Computer Animation:

- Computer Animation refers to any time sequence of visual changes in a scene.
- A computer generated animation scene could display time variations in object size, color, transparency or surface texture.
- Advertising animations –morphing
- Computer animations can be generated by changing camera parameters such as position, orientation and focal length.
- Visual Effects—producing exaggerated shapes and unrealistic motions and transformations

Steps for Design of Animation Sequences:

1. Storyboard layout
2. Object definition
3. Key - frame specifications
4. Generation of in- between frames

The ***storyboard*** is an outline of the action. It defines the motion sequence as a set of basic events that are to take place .Depending on the type of animation to be produced, the storyboard could consist of a set of rough sketches or it could be a list of the basic ideas for the motion.

An ***object definition*** is given for each participant in the action. Object can be defined in terms of basic shapes, such as polygons or splines In addition, the associated movements for each object are specified along with shape.

A ***key*** **frame** is detailed drawing of the scene at a certain time in the animation sequence. Within each key frame, each object is positioned according to the time for that frame. Some key frames are chosen at extreme positions in the action; others are spaced so that the time interval between key frames is not too great.

In-*betweens* are the intermediate frames between the key frames. The number of in - betweens needed is determined by the media to be used to display the animation .Film requires 24 frames per second, and graphics terminals are refreshed at the rate of 30 to 60 frames per second .

General Computer Animation Functions:

1. store and manage the object database
2. functions for motion generation and for object rendering
3. functions that simulates camera movements
4. zooming, panning, and tilting

Some steps in development of animation sequence are well - suited to computer solution. These include object manipulation and rendering camera motions, and the generation of in - betweens. One function available in animation packages is provided to **store and manage the object database.** Object shapes and associated parameters are stores and updated in the database. Other object functions include those for **motion generation** and those for **object rendering**. Motion

can be generated according to specified constraints using two dimensional or three dimensional transformations.
Another typical function **simulates camera movements**. Standard motions are **zooming, panning, and tilting.**

<u>Computer Animation Languages:</u>

1. C
2. Lisp
3. Pascal
4. FORTRAN
5. Special animation packages like Adobe Flash , autodesk maya etc.

Design and control of animation sequences are handled with a set of animation outlines. A general purpose language, such as **C, Lisp, Pascal or FORTRAN** is often used to program the animation functions, but several **specialized animation languages** have been developed. Animation functions include a **graphics editor**, **a key frame generator, an in-between generator and standard graphics routines.** The graphics editor allows us to design and modify object shapes, using spline surface, constructive solid geometry methods, or other representation schemes.

A typical task in an animation specification is **scene description**. This includes the positioning of objects and light sources, defining the photometric parameters, and setting the camera parameters (position, orientation, and less characteristics). Another standard function is **action specification**. This involves the layout of motion paths for the object and camera. And we need the usual graphics routines: viewing and perspective transformations, geometric transformations to generate object movements as a function of accelerations or kinematic path specification, visible-surface identification and the surface rendering operations.

key-frame systems are specialized animation languages designed simply to generate the in betweens from the user-specified key frames. Usually , each object in the scene is defined as a set of rigid bodies connected at the joints and with a limited number of degrees of freedom.

Parameterized systems allow object motion characteristics to be specified as part of the object definitions. The adjustable parameters control such object characteristics as degree of freedom, motion limitations, and allowable shape changes.

Scripting systems allows object specifications and animation sequences to be define with a user-input script. From the script, a library of various objects and motions can be constructed.

Raster Animations:

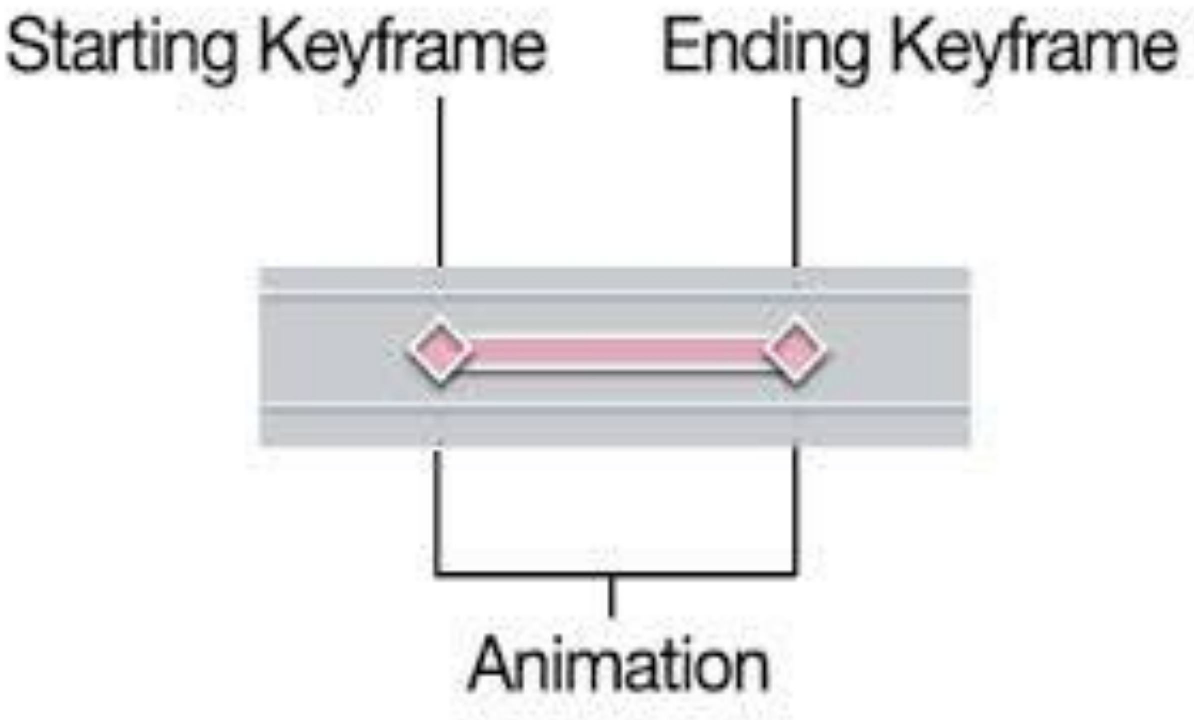
On raster scan systems we can generate real time animation in limited applications using **raster operations**. A simple method for translation in the xy plane is to transfer a rectangular blocks of pixels through arbitrary angles using anti-aliasing procedures. To rotate a block of pixels we need to determine the percent of area coverage for those pixels that overlap the rotated block. Sequences of raster operations can be executed to produce real time animation of either two-dimensional or three-dimensional objects, as long as we restrict the animation to motions in the projection plane. Then no viewing or visible surface algorithms need be invoked.

Color table transformations also produce animation. Just by changing the intensity values of pixels to on and off, we can produce animation on raster systems.

Key Frame Systems:

- Generate set of in-betweens from specification of two key frames
- Motion paths can be given with kinematic or dynamic descriptions
- Cels
- Animation paths—interpolation
- Morphing(transforming shape from one form to another form).
- Linear Interpolation
- Simulating Accelerations

A key frame is a detailed drawing of the scene at a certain time in the animation sequence. Within each key frame each object is positioned according to the time for that frame.In-betweens are intermediate frames between the key frames.

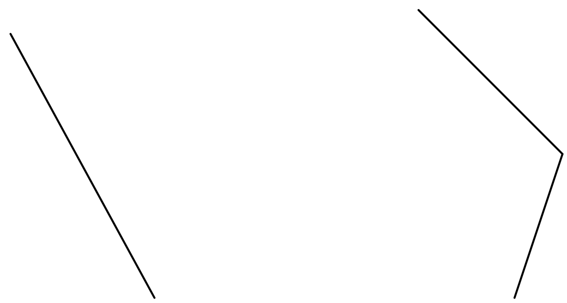


Morphing

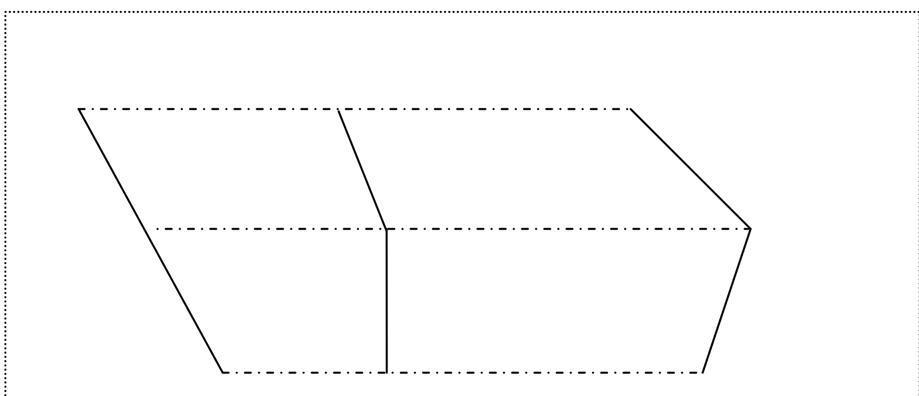
Transformation of object shapes from one form to another is called morphing. We generate set of in-betweens from the specification of two or more key frames. Given the animation paths we can interpolate the positions of individual objects between any two times or key frames. With complex object transformations the shapes of the object may change over time. If all surfaces are described with polygon meshes then the number of edges per polygon can change from one frame to the next. Thus the total number of line segments can be different in different frames.

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 of shape.

Given two key frames for an object transformation we first adjust the object specification in one if the frames so that the number of polygon edges (or the number of vertices) is the same for the two frames. This is illustrated below



Frame K   Frame K+1



Frame K   Halfway Frame   Frame K+1

A straight-line segment in key frame k is transformed into two line segments in key frame k+1. Since key frame k+1 has an extra vertex, we add a vertex between 1 and 2 in key frame K to balance the number of vertices and edges in the two key frames. Using linear interpolation to generate the in betweens we transition the added vertex in key frame k into vertex 3’ along the straight-line path as shown.We can state general preprocessing rules for equalizing key frames in terms of either the number of edges or the number of vertices to be added to a key frame.

Case 1: Equalizing edge count.

Let the parameters L_k and L_{k+1} denote the number of line segments in two consecutive frames. We then define

$$L_{max} = \max(L_k, L_{k+1})$$

$$L_{min} = \min(L_k, L_{k+1})$$

$$N_e = L_{max} \bmod L_{min}$$

$$N_s = \mathrm{int}(L_{max} / L_{min})$$

Then the preprocessing is accomplished by

1. Dividing the N_e edges of $keyframe_{min}$ into N_s sections.
2. Dividing the remaining lines of $keyframe_{min}$ into N_s sections

Case 2: Equalizing vertex count

Let the parameters be V_k and V_{k+1} denote the number of vertices in the two consecutive frames. We define

$$V_{max} = \max(V_k, V_{k+1})$$

$$V_{min} = \min(V_k, V_{k+1})$$

$$N_{ls} = (V_{max}-1) \bmod (V_{min}-1)$$

$$N_p = \mathrm{int}\ ((V_{max}-1) / (V_{min}-1))$$

Preprocessing using vertex count is performed by

1. Adding N_p points to N_{ls} line sections of $keyframe_{min}$

Adding N_p-1 points to the remaining edges of $keyframe_{min}$

Motion Specification in Computer Animation

There are several ways in which the motions of objects can be specified in an animation system. We can define motion in very explicit terms or We can use more abstract or more general approaches.

Ways for defining motion of objects are:

- **Direct motion specification**
- **Goal-directed systems**
- **Kinematics, Inverse kinematics and dynamics, Inverse Dynamics**

Direct motion specification : -

The most straightforward method for defining a motion sequence is direct specification of the motion paremeters. Here, We explicitly give the rotation angles and translation vectors. Then the geometric transformation matrices are applied to transform co-ordinate positions. Alternatively, We could use an approximating equation to specify certain kinds of motions. These methods can be used for simple user programmed animation sequencs.

Goal-directed systems : -

At the opposite extreme, We can specify the motions that are to take place in general terms that abstractly describe the actions. these systems are referred to as goal directed because they determine specific motion parameters given the goals of the animation. For example, We could specify that we want an object to "walk " or to "run" to a particular destination. Or We could state that we want an object to "pick up " some other specified object. The inpute directive are then interpreted in term of component motions that will accomplish the selected task. Human motion, for instance, can be defined as a heirarchical structure of sub motion for the toros, limbs,and so forth.

Kinematics and dynamics : -

We can also construct animation sequences using kinematic or dynemic descriptions. With a kinematic description, we specify the animation by giving 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 objects.

An alternate apporach is to use inverse kinematics. Here, we specify the initial and final positions of objects at specified times and the motion parameters are computed by the system . For example, assuming zero acceleration , we can determine the constant velocity that will accomplish the movement of an object from the initial position to the final position.

Dynamic descriptions on the other hand, require the specification of the forces that produce the velocities and acceleration. Descriptions of object behavior under the are generally referred to as a physically based modeling. Example of forces affecting object motion include electromagnetic, gravitational, friction, and other mechanical forces.

Object motion are obtained from the forces equations describing physical laws, such as newton's law of motion for gravitational ang friction processes, euler or navier-stokes equations describing fluid flow, and maxwell 's equations for electromagnetic forces. For example, the general form os newton's second law for a particle of mass m is
$F = d(mv)/dt$

with F as the force vector, and v as the velocity vector. If mass is constant, we solve the equation $F=ma$, where a is the acceleration vector. otherwise, mass is a function of time, as in relativistic motions of space vehicles that consume measurable amounts of fuel per unit time. We can also use inverse dynamics to obtain the forces, given the initial and final positions of objects and the type of motion.

Application of physically based modeling include complex rigid-body systems and such non rigid systems as cloth and plastic materials. Typically, numerical methods are used to obtain the motion parameters incrementally from the dynamical equations using initial conditions or boundary values.