

Animation Sketching: An Approach to Accessible Animation

Tomer Moscovich, John F. Hughes
Department of Computer Science
Box 1910, Brown University
Providence, RI 02912
tm@cs.brown.edu, jfh@cs.brown.edu

ABSTRACT

Sketching rough diagrams is a useful tool for communication and planning. However, static diagrams are often insufficient for expressing motion. While for many applications animation provides an effective depiction of movement and temporal relationships, it is difficult and often tedious to create. We explore the use of motion recording techniques to simplify the animation process. Control of event timing is accomplished by providing additional context cues and through time warping. The approach is then extended to more complex animation using free-form control skeletons, multi-track motion recording and layering.

KEYWORDS: Animation, sketching, accessible animation, coordination.

INTRODUCTION

When people communicate, they frequently use hand-gestures and props to show movement. For example, after a contested boat race, sailors hold protest meetings in which they use plastic boats to illustrate the events that occurred. These techniques are an effective means of communicating motion and temporal data, but they fail when participants are physically separated, and do not scale to more complex motion.

Expressing motion with static diagrams can be difficult. The goal of animation sketching is to make it possible for a non-expert user to quickly create an animation that conveys something meaningful¹. Animated diagrams can aid in communication, planning and education. Animation sketching may also be used for rapid prototyping of animated sequences, or as an alternative to traditional animation.

Even very simple animations can be expressive. In a study described by Heider[3], subjects were shown a film of geometric shapes moving about. When asked to describe what they saw, the subjects referred to the triangles and circles as people, assigning them gender, intention and personalities. One subject describes the film as follows: “A man has planned to meet a girl and the girl comes along with another man. The first man tells the second to go; the second tells the first, and he shakes his head. Then the two men have a fight....”

¹This is distinct from fine-arts animation where an aesthetic effect is sought in conjunction with the expression of an idea.

In this paper we describe several methods we have developed to aid sketching of animation, and the system we have implemented as a test-bed for these methods. The root of our approach lies in shifting from frame-based techniques to one based on time. This technique is motion-by-example. It is a simple form of motion recording. The animator simply moves the object as desired, and the timing and position information are recorded. The motion of the user’s hand can then be played back as the object’s motion producing a simple animation. Motion-by-example has been used in applications such as Kid-Pix[13] as a method of easily producing low-quality animation, but several issues such as object coordination and limitations on complexity have prevented the approach from being used for interesting animation. Our research addresses these issues to increase the utility of motion recording.

BACKGROUND

Traditionally, animators create animation by drawing a series of frames. Playing these frames in rapid succession gives the impression that the objects they depict are moving. Since drawing a large number of frames is both time-consuming and tedious, several techniques have been developed to assist the animator in this task. The most common of these is *key-frame* animation in which the head animator specifies important points of the animation. “In between” frames are then filled by a second animator, or interpolated by a computer[16, 1, 2]. Another approach is taken by *procedural animation techniques*. These methods describe motion using scripts, rules and simulations that are then used to generate animation. These work well for certain applications, but must always be tailored to a specific class of motion. The technique of *motion capture* is the most similar to our approach. In motion capture, positional trackers are attached to the joints of an actor, and their positions through time are recorded. The recorded motion can then be played back on the simulated body of an animated character[15].

RELATED WORK

The idea behind the sketching metaphor is that one is willing to sacrifice some precision and detail for ease of use and shorter completion time. Gesture based 3D modelers built around this metaphor have been highly successful. Sketch [17] can be used to quickly rough out complex scenes, while Teddy [4] allows even novice users to model free-form three dimensional shapes.

An early example of animation sketching is Scott Snibbe's Motion Phone[12, 11]. It allows a pair of users to draw on a running tape loop. Users of Motion Phone can quickly create abstract animations of shape and color.

Pickering's Coach program[9] uses a gesture based interface to interpret and animate user drawn diagrams of football plays. Motion and timing is generated by the program using a simple physical simulation. A related application is Coach's Edge [14], a commercially available system that provides simple motion-recording facilities for animating sports plays. While the software does not directly address synchronization issues it shows that motion-by example is a promising approach to accessible animation.

APPROACH

Motion by Example

The basic element of our approach to animation sketching is motion-by-example. In motion-by-example, the animator simply grabs an object of interest, and moves it about as she likes. The position and timing information is recorded. The motion can then be played back, duplicating the motion of the animator's hand. This is an easy way to create motion. We found that users can produce reasonable motion using a mouse. The use of input devices with a greater dynamic range, such as a large tablet, can improve the quality of the motion.

Motion recording can be extended to control multiple objects using in-context multi-track recording. Just as multi-track recording of music can record each instrument on a separate track, we can record multiple tracks of motion one at a time. For example, in an animation of balls being juggled, the animator begins by recording the path of the first ball. She then rewinds the animation to the point when the next ball is to be thrown, and records the path of the next ball. The motion of the previous ball is played back during recording, providing context that lets the animator know where she is in the animation and decide how the object should be moving at that point. The animator can record the motion in short sections or longer paths, and is free to choose how to break up the motion to make recording it easier.

Motion Coordination

One problem the user runs into when using motion recording is that of coordination. In frame-based techniques, the animator can cause things to occur simultaneously, or in some other temporal relationship such as cause-and-effect simply by drawing them in the appropriate frame. In motion recording, the animator must rely on his sense of timing and motor control ability. For example, in Figure 1 the animator attempts to have the boat on the right turn at the same time as the one on the left². However, by the time he sees the left boat turn it is too late to start turning the right one. We help the animator succeed in such tasks by providing additional context cues, and through the use of time-warping tools.

Providing a temporal context

One cause of the animator's quandary is the difficulty of visualizing time. The information available in the present is not

²An early driving application of this project was for illustrating the events of a boat-race during a followup protest meeting.

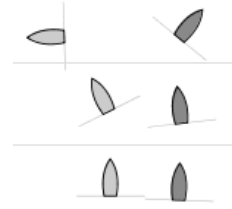


Figure 1: Without a cue, it is difficult to synchronize the boats' turn.

always enough for the animator to tell when a future event will occur. We have explored two approaches that provide the animator with additional context regarding timing and sequence.

The first approach borrows from the work of E. J. Marey who used multiple exposure photography to visualize motion through time[7, 10]. Preview ghosts, shown in Figure 2, reveal where the boat will be within the next second. They function as a cue that alerts the animator to near future events. This allows the user to plan ahead and respond appropriately. With a bit of practice users can learn to use the ghosts to help them synchronize events, but results are often unsatisfactory. Not only are the animated ghost images distracting, but they also cause a divided attention problem[5]. The user pays more attention to the ghosts than to the object he is animating, making it difficult to draw the desired motion.



Figure 2: Precognition: Preview-ghosts provide the animator with a cue before important events.

Our next approach is to simply display the path each object follows. The paths work much like the preview ghosts, as the user can see when objects approach interesting events, while getting a good sense of the timing of the animation. Unlike ghosts, paths are static and do not require constant attention. The static nature of motion paths is also much less distracting. Since events that are close to the present are more important for establishing timing, increasing the length of the curves does not improve their utility and may make them confusing. We therefore restrict the paths to a user defined time interval. This has the added benefit of creating a time-window cue that helps the user understand relative speed. To avoid the distraction caused by animating the paths in this way the system gradually fades them in and out.

Event Synchronization

The temporal context discussed above is often all the animator needs to make the timing look reasonable. However, the timing for certain events can be a bit tricky to get right. For example, for a character walking through a door, or two objects colliding, approximate timing may still not look right. It

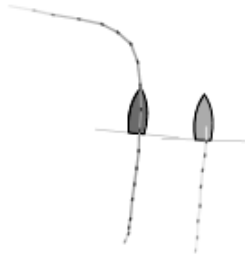


Figure 3: Paths serve as additional context without being distracting.

would not do for the door to open just *after* someone walks through it. To improve precision we transfer the load from the user to the computer by providing an event synchronization facility.

To synchronize two events, the animator simply picks a point on each object's path that should be reached at the same time as the other. The system warps each object's time line to align the events in time. This is done as follows: Time is reparameterized for both objects so that each will reach the specified point at the same time t_* . To reduce distortion, t_* is chosen half-way between the two events. If the time to be warped is in the interval $(0, 1)$ and t_o represent the point specified for one object, the reparameterization is:

$$f(t) = \begin{cases} \frac{t_*}{t_o} \cdot t & 0 < t < t_o \\ 1 + (t - 1) \frac{t_* - 1}{t_o - 1} & t_o < t < 1 \end{cases}$$

This is shown in Figure 4. The two events occur at different points on the original time line. The reparameterized time is shown in the lower part of the figure – both events occur simultaneously at t' .

Time-warping works well when the events to be synchronized are fairly close. Synchronizing events that are distant in time can cause artifacts near the synchronization point. Using a smoother reparameterization function may help a bit, but the character of the animation will still be damaged. For example, warping the path of a bouncing ball can cause the ball to move more slowly on the way down than at the peak of its bounce.

Free-form Skeleton controls

Motion-by-example has previously been used to create complex animations in the guise of motion-capture. However, by recording each track of motion separately, we can control compound motion without an expensive motion-capture studio. We are also not limited by the physical constraints and structure of an actor. Moreover, we can use motion recording to overlay tracks of motion in order to modify previously recorded movement, or to add detail.

The animator begins by creating a free-form skeleton to which the form to be animated is attached. Figure 5 shows the skeleton for a simple humanoid. The animator then

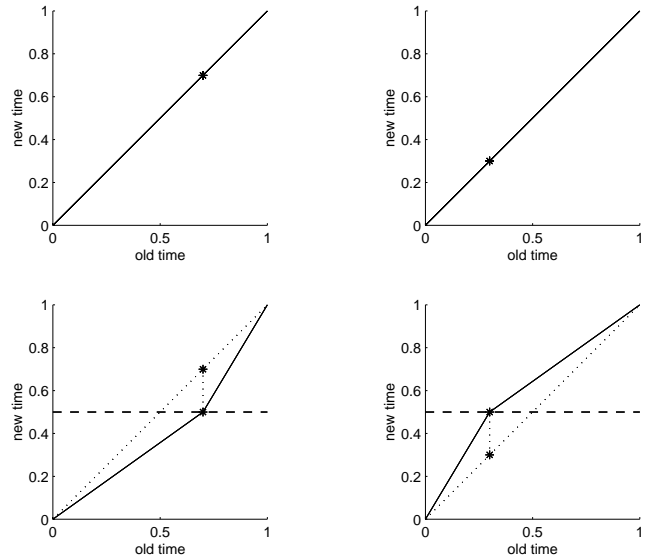


Figure 4: Synchronizing events through time-warping.

records the motion of the figure by tugging on the control points.

Simple animations such as the walking character shown in Figure 5, can be created quickly and easily. However, the control-points in this example are fairly independent of each other, making it easy for the animator to specify the motion of each one separately. When the moving parts of a character are interrelated it becomes exceedingly difficult for the animator to coordinate them. For example, to animate the bird shown in Figure 6 the animator has to move all three points along the same path, while moving the wings up and down relative to that path. Most people find this task nearly impossible. We overcome this problem by allowing the animator to overlay motion, allowing him to record its components separately. In the flapping bird, the animator first records the gross flight motion by selecting the entire control structure and moving it along the flight path. He then selects the points controlling the wings, and records the flapping motion by moving the mouse up and down. The flapping motion is added to the flight motion to produce the final animation. This method can be used to animate complex characters that would otherwise be very difficult to control. Figure 7 illustrates such a character. Overlaying tracks can also be used to

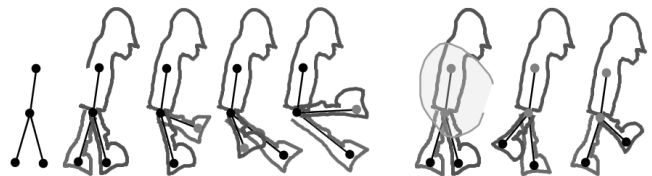


Figure 5: The above series depicts an animator creating a simple walking figure. He begins by creating the skeleton and then fleshes out the figure. The animator then drags the feet to make the character walk. Next, the animator selects the control structure of the upper-body and records its motion.

edit motion. For example, the walking character in Figure 5 is a bit stiff. It could easily be made to sway as it walks by adding a slight side to side motion to its head or waist.

The two key components of this approach are the ability to select multiple points and motion overlay. Multi-point selection allows for rigid motion and for the application of the same movement to several parts of a model. We are also experimenting with partial-selection and warping techniques for controlling multiple points. Motion can be overlaid in many ways. For example, a hierarchical technique may be useful for jointed characters. However, simply adding motion together is not only a less constrained method, but also allows the user to freely switch between and create new reference frames as the need arises.

EVALUATION

Our initial driving application was sketching boat races. However, discussion with experts in the field led us to believe that animation sketching is not as applicable as we expected, as they wanted precise control of boat position and were not so interested in timing. A more successful application is basketball diagramming. The computer science department basketball team has used our system to plan and illustrate plays. The players were unsatisfied with their previous method which involved drawing tangled lines on a white-board. These drawings were confusing and contained no timing information. The players said that animation sketching is “exactly what we need,” as it provided them with a quick and easy way to illustrate the sequence and timing of a basketball play.

To compare animation sketching to traditional animation, we consulted professional animator and animation professor Tammy Dudman, and asked her to review our system. Dudman commented that animation sketching provides animators with a different aesthetic choice than is currently available. One example she cited is the “Flash look” that animations authored with Macromedia’s Flash tool exhibit[6]. This look is an artifact of the software, and not necessarily an intentional choice of the artist. Dudman mentioned that animation sketching may also better preserve the animator’s personal timing. However, she would not use an animation sketching tool to teach animation. Learning to translate motion and time into frames is a difficult skill to learn, and students may be less enthusiastic about learning it when presented with an easier alternative. Another feature Professor Dudman said she appreciated was the use of a skeletal structure to control characters. This technique is common in 3D animation programs, but absent from popular 2D animation software.

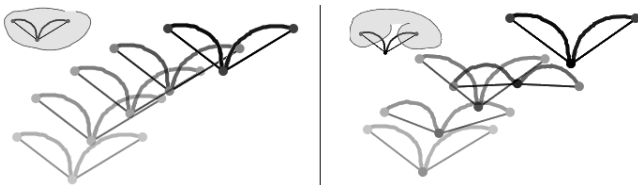


Figure 6: After recording the flight path, the user overlays a flapping motion by moving the wing controls.

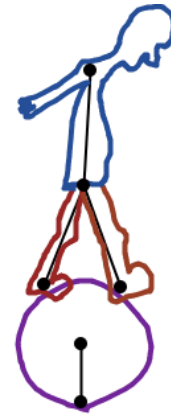


Figure 7: Without motion layering, it would be difficult to roll the ball and control the character while the whole assembly is moving forward.

FUTURE WORK

Our work takes several steps towards an interface for general animation sketching. However, it leaves many open questions. Our solution for object coordination is still incomplete. How does one synchronize several objects, or several events for the same pair of objects? How do we coordinate the layered motion tracks to ensure, for example, that a bird’s flight is correlated with the flapping of its wings? Controlling numerous control points can also be difficult. The addition of inverse-kinematics and other constraints may ease the task.

We need to extend the domain of things we can animate beyond gross motion and skeletal structures in order to enhance the expressiveness of the system. One path towards accomplishing this is the integration of animation sketching with existing methods. For example, one could use motion recording to drive the movement of characters, while using key-frame techniques for controlling their shape. It may also be convenient to refine animation-sketches using traditional methods. How would this be done? Is it possible to automatically extract key-frames from an animation in a manner advantageous to the user?

DISCUSSION

Animation sketching allows an inexperienced animator to quickly create simple animations. It is important to make a distinction between an animation created by a professional animator that attempts to evoke a certain response in the viewer, and an animation that serves as a diagram or communication aid. One key to creating a convincing animation is understanding the motion one is trying to convey. Professional animators spend much time studying how things move, acting out the movements of their characters, and understanding concepts (such as anticipation and squash-and-stretch) that help express motion. These skills are equally important in animation sketching as they are in traditional techniques, however, they are not as essential for explanatory animation. Traditional animation techniques are well established, and it is still too early to judge how motion recording may be incorporated into the animator’s tool-box. Accessible animation for the lay user, however, is a largely unexplored

field [8]. Explanatory animations do not always warrant the effort required by traditional methods, the availability of simpler techniques may make them more common.

ACKNOWLEDGMENTS

We thank everyone who has provided helpful comments and ideas on this project. In particular David Laidlaw, Nancy Pollard, Robert Zeleznik, and Joseph LaViola. Daniel Keefe used the system to animate basketball plays and provided useful feedback. Tammy Dudman presented us with the perspective of a professional animator.

REFERENCES

1. N. Burtnyk and M. Wein. Computer generated keyframe animation. *Journal of the Society of Motion Picture and Television Engineers*, pages 149–153, March 1971.
2. N. Burtnyk and M. Wein. Interactive skeleton techniques for enhancing motion dynamics in key frame animation. *Communications ACM* 19(10), 1976.
3. F. Heider and M. Simmel. An experimental study of apparent behavior. *American Journal of Psychology*, 13, 1944.
4. Takeo Igarashi, Satoshi Matsuoka, and Hidehiko Tanaka. Teddy: a sketching interface for 3D freeform design. *Computer Graphics*, 33(Annual Conference Series):409–416, 1999.
5. P. Kabbash, W. Buxton, and A. Sellen. Two-handed input in a compound task. *Proceedings of Human Factors in Computer Systems*, pages 417–423, 1994.
6. Macromedia. Flash. <http://www.flash.com>.
7. Etienne-Jules Marey. *Movement*. New York, D. Appleton and Company, 1895.
8. Tom Ngo, Doug Cutrell, Jenny Dana, Bruce Donald, Lorie Loeb, and Shunhui Zhu. Accessible animation and customizable graphics via simplicial configuration modeling. *Proceedings of SIGGRAPH 2000*, pages 403–410, July 2000. ISBN 1-58113-208-5.
9. J. Pickering, D. Bhuphaibool, J. LaViola, and N. Pollard. The coach’s playbook. Technical report, Brown University, Department of Computer Science, 1999.
10. S. Snibbe. A direct manipulation interface for 3D computer animation. *Computer Graphics Forum*, 14(3):271–284, 1995.
11. S. Snibbe and G. Levin. Interactive dynamic abstraction. *Proceedings of the Symposium on Nonphotorealistic Animation and Rendering*, June 2000.
12. Scott Sona Snibbe. Motion phone. <http://www.snibbe.com/scott/mphone/>.
13. Brøderbund Software. Kid-Pix, 1991. www.kidpix.com.
14. Coach’s Edge Software. Coach’s edge. <http://www.coachsedge.com>.
15. B. Bodenheimer *et. al.* The process of motion capture: Dealing with the data. In *Computer Animation and Simulation ’97*, pages 3–18, September 1997.
16. F. Thomas and O. Johnson. *Disney Animation: The Illusion of Life*. New York, Abbeville, 1984.
17. Robert C. Zeleznik, Kenneth P. Herndon, and John F. Hughes. SKETCH: an interface for sketching 3D scenes. *Computer Graphics*, 30(Annual Conference Series):163–170, 1996.