UNIT-6

Animations &Realism: Design of Animation sequences – animation function – raster animation – key frame systems – motion specification – morphing – tweening.

Computer Graphics Realism: Tiling the plane – Recursively defined curves – Koch curves – C curves – Dragons – space filling curves – fractals

– Grammar based models – fractals – turtle graphics – ray tracing.

Common steps of designing the animation sequence, Computer Graphics

Common Steps of Designing the Animation Sequence

Common Steps of designing the animation sequence are as given:

1) Layout of Storyboard: Storyboard layout is the action outline utilized to illustrate the motion sequence as a set of basic events which are to acquire place. This is the kind of animation to be produced that selects the storyboard layout. So, the storyboard comprises a set of rough sketches or a list of basic concepts for the motion.

2) Definition of Object: The object definition is specified for all participant objects in action. The objects can be explained in terms of fundamental shapes, related movements or movement with shapes.

3) Specification of Key Frame: this is the detailed drawing of the scene at an exact time in the animation sequence. Inside each key frame, all objects are positioned as per to time for that frame. Several key frames are selected at the extreme positions in the action; others are spaced hence the time interval among key frames is not as great. More key frames are given for intricate motion than for easy, slowly varying motions.

4) In-between frames Generation: In-among frames are the middle frames among the key frames. The number of among frames is based on the media to be utilized to display the animation. In common, film needs twenty-four frames per second, and graphic terminals are refreshed on the rate of 30 to 60 frames per second. Classically the time interval for the motion is set up hence there are 3 to 5 among for each pair of key frames. Based upon the speed identified for the motion, several key frames can be duplicated.

Animation Functions

1. Morphing: Morphing is an animation function which is used to transform object shape from one form to another is called Morphing. It is one of the most complicated transformations. This function is commonly used in movies, cartoons, advertisement, and computer games.

For Example:

1.Human Face is converted into animal face as shown in fig:

2. Face of Young person is converted into aged person as shown in fig:

The process of Morphing involves three steps:

1. In the first step, one initial image and other final image are added to morphing application as shown in fig: Ist & 4th object consider as key frames.
2. The second step involves the selection of key points on both the images for a smooth transition between two images as shown in 2nd object.

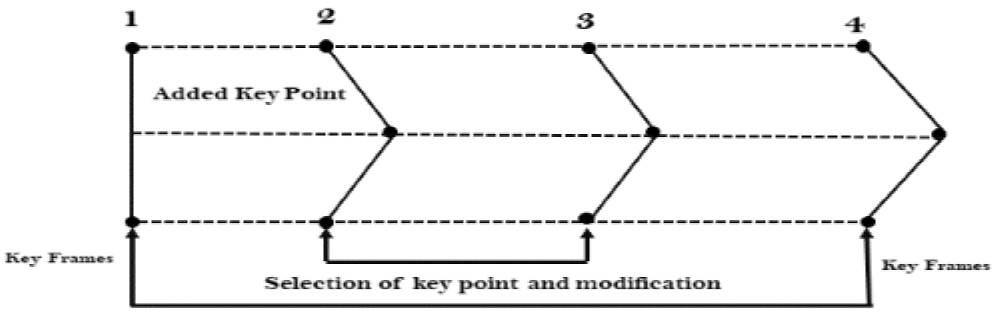

Fig: Process of Morphing

3. In the third step, the key point of the first image transforms to a corresponding key point of the second image as shown in 3rd object of the figure.

2. **Wrapping:** Wrapping function is similar to morphing function. It distorts only the initial images so that it matches with final images and no fade occurs in this function.

3. **Tweening:** Tweening is the short form of 'inbetweening.' Tweening is the process of generating intermediate frames between the initial & last final images. This function is popular in the film industry.

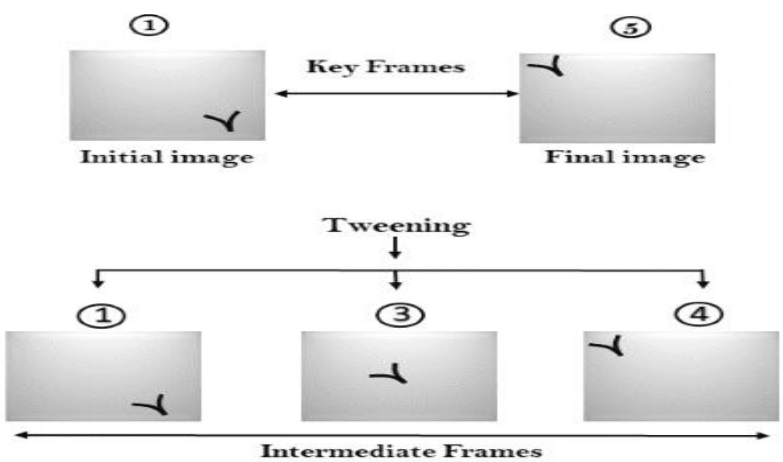

Fig: Tweening

4. **Panning:** Usually Panning refers to rotation of the camera in horizontal Plane. In computer graphics, Panning relates to the movement of fixed size window across the window object in a scene. In which direction the fixed sized window moves, the object appears to move in the opposite direction as shown in fig:

Fig: Panning

If the window moves in a backward direction, then the object appear to move in the forward direction and the window moves in forward direction then the object appear to move in a backward direction.

5. **Zooming:** In zooming, the window is fixed an object and change its size, the object also appear to change in size. When the window is made smaller about a fixed center, the object comes inside the window appear more enlarged. This feature is known as **Zooming In**.

When we increase the size of the window about the fixed center, the object comes inside the window appear small. This feature is known as **Zooming Out**.

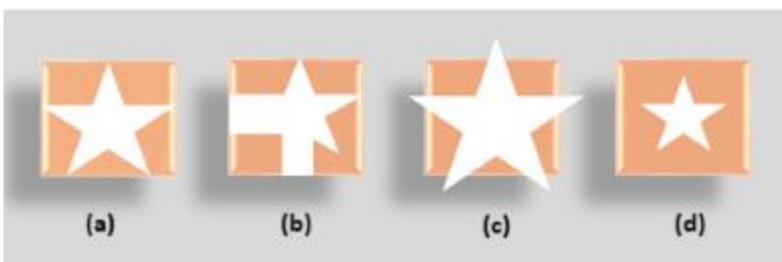

Fig: Zooming in & Zooming Out

6. **Fractals:** Fractal Function is used to generate a complex picture by using Iteration. Iteration means the repetition of a single formula again & again with slightly different value based on the previous iteration result. These results are displayed on the screen in the form of the display picture.

Raster Animations

On raster systems, real-time animation in limited applications can be generated using raster operations. Sequence of raster operations can be executed to produce real time animation of either 2D or 3D objects. We can animate objects along 2D motion paths using the color-table transformations. Predefine the object as successive positions along the motion path, set the successive blocks of pixel values to color table entries-Set the pixels at the first position of the object to „on" values, and set the pixels at the other object positions to the background color. The animation is accomplished by changing the color table values so that the object is „on" at successive positions along the animation path as the preceding position is set to the background intensity.

Computer Animation Languages

Animation functions include a graphics editor, a key frame generator and standard graphics routines. The graphics editor allows designing and modifying object shapes, using spline surfaces, constructive solid geometry methods or other representation schemes. Scene description includes the positioning of objects and light sources defining the photometric parameters and setting the camera parameters. Action specification involves the layout of motion paths for the objects and camera. Keyframe systems are specialized animation languages designed dimply to generate the in betweens from the user specified keyframes. Parameterized systems allow object motion characteristics to be specified as part of the object definitions. The adjustable parameters control such object characteristics as degrees of freedom motion limitations and allowable shape changes.

Scripting systems allow object specifications and animation sequences to be defined with a user input script. From the script, a library of various objects and motions can be constructed. Keyframe Systems Each set of in betweens are generated from the specification of two keyframes. For complex scenes, we can separate the frames into individual components or objects called cells, an acronym from cartoon animation.

Morphing

Transformation of object shapes from one form to another is called Morphing. Morphing methods can be applied to any motion or transition involving a change in shape. The general preprocessing rules for equalizing keyframes in terms of either the number of vertices to be added to a keyframe. we equalize the edge count and parameters Lk and Lk+1 denote the number of line segments in two consecutive frames. We define,

Lmax= max (Lk, Lk+1)

Lmin= min(Lk, Lk+1)

Ne= Lmax mod Lmin

Ns= int (Lmax/Lmin)

The preprocessing is accomplished by

1. Dividing Ne edges of keyframe min into Ns+1 section.

2.Dividing the remaining lines of keyframe min into Ns sections.

For example, if Lk= 15 and Lk+1= 11, we divide 4 lines of keyframek+1into 2 sections each. The

remaining lines of keyframek+1 are left infact. If the vector counts in equalized parameters Vk and Vk+1 are used to denote the number of vertices in the two consecutive frames. In this case we define

Vmax = max(Vk,Vk+1),

Vmin = min(Vk,Vk+1) and

Nls= (Vmax-1) mod (Vmin–1)

Np= int ((Vmax–1)/(Vmin–1))

Preprocessing using vertex count is performed by

1. Adding Np points to Nls line section of keyframe min

2. Adding Np - 1 points to the remaining edges of keyframe min

Simulating Accelerations

Curve- fitting techniques are often used to specify the animation paths between key frames. Given

the vertex positions at the key frames, we can fit the positions with linear or nonlinear paths.

frame positions. This determines the trajectories for the in betweens. To simula te accelerations, we can adjust the time spacing for the in betweens. For constant speed (zero acceleration), we use equal interval time spacing for the in betweens. Suppose we want n in betweens for key frames at times t1 and t2. The time interval between keyframes is then divided into n + 1 subintervals, yielding an in-between spacing of Δ= t2-t1/n+1

we can calculate the time for any in-between as tBj = t1+j Δt, j = 1,2, ... n

Motion Specification

These are several ways in which the motions of objects can be specified in an animation system.

Direct Motion Specification

Here the rotation angles and translation vectors are explicitly given.

Then the geometric transformation matrices are applied to transform coordinate positions.

We can approximate the path of a bouncing ball with a damped, rectified, sine curve

$y(x) = A / \sin(\omega x + \theta 0) / e\text{-}kx$

where A is the initial amplitude, ω is the angular frequency, θ is the phase angle and k is the damping constant.

Goal Directed Systems

We can specify the motions that are to take place in general terms that abstractly describe the actions. These systems are called goal directed. Because they determine specific motion parameters given the goals of the animation.

Eg., To specify an object to „walk‟ or to „run‟ to a particular distance.

Kinematics and Dynamics

With a kinematics description, we specify the animation by motion parameters (position, velocity and acceleration) without reference to the forces that cause the motion.

For constant velocity (zero acceleration) we designate the motions of rigid bodies in a scene by giving an initial position and velocity vector for each object.

We can specify accelerations (rate of change of velocity), speed up, slowdowns and curved motion paths.

An alternative approach is to use inverse kinematics; where the initial and final positions of the object are specified at specified times and the motion parameters are computed by the system.