity. Where people appear on the screen. Where the lines of action, look, movement project on the screen. (5 %)

Eye-trace. Respect of the audience's focus of interest before and after the cut. (7 %)

Rhythm. Cut at a moment which is both right and interesting. (10 %)

Story. Cut in a way that advances the story. (23 %)

Emotion. Cut in a way that is true to the emotion of the moment. (accounts for 51 % of what makes a good cut).

In 3-D animation, the three-dimensional space of action is always in continuity as long as we perform live editing. So we only really need to be concerned with the other five criteria. We can attempt to build a computational theory of film editing based on this reduced rule of five if we know how to evaluate each of the five criteria AND find a consistent way to rank possible cuts and shots using a combination of them.

4.2. Two-dimensional continuity.

Two-dimensional continuity is easiest to evaluate by computer. All the programmer has to do is project the various lines (of action,

of looks, of movements, etc) to the camera plane and check that they remain consistent. This is a direct application of projective geometry.

Two-dimensional continuity can be insured by adhering to the following rules of the so-called *classical continuity style*:

Line of action The relative ordering of characters must remain the same in the two shots. This is the basis for the 180 degree rule, which forbids cuts between cameras situated across a line between the two characters - the line of action.

Screen continuity Characters who appear in both shots must not appear to jump around too much.

Motion continuity Moving characters who appear in both shots must appear to move in the same screen direction. This is the basis for another variant of the 180 degree rule, which forbids cuts between cameras situated across a line along the actor's trajectory - the line of action in that case. Motion continuity also requires that the screen position of the actor in the second shot should be "ahead", rather than "behind" its position in the first shot to prevent repetition ("hiccup" effect).

Jump cut Characters who appear in both shots must not appear to jump around too little. Small changes in screen coordinates are interpreted as actor movements, rather than camera changes, as an effect of human perception. They should be avoided, or used systematically to obtain a stylistic effect (Godard).

Look The gaze directions of characters seen in separation should match. If they are looking at each other, their images should also be looking at each other. If the two characters are NOT looking at each other, their images should NOT be looking at each other

Distance The sum of apparent distances to two characters shown in separation should be at least twice the actual distance between them (as if the two images were taken from the same camera position). This prevents the use of close-ups for two characters very far apart.

Size The shot size relative to a character should change smoothly, rather that abruptly. Cutting from a long shot directly to a close-up makes it harder for the viewer to understand the relation between the two shots. Instead, the editor should prefer to first cut to a medium-shot, then to a close-shot.

4.3. Eye-trace.

Eye-trace refers to the expected trajectories ,of the eyes of the audience. Where on the screen is the audience looking in the first shot ? What happens there during the cut ? Where will the audience look in the second shot ?

A popular heuristic is to use the actors' eyes in the image. This is a well established principle confirmed by many film editors. But predicting where the audience is looking remains hard even for editors. Film director James Cameron (who also edits his own movies) phrased it as follows: "You can only converge to one image plane at a time – make sure it is the place the audience (or the majority of the audience) is looking. If it's Tom Cruise smiling, you know with 99 % certainty where they're looking. If it's a wide shot with a lot of characters on different depth-planes doing interesting things, your prediction rate goes down." [Zon05]. Current research in vision science attempts to predict the focus of attention in an image, based on the computation of local image features. The most established theory is the "saliency-based" model of Itti and Koch at Caltech [IKN98]. Their model was used by Santella et al. for the purpose of evaluating the composition while cropping and reframing images [SAD*06]. Their conclusion was that better predictions were obtained by considering the eyes and gaze of people in the image. More recent work in video saliency uses deep learning to better mimic human perception [GC18] but predicting the spectator's gaze while viewing cinematographic contents remains a challenging task [TKWB20], further complicated by high level narrative engagement [LLMS15].

4.4. Rhythm.

Rhythm refers to the tempo of the scene (how fast the film is cut). But we should be aware that the perceived duration of a shot depends on its content. Thus a shot that we have already seen many times will seem to last longer than it really is. A close-up will also seem to last longer than it really is. We should cut from any given shot only after the audience has been able to fully see what we intend them to see. We should also cut before the shot becomes redundant or boring.

One further complication is that the perceived length of a shot depends on its size, its novelty and the intensity of the action. Thus, a close-up will be perceived as taking longer than a long shot. A recurring shot will be perceived as taking longer than a new shot. And a shot of a static scene will be perceived as taking (much) longer than a shot of a fast action. A reasonable approximation may be to set the average shot length as a function of shot size, so that close-ups are cut faster and long shots are cut slower. This is a reasonable first approximation.

Another important factor is to choose a *natural* distribution of shot durations. Automated editing should not "get in the way". As a very simple illustrative example, cutting at regular intervals (as with a metronome) can be very annoying because it distracts the viewer from the experience of the movie. Cutting shots with randomized durations is usually a better idea. Even better editing can be computed by following the distribution of shot durations in real movies.

Film scholars Barry Salt [Sal03] and James Cutting [CC15] (among others) have extensively studied shot durations in cinema and found it to be an important parameter of film style. An empirical finding by Barry Salt is that the distribution of shot durations in a movie sequence is correctly represented by a log-normal distribution. This is also the distribution of sentence lengths in a book chapter. This is non-symmetric distribution with a smaller probability for very short durations and a relatively larger probability for longer shot durations. Galvane et al. set the editing rhythm by choosing an *average shot length* or ASL for the sequence, and cut according to a log-normal distribution [GRLC15].

4.5. Story advancement.

Story advancement can be measured by checking that all changes in the story line are correctly presented in the image. Thus, actors should only change places on-screen (not off-screen). We should

see (or hear) their reactions. We should see entrances and exits of all characters. We should see them when they sit down or stand up, when they dress or undress, when then they put on or take off their hats, etc. Of course, real directors and editors break this rule all the times, with interesting effects. But it seems to be a safe bet to adopt the rule that the best editing is the one that *presents the entire action in the scene from the best angle at all times*.

An even stronger principle was proposed by Hitchcock in an interview with Truffaut [Tru85]: "screen size and visibility of actors and objects should be proportional to their importance in the plot at any given time" (Hitchcock principle). This is useful principle to keep in mind because it allows the programmer to define mathematically what makes a good editing. Computing the screen size and visibility of actors and objects in a shot is the easy part. Computing their importance in the plot is the really difficult part.

In a scripted sequence, it seems reasonable to assume that the scripted actions are all equally important. Thus at any given time, the importance of actors and objects can be approximated as the number of actions in which they are taking part, divided by the total number of actions being executed in the scene at that time. Other approximations are of course possible. For instance, it may be preferable to assign all the attention to a single action at all times. This may be implemented with a "winner takes all" strategy.

4.6. Emotion.

For the purpose of editing, evaluating the emotional impact of any given shot or cut appears to be very difficult. Emotional cues can be received from the screenplay or from the director's notes. They assert which emotions should be conveyed at any given point in time. Given such emotional cues, we can then apply simple recipes such as separating actors or showing them closer together; changing editing rhythm to show increasing or decreasing tension; changing shot sizes to show increasing or decreasing tension; using lower camera angles to show ceilings and feel oppression; using higher camera angles to hide ceilings and feel freedom; using longer lenses to slow down actor movements and isolate them from the background; using wider lenses to accelerate actor movements and put them in perspective, etc. Similar to other criteria, such emotional impacts need to be planned during decoupage, implemented during mise-en-scène, and evaluated during montage. This is one of the outstanding challenges in automated film-making.

5. A taxonomy of film directing techniques

After having explained the theory of decoupage, mise-en-scène and montage, we now turn to actual implementations of working systems. We review procedural, declarative, optimization and learning approaches separately. Automatic film directing has a long history, dating back at least to John Carroll's book in Gilles Bloch's PhD thesis in 1986 [Blo86]. In the following section, we present a taxonomy of approaches for automatic film directing including decoupage, mise-en-scène and montage, which includes procedural, declarative, optimization and learning approaches. A procedural approach to movie editing builds an explicit solution. A good example of that is the Virtual Cinematographer system (VC) where each idiom is implemented as finite state machine. A reactive approach is essentially a procedural approach where multiple courses of events can be taken into account. A declarative approach states the constraints and rules and lets a separate solver find a solution that meets all the constraints, and/or maximizes a measure of quality. An optimization approach builds an explicit measure of the quality of a montage, which then needs to be maximized to find an optimal montage. A (supervised) learning approach builds a procedural solution from a large dataset of examples by maximizing the agreement between the predicted montages and the examples.

From the vast litterature on automated film directing in the last 40 years, we selected 30 papers, based on their novelty at the time of publication and their impact and influence. We tried to maintain a balance between the main categories of approaches and the three tasks of decoupage, montage and mise-en-scène-en-scène. The chronology of the papers is illustrated in Fig. 1.

5.1. Declarative approaches

In the beginning, automatic editing was attempted with traditional, rule-based systems. Indeed, most early in automated film directing originated from the artificial intelligence community, rather than the graphics community. We review important papers focusing on automatic montage from annotated live action rushes dating from the 1980s, because of their long lasting influence on more recent work, then continue our survey of declarative approaches in computer games and animation, starting from the 1990s and to the present.

Declarative approaches present an excellent overview of many important aspects of automated film editing, but the results are not always convincing for lack of a sufficient integration with advanced camera control techniques. Another drawback of declarative approaches is that they require an in-depth semantic analysis of the storyline, which is not always readily available in practical applications, especially in real-time games. More importantly, those methods usually return a (usually large) list of possible solutions, even in simple cases. As a result, they usually do not scale very well with larger vocabularies of plot actions, films idioms and shot categories.

5.1.1. Generative Cinema Grammar [Car80]

In his book and several related papers [Car77, Car81, Car82], John Carroll proposes an extension of Chomskyan theories of transformational generative grammars from natural language to cinema. The high level structure of his transformational generative cinema grammar (TGCG) is to decompose a movie into an event structure by way of "semantic rules"; then further decompose this event structure into a shot structure by way of scene transformations (montage) and shot transformations (decoupage), and finally decompose the shot structure into image frames by way of "photographic rules" (mise-en-scène). Examples of semantic rules are the decomposition of an event into actions and reactions of different actors, and the decomposition of an action into a preparation and a focal subaction. Examples of transformations are the rewriting of an action into a sequence of shots, the rewriting of an action sequence into a single shot, and the deletion of an action or a shot. Deletion plays an important role in Carroll's theory as it provides

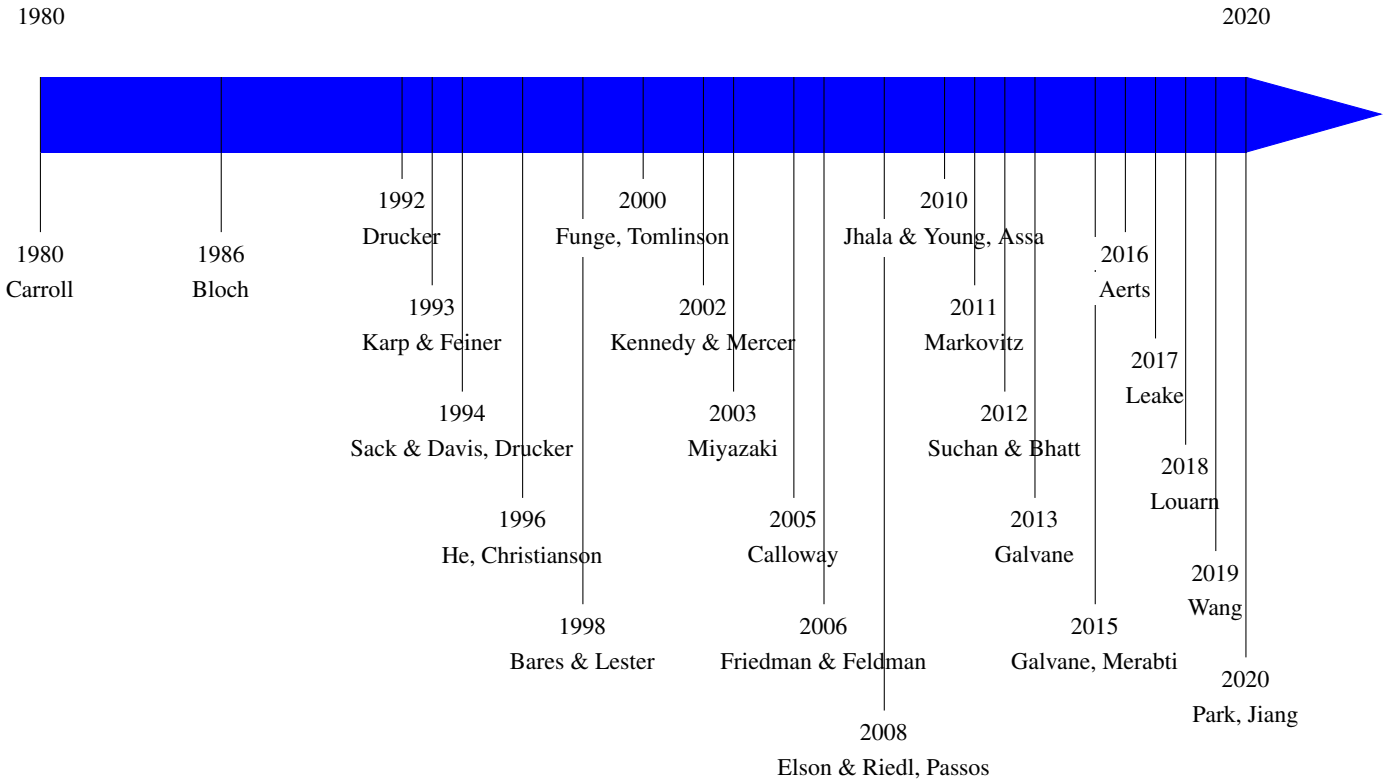

Figure 1: *Chronology of 30 landmark papers in automatic film directing from 1980 to 2020. Most papers directly address film directing in computer graphics and digital games. Some papers targeting live-action movie-making are also included when they introduce important ideas and methods.*

	Procedural	Declarative	Optimization	Learning
Decoupage	Camera creatures [TBN00], CAMBOT [ER07, RRE08], Virtual director [AWCO10], Steering [GCR*13]	ESPLANADE [KF93], DCCL [CAH*96], ConstraintCam [BGL98, BL99], CML [Fun00], Planning cinematography [KM02], DMP [SMAY03a, SMAY03b], GLAMOUR [CNN*05], MARIO [FF04, FF06], Darshak [JY05, JY06, JY10], Dynamic camera control [SB12]	Write-a-video [WYH*19]	Virtual director [MCB15]
Mise-en-scène	CINEMA [DGZ92], Camera creatures [TBN00], CAMBOT [ER07, RRE08], Steering [GCR*13]	Planning cinematography [KM02], MARIO [FF04, FF06], Staging [LCL18]	CAMDROID [DZ95]	Camera behaviors [JWW*20]
Montage	Virtual cinematographer [HCS96a], CINEMA [DGZ92], Camera creatures [TBN00], Behavior trees [MKSB11], Intelligent FSM [Par20]	Cinema grammar [Car80], Montage machine [Blo86], IDIC [SD94], Planning cinematography [KM02], DMP [SMAY03a, SMAY03b], GLAMOUR [CNN*05], MARIO [FF04, FF06], Darshak [JY05, JY06, JY10], Dynamic camera control [SB12], CAMETRON [AGV16],	CAMBOT [ER07, RRE08], Continuity [GRLC15], Dialogue [LDTA17], Write-a-video [WYH*19]	Neuronal editor [PMC*10]

Table 1: *Taxonomy of film directing techniques covered in our survey. Papers are classified along two axes, based on the directing tasks that they address (decoupage, mise-en-scène or montage) and the methodology that they propose (procedural, declarative, optimization or learning). The table lists the papers included in the survey for each class in the taxonomy.*

Title	Type	Story	Decoupage	Mise-en-scène	Montage	Time	Domain
Cinema grammar [Car80]	DEC	events and actions	shot grammar	photographic grammar	scene grammar	offline	live action
Montage machine [Blo86]	DEC	conceptual dependencies	no	no	pattern matching	offline	live action
CINEMA [DGZ92]	PROC	no	no	through the lens	match cuts	real time	film studies
ESPLANADE [KF93]	DEC	script + goals	rule-based	no	rule-based	offline	industrial scenes
IDIC [SD94]	DEC	annotated scripts and movie shots	no	no	forward planning	offline	movie trailers
CAMDROID [DZ95]	OPT	no	scripted	through-the-lens	scripted	near real time	virtual football game
Virtual Cinematographer [HCS96a]	PROC	subject + verb + object	idioms	fixed cameras	hierarchical finite state machines	real time	third person games
DCCL [CAH*96]	DEC	subject + verb + object	generate and test	though the lens	film tree	offline	third person games
ConstraintCam [BGL98, BL99]	DEC	character goals, plot points	cinematic goal selector	constraints	constraints	real time	interactive fiction
CML [Fun00]	PROC	situation calculus	golog programs	primitive actions	golog programs	real time	undersea world
Camera Creatures [TBN00]	DEC	emotions + actions	generate and test	dramatic angles and lighting	action selection	real time	artificial life
Planning cinematography [KM02]	DEC	actions, themes, moods	shot maker	renderer	RST planner	offline	cartoon animation
DMP [SMAY03a, SMAY03b]	DEC	semantic scene graph	rule-based	scripted (TVML)	rule-based	offline	shared virtual environments
GLAMOUR [CNN*05]	DEC	rhetorical tree, discourse plan	Parallel NLG and video planner	pan and scan	rhythm of discourse	offline	video documentaries
MARIO [FF04, FF06]	DEC	screenplay + floorplan	rule-based	rule-based	constraint resolution	offline	telenovela
CAMBOT [ER07, RRE08]	OPT	dramatic beats	shot per beat	game engine	dynamic programming	offline	machinima
Neuronal Editor [PMC*10]	LEARN	scene geometry	no	no	feed-forward neural network	real time	race game
Darshak [JY05, JY06, JY10]	DEC	mood, intensity, tempo, emotions, actions	rule-based	game engine	partial order causal link planning	offline	machinima
Virtual Director [AWCO10]	PROC	character motion	canonical correlation analysis	film idioms	accumulated view erosion	offline	highlight cinematics
Behavior trees [MKSB11]	PROC	smart events	reactive	no	scripted	real time	computer game
Dynamic camera control [SB12]	DEC	situation calculus	rule-based	pan-tilt-zoom camera	rule-based	real time	meetings
Steering [GCR*13]	PROC	crowd simulation	scouting behavior	tracjing behavior	no	real time	machinima
Virtual Director [MCB15]	LEARN	events	HMM	no	HMM	offline	machinima
Continuity [GRLC15]	OPT	parallel actions	Hitchcock rule	no	semi-markov model	offline	machinima
CAMETRON [AGV16]	DEC	actors +actions	no	no	sampling	real time	live action
Dialogue [LDTA17]	OPT	script + speech transcription	no	no	HMM	offline	live action
Staging [LCL18]	DEC	prose storyboard	no	actors + cameras	no	offline	text-to-movie
Write-a-video [WYH*19]	OPT	voice over narration	visual semantic matching	no	dynamic programming	offline	video documentaries
Intelligent FSM [Par20]	PROC	actions and roles	Hitchcock rule	FSM	FSM	real time	VR
Example [JWW*20]	LEARN	actor positions	no	deep network	no	delay	VR

Table 2: *Chronology of important papers in film directing from 1980 to 2020, classified into procedural, declarative, optimization and learning methods. We indicate how each method represents the story to be directed and how is deals with the separate tasks of decoupage, mise-en-scène and montage. We also distinguish real-time methods from offline methods and the targeted application domains.*

an explanation of spatial and temporal discontinuities in movies. While TGCG is not a complete theory, it anticipates and encompasses many later attempts in automatic movie generation.

5.1.2. Montage machine [Blo86, Blo87]

The montage machine is the first implementation of a completely automatic method for film editing. It was developed during Gilles Bloch's PHD thesis [Blo86]. The montage machine takes annotated video rushes as an input and pieces them together into a movie. Bloch provides a careful analysis of continuity and discontinuity of movement, gaze and screen positions borrowed from film theory [Bur81] and implements them as constraints on a generative discourse grammar. He implements pattern matching methods for generating short film sequences narrating physical actions of two characters (climbing stairs, opening and closing doors, exchanging glances, picking up objects). This is one major step forward between Carroll's theoretical investigations and the computer generated movies of the following decade.

5.1.3. ESPLANADE [KF93]

The Expert System for PLANning Animation, Design, and Editing by Karp and Feiner is one of the first complete systems for generating 3D animated movies from a symbolic input of actions and communicative goals. The system consists of an action planner and an animation planer, both using a large domain database encoding dramatic and cinematographic knowledge. The animation planner is responsible for the decoupage and montage of shots into scenes. ESPLANADE chooses between seven basic scene structures borrowed from Steven Sharff's Elements of cinema [Sha82]. Each scene structure is based on a single cinematic device - separation, parallel action, slow disclosure, familiar image, moving camera, multi-angularity, and master shot. This brings coherence in each scene and diversity over an entire movie. The system is demonstrated in walk through scenarios in virtual industrial scenes and targets narrative games.

5.1.4. IDIC [SD94]

IDIC by Sack and Davis [SD94] follows Bloch's path with another attempt in automatic film editing from annotated movie shots. Mostly a sketch of what is possible, it was based on the general problem solver (GPS), a fairly primitive forward planner [RN02]. IDIC was demonstrated in the task of generating Star Trek movie trailers from annotated shots. Despite its (assumed) limitations, IDIC makes an important contribution to algorithmic montage theory by reformulating it explicitly as a planning problem. A cut between two shots is viewed as a planning operator with a list of pre-conditions, and add-list and a delete-list. The pre-conditions represent what was shown in the first shot, the add-list represents what is shown in the second shot, and the delete-list represents what is supplied by the inferential activity of the viewer during the cut between the two shots. On the positive side, IDIC allows a much larger variety of spatial and temporal discontinuities between shots. On the negative side, montage becomes a NP hard problem and can only be solved for short sequences with a small number of candidate shots. n future work, It would be useful to resume work along the same lines using more efficient planning approaches.

5.1.5. DCCL [CAH*96]

A companion paper to the virtual cinematographer paper by the same authors, "Declarative Camera Control for Automatic Cinematography" is a much more elaborate attempt at formalizing the editing of an animated movie, this time using modern planning techniques [CAH*96]. In that paper, idioms are not described in terms of cameras in world coordinates but in terms of shots in screen coordinates, through the use of the DCCL language. DCCL is compiled into a film tree, which contains all the possible editings of the input actions. Actions are represented as subject-verb-object triples. As in the Virtual Cinematographer companion paper, the programming effort for implementing an idiom is important.

5.1.6. ConstraintCam [BGL98, BL99]

Bares and Lester designed and built the ConstraintCam system for generating multi-shot presentations of interactive fiction. They use a rich story representation from their own narrative planner, which includes an explicit representation of character goals and plot points in the story. They implement a cinematic goal selector for solving the decoupage problem, based on a repository of common cinematic goals. They approach the mise-en-scène and montage problems using constraint satisfaction.

5.1.7. Cognitive modeling [Fun00]

Funge and Terzopoulos propose a formal treatment of film idioms as programs written in their Cognitive Modeling Language (CML), a variant of the GOLOG programming language which allows for the specification and execution of complex actions in dynamical domains [LRL*97]. GOLOG and CML are both rooted in the situation calculus, a logical framework allowing to reason about properties (named fluents) whose truth values change over time [Rei01]. In this context, a film idiom consists of some (hard coded) primitive actions corresponding to common shots, and (algorithmic) complex actions for choosing and sequencing them at runtime. As a result, the mise-en-scène remains procedural but the decoupage and montage become declarative.

In contrast to the state machines used by He et al., CML programs can take advantage of the entire history of situations encountered during a virtual world simulation, and take more informed decisions at least in principle. Unfortunately, their paper does not offer a very convincing case that the increased expressivity of the language results in better movies. It is left for future research to determine whether a virtual film director written in GOLOG or CML could lead to superior performances in more complex real-time digital games.

5.1.8. Planning cinematography [KM02]

Kennedy and Mercer use the LOOM knowledge representation language to encode different communicative acts in the rhetorical structure theory. By mapping the story-line into communicative goals, stated in terms of themes and moods, they are able to plan the choice of camera and editing. Their system separately solves the decoupage, montage and mise-en-scène problems sequentially. First a "shot maker" chooses the appropriate shots given the input actions and communicative goals. Then a "rhetorical structure planner" chosen the temporal ordering of those shots using rhetorical

structure theory [MT88]. Finally, a "renderer" generates the animation for all shots in the resulting montage. They demonstrate their system with expressive cartoon animations created with different themes and moods (such as happy or scary).

5.1.9. DMP [SMAY03a, SMAY03b]

Miyazaki et al. describe a complete film-making production system [SMAY03a, SMAY03b]. Their input is a semantic scene graph encoded in the CLIPS/COOL knowledge representation framework. Cinematic rules for choosing cameras (decoupage) and editing them (montage) are also encoded in CLIPS/COOL for several common film idioms. Decoupage and montage are solved simultaneously using RETE planning algorithm implemented in JESS. This produces an abstract shot list which is is sent to the NHK TVML animation system [HDH04]. Contrary to other systems, DMP does not separate decoupage and montage, and therefore relies on the metaphor of "cutting in the head" without re-evaluation of the quality of the mise-en-scène. It is demonstrated in the context of a shared virtual environment.

5.1.10. GLAMOUR [CNN*05]

GLAMOUR generates video documentaries with synchronized voice-over narratives. Given a set of communicative goals, encoded as rhetorical structures, GLAMOUR separately generates a voice over narration using natural language generation techniques, and a set of pan-and-scan animations (a.k.a. Ken Burns effects) computed from a database of annotated still pictures to illustrate the narration. This solves the decoupage problem. Then they perform a montage step where they synchronize the voice and the pan and scan animations to achieve an appropriate rhythm of discourse. While their approach is quite specific, it provides useful hints for controlling the rhythm of a montage sequence and its synchronization with a voice over, which are not addressed by other systems.

5.1.11. MARIO [FF04, FF06]

Friedman and Feldman present another knowledge-rich approach for editing sitcoms and telenovelas in 3D animation [FF06]. Their system assumes an existing animation scene. It takes as input a timeline of (possibly overlapping) actions and a floor plan, and produces a list of camera parameters for each frame of the input animation. Rather than relying on a small set of film idioms, they implement several important rules of continuity editing (line of action, 60 degree rule, prevention of jump cuts) geometrically and choose shots and cuts respecting those rules with a system of defaults, preferences and assumptions. Their system was evaluated by expert film makers with the conclusion that it achieves the same quality as a novice film maker.

5.1.12. Darshak [JY05, JY06, JY10]

Jhala and Young have used text generation techniques to automatically edit shots together using "plan operators" [JY05]. In another paper, Jhala and Young have used examples from the movie "The Rope" by Alfred Hitchcock to emphasize stronger requirements on how the story line AND the director's goal should be represented to an automatic editing system [JY06]. They use 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 do not attempt to combine those strategies and instead prefer to demonstrate the capability of their solver to present the same sequence in different editing styles.

5.1.13. Dynamic camera control [SB12]

Suchan and Bhatt describe an original architecture for generating a movie of a meeting with multiple speakers and audience participation. Cameras and microphones are equipped with HMM-based probabilistic activity recognition. They build an approximate, topological model of the scene. Based on this qualitative information, camera actions (cut, pan, tilt, zoom) are generated by stochastic GOLOG programs guided by choices and preferences. This is one rare example of a declarative, rule-based system which achieves real-time performance, although in the limited scope of video recording of meetings.

5.1.14. CAMETRON [AGV16]

CAMETRON is a live action video production system which focuses on the task of automating the montage of all available cameras during a live broadcast. It is based on a causal probabilistic model of events and shots (film idioms) encoded in CP-logic/Problog [VDB09]. CAMETRON is one of the few systems in this survey which correctly handle shot duration and rhythm of montage. Another original feature of their approach is that they sample the probability distribution of all possible montages, which allows them to make decisions in near real time. As a result, their method is highly non deterministic. They tested their system on lectures with two speakers, three cameras and three actions (walking, speaking and pointing), where they achieved near real time performance of 4 FPS. Unfortunately, they do not provide a comparison of their results with the maximum probability solution, which can be computed offline. One promising avenue for future research in this direction is to learn such probabilistic programs from examples.

5.1.15. Automated Staging [LCL18]

Louarn et al. describe a method for staging actors and cameras simultaneously, given a symbolic description of the desired shot composition as a "prose storyboard" [RGBM20]. While their system is only concerned with the problem of mise-en-scène, and does not cover decoupage or montage, it focuses on the important aspect of staging actors, which is not covered by other methods. Previous work has focused on the simpler problems of staging cameras relative to the given actors. Here, they provide a complete solution to the mise-en-scène problem of staging actors and cameras, at least in the case of static cameras and actors. The more difficult case of dynamic cameras and actors remains an open issue and a central challenge for future research in film directing.

5.2. Procedural approaches

Declarative approaches suffer from high algorithmic complexity, which makes them ill-suited to real-time graphics. As an alternative, researchers in computer games and real-time animation have proposed procedural methods, which are amenable to real-time implementations. We review the most important academic papers in

this category, and refer the reader to existing books [Haw05,HH09] and conferences [GDC, GAM] on game cinematography for more specialized and game-specific variants.

5.2.1. CINEMA [DGZ92]

CINEMA combines earlier work in keyframed camera movements [Koc84], navigation in virtual worlds [Bro86], 3D interaction [WO90] and synthetic visual narrative [DSP91] into the first real-time procedural film directing system capable of generating coordinated shot sequences that illustrate very short "stories". This procedural approach was later abandoned by the authors in favor of a more declarative approach. Yet, it remains an important landmark in research of film directing because if demonstrates for the first time the possibility of performing decoupage, mise-en-scene and montage simultaneously in a real time application.

5.2.2. The Virtual Cinematographer [HCS96a]

The Virtual Cinematographer by He et al. [HCS96a] relies on the use of film idioms, which are recipes for obtaining good framing and editing in a given situation. The general approach is similar to the old-fashioned AI principle of case-based reasoning - if a conversation starts in a game, use the conversation idiom; if a fight start, use the fight idiom; etc.

Each idiom has two components - a set-up (blocking) of the cameras relative to the actors; and a state machine for switching automatically between cameras in that setup. This is a powerful paradigm, that easily allows for gradually building up a complex cinematography system from simple building blocks.

Each idiom is very easy to program - the set-up of the cameras is defined in terms of world coordinates - relative to the actors. The VC takes as input strings of simple sentences : SUBJECT+VERB+OBJECT representing the action taking place in the scene. The VC also takes as input a continuous stream of bounding boxes and orientation, representing the relative geometric positions and orientations of the virtual actors, objects and scene elements.

Idioms are usually chosen based on the next action string. More complex editing patterns can also be achieved by defining hierarchical state machines, encoding the transitions between idioms. While powerful, this scheme has yet to demonstrate that it can be used in practical situations. One reason may be that there is a heavy burden on the application programmer, who must encode all idioms for all narrative situations. Another reason may be that the resulting editing may be too predictable. In a finite state machine, the switching of a camera is triggered by the next action string. This may have the undesirable effect that the switching becomes too predictable. A good example is the "dragnet" style of editing [Mur86] where the camera consistently switches to a close-up of the speaker on each speaker change; then back to a reaction shot of the other actors being spoken to. This can become especially annoying when the speakers alternate very quickly. While it is possible to use the dragnet style of editing as a separate film idiom, this causes the number of idiom to explode since every configuration can be filmed in dragnet style. A better solution separates the camera set-ups from the state machines - for each set-up, different styles can then be encoded with different state machines. But the same "style" must still be separately re-encoded for each set-up.It is not obvious how to "generalize" film idioms. This is an open problem for procedural approaches in general.

5.2.3. Camera Creatures [TBN00]

Tomlinson et al. describe a system where cameras and lights are autonomous creatures governed by goals and motivations, which interact with other virtual actors in the scene. A key component of their system is a generic algorithm for action selection, used by actors, lights and cameras alike [KB99]. Camera actions consist in choosing shot values and lighting patterns to maintain relations between actors, sets and participants and to express one of six emotions (happy, sad, angry, fearful, surprised and disgusted). Decoupage is performed by evaluating available shots in the light of the current camera goals and motivations. Actors can also request shots directly from the camera. Montage is then performed in real-time though action selection. The system was demonstrated live at Siggraph 1999 and 2000 and evaluated from subjective audience reactions.

5.2.4. Camera behavior trees [MKSB11]

As an alternative to finite state machines, Markowitz et al. have proposed to model cinematography and editing using behavior trees [MKSB11]. Behavior trees have become popular tools in game engines for programming the behavior of non player characters (NPC). They also form the computational basis for important work in interactive storytelling by the same authors [SMKB13]. In their system, behaviors are stored in and triggered by smart events [SSH*10] in the virtual world. They are encoded and executed procedurally by the camera as behavior trees, resulting in real-time decisions which are at once goal-driven and hierarchically organized. Compared to finite state machine, behavior trees result in more sophisticated, less predictable implementations of common film idioms that automatically adapt to the virtual environment at runtime.

5.2.5. Steering behaviors for autonomous cameras [GCR*13]

Galvane et al. extend Reynold's steering approach [Rey99] to cameras by adding an explicit control of the viewing direction, which is governed by torques, independently of the moving direction. They define scouting and tracking camera behaviors, which they use to provide real-time coverage of events during crowd simulations. The scouting behavior searches for interesting events (decoupage) and the tracking behavior computes suitable camera movements for those events (mise-en-scène). Montage can be performed by collecting images captured by a flock or herd of autonomous cameras.

5.2.6. Intelligent FSM [Par20]

Park describes a modern implementation of the virtual cinematographer [HCS96b] with a more elaborate event model including thematic roles (location, tool, target, destination). The proposed system was tested and evaluated subjectively on 10 film idioms with limited success (80 % positive evaluations for one actor, 60 % for two actors, 35 % for three actors). FSM implementations of film idioms are attractive because they cover the three problems of decoupage, mise-en-scène and montage in a unified framework which

is easy to implement. But those new results suggest that they typically produce predictable results with little aesthetic value. This raises an important issue in film directing, that few of the proposed methods have been evaluated seriously, and that there is no common dataset on which those methods could be evaluated and compared to each other.

5.3. Optimization approaches

To overcome the problems of procedural and declarative approaches, it seems natural to rephrase the three tasks of film directing as optimization problems. The general idea is to compute a cost function that measures the aesthetic quality of any given decoupage, mise-en-scène or montage, and to propose methods for finding the minimum cost solution from a large enough list of candidates.

Decoupage is the hardest problem to solve using optimization methods because of the exponential number of possible camera shots that can produced for any given story. Mise-en-scène is a more tractable problem where the given decoupage give constraints on the screen positions and motions of actors, and the remaining degrees of freedom (including camera positions, orientations and focal lengths) can be determined by optimizing aesthetic measures based on "rules of composition" which are common practice in cinematography [Mas65, Ari76, Kat91, Tho98]. By choosing suitable (e.g. convex) cost functions, it is possible in some cases to guarantee a global optimum. Montage is a hard combinatorial problem in general, but can be simplified my making suitable assumptions. If the candidate camera shots are all aligned on the same timeline for instance, as in the case of multiple cameras shooting the same events, and the resulting montage is also constrained to follow this timeline, then dynamic programming methods can be used to find a minimum cost solution if the cost function is chosen carefully, for example assuming a (frame by frame) markovian or (shot by shot) semi-markovian cost function.

5.3.1. CAMDROID [DZ95]

Following up on the CINEMA system, CAMDROID mixes a procedural approach to decoupage and montage with a constrained optimization approach to mise-en-scène to achieve near real time performance in a virtual football game. Shots are generated by a network of camera modules, encoding cinematographic knowledge borrowed from Katz's book "Film directing shot by shot" [Kat91].

5.3.2. CAMBOT [ER07, RRE08, ORN09]

CAMBOT is a machinima generation system which creates movies in the Unreal game engine from a symbolically encoded script and a set, i.e. a story world in the game engine annotated with semantic labels. CAMBOT provides original solutions to the decoupage and montage steps which are inspired by real filmmaking practice. First of all, they assume that the script has been broken down into a sequence of dramatic "beats" a concept drawn from screenwriting practice [McK97]. Beats are the smallest divisible segments of a CAMBOT scene, typically one line of dialogue or a moment of action. For each beat, CAMBOT searches the set for all possible locations (stages), blockings of the actors and cameras and shots compatible with the input script. This provides a decoupage of the scene with a small number of shots per beat. CAMBOT then queries the game engine to render those shots in the appropriate locations (mise-en-scène). Finally, CAMBOT computes all possible montages of those shots using dynamic programming, under the assumption that the quality of a montage is the sum of the qualities of all shots and transitions between shots. As a result, CAMBOT can choose the overall best montage sequence, something that previous declarative approaches were not able to guarantee. They report a library of approximately 50 shots, two stages and half a dozen blockings. Stages and blockings play a similar role to film idioms in previous work. A key contribution of CAMBOT is that a better montage can be discovered efficiently in the case of an (offline) machinima production. A drawback of their method is that it cannot run in real time since it requires a complete evaluation of each scene.

5.3.3. Virtual Director [AWCO10]

Assa et al. proposed a generate-and-test approach for camera selection (decoupage) and editing (montage) for the case of creating movies from 3D animation where the focus is on the body motion of the virtual actors. This comes as a welcome complement to this survey which is otherwise dominated by talking faces and conversations. They propose to evaluate the quality of a camera by measuring the correlation between human motion in the scene and in the camera (the higher the correlation the better). They use Canonical Correlation Analysis (CCA) as a measure of correlation between the scene and the view. They use as a criterium for choosing views (decoupage). In the montage step of their approach, they introduce the notion of "accumulated view erosion" so that the current camera choice gradually loses interest and is eventually abandoned in favor of a new camera view. They use this mechanism for switching vies back and forth during interaction between two virtual characters. Their approach is validated by an extensive user study demonstrating the benefits of their camera selection mechanism in the particular application of generating highlight cinematics in sports games.

5.3.4. Continuity Editing [GRCS14, GRLC15, GR17]

Galvane et al. focus on the problem on montage in the case of machinima generation and extend the optimization approach of Elson and Riedl in several ways. First of all, they allow camera transitions at arbitrary times, rather than at the boundaries between dramatic beats. This make it possible to control the rhythm of the montage independently of the rhythm of actions and dialogues. They are also not limited by the number of actors in a scene. In fact, their system is applicable to an unlimited number of actors engaged in parallel actions. They allow four main categories of actions (speaking, reacting, moving and manipulating) and compute suitable shot compositions for each category based on the visibility of their body parts. As a result, their method does not rely on a catalog of existing film idioms but instead chooses shot compositions at runtime, based on the generic Hitchcock principle that the visibility of actors should be proportional to their importance in the story. Finally, they compute the quality of shot transitions using an exhaustive list of continuity editing rules (continuity of screen positions, gaze directions, movements, relative positions). They demonstrate their system on a synthetic re-creation of a famous scene from the

movie "Back to the future", which they make publicly available. A subjective ablation study shows that the three components of their optimization function (shot quality, transition quality, rhythm quality) are equally important. In related work, they apply their method for the special case of generating cinematic replays of narrative games [GRCS14] using the rich set of narrative actions generated by the IDTENSION narrative engine [SBK07].

5.3.5. Dialogue scenes [LDTA17]

Leake et al. also train HMM models of video editing for solving the montage problem in the case of dialogue scenes between two characters. Contrary to Merabti et al. they train multiple HMM models corresponding to different film idioms and use a mixture of all idioms at runtime. Their approach can be seen as a probabilistic reformulation of the virtual cinematographer paper [HCS96b] with non deterministic idioms. It was applied to real footage of dialogue scenes and evaluated positively in this context. It would be interesting to see if it can be extended to the case of computer games and animation and evaluated in this context as well.

5.3.6. Write-a-video [WYH*19]

Wang et al. propose a method for automatically generating a video montage illustrating a given voice-over narration. In the decoupage step, they search a video database for suitable shots for the input narration, based on visual semantic matching. Then in a montage step, they choose an optimal sequence of the chosen shots using dynamic programming, based on a quantitative model of the shot and transition qualities.

5.4. Learning approaches

The perspective of automatic film directing using machine learning (AI) has been raised when IBM Watson was used to generate a trailer for the movie Morgan. In reality, IBM Watson was programmed to find areas of high action, or high emotion from the movie (decoupage) and make them available to an experienced (human) trailer editor who created the trailer's montage [SJH*17]. Learning film directing from real movies is in fact a difficult task because in most cases, only the selected shots are available for study. The rejected shots would provide valuable information as well but they are usually not given. Previous work has alleviated this difficulty by making several simplifying assumptions, which are reviewed in this subsection. We expect this new direction of research will continue to expand and contribute to the state of the art as larger shared datasets and evaluation methods become available.

5.4.1. Neuronal Editor [PMC*10]

Passos et al. describe a neuronal film editor applicable to real-time games [PMC*10]. Their system uses a cinematographer agent to generate several possible shots in real time, and an editor agent is trained to choose shots at runtime from training sessions with an "editor's choice" feedback using a fairly simple feedforward neural network. The system was tested on a simple race game with three cameras (chasing, front, high-view). We include it in our survey because it is the first reported learning method for film directing. Compared to other approaches in this survey, the results appear quite limited and many outstanding issues still need to be resolved for learning film directing from examples in more realistic scenarios.

5.4.2. Virtual Director [MCB15]

Merabti et al. propose to train models of decoupage and montage directly from examples of movies and movie scripts. Similarly to previous work such as [Car80] they assume the problem of film directing is to translate a sequence of events into a sequence of shots. They make the simplifying assumption that this translation is governed by a hidden Markov model (HMM) with a finite vocabulary of events and shots. They train their model from examples of events and shots from real movies and their scripts, and use the trained model to generate novel shot sequences from novel event sequences. The finite vocabulary assumption limits their approach to specialized domains.

5.4.3. Example-based camera behaviors [JWW*20]

The most recent attempt in learning film directing from examples deals exclusively with the problem of mise-en-scène, and more specifically the problem of generating expressive and meaningful camera movements reacting to the actor's actions and movements. The problem is challenging in the context of real-time computer graphics and games because cinematographic camera movements need to be planned in advance and are not easily amenable to real-time implementations. Previous work on this topic has proposed the notion of a camera motion graph, where camera movements are extracted from existing movies and pieced together at runtime [SDM*15]. Here, the authors propose to learn camera motion from examples of real movies and to retarget the learned camera motions to new animation sequences based on similarity of actor poses, movements and actions. To do this, they train a hybrid neural network (LSTM + gating) on actor poses extracted from real movie scenes, and transfer them to actor positions in a virtual world. Their system produces aesthetic camera movements in 3D games in near real time, with a delay. Future work is needed to produce similar results in real time without delay using predictive deep networks. Another promising direction for future research is to take into account other elements of the story than relative actor positions.

6. Applications

Film directing techniques covered in this survey are applicable to many domains in computer graphics, and we review some of them here. They include interactive storytelling, where the story itself is generated in real time and must be directed immediately; text-to-movie generation, where a written screenplay is used to produce an animation movie; machinima production, where a real-time game engine is used to produce a narrative movie; cinematic replay, where a game session is recorded and replayed with new camera angles for an audience; and immersive virtual reality, where transitions between shots become teleportations between virtual spaces under the film director's control.

We can classify the papers in our survey according to their targeted applications. This reveals important differences in the kind of input they take and the difficulty of the task which is expected of

them. In earlier work, methods were proposed to generate movies from existing footage, as in live cinema and brodacast applications. This involves montage, but not decoupage or mise-en-scene. Then with advances in computer graphics and video games, new methods were proposed to generate movies from existing animation, allowing novel applications such as virtual production, machinima, cut scene generation, and cinematic replays. This involves decoupage and limited mise-en-scene, where only the placement and motion of the camera needs to be computed. The new generation of automated film-making is now starting to propose methods to generate movies from screenplays, including the placement and animation of virtual actors. This makes them applicable to text-to-movie generation and interactive drama.

Table 3 below summarizes the classification of methods and applications based on two criteria. On the horizontal axis, we separate methods and applications based on the required input (footage, animation or screenplay). On the vertical axis, we separate methods and applications which are performed offline or in real-time. This creates a matrix-like representation with increasing requirements from left to bottom.

6.1. Live cinema

Live cinema is a concept forged by Francis Ford Coppola to describe a new art form borrowing form live theater, television and movie making, with the goal of creating complete movies with live actors in real time in front of an audience [Cop17]. While Coppola is most interested in playing the part of the director himself, live cinema provides great opportunities for automated film directing as well, with possible applications to cinematographic broadcasts of live performances.

6.2. Virtual production

Virtual production is the process of recording live actor performances using motion capture, rather than video cameras, so that different option in the decoupage, mise-en-scene and montage of the recorded scene can be computed offline during post-production. Many techniques described in this survey are directly applicable to virtual production, and will likely be used in future work to overcome the combinatorial explosion of choices now facing the film director.

6.3. Machinima

Machinima has been a motivation for many techniques in this survey and some commercial machinima systems are beginning to include limited support for automatic film editing. For instance, MovieStorm [Mov] includes "through-the-lens" camera control with two types of cameras. A "free" camera can pan, tilt or roll around the camera axis in all directions. A "sticky" camera is attached to an actor's eye line. The camera can pan, tilt or roll around the actor's eye axis. In principle, it is possible to implement sticky cameras on other targets, and multiple targets. This can be extended to compute dolly paths as well. For two-shots, the camera can move along a circle while keeping the two actors in any given screen position. Few machinima systems include support for automatic decoupage or montage. One may expect that this will change in the near future. One difficult that needs to be resolved is the need for high level description of actions and events to motivate the choice of shots and transitions. In a game engine, such high-level descriptions can in principle be inferred from the player's or puppeteer's actions. But player's intentions cannot easily be inferred from their movements. Non-player characters (NPC) have a more formalized vocabulary of intentions and actions. This could be used to motivate the decoupage and the montage of machinima sequences. We therefore expect to see automatic film directing techniques gradually become integrated in the next generation of computer games for machinima production.

6.4. Highlight cinematics

Many computer games, provide methods for recording and past moments (highlights) and replaying them with new camera angles (cinematics). For example, first-person games may be replayed as third-person movies. And online multiplayer games may be replayed by alternating between players. This is a good application area for automatic film directing techniques, and several methods surveyed in this paper are devoted to solving this class of problems. Assa et al. target the creation of highlight cinematics in sports games [AWCO10]. Dominguez et al. [DYR11] apply Darshak [JY10] to recognize and generate highlight cinematics. A related application is the automatic broadcast of sports games, which requires to be computed in real time. We anticipate that this class of applications will continue to drive research in automated film directing in years to come.

6.5. Text-to-movie generation

Text-to-movie generation combines natural language processing and the staging of virtual actors (text-to-scene) with the film directing techniques reviewed in this survey to automatically produce movies from written scripts. An example is the text-to-movie system by Nawmal Technologies [Naw]. This includes various declarative shots - one-shots and two-shots with a variety of camera angles and compositions. Camera placement is automated for declarative shots. Editing is fully automated and makes use of both declarative shots (idioms) and free cameras. This overcomes the traditional limitations associated with a purely idiom-based system. Visibility is taken into account through the use of "stages", i.e. empty spaces with unlimited visibility, similar to [ER07]. Both systems use a simple algebra of "stages", i.e. intersections and unions of stages, allowing for very fast visibility computation against the static elements of the scene. Occlusion between actors is handled separately by taking pictures through the eyes of the actors. The Nawmal text-to-scene system is currently limited to short dialogue scenes, although with a rich vocabulary of gestures, facial expressions and movements. But we can expect future improvements and extensions with similar systems capable of generating other scene categories, including action and mood scenes.

6.6. Interactive drama

Interactive drama promises to become a hybrid between film and game, with a very strong need for fully automated real-time film directing, so that all possible navigation paths through the "story

Table 3: *Applications and requirements. From left to right : the methods covered in this survey differ in the required input, working from existing footage, or existing animation, or existing script. Top to bottom: we also distinguish between methods that work offline, and methods that work in real-time.*

	From footage	From animation	From screenplay
Offline methods	Automated film editing [Car80] [Blo86] [CNN*05] [LDTA17] [WYH*19]	Machinima [DGZ92] [ER07, RRE08] [JY05, JY06, JY10] [GCR*13] [MCB15] [GRLC15], Cinematic replay [AWCO10]	Text-to-movie [KM02] [FF04, FF06] [LCL18]
Real-time methods	Live-cinema [SB12] [AGV16]	Third person games [DZ95] [HCS96a] [CAH*96] [PMC*10] [MKSB11] [Par20] [JWW*20]	Interactive drama [BGL98, BL99] [Fun00] [TBN00] [SMAY03a, SMAY03b]

graph" generate movies that are aesthetically pleasing. FACADE by Mateas and Stern is a good example, although with a very simple cinematic look and feel [MS03].

7. Discussion and Open Issues

In this section, we review alternative directions for future research. Our survey shows that few methods cover the three tasks of decoupage, mise-en-scene and montage, and we examine possible ways to better integrate them in future work. Our survey also reveals some severe imitations in the spatial and temporal complexity that can be handled by the state of the art, and we review possible directions to overcome them. We then review the challenges and opportunities for learning methods. We propose novel directions of research to make automated film directing more expressive and more creative. We conclude this section with a critical review of the difficult problem of experimental validation.

7.1. Integrated solutions

Integrated solutions to decoupage, mise-en-scene and montage are still uncommon. Multi-agent architectures have been proposed as a promising direction of research [Haw05]. But they raise difficult issues in coordinating the work of different agents with different goals, and no complete and convincing implementation of multi-agent film directing have been proposed yet. A promising approach that combines AI-based planning with optimization is hierarchical task network (HTN) planning with preferences [SBM09], which has been used in game AI to solve military team planning. While we are not aware of it being used in film directing, it appears to be a likely candidate for future work in this direction.

7.2. Spatial and temporal complexity

We can classify the papers in this survey based on the spatial and temporal complexity that they can handle. On the temporal scale, we distinguish methods that work at the level of individual shots, methods that work at the level of individual scenes, and methods that work at the level of complete movies. On the spatial scale, complexity can be measured with the number of actors in each camera shot. The vast majority of methods in our survey is limited to a dozen camera shots with two or three actors. How can future work address the case of the long term narrative structure of movies with a large cast of characters ? One possible direction of research is to leverage the accumulated knowledge in decoupage, mise-en-scène and montage of short scenes with two or three main actors, and to use them as building blocks for the long term structure of movies. This is consistent with some combinatorial approaches in dramaturgy, where a story is built up by assembling narrative events as in a game of dominoes [Bal83]. Another possible route for future research is to take the opposite direction and first attempt to break the spatial limitations of the state of the art, i.e. make it possible to generate complex scenes with unlimited numbers of actors and actions, based on more general principles of decoupage, mise-en-scene and montage. Long-term structure in this case would be obtained by making the scenes increasingly long and complex until they become complete movies. Those two directions of research appear to be equally valid and necessary.

Video summarization is a related line of research which can shed light on the long term structure of movies and help alleviate some of the temporal limitations of current work [dMTLT17]. The works of Friedman et al [DFD04], Lu et al. [LG13] and Arev et al. [APS*14] are especially relevant in the context of this survey.

7.3. Learning film directing

In recent years, deep learning approaches have been used with great success in the task of video classification and captioning [WYFJ17, CYJ19]. Deep learning is also actively researched in the related task of analyzing film style and authorship [SSS*18]. In contrast, our survey has found only few instances of deep learning methods being used to generate movies either from existing footage, or from existing animation, or from screenplays.

In this subsection,we propose some explanations and hypothetical directions of research for learning film directing, including montage, decoupage and mise-en-scène from examples using deep learning methods.

Learning montage. Continuity editing and its classical rules appear to be a good candidate for deep learning methods. A major difficulty is the lack of datasets including both the available camera shots and the edited shots. This is likely a vey rich and promising area for future research, which could benefit from shared datasets and competitions.

Learning decoupage. Decoupage is a hugely difficult problem requiring long term structure understanding. A suitable goal would be to learn to generate storyboards of existing movies from their screenplays. This would require a large dataset of paired screenplays and storyboards, which are not readily available. As a remedy, it should be possible in principle to reverse engineer storyboards of existing movies [GCSS06] and use them to train deep learning methods. This is a promising direction for future work.

Learning mise-en-scène . Many aspects of mise-en-scène can be learned from real movie examples. A suitable goal would be to learn to generate 3D animation, including character and camera animation, from storyboards. The virtual director would now need to place actors and direct them as well. This is a hugely difficult problem, beyond the current state of the art, but one that can be solved at least in principle with deep learning. Even a simplified, symbolic version of the storyboard could be used in this task [RGBM20]. This would require large datasets of paired storyboards and movie shots, which are not readily available. Again, the storyboards could be reverse-engineered from the existing movies, which is a promising direction for future work. As an alternative, animation studios could use their own storyboards to train deep learning methods in a self-supervised fashion.

End-to-end learning. Learning to generate complete movies from screenplays has never been attempted and appears to be well beyond the state of the art in both deep learning and automated film directing. Given the recent success of end-to-end deep learning methods in both computer graphics and computer vision, it is legitimate to ask whether this is a viable future direction for film directing. On the positive side, large datasets of paired screenplays and movies are readily available. On the negative side, the problem of generating consistent and appealing video sequences at the scale of an entire movie would require control of the long term structure of the movie, which is far beyond the state of the art in deep learning at the time of writing. Instead, a more likely path to the grand challenge of automated film directing from screenplay to movie may be to first learn mise-en-scène from storyboards, then use this to generate large amounts of synthetic camera shots, and use this synthetic data to train self-supervised methods of decoupage and montage.

7.4. Expressive film directing

Computational analysis of film style is a new and important area of research [Sal03, Cut14, CC15, PISM*20] which can have a huge impact on film directing. Most previous work reported in this survey has relied on more or less general rules, resulting in more or less style-less movies. Larger datasets of movies in different styles could be used to produce much more interesting and creative movies. Yet many problems also need to be resolved to pursue this new direction, starting with the core theoretical problem of separating film content from film style. In the light of this survey, we conjecture that it may be beneficial to examine separately decoupage styles, mise-en-scene styles and montage styles.

Furthermore, the emotional content of a shot or a transition between shots has been little explored [SG03, Smi05] and is a promising avenue for building more expressive movies.

7.5. Creative film directing

Automated film-making is most needed in cases where movies must be create in real time, which prevents human intervention. This is the case in video games and interactive drama. Much work has been devoted to the generation of third-person games where the rules of classical cinema are directly applicable. Another promising application is the generation of first-person games and immersive virtual reality, where new cinematic rules need to be discovered. In those cases, future work will be needed to separate the planning steps from the execution steps and provide creativity tools allowing to discover suitable film idioms and automate their execution. In this case, the distinction between decoupage, mise-en-scene and montage must be revisited slightly, because all of them need to be executed multiple times, to account for player actions.

In first-person immersive VR, shots are replaced by immersive experiences in virtual worlds where the player takes control of the camera, and cuts are replaced by teleportations between worlds. In addition to the duration and ordering of shots, this new kind of "spatial montage" may then include decisions on the relative positions, directions and sizes of those worlds, with new requirements to maintain consistency in spatial orientation during teleportations. Mise-en-scene is primarily in the hands of the player in those applications, which can make them an excellent playing ground for future research in creative decoupage and montage.

7.6. Evaluation

Evaluation of automated film directing has been attempted by only a few researchers [LRGG14, RET*20]. The result seems to be that it is relatively easy to emulate an "amateur" director, but very hard to emulate even a modest "professional" director. In other words, empirical evaluations show that a professionally directed scene is always preferred to a machine-generated scene. But a machine-generated scene can be preferred (or found comparable) to an amateur-directed scene. Another possible evaluation criteria is *ease of use*. For example, it would be useful to compare the time needed for generating a movie scene with different inputs and methods. Future work is needed to organize competitions in automated film directing with well-defined goals that can be evaluated quantitatively, so that the proposed techniques can be compared in a more principled fashion.

8. Conclusion

Many automatic film directing techniques have been proposed in the last forty years, but they have never been reviewed together. In this survey, we have proposed to classify them according to three classes of problems (decoupage, montage and mise-en-scène) and four categories of approaches (declarative, procedural, optimization and learning). The survey reveals the diversity of situations where automatic film directing has been investigated, measures the progress made towards the grand challenge of automated filmmaking, and outlines the shortcomings of existing methods and the challenges ahead for the computer graphics community in this important research field.

References

[AGV16] AERTS B., GOEDEMÉ T., VENNEKENS J.: A probabilistic logic programming approach to automatic video montage. In *Proceedings of the Twenty-second European Conference on Artificial Intelligence* (2016), ECAI'16, pp. 234–242. 6, 7, 9, 14

[APS*14] AREV I., PARK H. S., SHEIKH Y., HODGINS J., SHAMIR A.: Automatic editing of footage from multiple social cameras. *ACM Trans. Graph. 33*, 4 (July 2014). 14

[Ari76] ARIJON D.: *Grammar of the Film Language*. Hastings House Publishers, 1976. 11

[AWCO10] ASSA J., WOLF L., COHEN-OR D.: The virtual director: a correlation-based online viewing of human motion. *Comput. Graph. Forum 29*, 2 (2010), 595–604. 6, 7, 11, 13, 14

[Bal83] BAL D.: *Backwards & Forwards: A Technical Manual for Reading Plays*. Southern Illinois University Press, 1983. 14

[Bar20] BARNARD T.: Découpage. In *Essays on film form*. Caboose, Montreal, 2020. 1

[BGGR17] BARES W. H., GANDHI V., GALVANE Q., RONFARD R. (Eds.):. *6th Workshop on Intelligent Cinematography and Editing, Lyon, France* (2017), Eurographics Association. 1

[BGL98] BARES W. H., GREGOIRE J. P., LESTER J. C.: Realtime constraint-based cinematography for complex interactive 3D worlds. In *Proceedings of AAAI-98/IAAI-98* (1998), pp. 1101–1106. 6, 7, 8, 14

[BL99] BARES W. H., LESTER J. C.: Intelligent multi-shot visualization interfaces for dynamic 3D worlds. In *Proceedings of the 4th international conference on Intelligent user interfaces (IUI 99)* (New York, NY, USA, 1999), ACM Press, pp. 119–126. 6, 7, 8, 14

[Blo86] BLOCH G.: *Eléments d'une machine de montage pour l'audiovisuel*. PhD thesis, T'elécom Paris, 1986. 5, 6, 7, 8, 14

[Blo87] BLOCH G.: *From concepts to film sequences*. Tech. rep., Yale University, 1987. 8

[Bra92] BRANIGAN E.: *Narrative comprehension and film*. Routledge, 1992. 3

[Bro86] BROOKS F. P.: Walkthrough—a dynamic graphics system for simulating virtual buildings. In *Proceedings of the 1986 Workshop on Interactive 3D Graphics* (New York, NY, USA, 1986), I3D '86, Association for Computing Machinery, p. 9–21. 10

[Bur81] BURCH N.: *Theory of film practice*. Princeton University Press, Princeton, New Jersey, 1981. 8

[CAH*96] CHRISTIANSON D. B., ANDERSON S. E., HE L., SALESIN D. H., WELD D. S., COHEN M. F.: Declarative Camera Control for Automatic Cinematography. In *Proceedings of the American Association for Artificial Intelligence (AAAI 96)* (1996), pp. 148–155. 6, 7, 8, 14

[Car77] CARROLL J. M.: A program for cinema theory. *The Journal of Aesthetics and Art Criticism 35* (1977), 337. 5

[Car80] CARROLL J. M.: *Toward a Structural Psychology of Cinema*. Mouton, The Hague and New York, 1980. 5, 6, 7, 12, 14

[Car81] CARROLL J. M.: A linguistic analysis of deletion in cinema. *Semiotica 34*, 1/2 (1981), 25–53. 5

[Car82] CARROLL J. M.: Structure in visual communication. *Semiotica 40*, 3/4 (1982), 371–392. 5

[CC15] CUTTING J. E., CANDAN A.: Shot durations, shot classes, and the increased pace of popular movies. *Projections 9*, 2 (2015), 40 – 62. 4, 15

[CNN*05] CALLAWAY C., NOT E., NOVELLO A., ROCCHI C., STOCK O., ZANCANARO M.: Automatic cinematography and multilingual nlg for generating video documentaries. *Artificial Intelligence 165*, 1 (2005), 57–89. 6, 7, 9, 14

[CON08] CHRISTIE M., OLIVIER P., NORMAND J.-M.: Camera control in computer graphics. *Computer Graphics Forum 27*, 8 (2008), 2197–2218. 1

[Cop17] COPPOLA F. F.: *Live cinema and its techniques*. Liveright Publishing Corporation, 2017. 13

[Cut14] CUTTING J. E.: Event segmentation and seven types of narrative discontinuity in popular movies. *Acta Psychologica 149* (2014), 69 – 77. 3, 15

[CWLG20] CHRISTIE M., WU H., LI T., GANDHI V. (Eds.):. *9th Workshop on Intelligent Cinematography and Editing, Norrköping, Sweden* (2020), Eurographics Association. 1

[CYJ19] CHEN S., YAO T., JIANG Y.-G.: Deep learning for video captioning: A review. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19* (7 2019), International Joint Conferences on Artificial Intelligence Organization, pp. 6283–6290. 14

[DFD04] D. FRIEDMAN A. SHAMIR Y. A. F., DAGAN T.: Automated creation of movie summaries in interactive virtual environments. In *IEEE Virtual Reality* (2004), pp. 191–290. 14

[DGZ92] DRUCKER S. M., GALYEAN T. A., ZELTZER D.: Cinema: A system for procedural camera movements. In *Symposium on Interactive 3D graphics* (New York, NY, USA, 1992), ACM Press, pp. 67–70. 6, 7, 10, 14

[dMTLT17] DEL MOLINO A. G., TAN C., LIM J.-H., TAN A.-H.: Summarization of egocentric videos: A comprehensive survey. *IEEE Transactions on Human-Machine Systems 47*, 1 (February 2017). 14

[DSP91] DAVENPORT G., SMITH T. A., PINCEVER N.: Cinematic primitives for multimedia. *Computer Graphics and Appliations 11*, 4 (July 1991), 67–74. 10

[DYR11] DOMINGUEZ M., YOUNG R. M., ROLLER S.: Automatic identification and generation of highlight cinematics for 3d games. In *Foundations of Digital Games* (New York, 2011), ACM, p. 259–261. 13

[DZ95] DRUCKER S. M., ZELTZER D.: Camdroid: A System for Implementing Intelligent Camera Control. In *Proceedings of the 1995 symposium on Interactive 3D graphics (SI3D 95)* (1995), pp. 139–144. 6, 7, 11, 14

[ER07] ELSON D. K., RIEDL M. O.: A lightweight intelligent virtual cinematography system for machinima production. In *Proceedings of the Third AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (2007), AIIDE'07, AAAI Press, pp. 8–13. 6, 7, 11, 13, 14

[FF04] FRIEDMAN D. A., FELDMAN Y. A.: Knowledge-Based Cinematography and Its Applications. In *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI 04)* (2004), IOS Press, pp. 256–262. 6, 7, 9, 14

[FF06] FRIEDMAN D., FELDMAN Y. A.: Automated cinematic reasoning about camera behavior. *Expert Systems with Applications 30*, 4 (May 2006), 694–704. 6, 7, 9, 14

[For20] FORESTIER L. L.: Montage. In *Essays on film form*. Caboose, Montreal, 2020. 1

[Fun00] FUNGE J.: Cognitive modeling for games and animation. *Commun. ACM 43*, 7 (July 2000), 40–48. 6, 7, 8, 14

[GAM] Gamastutra, the art and business of making games. https://www.gamasutra.com/. 10

[GC18] GORJI S., CLARK J. J.: Going from image to video saliency: Augmenting image salience with dynamic attentional push. In *Computer Vision and Pattern Recognition* (2018), p. 7501–7511. 4

[GCR*13] GALVANE Q., CHRISTIE M., RONFARD R., LIM C.-K., CANI M.-P.: Steering behaviors for autonomous cameras. In *Proceedings of Motion on Games* (2013), MIG '13, p. 93–102. 6, 7, 10, 14

[GCSS06] GOLDMAN D. B., CURLESS B., SALESIN D., SEITZ S. M.: Schematic storyboarding for video visualization and editing. *ACM Trans. Graph. 25*, 3 (July 2006), 862–871. 15

[Gd07] GERMEYS F., D'YDEWALLE G.: The psychology of film: perceiving beyond the cut. *Psychological Research 71*, 4 (2007), 458–466. 3

[GDC] Game developers conferences. https://gdcvault.com/. 10

[God56] GODARD J.-L.: Montage, mon beau souci. *Les cahiers du cinéma 11*, 65 (décembre 1956). 3

[GR17] GALVANE Q., RONFARD R.: Implementing hitchcock: The role of focalization and viewpoint. In *Proceedings of the Eurographics Workshop on Intelligent Cinematography and Editing* (2017), WICED '17, pp. 5–12. 11

[GRCS14] GALVANE Q., RONFARD R., CHRISTIE M., SZILAS N.: Narrative-driven camera control for cinematic replay of computer games. In *Proceedings of the Seventh International Conference on Motion in Games* (2014), MIG '14, pp. 109–117. 11, 12

[GRLC15] GALVANE Q., RONFARD R., LINO C., CHRISTIE M.: Continuity editing for 3d animation. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence* (2015), AAAI'15, pp. 753–761. 4, 6, 7, 11, 14

[Haw05] HAWKINS B.: *Real-time cinematography for games*. Charles River Media, 2005. 10, 14

[HCS96a] HE L., COHEN M., SALESIN D.: The virtual cinematographer: a paradigm for automatic real-time camera control and directing. In *SIGGRAPH '96* (1996), pp. 217–224. 6, 7, 10, 14

[HCS96b] HE L.-W., COHEN M. F., SALESIN D. H.: 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* (1996), SIGGRAPH '96, pp. 217–224. 10, 12

[HDH04] HAYASHI M., DOUKE M., HAMAGUCHI N.: Automatic tv program production with apes. In *Proceedings. Second International Conference on Creating, Connecting and Collaborating through Computing, 2004.* (2004), pp. 20–25. 9

[HH09] HAIGH-HUTCHINSON M.: *Real Time Cameras: A Guide for Game Designers and Developers*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2009. 10

[IKN98] ITTI L., KOCH C., NIEBUR E.: A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence 20*, 11 (1998), 1254–1259. 4

[JWW*20] JIANG H., WANG B., WANG X., CHRISTIE M., CHEN B.: Example-driven virtual cinematography by learning camera behaviors. *ACM Trans. Graph. 39*, 4 (July 2020). 6, 7, 12, 14

[JY05] JHALA A., YOUNG R. M.: A discourse planning approach to cinematic camera control for narratives in virtual environments. In *AAAI'05: Proceedings of the 20th national conference on Artificial intelligence* (2005), AAAI Press, pp. 307–312. 6, 7, 9, 14

[JY06] JHALA A., YOUNG R. M.: Representational requirements for a plan based approach to automated camera control. In *AIIDE* (2006), pp. 36–41. 6, 7, 9, 14

[JY10] JHALA A., YOUNG R.: Cinematic visual discourse: Representation, generation, and evaluation. *IEEE Transactions on Computational Intelligence and AI in Games 2*, 02 (2010), 69–81. 6, 7, 9, 13, 14

[Kat91] KATZ S.: *Film Directing Shot by Shot: Visualizing from Concept to Screen*. Michael Wiese Productions, 1991. 2, 11

[KB99] KLINE C., BLUMBERG B.: The art and science of synthetic character design. In *Symposium on AI and Creativity in Entertainment and Visual Art* (Edinburgh, Scotland, 1999), AISB. 10

[Kes20] KESSLER F.: Mise en scène. In *Essays on film form*. Caboose, Montreal, 2020. 1

[KF93] KARP P., FEINER S.: Automated presentation planning of animation using task decomposition with heuristic reasoning. In *Graphics Interface* (1993). 6, 7, 8

[KM02] KENNEDY K., MERCER R. E.: Planning animation cinematography and shot structure to communicate theme and mood. In *Smart graphics* (New York, NY, USA, 2002), ACM, pp. 1–8. 6, 7, 8, 14

[Koc84] KOCHANEK D.: Interpolating splines with local tension, continuity, and bias control. *Proceedings of SIGGRAPH 18*, 3 (1984), 33–42. 10

[LCL18] LOUARN A., CHRISTIE M., LAMARCHE F.: Automated staging for virtual cinematography. In *Proceedings of the 11th Annual International Conference on Motion, Interaction, and Games* (2018), MIG '18. 6, 7, 9, 14

[LDTA17] LEAKE M., DAVIS A., TRUONG A., AGRAWALA M.: Computational video editing for dialogue-driven scenes. *ACM Trans. Graph. 36*, 4 (July 2017). 6, 7, 12, 14

[LG13] LU Z., GRAUMAN K.: Story-driven summarization for egocentric video. In *Computer Vision and Pattern Recognition* (2013), pp. 2714–2721. 14

[LLMS15] LOSCHKY L., LARSON A., MAGLIANO J., SMITH T.: What would jaws do? the tyranny of film and the relationship between gaze and higher-level narrative film comprehension. *PLoS ONE 10*, 11 (2015). 4

[LRGG14] LINO C., RONFARD R., GALVANE Q., GLEICHER M.: How Do We Evaluate the Quality of Computational Editing Systems? In *AAAI Workshop on Intelligent Cinematography And Editing* (Québec, Canada, July 2014), AAAI, pp. 35–39. 15

[LRL*97] LEVESQUE H. J., REITER R., LESPÉRANCE Y., LIN F., SCHERL R. B.: Golog: A logic programming language for dynamic domains. *The Journal of Logic Programming 31*, 1 (1997), 59 – 83. 8

[Mas65] MASCELLI J.: *The Five C's of Cinematography: Motion Picture Filming Techniques*. Cine/Grafic Publications, Hollywood, 1965. 2, 3, 11

[MCB15] MERABTI B., CHRISTIE M., BOUATOUCH K.: A Virtual Director Using Hidden Markov Models. *Computer Graphics Forum* (2015). 6, 7, 12, 14

[McK97] MCKEE R.: *Story: Style, Structure, Substance, and the Principles of Screenwriting*. Harper Collins, New York, 1997. 11

[MKSB11] MARKOWITZ D., KIDER J. T., SHOULSON A., BADLER N. I.: Intelligent camera control using behavior trees. In *Proceedings of the 4th International Conference on Motion in Games* (2011), MIG'11, pp. 156–167. 6, 7, 10, 14

[Mov] Moviestorm filmmaker. http://www.moviestorm.co.uk/. 13

[MS03] MATEAS M., STERN A.: Facade: An experiment in building a fully-realized interactive drama. In *Game Developers Conference* (2003). 14

[MT88] MANN W. C., THOMPSON S. A.: Rhetorical structure theory: Toward a functional theory of text organization. *Text & Talk 8*, 3 (1988), 243 – 281. 9

[Mur86] MURCH W.: *In the blink of an eye*. Silman-James Press, 1986. 3, 10

[Naw] Nawmal text-to-movie. http://nawmal.com. 13

[ORN09] O'NEILL B., RIEDL M. O., NITSCHE M.: Towards intelligent authoring tools for machinima creation. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems* (New York, 2009), ACM, p. 4639–4644. 11

[Par20] PARK W.: Intelligent camera using a finite-state machine (fsm). In *2020 14th International Conference on Ubiquitous Information Management and Communication (IMCOM)* (2020), pp. 1–9. 6, 7, 10, 14

[Pep05] PEPPERMAN R.: *Setting Up Your Scenes: The Inner Workings of Great Films*. Michael Wiese Productions, Studio City, CA, 2005. 2

[PISM*20] PUSTU-IREN K., SITTEL J., MAUER R., BULGAKOWA O., EWERTH R.: Automated visual content analysis for film studies: Current status and challenges. *Digital Humantities Quaterly 14*, 4 (2020). 15

[PMC*10] PASSOS E. B., MONTENEGRO A., CLUA E. W. G., POZZER C., AZEVEDO V.: Neuronal editor agent for scene cutting in game cinematography. *Comput. Entertain. 7*, 4 (Jan. 2010). 6, 7, 12, 14

[Pro08] PROFERES N. T.: *Film Directing Fundamentals, See Your Film Before Shooting*, third edition ed. Focal Press, Boston, 2008. 2

[RBJ14] RONFARD R., BURELLI P., JHALA A. (Eds.):. *3rd Workshop on Intelligent Cinematography and Editing, Québec, Canada* (2014), vol. WS-14-06 of *AAAI Workshops*, AAAI Press. 1

[RCB15] RONFARD R., CHRISTIE M., BARES W. H. (Eds.):. *4th Workshop on Intelligent Cinematography and Editing, Zurich, Switzerland* (2015), Eurographics Association. 1

[RCG16] RONFARD R., CHRISTIE M., GALVANE Q. (Eds.):. *5th Workshop on Intelligent Cinematography and Editing, Lisbon, Portugal* (2016), Eurographics Association. 1

[Rei01] REITER R.: *Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems*. The MIT Press, 2001. 8

[RET*20] RADUT M., EVANS M., TO K., NOONEY T., PHILLIPSON G.: How good is good enough? the challenge of evaluating subjective quality of ai-edited video coverage of live events. In *Intelligent Cinematography and Editing* (2020), Eurographics Association, pp. 17–24. 15

[Rey99] REYNOLDS C.: Steering behaviors for autonomous characters. In *Game developers conference* (1999). 10

[RGBM20] RONFARD R., GANDHI V., BOIRON L., MURUKUTLA V. A.: The prose storyboard language: A tool for annotating and directing movies. https://arxiv.org/abs/1508.07593v4, 2020. 9, 15

[Rie14] RIEDL M.: Toward the grand challenge of automated filmmaking. https://www.cc.gatech.edu/~riedl/talks/aaai-wiced-ws.pdf, 2014. 1

[RN02] RUSSELL S., NORVIG P.: *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2002. 8

[RRE08] RIEDL M. O., ROWE J. P., ELSON D. K.: Toward intelligent support of authoring machinima media content: story and visualization. In *Proceedings of the 2nd international conference on INtelligent TEchnologies for interactive enterTAINment* (2008), INTETAIN '08. 6, 7, 11, 14

[SAD*06] SANTELLA A., AGRAWALA M., DECARLO D., SALESIN D., COHEN M.: Gaze-based interaction for semi-automatic photo cropping. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems* (New York, NY, USA, 2006), ACM, pp. 771–780. 4

[Sal03] SALT B.: *Film Style and Technology: History and Analysis (2nd edition)*. Starword, 2003. 2, 4, 15

[SB12] SUCHAN J., BHATT M.: Toward high-level dynamic camera control: An integrated qualitative-probabilistic approach. In *International Workshop on Qualitative Reasoning (QR)* (2012). 6, 7, 9, 14

[SBK07] SZILAS N., BARLES J., KAVAKLI M.: An implementation of real-time 3d interactive drama. *ACM Computers in Entertainment 5*, 1 (2007). 12

[SBM09] SOHRABI S., BAIER J. A., MCILRAITH S. A.: Htn planning with preferences. In *IJCAI'09: Proceedings of the 21st international jont conference on Artifical intelligence* (San Francisco, CA, USA, 2009), Morgan Kaufmann Publishers Inc., pp. 1790–1797. 14

[SD94] SACK W., DAVIS M.: Idic: assembling video sequences from story plans and contentannotations. In *Multimedia Computing and Systems* (1994), pp. 30 – 36. 6, 7, 8

[SDM*15] SANOKHO C., DESOCHE C., MERABTI B., LI T.-Y., CHRISTIE M.: Camera motion graphs. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Goslar, DEU, 2015), SCA '14, Eurographics Association, p. 177–188. 12

[SG03] SALWAY A., GRAHAM M.: Extracting information about emotions in films. In *ACM Conference on Multimedia* (2003), pp. 299–302. 15

[Sha82] SHARFF S.: *The Elements of Cinema: Toward a Theory of Cinesthetic Impact*. Columbia Press, 1982. 2, 8

[SJH*17] SMITH J. R., JOSHI D., HUET B., HSU W., COTA J.: Harnessing a.i. for augmenting creativity: Application to movie trailer creation. In *Proceedings of the 25th ACM International Conference on Multimedia* (New York, 2017), ACM, p. 1799–1808. 12

[SMAY03a] SHEN J., MIYAZAKI S., AOKI T., YASUDA H.: Intelligent digital filmmaker dmp. In *Computational Intelligence and Multimedia Applications* (2003), pp. 272 – 277. 6, 7, 9, 14

[SMAY03b] SHEN J., MIYAZAKI S., AOKI T., YASUDA H.: Representing digital filmmaking techniques for practical application. In *Information and Knowledge Sharing* (2003). 6, 7, 9, 14

[Smi05] SMITH T. J.: *An Attentional Theory of Continuity Editing*. PhD thesis, University of Edinburgh, 2005. 3, 15

[SMKB13] SHOULSON A., MARSHAK N., KAPADIA M., BADLER N. I.: Adapt: The agent development and prototyping testbed. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (2013), I3D '13. 10

[SSH*10] STOCKER C., SUN L., HUANG P., QIN W., ALLBECK J. M., BADLER N. I.: Smart events and primed agents. In *Proceedings of the 10th International Conference on Intelligent Virtual Agents* (Berlin, Heidelberg, 2010), Springer-Verlag, p. 15–27. 10

[SSS*18] SVANERA M., SAVARDI M., SIGNORONI A., KOVÁCS A. B., BENINI S.: Who is the director of this movie? automatic style recognition based on shot features. *CoRR abs/1807.09560* (2018). 14

[Tan18] TAN E. S.: A psychology of the film. *Palgrave Communications 4*, 82 (2018). 3

[TBN00] TOMLINSON B., BLUMBERG B., NAIN D.: Expressive autonomous cinematography for interactive virtual environments. In *Proceedings of the Fourth International Conference on Autonomous Agents* (Barcelona, Catalonia, Spain, 2000), ACM Press, pp. 317–324. 6, 7, 10, 14

[Tho98] THOMPSON R.: *Grammar of the Shot*. Focal Press, 1998. 11

[TKWB20] TANGEMANN M., KUMMERER M., WALLIS T. S. A., BETHGE M.: Measuring the importance of temporal features in video saliency. In *European Conference on Computer Vision* (2020), p. 667–684. 4

[Tru85] TRUFFAUT F.: *Hitchcock-Truffaut (Revised Edition)*. Simon and Schuster, 1985. 1, 5

[VDB09] VENNEKENS J., DENECKER M., BRUYNOOGHE M.: Cp-logic: A language of causal probabilistic events and its relation to logic programming. *Theory and Practice of Logic Programming 9*, 3 (2009), 245–308. 9

[WO90] WARE C., OSBORNE S.: Exploration and virtual camera control in virtual three dimensional environments. In *Proceedings of the 1990 Symposium on Interactive 3D Graphics* (1990), p. 175–183. 10

[WPRC18] WU H.-Y., PALÙ F., RANON R., CHRISTIE M.: Thinking like a director: Film editing patterns for virtual cinematographic storytelling. *ACM Trans. Multimedia Comput. Commun. Appl. 14*, 4 (Oct. 2018). 2

[WSJ18] WU H., SI M., JHALA A. (Eds.):. *7th Workshop on Intelligent Cinematography and Editing, Edmonton, Canada* (2018), vol. 2321 of *CEUR Workshop Proceedings*, CEUR-WS.org. 1

[WYFJ17] WU Z., YAO T., FU Y., JIANG Y.-G.: *Deep Learning for Video Classification and Captioning*. Association for Computing Machinery and Morgan & Claypool, 2017, p. 3–29. 14

[WYH*19] WANG M., YANG G.-W., HU S.-M., YAU S.-T., SHAMIR A.: Write-a-video: Computational video montage from themed text. *ACM Trans. Graph. 38*, 6 (Nov. 2019). 6, 7, 12, 14

[Zon05] ZONE R.: *3-D Filmmakers: Conversations with Creators of Stereoscopic Motion Pictures*. Scarecrow Press, Oxford, 2005. 4