Motion Overview of Human Actions

Jackie Assa Daniel Cohen-Or
Tel Aviv University
I-Cheng Yeh Tong-Yee Lee
National Cheng-Kung University, Taiwan

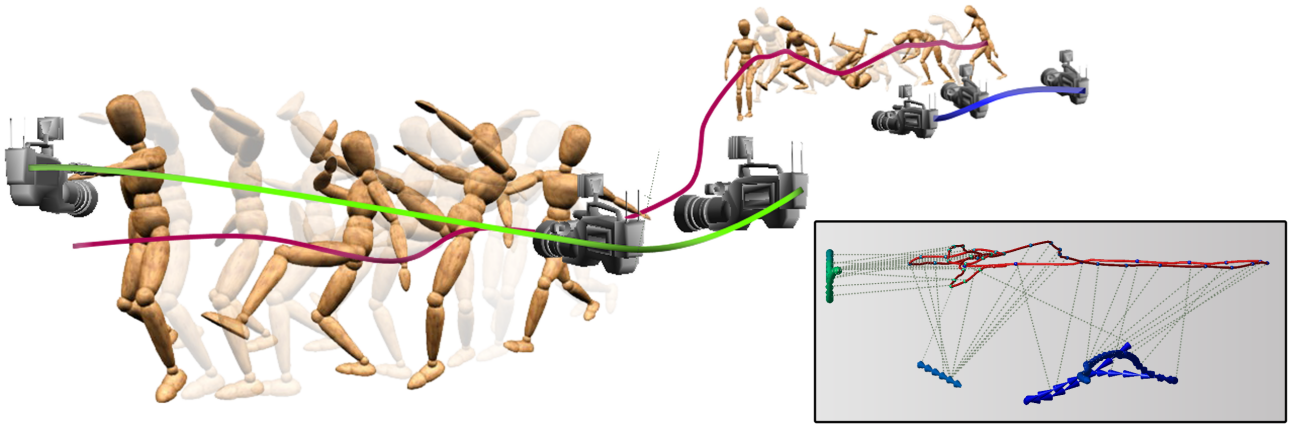


Figure 1: *Our algorithm analyzes human motion data to generate an overview video clip. On the right, a bird's eye view on the character reference location (shown in red) and our generated multi-shot camera control path (shown in blue-green)*

Abstract

During the last decade, motion capture data has emerged and gained a leading role in animations, games and 3D environments. Many of these applications require the creation of expressive overview video clips capturing the human motion, however sufficient attention has not been given to this problem. In this paper, we present a technique that generates an overview video based on the analysis of motion capture data. Our method is targeted for applications of 3D character based animations, automating, for example, the action summary and gameplay overview in simulations and computer games. We base our method on quantum annealing optimization with an objective function that respects the analysis of the character motion and the camera movement constraints. It automatically generates a smooth camera control path, splitting it to several shots if required. To evaluate our method, we introduce a novel camera placement metric which is evaluated against previous work and conduct a user study comparing our results with the various systems.

Keywords: mocap, animation, salient action, animation summary, viewpoint selection, camera

1 Introduction

Over the last three decades researchers from vision, computer graphics and robotics had been presenting methods for automatically generating camera control paths which can be used in a large set of 3D applications. This problem is considered hard because of the large configuration space, as well as the huge number of factors that can affect the camera control [Christie and Olivier 2006; McCabe and Kneafsey 2006; Pickering 2002; He et al. 1996; Halper and Olivier 2000]. The universal popularity of 3D games that are based on human animation and 3D virtual environments poses the camera control algorithms an even more difficult problem: the capturing of human action scenes. This problem is more difficult mainly since the human visual perception is subjective for this specialized task, but also due to the following reasons:

1. The human character is an articulated object with many degrees of freedom. As a result, human motions are complex and its analysis is considered hard.

2. Human actions generally include several participating limbs which have a prominent role in expressing the actual action. As a result, unlike in movement of simple geometric objects, the selection of expressive viewpoints should be affected by analyzing the motion and visibility of the body parts.

3. Small changes in the character pose often implies significant changes in the desired viewpoints for capturing this motion well.

4. The significance of the human actions is non-uniform over time. For example, routine actions such as a walk are visually less significant than a high action karate kick.

Many of the previous studies of this problem had formulated it as an optimization problem which maximizes viewpoint properties such as the subject's visibility, angle to its movement axis, while considering global properties such as the camera speed and director guidelines [Christie et al. 2005]. Applying such methods to human action scenes proves to be ineffective due to the reasons listed above, and generates results which are unsatisfactory. For example, the viewpoints are shown in Figure 2, and in extreme cases impose movements which include fast or shaky camera control path.

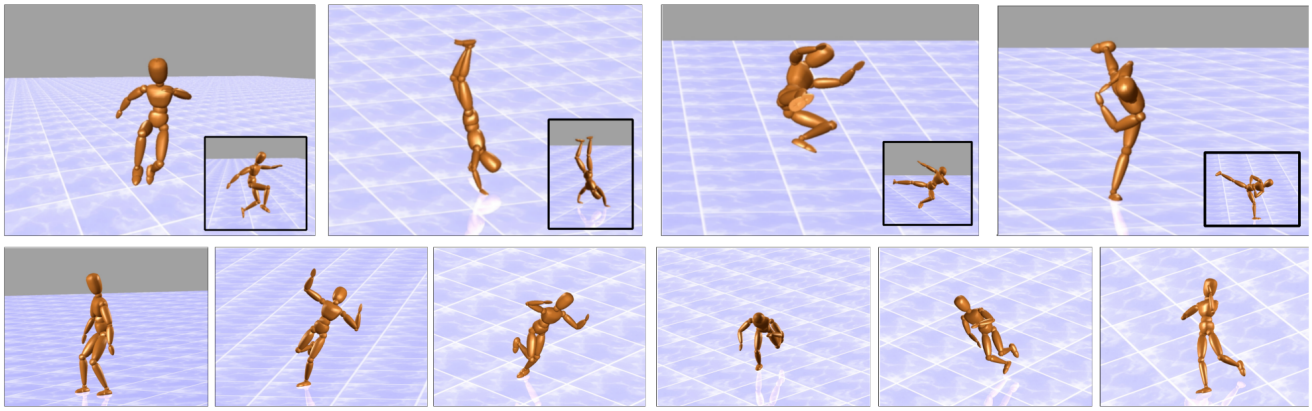


Figure 2: *Examples of poor selections from camera control algorithms which do not consider the human actions. (top row) Poor selection of viewpoint and occlusion of significant body parts for the actions. (bottom row) An image sequence caused by a abrupt camera movement does not illustrate well the action (fall).*

The goal of our method is to automatically generate a video which expresses and highlights the detailed human actions by using static and moderate camera movements. Our technique extends the concepts of existing methods and introduces a coherent and consistent framework for generating an automatic camera path for human animation scenes. Our method is effective for many applications such as 3D authoring tools, 3D environments, motion capture data preview and most human animation games. Nevertheless, as one of the early steps in this direction we focus on the post-processed overview of actions in games and 3D scenes. Other applications include also using it in a more professional scenario for assisting in reviewing motion capture library files. Christie and Olivier [2006] emphasize the high importance of the games recording and replay functions. Using our technique, these applications can achieve better results from considering the complete motion segment and the global analysis.

The novelty of our work is the introduction of a technique extending the scope of previous work to handle the challenging data of human motion and to define an automatic camera control for human character action overview. More specifically, this includes the fusion of segments saliency, motion capture analysis, limb saliency and limb viewpoint selection. Our optimization method considers the camera constraints and shot separations in a single framework to allow a better capturing of human action scenes.

Gameplay videos are used today in many of the gaming web sites, however in most cases, the generated video does not necessarily allow appreciation of the motion of the characters. Especially in 'first-person shooter' games, the default camera location for playing the game is designed to allow intuitive control. This camera restriction is no longer required to generate an overview of the game. Selecting a better viewpoint may increase the understanding of the scene motion. Moreover, many games implement a 'replay' mode, in which recently played actions are being presented again to the user using different viewpoints. Currently, these camera paths are predetermined and are not action related. Our algorithm can assist in generating a better camera control path for such cases, and provide a better overview of the scene. Other applications for our method, which benefit from its properties, include assistance in creation of camera control path in 3D authoring tools, and overview of clips motion capture data libraries.

Overview A camera control path is a set of seven dimensional points which describes for each frame the camera location, look-at vector and its field of view. As stated in [Christie et al. 2005] the construction of a path is treated by many as an optimization problem, which maximizes a viewpoint quality function composed of several scene attributes. In our case, we consider the viewpoint quality function as a linear combination of motion, orientation and human-pose related attributes. This combination yields a metric that quantifies the expressiveness of a viewpoint for a given pose. The metric is extended to describe the quality of the chosen path as the sum of the frames viewpoints. Our technique focuses on the camera location, whereas the other camera parameters such as the camera field of view, and the camera up vector are calculated based on the determined location and visible scene.

The expressiveness of a motion clip consisting of a set of frames can be naively considered as a selection of the best viewpoint for each frame. This usually generates a highly noisy camera path, since slight changes in the character orientation or pose may result in significant changes in the resulting viewpoint quality metric. These constraints require the optimization of the quality of the camera movement and the selected viewpoints it traverses. Since not all poses have the same significance (for example walking cycle poses are usually less significant than poses of a karate kick), we employ a non-uniform weight function based on the saliency of the motions.

The optimization method we propose is based on quantum annealing. This method is a simulated annealing algorithm which is suited for locating a minimal solution in a large configuration space [Apolloni et al. 1989]. This technique evaluates an energy term, defined by the external force of the viewpoint expressiveness metric and by the internal forces which express global camera movement constraints such as speed, acceleration, and panning changes. The usage of internal and external forces to calculate a path determines the sequence of viewpoints which presents a local minimum of the resulting forces. The same energy term is used to determine cases which can be expressed better using multiple camera shots, and to indicate in which frame the shots should be split. By splitting the shot into two different shots, we reduce the internal energy in the resulting path and potentially improve the overall expressiveness of the result. Once a path is split, each section of the path is optimized independently. This hierarchical construction guarantees a high viewpoint quality with a good balance over the number of shots.

Our technique consists of the following stages (shown in Figure 3): first we analyze a motion clip data for salient segments and de-

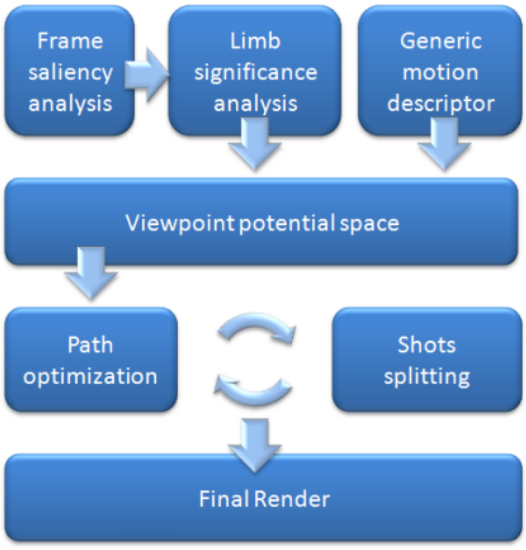


Figure 3: *System overview. We analyze the motion for saliency and significance, and combine them with known generic motion descriptors to generate the viewpoint potential space. Next we optimize the generated path and separate the result into multi-shots iteratively. Finally, we define the field of view and render the overview.*

tect salient body parts within these segments. Next we calculate the viewpoint potential metric and fuse it with the saliency information to generate a viewpoint potential space. Our optimization technique processes this potential space and designs a multi-shot path following a predefined set of camera movement constraints. The last stage of our algorithm sets the field of view and renders the overview video clip.

In the following sections we present some of the previous work and foundations for the presented technique (Section 2). In Section 3 our metric and saliency analysis is described, and the optimization technique is described in Section 4.

In the last sections of the paper we evaluate the performance of our method by comparing it to other methods using the predefined metric, and by presenting the results of a user study. We conclude with a description of the strong points and limitations of our method and future work.

2 Related Work

Camera control in computer graphics is a well established problem focusing on searching for a suitable camera configuration for capturing a scene narrative, while obeying a set of cinematographic rules [Mascelli 1965], as well as other constraints such as occlusion, objects visibility, layout in the resulting image [Gleicher and Witkin 1992], and orientations [Christie et al. 2005]. In this section, we survey only those studies which closely relate to our work. For a more comprehensive study on camera control, we refer the reader to the overview of [Christie and Olivier 2006].

Over the years, the art of cinematography has refined a set of standard principles of camera configurations and transitions [Arijon 1976; Mascelli 1965; Katz 1991], such as establishing the scene configuration, avoiding jump cuts, crossing movement lines and other rules. Following these principles, several studies have generated coherent camera control systems. For example, the Virtual Photographer of Li-Wei et al. [1996] constructs a state machine expressing cinematography idioms to control the camera. Others used numerous constraint-based approaches (e.g., [Drucker and Zeltzer 1994; Christianson et al. 1996; Bares et al. 2000; Halper et al. 2001; Lin et al. 2004; McCabe and Kneafsey 2006]) for investigating means for moving the camera around the scenes with different constraints. While these studies focus on the interaction between scene actors and the cinematic idioms that should be applied in these cases, they give little or no attention to the specific actions of the actors, which is a focus of our work.

Following [Christie and Olivier 2006], our method can be categorized as a hybrid method which fuses both constraint based and optimization based methods over a discrete space of possible camera configurations. This category has the flexibility of easily adding complex constraints, but incurs a large search space. It was shown that a solution to the camera configuration requires an exhaustive search over a vast configuration space [Bares et al. 2000; Halper and Olivier 2000]. Various studies aimed to reduce the size of the search space by adopting hierarchical data structures and fast elimination of irrelevant configurations [Jardillier and Languénou 1998; Benhamou et al. 2004], by using stochastic search methods over the configuration space [Halper and Olivier 2000], or by restricting the solution to only a small set of camera configurations and idioms [Jardillier and Languénou 1998; Halper et al. 2001; Lin et al. 2004]. Our method uses quantum annealing which is proven to be effective over a large number of similar applications. Nevertheless our optimization method can incorporate similar strategies as a bias to the random selection of the quantum annealing technique.

Dynamic scenes and searching for a solution over an entire sequence of frames, while keeping temporal coherence between consecutive frames requires intensive calculations [Christie and Olivier 2006]. Usually to reduce the calculation load, the viewpoint attributes are calculated only on a uniform subset of frames. The remaining frames are then interpolated by continuous quaternion splines [Shoemake 1985; Barr et al. 1992]. In contrast, our work is based on analysis of the motion data and focuses on significant actions for expressing the motion. We show that the careful analysis of the motion capture data proves to be effective and generates better results with significantly less effort than naïve methods.

The work of Halper *et al.* [2000; 2001], suggests the implementation of a camera engine within a game pipeline, for the purpose of generating better viewpoint selection, and game summarization. Their system is based on solving constraints which consider a set of viewpoint quality attributes, as well as the camera control path quality. Here, we extend their work by introducing attributes which relate to the subject's action. As shown next, this addition is sufficient to significantly improve the resulting camera control path for such human animation based games. While their work focuses on generating results in real-time, it lacks the ability to analyze the data and search for global solutions, as our method suggests.

Recently, Kwon and Lee [2008] introduced a camera control technique for character based scenes. Their approach is based on the measuring the motion area, that is, the integrated area spanned by the character bones motion. Their work focus on a selection of static camera positions which are then extended into a camera path by interpolation. Our method is based on a global optimization approach to calculate the required camera movement along all the scene frames, while considering the camera motion constraints. Our global approach allows handing multiple shots conditions, while considering the saliency of the various poses to better illustrate the motion of the significant actions.

One of the main problems in camera path planning is occlusions between objects. This problem can be integrated with ease into such a framework as shown in [Bares et al. 2000; Pickering 2002; Halper

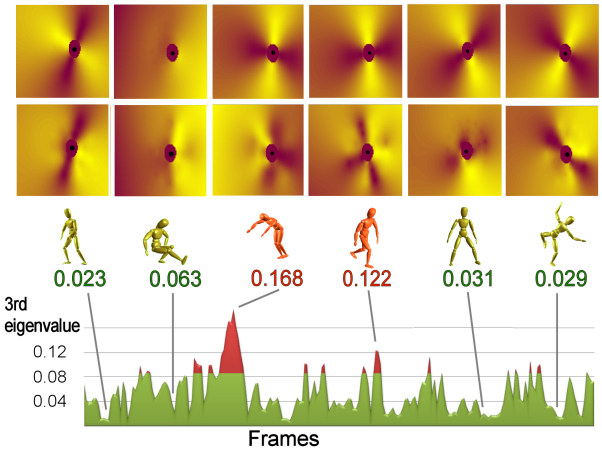


Figure 4: *Comparison of the widest aspect descriptor (top row) and the silhouette size descriptor (second row), for difference poses (third row) along the frames timeline. We show the similarity of the descriptors with a top view of the descriptors quality maps (yellow describes high descriptor value, the character root node is located at black circle). In the bottom row, we show the third eigenvalue function over the sequence of frames. The two red characters exhibit a high third eigenvalue which causes a poor estimation. Note however, that the number of frames with high third eigenvalue (the red regions) is relatively small.*

and Olivier 2000]. In addition to such static scene related occlusions, here we focus on examining self occlusions of the various human limbs. To the best of our knowledge, this extension of standard occlusion constraint methods [Halper and Olivier 2000] has not yet been addressed in the context of camera control.

During the last few years, the study of video and animation summarization has been gaining traction. Such methods, based on the detection of salient segments in the movie, generate shorter clips which summarize the result [Laptev and Lindeberg 2003], or are being used to index the movie for fast browsing [Assa et al. 2005]. In this paper, although we focus on generating an overview of the action rather than a summary, we use a similar technique for boosting the quality and speed of the camera configuration generation. Similar work for selection of key frames for the purpose of summarization is demonstrated using image space features in [DeMenthon et al. 1998] and analyzing motion capture data [Park and Shin 2004; Lee et al. 2002; Assa et al. 2005].

Many algorithms have been proposed to compute the quality of various viewpoints of an object. Early works used simple heuristics such as the "three-quarter view" [Palmer et al. 1981; Blanz et al. 1999], minimizing degenerated projection of polygon [Kamada and Kawai 1988; Gómez et al. 2001], maximizing the visible projected area on the screen [Sokolov and Plemenos 2008], or artistic composition [Gooch et al. 2001]. Later, more advanced methods from scene understanding utilized the entropy functions and selected good viewpoints by maximizing the visible saliency of an object [Vázquez et al. 2003; Page et al. 2003; Sokolov and Plemenos 2008; Lee et al. 2005]. The work of Polonsky *et al.* [2005] categorizes such viewpoint descriptors and examines their effectiveness in determining the viewpoint quality. Here we use some of the descriptors shown to be effective by their work, and evaluate a rough estimate for these descriptors, which requires significantly less effort to compute.

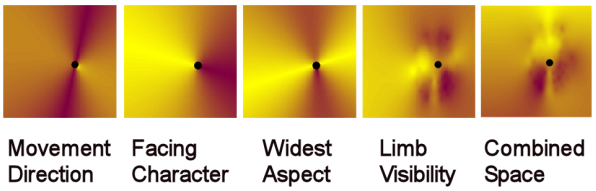


Figure 5: *We measure the viewpoint quality as a combination of generic descriptors and pose specific ones. We visualize each of the resulting descriptors quality maps (yellow describes high descriptor value, the character root node is marked as black circle). The combined viewpoint quality map for that frame is on the right.*

3 Motion analysis and viewpoint metric

Many camera control systems use desired attributes to motivate the camera viewpoint selection. For example, Bares *et al.* [2000] describe four major constraints that affect camera control: subject inclusion, vantage point, shot distance and occlusion avoidance. Halper *et al.* [2001] implement similar conditions as distance to the subject, subject size in the frame and visibility, but also add terms for keeping the angles to the line of motion and to the object front line. Using only these attributes on human motion animation clips does not generate pleasing results, as shown in Figure 2, we therefore add to the set of such general attributes also human motion related attributes.

Based on the work of Polonsky *et al.* [2005], we include a more elaborate descriptor to express the human pose viewpoint quality. In their work, they examined seven types of descriptors, from which several are applicable to our case. They conclude that each of these descriptors can be used in determining the viewpoint quality to a high degree. The descriptors include properties of the visibility, subject silhouette and frame entropy of the resulting image, all of which require rendering the scene from each viewpoint. To achieve a high degree of confidence in the descriptor behavior for different viewpoints, the viewpoint space should be sampled and the scene should be rendered many times for each frame. The overall required computational cost for a clip of frames is therefore too high for any reasonable implementation. We use an estimation for the silhouette size descriptor, which requires significantly lower computational effort following the lines of the work of Gomez *et al.* [2001].

We consider pose joint locations as a point cloud, where each joint location is a point in 3D. The first two eigenvectors of this cloud form a plane which best represents the largest projection of the points. The third eigenvector, being perpendicular to the first two, is perpendicular to the plane with the largest projection of the character. We therefore apply principal components analysis (PCA) and consider the angle between the viewpoint angle and the third eigenvector. We refer to this property as *widest aspect descriptor*. The results of this estimation are proven to be sufficiently similar to the real size of the silhouette, except in the following cases: viewpoints too close to the point cloud and point cloud configurations which have similar values of the second and third eigenvalues. In these cases, there is no preferred projection plane in which the silhouette would be significantly large. These exceptions are acceptable in our case, as we do not expect to place the camera too close to the character, and in cases where the third eigenvalue is significantly high, the actual descriptor preference becomes less significant as the projection viewpoint preferences becomes undetermined. Therefore in these situations the descriptor is not considered. We compare our estimation and the ground truth calculation in Figure 4.

Although the widest aspect attribute improves the resulting metric,

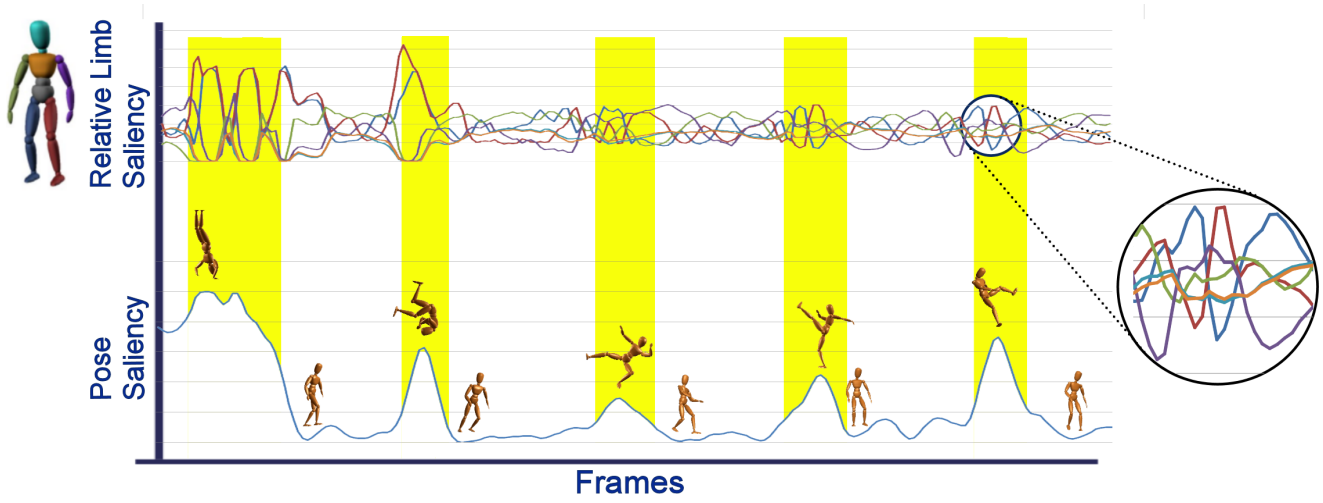


Figure 6: *The generated saliency function for a given clip. The lower graph indicates frame saliency, some of the keyframes are shown with their respective pose. The upper graph illustrates the relative significance of the various body parts (shown in the colored character on the left). The body relative significance of the body parts is considered only during the salient motion segments, shown in yellow.*

its effectiveness deteriorates during fast actions, where the viewpoint should mainly express the participating body parts and limbs. For example, during a boxing punch, special attention should be directed to the punching arm which should be presented in the most informative way. Following the concept of maximizing the viewpoint entropy [Polonsky et al. 2005], we express the information of the participating body parts by selecting viewpoints in which they can be better seen. The previous estimation method cannot be applied to this case, mainly due to occlusions of different body parts, as shown in Figure 2. However, since such fast actions occur in a small number of frames, we can calculate the limb silhouette surface area E_{limb}, referred to as the *limb visibility descriptor*, without requiring huge calculation efforts. We calculate this descriptor by examining the resulting silhouette of each of six main body parts (head, torso, two legs and two arms) from various locations. To speed up the calculation we use character model simplification to a set of ellipsoids, each colored differently. At each render, the number of pixels in each color indicates the visibility of that silhouette. An example of the limb visibility is shown in Figure 7.

The ratio of the widest aspect and the limb visibility descriptors is controlled by the saliency of the motion, and the significance of the limbs. During salient action, the viewpoint should be affected mainly to best represent the significant limbs for the action; otherwise, the widest aspect descriptor on the full body is used.

We therefore locate the segments of high motion saliency by using the method described in [Assa et al. 2005]. We extract two motion aspects of the character: its relative joint location and speed, which generate a high dimensional point for each frame. Then we employ a non-linear multi-dimensional scaling (MDS) to reduce the high dimensional space curve to generate a low dimensional curve, which well describes the attributes of the original motion. Lastly, instead of selecting a set of single frames, we measure the distance of the low dimensional curve to a smooth average curve, and locate the segments in which the distance between the curves is significantly high. This corresponds to the salient segments, as shown in Figure 6. To calculate the significance of the body parts participating in each segment, we apply the same method on the details of the main six body parts and only consider cases where there are large differences between the saliency of the different body parts, as shown in Figure 6.

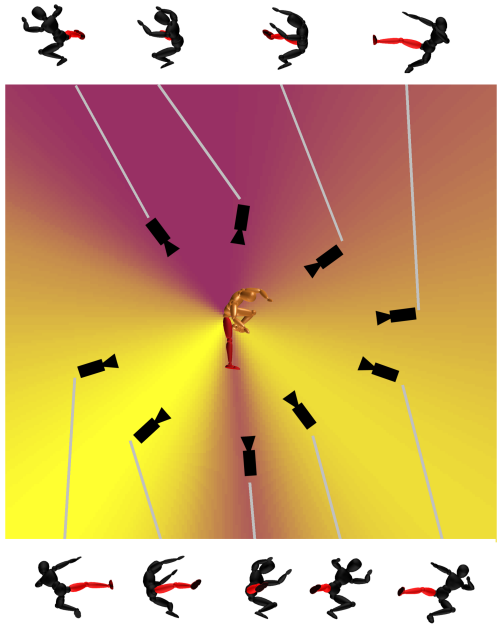


Figure 7: *An example of the limbs visibility descriptor map generated by a right leg kick. Each camera location presents the relative right leg silhouette size.*

We define the viewpoint quality metric V as the linear combination of generic descriptors, and pose saliency S_p controlled limbs E_{limb} and wide aspect E_{wide} descriptors. More formally the resulting viewpoint metric is described in the following equation:

$$V = \sum_{\substack{Generic \\ desc}} E_{desc} + (1 - S_p) \cdot E_{wide} + S_p \cdot \sum_{limbs} S_{limb} \cdot E_{limb},$$

where in our implementation we used E_{desc} as the following standard descriptors described in [Halper et al. 2001] including orienta-

tion descriptors: 3/4 view or facing character descriptor, speed vector perpendicular descriptor, and region descriptors: distance from the subject and static scene object occlusions. The limb visibility descriptor E_{limb} is the normalized visibility (in pixels) of the various limbs. S_p and S_{limb} describe the pose saliency measure and the normalized limb saliency for each frame.

Due to space limitations, we will not discuss here in details the standard descriptors used. In short, the orientation descriptors promote viewpoints with certain angles for the character front, and its main speed direction, and the distance and occlusion descriptors promote the viewpoints which are not occluded and at a certain distance from the character. All of these descriptors can be pre-processed or approximated in real-time [Halper et al. 2001; Drucker and Zeltzer 1994; Christie and Olivier 2006]. We refer the reader to these works for more details about these generic attributes.

The viewpoint potential space, which describes the viewpoint quality for each point in space-time together with the motion saliency S_p are used by the optimization to determine the camera control path in the optimization phase. An example of the descriptors and the resulting V for a given frame are illustrated in Figure 5. Note that although our algorithm calculates a path in 3D space, due to visualization constraints, throughout this work we present the space by showing a top view of a horizontal slice cut at the mid height of the space cube.

4 Camera control optimization

The goal of the optimization is twofold: to generate a camera control path, and to determine frames which can be used to split the current shot. As noted in [Bares and Lester 1999] the separation into shots relaxes some of the optimization constraints, which allows further improving of the camera control path, by applying the optimization again on each shot separately.

The optimization seeks a camera path in space-time 4D space by balancing between two forces: an external force which describes the viewpoint quality at each location and time, and an internal force which enforces smoothness on the generated path. More specifically, the internal forces should include better control over the camera speed, preferring a static camera if possible. In cases where some movement is required, camera speed should be as constant as possible and should not exceed a maximal predefined speed limit. The internal forces should also consider the camera panning speed which should be sufficiently small with minimal acceleration. Note that in our work we introduce motion saliency into the optimized energy function. This allows significant actions to influence the generated path more than other less significant ones. The resulting energy function is described formally in the following equation:

$$\begin{aligned}
&E = E_{internal} + E_{external} \\
&E_{external} = -\sum_t S_p(t) \cdot V(p_t) \\
&E_{internal} = \sum_t c_1 (p_{t-1} - 2p_t + p_{t+1})^2 + \\
&c_2 \left\lfloor \tfrac{p_{t+1}-p_{t-1}}{2\dot{S}_{\max}} \right\rfloor_{>1}^{4} + c_3 (\alpha_{t-1} - 2\alpha_t + \alpha_{t+1})^2 + \\
&c_4 \left\lfloor \tfrac{\alpha_{t+1}-\alpha_{t-1}}{2\dot{\alpha}_{\max}} \right\rfloor_{>1}^{4} - c_5 \left\lfloor \tfrac{p_{t+1}-p_{t-1}}{2\dot{F}_{\max}} \right\rfloor_{<1}^{2},
\end{aligned}$$

where:

$$\lfloor x \rfloor_{condition} = \left\{ \begin{array}{cc} x & \text{condition is true} \\ 0 & \text{otherwise} \end{array} \right\},$$

and p_t,α_t are the camera location and viewing angle to the target in frame t, $\dot{S}_{\max}$, $\dot{\alpha}_{\max}$, $\dot{F}_{\max}$ are the camera maximal speed, angular speed and minimal friction, in all of our examples set to character height/6, 20 degrees, character height/20 correspondingly. $c_{1..5}$ are the coefficients of the various terms, in all of our examples set to 10,10,5,5,2 correspondingly. Note the pose saliency S_p is used both to control the ratio of E_{limb} and E_{wide} within the calculation of V and also here as for considering the frame relative significance.

The internal energy term $E_{internal}$ includes minimization of the acceleration and angular acceleration terms, hard constraints on speed and panning rates, and a friction term which counteracts forces and promotes static camera up to a certain force. These terms introduce functions with $C1$ discontinuities marked with the $\lfloor x \rfloor_{condition}$ operator. The external energy term $E_{external}$ is non-smooth over time as small changes in the character orientation and speed may result in large changes in the viewpoint potential space. Note that unlike a standard snake algorithm which is driven by image-space edges, here there are no coherent edges that can be used. We therefore use a suitable optimization method, which is based on a variant of simulated annealing called *quantum annealing*, usually used for searching discrete spaces with large configuration space, and many local minima [Apolloni et al. 1989].

At each iteration in quantum annealing the selected path is replaced by a randomly selected neighboring path whenever the latter has a lower energy E. The process is controlled by a parameter that determines the extent of the neighborhood explored by the method. The neighborhood size starts high, and is slowly reduced through the computation, until it is below a certain discretization threshold. As an initial solution path we used the degenerated path, a static camera placed in the best global location p:

$$\max_p \sum_t S_p(t) \cdot V(p)$$

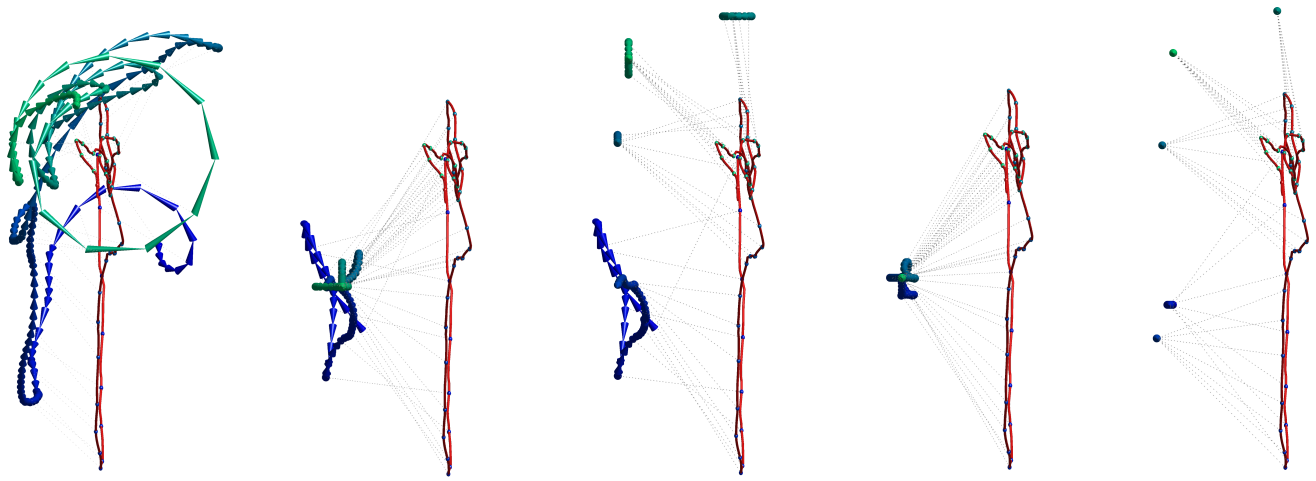
The neighboring path is defined by L1 distance metric between path positions. The resulting path minimizes the energy function shown above and effectively provides a local minima for the camera path.

Scenes with large character movements, yield poor viewpoint locations over many frames. The reasons for these selections are the camera movement constraints which prohibits fast camera movement and rotation, and thus restricts the selection of viewpoints to ones with comparatively low quality. To alleviate this problem, as suggested in [Bares and Lester 1999], our method iteratively splits the camera path into shots which introduce less constraints, and potentially can improve the total energy. After the split, the optimization is repeated separately for each shot.

The relative quality of a given camera location for a given frame $Q(p_t)$ is expressed by the ratio of the selected viewpoint quality versus the best viewpoint quality value for that frame as expressed by $V_t(p_t) / \max_p (V_t)$. The quality of the path is therefore expressed as $\sum_t Q(p_t)$. A low path quality (in our case below 70% of the average saliency) motivates the separation into shots.

The selection of the frame which splits the shots is motivated by two factors: we would like to reduce the likelihood of dividing a significant action into two shots and secondly we prefer frames with low quality viewpoints, in which the split will reduce their internal forces and would allow improvement of their quality score. The shots split frame t_{split} is selected by locating the lowest quality $Q(p_t)$ frame among the sufficiently low saliency (in our case below 80%) frames. We avoid splits which generate short shots to minimize the camera switches as described in [Arijon 1976].

Next we reapply the optimization on the resulting shots separately to improve the overall viewpoint selection quality grade, as shown



(a) (b) (c) (d) (e)

Figure 8: *Top view of camera control paths for Halper et al. and our method. Character center movement is shown in red. The camera path is illustrated using blue to green cones. (a) Halper et al. results. Note the abrupt large camera movements. (b-e) the results of our algorithm, for the same action sequence. (b) shows the camera control path created by a single shot (quality = 66%). (c) the final path after splitting into five shots (quality = 82%) (d,e) same as b,c but with low $\dot{S}_{\max}$ values, which result in a more static camera placement.*

in Figure 9. Using multiple cameras may violate one of the basic camera idioms defined in [Mascelli 1965; Arijon 1976]. Using several shots, the camera locations should not cross the 'line of movement'. Violating the guideline may cause the viewer to feel disoriented, as shown in the accompanying video of this paper. We therefore enforce this restriction to the camera locations in neighboring frames in the two shots by adding a hard 'line of movement' side prior to $E_{internal}$ in frames $t_{split-1}$ and $t_{split+1}$. To select which side of the 'line of movement' to use, we apply the optimization with priors on both sides, and select the solution with the better viewpoint quality. The number of iterations and number of shots required to improve the quality sufficiently up to 80%, is relatively small (in our case up to 6 shots for 1000 frames clip). An example of the resulting quality change by path splitting is shown in Figure 9. An example of the resulting paths with high and low $E_{external}$ is presented in Figure 8.

As a final stage we compute the camera field of view and center of view and render the scene. The camera field of view is calculated separately for each shot by using the largest character silhouette bounding box in every frame in the shot, and setting the lookat vector to point to the character, as shown in the accompanying video.

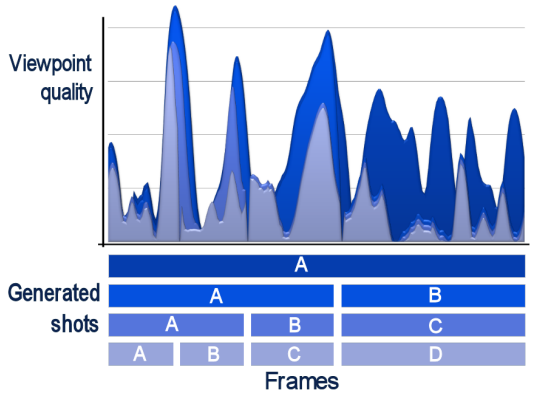
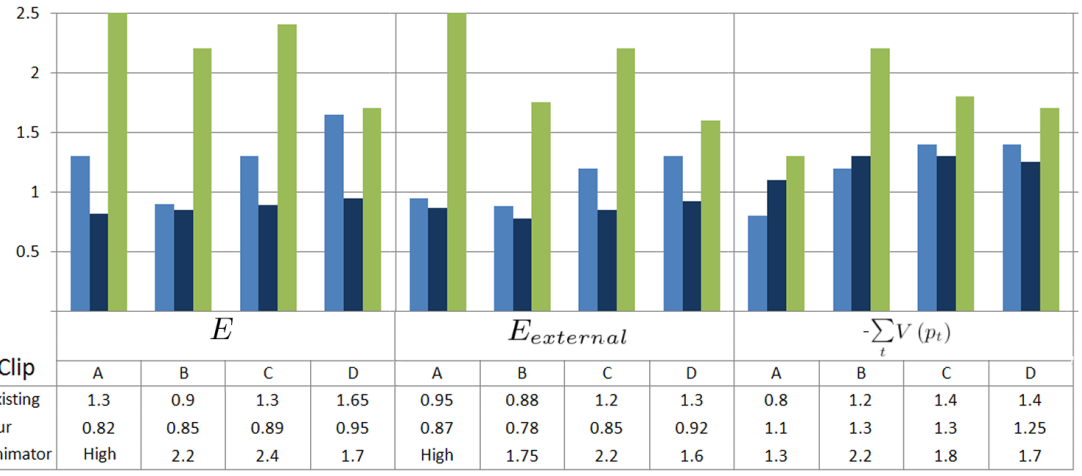


Figure 9: *Shows the quality improvement as a result of the shot splitting. The graph indicates the quality $Q(p_t)$ of each frame. The different blue hue shows different splitting and optimization iterations. Shots segments are presented below.*

5 Results and Discussion

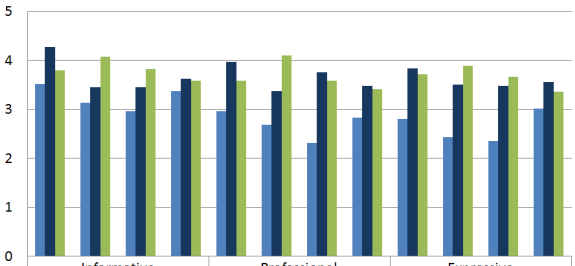
We implemented our method using C# and Matlab code. The current execution time for clips with 700-1200 frames takes up to 50 seconds on a mobile workstation running Intel Pentium M 2.13GHz. Shorter clips of 300 frames are calculated in a few seconds. About 70% of the processing time is required for the calculation of the body parts visibility and establishing the space-time values of E_{limb}, and the rest is for the viewpoint quality metric generation and optimization iterations. The methods description included several coefficients used to scale the various terms. The values used for these coefficients are described in the various sections and remained the same throughout the result clips.

Using the metric defined in Section 3, we measure the generated path quality of our method and compare it to other leading methods such as [Halper et al. 2001], and to a path that was designed by a professional animator, without any restrictions. The comparison was done using 4 different motion capture clips, shown in Figure 12. We compare these different paths using the following attributes: E defines the overall energy term of the path, the viewpoint potential space V is used to determine only the quality of the path locations and $E_{external}$ which considers also the saliency weight for the various frames. To normalize the results in each of the terms, we selected the best attribute score that can be achieved without any camera movement for each clip and checked the performance of the various methods compared to it. The results, shown in Figure 10, express that our method generated better results of E and $E_{external}$ for each of the clips. The V value comparison is inconclusive, mainly due to the fact that it does not consider the resulting path properties (for example the path smoothness).



Clip	E A	E B	E C	E D	$E_{external}$ A	$E_{external}$ B	$E_{external}$ C	$E_{external}$ D	$-\sum_t V(p_t)$ A	$-\sum_t V(p_t)$ B	$-\sum_t V(p_t)$ C	$-\sum_t V(p_t)$ D
Existing	1.3	0.9	1.3	1.65	0.95	0.88	1.2	1.3	0.8	1.2	1.4	1.4
Our	0.82	0.85	0.89	0.95	0.87	0.78	0.85	0.92	1.1	1.3	1.3	1.25
Animator	High	2.2	2.4	1.7	High	1.75	2.2	1.6	1.3	2.2	1.8	1.7

Figure 10: *Results comparison. We compare our work with an example of existing work of Halper et al. and a camera path designed by a professional animator, with the following attributes (left to right): The energy E of the generated camera path. The $E_{external}$ term and on the right the standard (not saliency biased) viewpoint potential metric V (right). We normalize the calculated results with the best static camera energy term for each scene so that in these graphs, lower values indicate better viewpoint selections, according to the different functions.*



	Informative				Professional				Expressive			
Clip	A	B	C	D	A	B	C	D	A	B	C	D
Existing	3.52	3.14	2.97	3.38	2.97	2.69	2.31	2.83	2.80	2.43	2.36	3.02
Our	4.28	3.45	3.45	3.62	3.97	3.38	3.76	3.48	3.84	3.50	3.48	3.55
Animator	3.79	4.07	3.83	3.59	3.59	4.10	3.59	3.41	3.71	3.89	3.66	3.36
Significant difference (p-value< 5%)	E-O E-A	E-O E-A	E-O E-A A-O	E-O	E-O E-A A-O	E-O E-A A-O	E-O E-A	E-O E-A	E-O E-A	E-O E-A A-O	E-O E-A A-O	E-O E-A

Figure 11: *The results of a user study grading the various methods in terms of how informative, professional and expressive they are for the 4 clips. Values vary between 1 to 5, where 5 indicates the best grade. Lower row presents the method grades pairs which have* significant difference *(p-value$< 5\%$).*

The worse metric scores of the professional animator generated scene suggests that the skilled animator did not consider our selected saliency measure, but rather used his personal experience and artistic skills. In our work, we present a computable metric that generates pleasing results, not claiming this metric is necessarily the only possible metric. Our algorithm is biased toward the salient points and therefore its overall viewpoint potential space (V) score may become worse than other techniques. However as presented in the user study, our algorithm achieves better viewpoints in the salient parts, and smooth camera path.

To further strengthen our comparison we introduce a user study which include 30 non-professional computer graphics students. The users were asked to grade attributes of video clips showing the same animation with different camera control methods. This test was blind, with counterbalancing to avoid order effects of the examined results. Users were asked to grade how informative each clip was (how well it describes the presented motion), how professional the camera control looks, and how expressive it is. Each grade ranges in a scale of 1-5 where 5 is the most informative, professional and expressive. The results of the user study (see Figure 11) indicate that our method usually generates satisfactory paths, matching up the informative, professional and expressiveness grades of the camera paths generated by the professional animator. Most of the results show *significant difference* (p-value$< 5\%$), between the existing method and ours (E-O), and the existing method and the animator path (E-A). Some of the results show *almost significant difference* (p-value$< 10\%$) between our method and the animator. For a better impression we recommend the reader to examine the supplementary user study video of this work.

6 Conclusion and Future Work

In this paper we have described our method and have demonstrated that conveying the motion is indeed a hard problem, especially due to the local and global conflicting constraints on the path, view points and camera speed. Our method resolves these conflicts by focusing on the significant poses and limbs. Our method can therefore be applied to automatically render motion capture clips, but can also be used in 3D environments and 3D authoring tools, as an automatic method and semi-automatic tool. The comparative professional animator camera path used in the comparison, for example, took about 30 minutes to complete. The optimization framework of our method can also be extended to additional domains as its properties can be generalized and are not human motion specific.

The introduction of spatial constraints which should be considered during the optimization can be used in scenarios where a novice user can affect the camera location, marking regions which define his viewpoint preferences. These hints can be easily translated into descriptors used by the external energy term. Similar priors can also be used to generate overview clips of motion capture libraries, allowing to comprehend their actions. The presented method demonstrates the following limitations that can be further explored:

1. The current main limitation is its speed. The method is designed for offline camera control. The analysis of the entire clip, its global nature and the current non-optimized realization does not allow real-time applications. The main portion of the processing load is the limb occlusion rendering. A

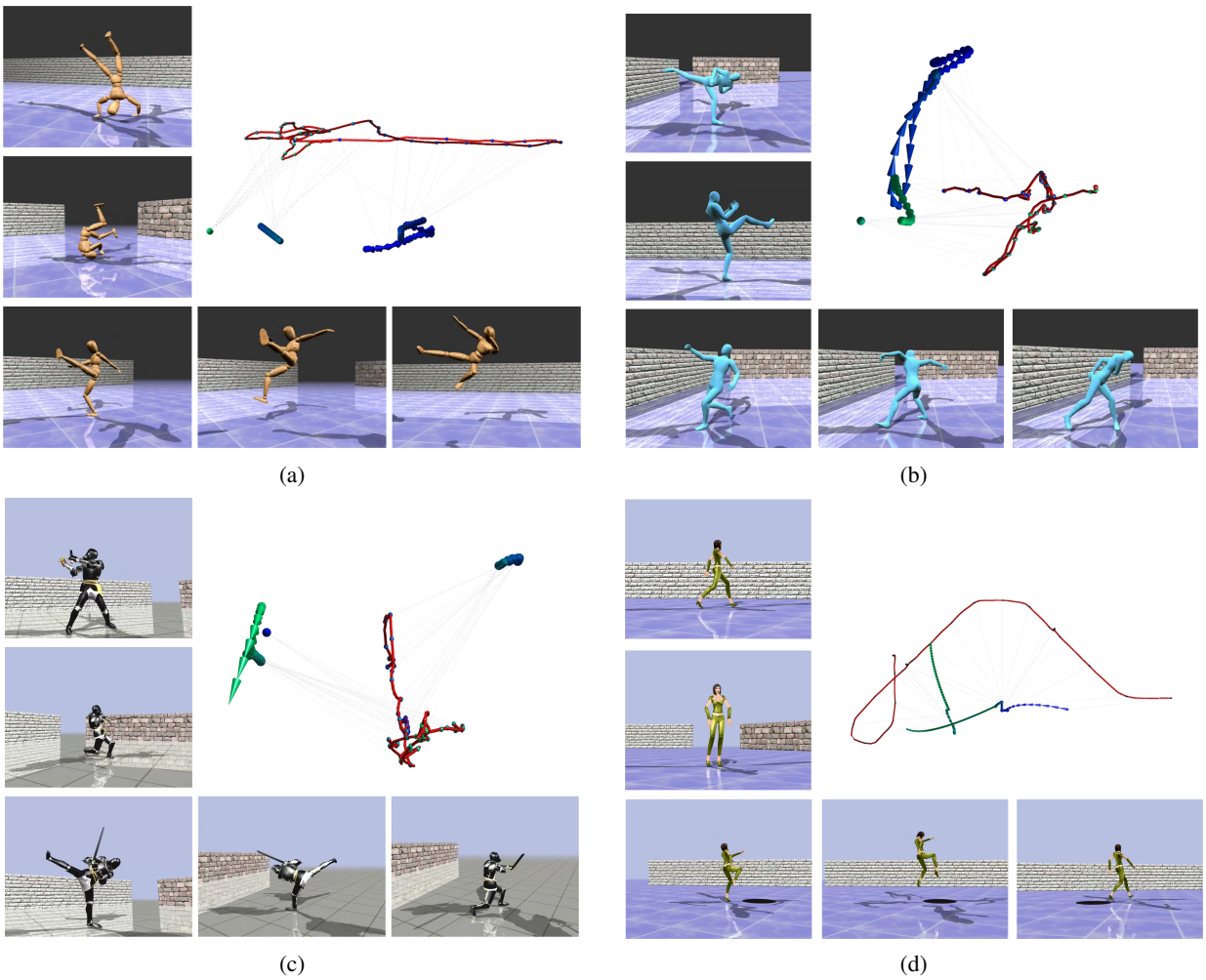


Figure 12: *Some of our results. Due to space limitations, we show a sample set of images taken from path viewpoint values with ratio of 80% and better. The clips correspond to the results table in Figure 10. For better impression of our method and its results, we refer the reader to the supplementary video of this paper, and the user study video.*

faster GPU implementation can accelerate this computation significantly without losing accuracy.

2. Currently, our application uses descriptors which focus on a single character motion. Analyzing the motion of several characters, and locating the best single viewpoint, requires closer consideration of cinematic idioms, and therefore is solved using our method with only varying success. Moreover, the presented method does not consider the significance of props, textures and events such as explosions, for the viewpoint selection, which are essential to the players experience.

3. Adding the effect of cinematic styles, scene mood and emotion, and use of additional information, such as moving the camera to show where the character is looking, can also improve the final result.

4. The global optimization which is based on the annealing technique, may in some cases result in a local minimum for frame sequences which are not sufficiently significant. This may introduce segments, which demonstrate non-optimal viewpoint selections for those frames. Such segments in our results are relatively short and do not include significant motions.

We consider this work to be among the first steps in camera path planning which is affected by the captured motion and the finer details of its actions. We argue that such a camera path is essential to the expressiveness of the generated path. Future work in this direction can extend this work in providing additional attributes that can be translated to potential maps, additional motifs and to techniques for evaluating complex scenes with more than a single character.

Acknowledgments

We thank Mr. Yueh-Han Lin at Department of Animation Design & Game Programming, TOKO University, Taiwan, for his work in preparing the animation scenes. This work is in part supported by the Landmark Program of the NCKU Top University Project under Contract B0008 and the National Science Council, Taiwan under NSC-96-2628-E-006-200-MY3, and by grants from the Israeli Ministry of Science and the Israel Science Foundation.

References

APOLLONI, B., CARVALHO, C., AND DE FALCO, D. 1989. Quantum stochastic optimization. *Stochastic Processes and their Ap-*

plications 33, 2 (Dec), 233–244.

ARIJON, D. 1976. *Grammar of the film language*. Silman-James Press.

ASSA, J., CASPI, Y., AND COHEN-OR, D. 2005. Action synopsis: Pose selection and illustration. In *SIGGRAPH 2005 Conference Proceedings*, ACM, vol. 24, 667–676.

BARES, W. H., AND LESTER, J. C. 1999. Intelligent multi-shot 3d visualization interfaces. *Knowl.-Based Syst. 12*, 8, 403–412.

BARES, W. H., THAINIMIT, S., AND MCDERMOTT, S. 2000. A model for constraint-based camera planning. In *Smart Graphics, Papers from the 2000 AAAI Spring Symposium*, AAAI Press, vol. 4, 84–91.

BARR, A. H., CURRIN, B., GABRIEL, S., AND HUGHES, J. F. 1992. Smooth interpolation of orientations with angular velocity constraints using quaternions. In *SIGGRAPH 1992 Conference Proceedings*, ACM, vol. 26, 313–320.

BENHAMOU, F., GOUALARD, F., LANGUÉNOU, E., AND CHRISTIE, M. 2004. Interval constraint solving for camera control and motion planning. *ACM Transactions on Computational Logic 5*, 4, 732–767.

BLANZ, V., TARR, M. J., AND BÜLTHOFF, H. H. 1999. What object attributes determine canonical views. *Perception 28*, 5, 575–600.

CHRISTIANSON, D. B., ANDERSON, S. E., WEI HE, L., SALESIN, D. H., WELD, D. S., AND COHEN, M. F. 1996. Declarative camera control for automatic cinematography. In *AAAI/IAAI, Vol. 1*, 148–155.

CHRISTIE, M., AND OLIVIER, P. 2006. Camera control in computer graphics. In *Eurographics 2006 Star Report*, 89–113.

CHRISTIE, M., MACHAP, R., NORMAND, J.-M., OLIVIER, P., AND PICKERING, J. 2005. Virtual camera planning: A survey. In *Smart Graphics*, 40–52.

DEMENTHON, D., KOBLA, V., AND DOERMANN, D. 1998. Video summarization by curve simplification. In *MULTIMEDIA '98: Proceedings of the 6th ACM international conference on Multimedia*, ACM, 211–218.

DRUCKER, S. M., AND ZELTZER, D. 1994. Intelligent camera control in a virtual environment. In *Proceedings of Graphics Interface '94*, 190–199.

GLEICHER, M., AND WITKIN, A. 1992. Through-the-lens camera control. In *SIGGRAPH 1992 conference proceedings*, ACM, New York, NY, USA, 331–340.

GÓMEZ, F., HURTADO, F., SELLARES, J. A., AND TOUSSAINT, G. T. 2001. Nice perspective projections. *Journal of Visual Communication and Image Representation 12*, 4, 387–400.

GOOCH, B., REINHARD, E., MOULDING, C., AND SHIRLEY, P. 2001. Artistic composition for image creation. In *Proceedings of the 12th Eurographics Workshop on Rendering Techniques*, Springer-Verlag, London, UK, 83–88.

HALPER, N., AND OLIVIER, P. 2000. Camplan: A camera planning agent. In *AAAI 2000 Spring Symposium on Smart Graphics*, AAAI Press, 92–100.

HALPER, N., HELBING, R., AND STROTHOTTE, T. 2001. A camera engine for computer games: Managing the trade-off between constraint satisfaction and frame coherence. In *EG 2001 Proceedings*, Blackwell Publishing, vol. 20(3), 174–183.

HE, L.-W., COHEN, M. F., AND SALESIN, D. H. 1996. The virtual cinematographer: a paradigm for automatic real-time camera control and directing. In *SIGGRAPH 1996 Conference Proceedings*, ACM, 217–224.

JARDILLIER, F., AND LANGUÉNOU, E. 1998. Screen-space constraints for camera movements: the virtual cameraman. *Computer Graphics Forum 17*, 3, 175–186. ISSN 1067-7055.

KAMADA, T., AND KAWAI, S. 1988. A simple method for computing general position in displaying three-dimensional objects. *Computer Vision, Graphics, and Image Processing 41*, 1, 43–56.

KATZ, S. D. 1991. *Film Directing Shot by Shot: Visualizing from Concept to Screen*. Michael Wiese Productions.

KWON, J.-Y., AND LEE, I.-K. 2008. Detemination of camera parameters for character motions using motion area. *The Visual Computer 24*, 475–483.

LAPTEV, I., AND LINDEBERG, T. 2003. Space-time interest points. In *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*, IEEE Computer Society, Washington, DC, USA, 432.

LEE, J., CHAI, J., REITSMA, P. S. A., HODGINS, J. K., AND POLLARD, N. S. 2002. Interactive control of avatars animated with human motion data. In *SIGGRAPH 2002 Conference Proceedings*, ACM, vol. 21, 491–500.

LEE, C. H., VARSHNEY, A., AND JACOBS, D. W. 2005. Mesh saliency. In *SIGGRAPH 2005 Conference Proceedings*, ACM, vol. 24, 659–666.

LIN, T.-C., SHIH, Z.-C., AND TSAI, Y.-T. 2004. Cinematic camera control in 3d computer games. In *WSCG*, 289–296.

MASCELLI, J. V. 1965. *The Five C's of Cinematography: Motion Picture Filming Techniques.* Cine/Graphic Publications.

MCCABE, H., AND KNEAFSEY, J. 2006. A virtual cinematography system for first person shooter games. In *Proceedings of International Digital Games Conference*, 25–35.

PAGE, D. L., KOSCHAN, A., SUKUMAR, S. R., ROUI-ABIDI, B., AND ABIDI, M. A. 2003. Shape analysis algorithm based on information theory. In *ICIP (1)*, 229–232.

PALMER, S., ROSCH, E., AND CHASE, P. 1981. Canonical perspective and the perception of objects. *Attention and Performance IX*, 135–151.

PARK, M. J., AND SHIN, S. Y. 2004. Example-based motion cloning. *Computer Animation Virtual Worlds 15*, 3-4, 245–257.

PICKERING, J. H. 2002. *Intelligent Camera Planning for Computer Graphics*. PhD thesis, University of York.

POLONSKY, O., PATANÈ, G., BIASOTTI, S., GOTSMAN, C., AND SPAGNUOLO, M. 2005. What's in an image: Towards the computation of the "best" view of an object. *The Visual Computer 21*, 8-10, 840–847.

SHOEMAKE, K. 1985. Animating rotation with quaternion curves. In *SIGGRAPH 1985 Conference Proceedings*, ACM, 245–254.

SOKOLOV, D., AND PLEMENOS, D. 2008. Virtual world explorations by using topological and semantic knowledge. *Vis. Comput. 24*, 3, 173–185.

VÁZQUEZ, P.-P., FEIXAS, M., SBERT, M., AND HEIDRICH, W. 2003. Automatic view selection using viewpoint entropy and its application to image-based modelling. *Computer Graphics Forum 22*, 4, 689–700.