Computer Animation
Computer Animation

Animation
The term animation has a Greek (animos) as well as roman (anima) root, meaning “to bring to life”
Life: evolution over time

Conventional Animation
Animation is a technique in which the illusion of movement is created by photographing a series of individual drawings on successive frames of film.

“The Illusion of Life” by Thomas Johnson and Ollie Johnson (From Disney Animation)
Computer Animation

Conventional Animation

Animation refers to the process of dynamically generating a series of frames of a set of objects, in which each frame is an alteration of the previous frame.

Restrictions
a) Frame by frame and not real time
b) 2 D only
Computer Animation

Conventional Animation

Process

• Story board
  Sequence of drawings with descriptions
• Key frames
  A few important frames as drawings
• Inbetweens
  Draw the rest of the frames
• Painting
  Redraw onto acetate Cels, color them
Computer Animation

Conventional Animation

The flour sack principle

Stretch and Squash using half filled bag of flour
Computer Animation

Conventional Animation

Stretch and Squash

Exaggeration with believability
Computer Animation

Real Time vs. Image by Image

a) Real Time: Compute - Draw
b) Image by Image: Compute – Store - Draw
c) Display rate: 30 fps or 25 fps

Animation characteristics

• Spatial (position, orientation, form )
• Temporal (velocity, acceleration )
• Visual (color, texture )
Computer Animation

Animation Techniques

• Rotoscopy
• Key Framing
• Parametric
• Algorithmic
Computer Animation

Rotoscopy

- Register (record) data for each frame
- Data intensive
- Useful for complex motion
- Realistic
- Brute-force (less creative)

*Data driven animation, Motion capture*
Computer Animation

Key Framing

- Selected (key) frames are specified
- Interpolation of intermediate frames
- Simple and popular approach
- May give incorrect (inconsistent) results
Computer Animation

Key Framing

Interpolation
At \( f_i \), the position of a point \( P_i \) for \( KF_1: P_1 \), \( KF_2: P_2 \)
\[ P_i = (1-t)P_1 + tP_2 \]
Computer Animation

Key Framing

Linear Interpolation
- Discontinuities: spatial, temporal
  Splines may be used
- Unrealistic results

\[ \text{KF}_1 \quad \text{KF}_2 \quad \text{KF}_3 \]
\[ \text{KF}_1 \quad \text{KF}_2 \quad \text{KF}_3 \]
Computer Animation

Key Framing

Interpolation

Linear
Computer Animation

Key Framing

Interpolation

Spline
Computer Animation

**Key Framing**

**Interpolation**
Using other functions (slow-in, slow-out)

\[ v_i = (1-f(t)) v_s + f(t) v_e \]

\( v_s \): attribute at start frame
\( v_e \): attribute at end frame
\( v_i \): attribute at intermediate frame
Computer Animation

Key Framing

Interpolation

At $t = 0.5$

$K_{F1}(t=0) \quad At \quad t = 0.5 \quad K_{F2}(t=1)$
Computer Animation

Key Framing

Interpolation

Incorrect Results (rotation of square by 180°)

$KF_1(t=0)$

At $t = 0.5$

$KF_2(t=1)$
Computer Animation

Parametric

• Characteristic parameters for motion are specified and interpolated.
• Less data is required e.g. for motion of an arm, the parameter could be rotation angle.

\[ \theta \]
Computer Animation

Algorithmic

Laws of motion: physical or procedural animation

Simulation
Computer Animation

Example
Computer Animation

Morphing

Transformation of object shapes from one form to another

- Each form may be considered as a key frame
- Establish common topology for the two key frames
- Interpolate the intermediate frames
Computer Animation

Image Morphing

Transformation of one image (source) to another image (target)

- Normalization of both images
- Feature correspondence
- Warping of the two images (spatial deformation)
- Color blending
Computer Animation

Image Morphing

Without feature correspondence (cross dissolving)

Source

Destination
Computer Animation

Image Morphing

Correspondence
Computer Animation

Image Morphing

Triangle Method

- Feature points are marked on source and target.
- These feature are given the correspondence.
- Triangulate the points.
- Interpolate triangulation for intermediate frames.
- Warp the images, and blend colors.
Computer Animation

Image Morphing

Triangle Method

Interpolation in triangular domain
How is $P$ related to $P_1$, $P_2$ and $P_3$?

$$P = uP_1 + vP_2 + wP_3$$

$$u = \frac{A_1}{A}, \quad v = \frac{A_2}{A}, \quad w = \frac{A_3}{A}$$

$A$: total area of triangle

$u$, $v$, $w$: Barycentric coordinates.
Computer Animation

Image Morphing

Triangle Method

With feature correspondence (Triangle Method)
Computer Animation

Image Morphing

Triangle Method

Example
Computer Animation

Image Morphing

Application
Computer Animation

Particle Systems

Williams T Reaves (1983) SIGGRAPH

Particle Systems “A Technique for Modeling a Class of Fuzzy Objects”
Computer Animation

Particle Systems

An object is represented as cloud of particles
Particles are not static; particle system evolves
Non deterministic

Particles are simple (computationally efficient) but can model complex amorphous objects and behaviors

*Dust, Water fall, Rain, Fire, Cloud, Stars, Grass, Fur, etc*
Computer Animation

Particle Systems
Computer Animation

Particle Systems

In a typical particle system

• Generate new particles with initial attributes
• Particles have lifespan: Kill off dead particles
• Modify particle attributes: color, position
• Render particles
Computer Animation

Particle Systems

Particle Generation

Stochastic

\[ N = \text{average} + \text{rand}() \times \text{var} \]

Particle Attributes

Determine motion status, appearance, and its life in the particle system

(position, color, opacity, size, speed, life-span etc.)

Initialized at the time of creation
Computer Animation

Particle Systems

Particle Termination

For each new frame, particle’s life time is decremented by one

When life time = zero, the particle is removed
Computer Animation

Particle Systems

Particle Animation

Particle dynamics

From force find acceleration

velocity

position

Other attributes (color, opacity, etc.) may also change with time
Particle Animation

Particle Systems

Particle Animation

Particle dynamics

\[ v_{i}^{\text{new}} = v_{i} + \Delta t \frac{F(x_{i}, v_{i}, t)}{m_{i}} \]

\[ x_{i}^{\text{new}} = x_{i} + \Delta t v_{i}^{\text{new}} \]
Computer Animation

Particle Systems

Particle Rendering

- Particles can be rendered as light sources
- Particles do not intersect with objects
- May ignore shadows
- These assumptions simplify the rendering and computation
Computer Animation

Particle Systems

Examples
Computer Animation

Particle Systems

Algorithm

for each video frame {
    generate new particles
    remove old particles
    for each particle {
        resolve forces by vector addition
        calculate a, v, x
        update other particle properties
        render particle
    }
}
Computer Animation

Particle Systems

Wrath of Khan

Particle systems generated in concentric rings

Number of systems based on ring circumference

New fire particles based on distance from impact crater

Fig. 2. Distribution of particle systems on the planet’s surface.
Computer Animation

Particle Systems

Wrath of Khan

Individual particle systems look like explosions
Computer Animation

Particle Systems
Computer Animation

Particle Systems

Wrath of Khan
Computer Animation

Particle Systems

More Examples

Smoke and Fire
Particle Systems

A grass clump is a particle system

A particle is a blade of grass

Draw parabolic streak over entire life time
Computer Animation

Particle Systems

Behavioral Animation
Flocking of birds

Deformable Objects
of springs: Cloth
of springs: Hair