Graphics and Image Processing

Interactive Skeleton Techniques for Enhancing Motion Dynamics in Key Frame Animation

N. Burtnyk and M. Wein National Research Council of Canada

A significant increase in the capability for controlling motion dynamics in key frame animation is achieved through skeleton control. This technique allows an animator to develop a complex motion sequence by animating a stick figure representation of an image. This control sequence is then used to drive an image sequence through the same movement. The simplicity of the stick figure image encourages a high level of interaction during the design stage. Its compatibility with the basic key frame animation technique permits skeleton control to be applied selectively to only those components of a composite image sequence that require enhancement.

Key Words and Phrases: interactive graphics, computer generated animation, key frame animation, interactive skeleton, skeleton control, stick figure animation

CR Categories: 3.41, 3.49, 4.9, 8.2

Author's address: National Research Council of Canada, Division of Electrical Engineering, Ottawa, Canada K1A OR8.

¹ Cel has been derived from celluloids, the material on which drawings are prepared in conventional cel animation. Component images that move separately are usually drawn on separate cels and stacked into a cel sandwich for filming.

Introduction

Previous work has demonstrated that key frame animation techniques constitute a successful approach to animation of free-form images [1–3]. Using this technique, the artist draws key images at selected intervals in an animation sequence and the playback program computes the in-between images by interpolation. Interpolation between related key images allows the animation of change of shape or distortion. It permits a direct and intuitive method for specifying the action, whereas mathematically defined distortion requires trial and error experimentation. One strength of key frame animation techniques is the analogy to conventional hand animation techniques, simplifying the transition when a classically trained animator adapts to using computers.

Figure 1 illustrates a typical image sequence generated using key frame animation. The first and last images were drawn by the artist, while the six intermediate images were selected from the 240 frames in the actual film sequence.

The animation package is implemented on a minicomputer based interactive graphics system. This package supports the four major phases of a production: (1) the drawing phase, (2) the assembly of drawings into key frame sequences, (3) the preview/modification phase, and (4) the final processing and recording of the sequences on film.

The drawing phase is carried out in two stages. The first stage is off-line at the drawing board. Analysis of the action depicted by the story board establishes key positions from which drawings are prepared. The second stage involves tracing these drawings on a graphic tablet at the display console. During this stage, the order in which strokes are traced to describe an image is important. Since the interpolation process is based on stroke to stroke mapping, this ordering of strokes between related images controls the form of the intermediate image.

The second phase consists of the interactive assembly of individual drawings or cels¹ into key frames, including a specification of the interpolation law for each cel and a key to key time interval. Concatenated key frames form a sequence. This process is repeated for all concurrent sequences that make up a composite sequence.

During the preview phase, playback of any individual sequence or concurrent sequences on the interactive display permits an assessment of the resulting animation. Modification involves returning to the interactive assembly phase to edit the sequences. In practice, direct playback assures only that the form of the interpolated images can be assessed, since it is difficult to achieve playback at the cine rate with complex images. Proper assessment of motion and timing requires further conversion to a raster format which maintains display at the cine rate independent

Communications of the ACM

Copyright © 1976, Association for Computing Machinery, Inc. General permission to republish, but not for profit, all or part of this material is granted provided that ACM's copyright notice is given and that reference is made to the publication, to its date of issue, and to the fact that reprinting privileges were granted by permission of the Association for Computing Machinery.

A version of this paper was presented at SIGGRAPH '76: The Third Annual Conference on Computer Graphics, Interactive Techniques, and Image Processing, The Wharton School, University of Pennsylvania, July 14-16, 1976.

of the image content.

This technique of animating free-form images has been directed mainly towards drawn images and hence two-dimensional. The image material, of course, attempts to represent a 3-D space much as in conventional cel animation. The basic capability includes a simplified solution to the problem of hidden surfaces by treating the image as a hierarchy of parallel planes. The simplification lies in the fact that the animatorspecified order of planes establishes the order of visibility computation, thus eliminating any programmed sorting of data by depth. The composite playback facility produces separately a composite line image with hidden lines removed and a composite surface sequence.

Consideration of Motion Dynamics

The greatest shortcoming in key frame animation results from incomplete control of motion dynamics, both in complexity and in smoothness or continuity. It is relatively simple to have good control over the dynamics in time. The amount of change from one frame to the next is determined by a weighting factor which is a single-valued function of time. Thus one can easily compute, or store precomputed, various functions representing different "tapers." However, the same value of weighting function is applied to an entire picture component. There is no "spatial weighting."

The shortcomings manifest themselves in the following ways: (a) the motion of each point in the image is along a straight line and the relative change from one frame to the next is the same for all points belonging to one picture element, and (b) there is a discontinuity at key frames in both the amount of frame to frame change and in the direction of apparent motion. Therefore it is difficult to synthesize smooth continuous motion spanning several key positions. There is a dilemma in that smoothness is achieved by having as few key images as possible (and therefore widely spaced in time), while close control requires many closely spaced keys. In addition, a large number of closely spaced drawings negates much of the economic advantage of using computers.

Various techniques have been examined for overcoming these problems. One technique provides an ability to include a rotational component as part of the image change, which in effect superimposes rotation on the interpolation process [2]. This permits some variation in the spatial dynamics but its application is limited and discontinuities at key frames remain.

Synthesis of complex motion could be achieved by using additional intermediate keys, but preparation of additional drawings by the animator is uneconomical. Skeleton techniques were developed to derive variations of existing key drawings to be used as intermediate keys [2]. This involves representation of a drawing by a simple skeleton and then extracting a distorted form of the image by modifying only the skeleton. Even when such additional keys are used, discontinuity in motion is difficult to avoid.

Another technique involves the use of smooth drawn paths to control the interpolation process. Motion along a path, as a method distinct from key frame animation, has been used extensively in computer animation [4-7]. A single path to control interpolation between key images offers a limited solution somewhat equivalent to the use of rotation—it tends to be satisfactory only when the distortion of the image is minimal. For a distorting image, different portions must follow entirely different paths. This immediately leads to a problem if several paths are to be drawn for different portions of the image. It is difficult to establish points of simultaneity on several paths such that one could easily perceive the shape of the image at any instant.

An examination of the methods used in conventional animation has led to a solution to this problem.

Fig. 1. Selected frames from a key frame animation sequence. The first and last images are drawn, the intermediate images are interpolated. Multilayer visibility is included in computing the composite image. From "Visage," a film by Peter Foldes



Communications of the ACM

To visualize a complex movement, the animator often sketches stick figure representations at equal-time intervals between key positions. He may use smooth curves through related skeletal points as a further guide. This set of stick figures achieves both objectives: the frame to frame spacing conveys the rate of movement and the shape of each skeleton represents the shape of the object at that instant. Thus the problem reduces to animating a stick figure representation of the image which will in turn impart the movement to the actual image sequence.

The system described in this paper incorporates the use of skeletons into the key frame technique to provide overall control in the playback process. As in the basic key frame animation system, the process of producing a sequence involves two steps. The first is the interactive stage at which the animator prepares the key images and establishes the stick figure representations at as many intermediate positions as desired. The intermediate skeletons define intermediate control keys. During playback the program selects those image components that are skeleton driven and applies the necessary deformation.

There are two significant aspects of the skeleton driven technique. First, the skeletons are simple images composed of only a few points, so that it is possible to provide a high level of interaction. The second aspect of this technique is its compatibility with basic key frame animation. Skeleton sequences are prepared only where necessary and the playback system identifies those image components that are skeleton driven and those that are not.

It should be noted that the concept of skeletons used in the context of this paper differs from that used by Blum [8]. Blum's skeletons are used for image representation in a compressed form and are derived automatically from the coordinate data. Our skeleton representation of an image provides a definition of some coordinate space within which the image, described in relative coordinates, is distributed.

Skeleton Coordinate System

The nature of the coordinate space that is used to define relative skeleton coordinates may be thought of as a network of polygons that form a mesh (Figure 2).



Each polygon has a relative coordinate range of 0 to 1.0 along each axis. Now the nodes in this mesh may be displaced relative to one another to change its geometry. However, because the relative coordinate system within each polygon is based on its geometry, coordinate values remain continuous across common edges between adjacent polygons. Thus any image whose coordinates are defined within this system will take on the overall distortion exhibited by the coordinate space (Figure 3).



Geometric transformation of contours from one coordinate space to another is, of course, well known in conformal mapping.

The notion of skeleton control implies a central core of connected "bones" with a surrounding image distribution. In order to restrict the transverse distance away from the core over which skeleton control will be active, delimiting boundaries must be specified. Consequently, the practical form of skeleton coordinate space spans two units in width, but may extend in length as desired (Figure 4).



For convenience, the central core always represents the L axis, which is also the W = 0 coordinate reference; the delimiting boundary which is specified first is the positive or W = 1.0 boundary, the other is the W = -1.0 boundary. The L coordinate range starts at L = 0

Communications of the ACM

and is incremented by one for each node on the central core. If desired, the L coordinate space may be separated at any coordinate boundary by providing a redefinition of that coordinate boundary before continuing the coordinate space. Of course, the related image will not normally continue through such a separation. In addition, ambiguities can occur if separated coordinate spaces overlap. In general, it is preferable to treat these as separate skeletons so that no restrictions are imposed. On the other hand, any given image need not fall entirely within the coordinate space of a skeleton. Those points which lie outside the skeleton space will remain unaffected by the distortion of the skeleton coordinate space.

Relative coordinates, denoted by (l, w), may be defined as the fractional distance along each axis which is occupied by a line passing through the point while intersecting the two opposing edges of the polygon at this fractional distance. In Figure 5(a), the coordinates of point P are (0.75, 0.5) by this definition. In order to minimize the computation involved in coordinate conversion, the simpler definition of Figure 5(b) is used.



Given the absolute vertex coordinates of the polygon and an image point P, the fractional distance of points P_5 and P_6 along lines P_1P_2 and P_3P_4 is expressed by l, giving

$$x_5 = l(x_2 - x_1) + x_1, \quad y_5 = l(y_2 - y_1) + y_1,$$

$$x_6 = l(x_4 - x_3) + x_3, \quad y_6 = l(y_4 - y_3) + y_3,$$

These expressions are substituted into $(x - x_6)/(y - y_6) = (x_5 - x)/(y_5 - y)$, the equation of the line P_5P_6 passing through point P, giving

$$[x - x_3 - l(x_4 - x_3)]/[y - y_3 - l(y_4 - y_3)]$$

=
$$[x - x_1 - l(x_2 - x_1)]/[y - y_1 - l(y_2 - y_1)].$$

This reduces to

$$(BH - DF)l^{2} + (CF - DE - AH - BG)l$$

where

 $\begin{array}{ll} A &= x - x_1, & E = x - x_3, \\ B &= x_2 - x_1, & F = x_4 - x_3, \\ C &= y - y_1, & G = y - y_3, \\ D &= y_2 - y_1, & H = y_4 - y_3, \end{array}$

The desired root for l has a value between 0 and 1,

the other root will be negative or greater than 1. The w-coordinate, expressing the fractional distance of point P along line P_5P_6 , is given by

$$w = (x - x_5)/(x_6 - x_5) = (A - Bl)/[A - E - (B - F)l].$$

To convert relative coordinates back to display coordinates, the *l*-coordinate is applied to the vertex coordinates to determine the coordinates of P_5 and P_6 from which P is found.

The effect of skeleton control is to take any specified area of the display plane and distort it into another area of the display plane as if it were made up of rubber sheet patches. In that sense, it is similar to the distorted raster scan technique used in Caesar [9], and equivalent to the mapping of images into curved surfaces described by Catmull [10]. While the skeleton coordinate space is distorting, however, the relative coordinates of the image itself may be undergoing a change. Relative coordinates may be treated in the same way as absolute coordinates, as if the reference coordinate space was always uniform and orthogonal, its particular shape being important only for display purposes (Figure 6). Therefore the key frame interpolation process may still be carried out even if key images are represented in relative coordinates. It is this compatibility with key frame animation that makes the skeleton control technique so powerful and attractive. No other practical techniques have been developed that offer a comparable degree of image control in computer generated animation.

Implementation within Key Frame Animation

The benefits that may be derived in practice from skeleton control are closely related to the method of implementation. While there is little doubt that any capability for enriching motion is useful, it is equally clear that the compulsory use of skeleton control for all parts of an animation sequence would be a great hindrance. The full advantage in the use of this technique is realized only if the animator can apply it selectively. In fact, it is most attractive if it can be used to improve motion dynamics of sequences which were previously created. This is the form in which it has been implemented in our system.

Component sequences are first assembled in the usual form and displayed as a composite image sequence for previewing. If, after assessment, the animator wants to modify or improve parts of it, he does so by adding skeleton control to those components only. This is accomplished by attaching a reference skeleton, which he has drawn, to each image which will be controlled. These image components, which are referenced to skeletons, are converted to relative coordinates and tagged during assembly of the sequence.

Communications	October 1976
of	Volume 19
the ACM	Number 10

Now the composite sequence is regenerated in this modified form. During playback, all coordinate data pass through the interpolation process, but those identified as being relative must be mapped to a particular skeleton reference for display. A skeleton reference defines a display space in absolute coordinates. These skeleton coordinate references for each frame are provided by assembling stick figure control sequences which are also played back as part of the composite sequence.

The design of the skeleton control sequence itself is developed interactively in a separate package. Stick figure representations of key images provide the starting point (Figure 7(a)). Two such skeletons are used to define a start and end frame. When the INBETWEEN display mode is active, intermediate frames are interpolated and presented on the screen as a superposition of many frames (Figure 7(b)). For convenience, the delimiting boundaries about the skeleton core are eliminated from display to prevent excessive clutter. Any frame may now be selected and modified using tablet interaction. In this mode, the modified coordinates of the selected frame are stored as a control frame within the sequence. The interpolated intermediate frames adjust accordingly in response to tablet interaction (Figure 7(c)). Additional frames may be modified in a similar manner (Figure 7(d)).

Frame to frame change is easily related to the spacing between stick figures. This interaction continues, giving the animator control over motion dynamics down to the frame level as in conventional animation, if desired. The user-modified control frames are preserved, whereas all other intermediate stick figures are recomputed when needed. Display of the final sequence of control frames is shown in Figure 7(e). The skeleton boundaries have been adjusted where required through similar tablet interaction to maintain their desired form. Alternatively skeleton keys which have been drawn in these desired positions may be brought in from the picture library and assembled as control frames in the same way.

In practice, the number of control frames that are used to generate a motion sequence will be kept to a minimum. Because of this, simple linear interpolation between specified control frames may not adequately reproduce a smooth continuous movement. This result is illustrated in Figure 7(f), where the dynamics of the movement suffer from excessive discontinuities in rate at two of the control frames. Additional intermediate frames have been interpolated and plotted for clarity.

This deficiency is removed if a smoothing function is applied during computation of intermediate frames (Figure 7(g)). This process maintains continuity of movement of corresponding points through successive control frames. The parametric method of curve fitting is adapted from the work of Akima [11]. Not only does it produce a smooth path for each point, but progression Fig. 6. (a), (c) two drawn images in absolute coordinates; (b), (d) same images with reference skeletons; (e), (f) the relative coordinates presented on an orthogonal coordinate system.



Fig. 7. Development of a skeleton control sequence: (a) start and end frames; (b) interpolated inbetweens presented for interaction; (c), (d) frames 5 and 9 modified in turn by the animator (control frames); (e) final sequence of control frames, shown with boundaries; (f) linearly interpolated inbetweens; (g), (h) with curve smoothing; (i), (j) images driven by the motion sequence.



Communications of the ACM

along each path is tapered to accommodate changes in rate in a complex movement. Successive positions of several points have been joined in Figure 7(h) to emphasize the effect. This process has all the characteristics of drawing smooth paths to control interpolation between images without any of its limitations.

Now the skeleton control information is complete for driving the original image sequence through the desired movement (Figure 7(i)). It may equally well be applied to drive any other compatible image sequence through the same motion. In Figure 7(j), although the image sequence itself specifies a HOLD during that interval, the skeleton sequence drives it through the same motion cycle. The relationship between a start/end frame skeleton pair and the number of key images in that interval is arbitrary. It should also be clear that the interpolation rate between key images is independent of the progression of its skeleton through a movement. Because of this, the smoothing process may span as many key images as required to complete a continuous movement.

Various program features assist the animator in developing a skeleton control sequence. Direct viewing of skeleton movement is obtained by requesting the ANIMATE display mode at any time. This causes the sequence of intermediate frames from the start to the end frame to be continuously cycled at the cine rate instead of being presented as a static ensemble. Companion skeletons that have been developed for several cels may be previewed together in this mode as well as during interaction. If the INBETWEEN display presentation contains confusing frame to frame overlap (e.g. walking on the spot), a positional offset may be introduced into successive control frames to remove the ambiguity. Similarly any number of intermediate frames may be skipped to simplify the overall display during interactive modification.

Proposed Extensions to the System

Because final processing of skeleton control sequences is performed in the composite playback program, the same camera control commands that apply pan, tilt, and zoom to an image sequence can be applied to the control sequence. With an extension of the system, more complex processing functions could be applied to the control skeleton. This approach may be useful for superimposing complex forms of movement control on the skeleton. It may also significantly reduce the processing time for rotation since only the skeleton coordinate references need to be rotated.

Another useful extension deals with the capability for creating a library of common movements. If sequences of control frames are saved in a normalized form, they can be retrieved and superimposed on any particular skeleton as a starting point for developing variations of that movement. Since the details of the image itself are not contained in the skeleton, this necessitates only that the form of this skeleton match the standard normalized form (i.e. it consists of the same connection of bones). All the physical characteristics of the particular skeleton being used (such as relative length and width of each section) will be retained while only the stored motion characteristics will be transferred. Although this capability has not been implemented in the present system, it does indeed offer an important potential reduction in animation production costs.

References

1. Burtnyk, N., and Wein, M. Computer generated key frame animation. J. Soc. Motion Picture and Television Engineers 80, 3 (1971), 149–153.

2. Burtnyk, N., and Wein, M. Towards a computer animating production tool. Proc. Eurocomp Conf., Brunel U., 1974, Online Pub. Co., 172–185.

3. Burtnyk, N., and Wein, M. Computer animation of free form images. Computer Graphics 9, 1 (1975), 78-80 (Issue of Proc. Second Ann. Conf. Computer Graphics and Interactive Techniques).

 Baecker, R.M. Picture-driven animation. Proc. AFIPS
1969 SJCC, Vol. 34, AFIPS Press, Montvale, N.J., pp. 273–288.
Csuri, C. Real-time animation. Proc. Ninth Ann. Meeting
UAIDE (Users of Automatic Inform. Display Equipment), Miami, 1970, pp. 289–305.

6. Burtnyk, N., et al. Computer graphics and film animation. Canadian J. Operational Res. and Inform. Processing (INFOR) 9, 1 (1971), 1-11.

7. Burtnyk, N., and Wein, M. A computer animation system for the animator. Proc. Tenth Ann. Meeting UAIDE (Users of Automatic Inform. Display Equipment), Los Angeles, 1971, pp. 3.5-3.24.

8. Blum, H. A transformation for extracting new descriptors of shape. In *Models of Speech and Visual Form*, MIT Press, Cambridge, Mass., 1967, pp. 362–380.

9. Honey, F.J. Computer animated episodes by single axis rotations—CAESAR. Proc. Tenth Ann. Meeting UAIDE (Users of Automatic Inform. Display Equipment), Los Angeles, 1971, pp. 3-210 to 3-226.

10. Catmull, E. Computer display of curved surfaces. Proc. Conf. Computer Graphics, Pattern Recognition and Data Structure, May 1975, pp. 11–17 (IEEE Cat. No. 75CH0981-IC).

11. Akima, H. A new method of interpolation and smooth curve fitting based on local procedures. J. ACM 17, 4 (1970), 589-602.