Animation

Dr. V Masilamani
masila@iiitdm.ac.in

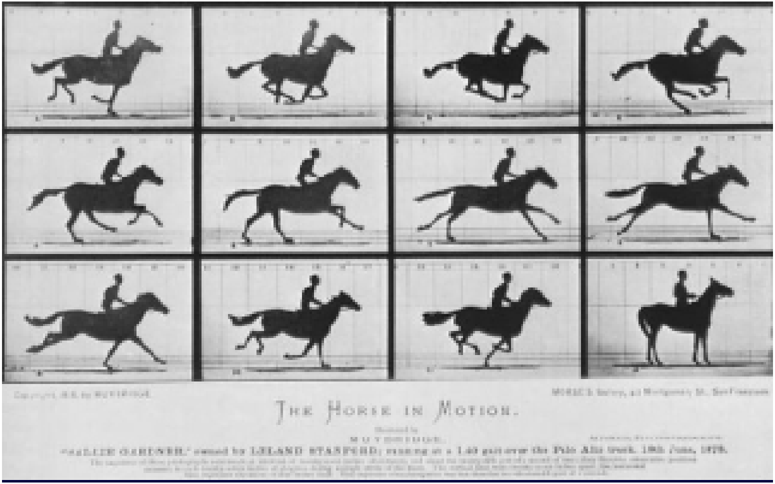
Department of Computer Science and Engineering
IIITDM Kancheepuram
Chennai-127

Overview

What is Animation?

- The term **animation** has a Greek (animos) as well as roman (anima) root, meaning "to bring to life"
  Life: evolution over time
- Animation is a technique in which the **illusion of movement** is created by a sequence of individual drawings with the property that each frame(drawing) is the alteration of previous frame.
- **Conventional Animation:** Sequence of frames, where each frame is drawn manually
  - **Example:** "The Illusion of Life" by Thomas Johnson and Ollie Johnson (From Disney Animation)
- **Computer Animation:** Sequence of frames, where frames are drawn using computer

Conventional Animation



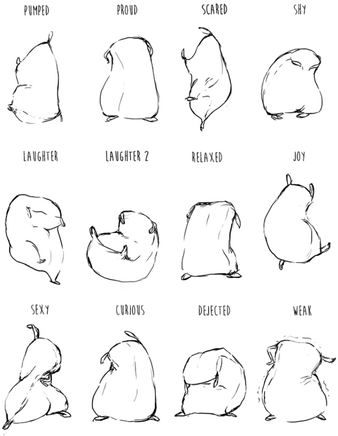
Restrictions in Conventional Animation

- As every frame is drawn by animators manually, the animation is not real time
- Mostly only 2D has been considered

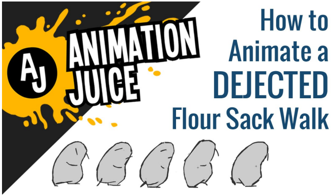
The steps of animation process

- 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

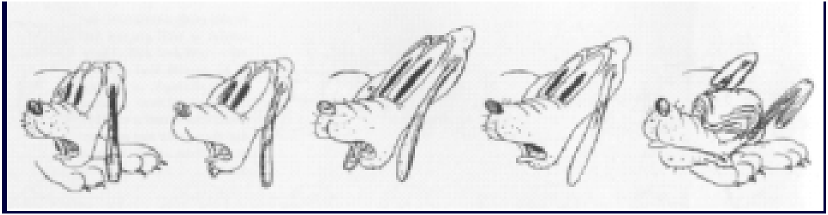
- **The flour sack principle:** Stretch and Squash using half filled bag of flour
- The half-filled flour bag will squash out to the fullest shape when it's dropped on the floor, and
- It will stretch out to the longest shape when it's picked up by the corners.
- But it will never change its volume.
- Animation technique that mimics deformation of half-filled flour bag is called as stretch and squash



- Anything composed of living flesh will be proportionally deformed (squashed or stretched) within its shape in processing through action.
- This rule could be also applied to still elements, like a ball or chair, to emphasize movement.
- For instance, a ball should squash when touching the floor during a bouncing animation.



Conventional Animation using Stretch and Squash



- **An attribute of animation:** Exaggeration with believability to some extent

Factors to be Considered in Animation

Real Time vs. Image by Image

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

Animation Characteristics

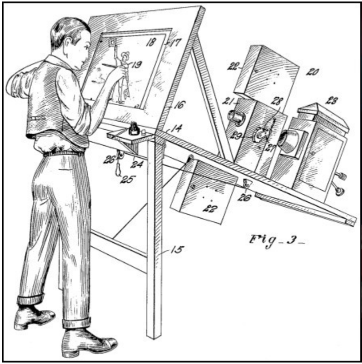
- Spatial (position, orientation, form)
- Temporal (velocity, acceleration)
- Visual (color, texture)

Animation Techniques

- Rotoscopy
- Key Framing
- Parametric
- Algorithmic

Animation Techniques (cont.)

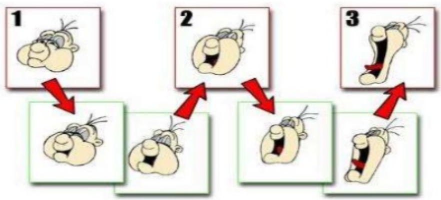
- **Rotoscopy: The following process of creating animation is called as rotoscopy**
  - Photographed images are projected onto a glass panel
  - Trace over the image with a drawing tool



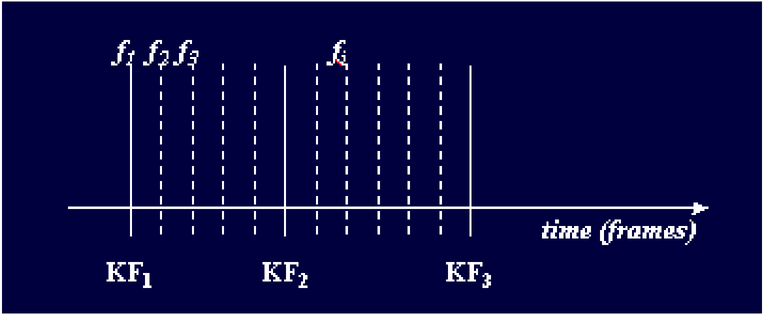
- The device used for projection is called as rotoscope
- Now the device rotoscope has been replaced by computer
- Register (record) data for each frame
- Data intensive
- Useful for complex motion
- Realistic
- Brute-force (less creative)

Key Framing

- Frames with very high deviations are called as key frames
- Draw such key frames
- Draw the frames between two consecutive key frames using interpolation
- May give incorrect (inconsistent) results in some cases



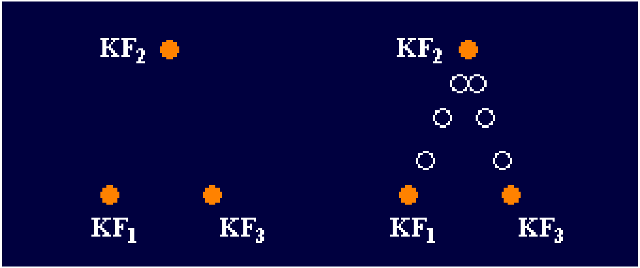
Key Framing



- Interpolation
  - Let $P_1 = (x, y, t_{kF_1})$, $P_2 = (x, y, t_{kF_2})$
  - The corresponding point in frame i, $P_i = (x, y, i)$ is calculated as $P_i = (1 - t)P_1 + tP_2$

Key Framing

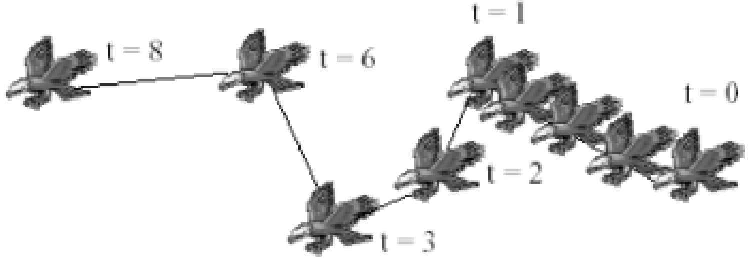
- Issues with Linear Interpolation
  - May not be smooth
  - Unrealistic results: it will produce the same upward velocity downward velocity



- Solution : Splines may be used

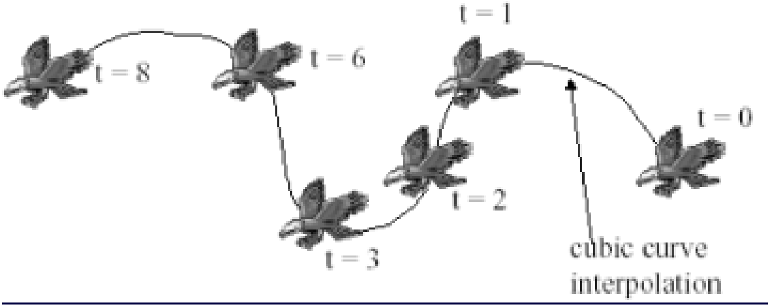
Key Framing

- Linear Interpolation



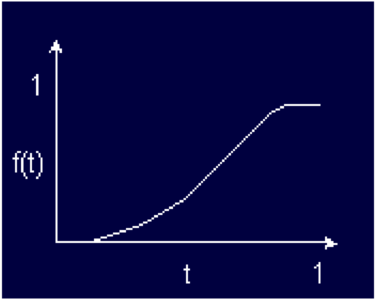
Key Framing

- Spline Interpolation



Key Framing

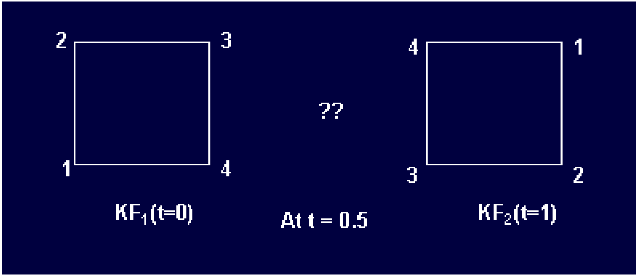
- Interpolation
  - Using other functions (slow-in, slow-out)



V_s: attribute at start frame
V_e: attribute at end frame
V_i: attribute at intermediate frame
$V_i = (1 - f(t))V_s + f(t)V_e$

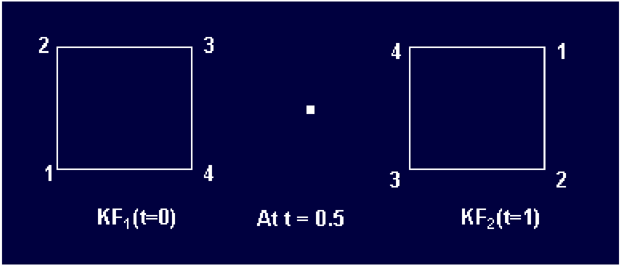
Key Framing

- Suppose a square at $t = 0$ has become rotated square with 180 degree at time $t = 1$
- If linear interpolation is applied to find a frame at $t = 0.5$, what will be the shape of the object?



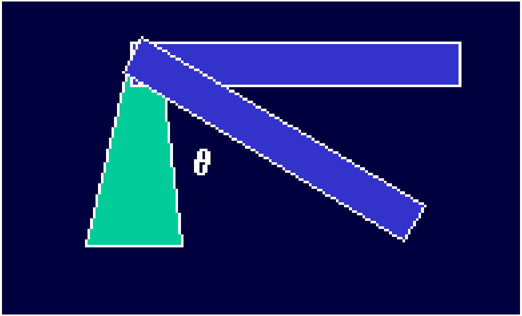
Key Framing

- The linear interpolation produces
  - Incorrect Results: Just a dot



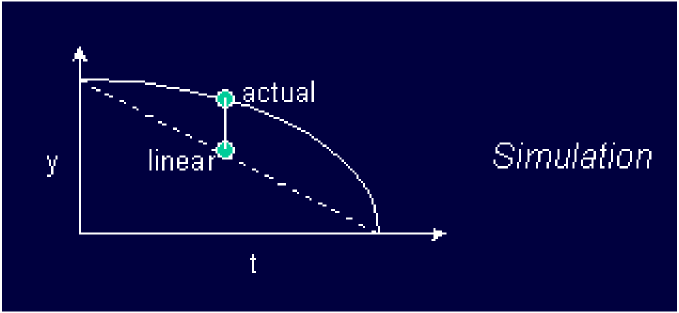
Parametric

- 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



Algorithmic

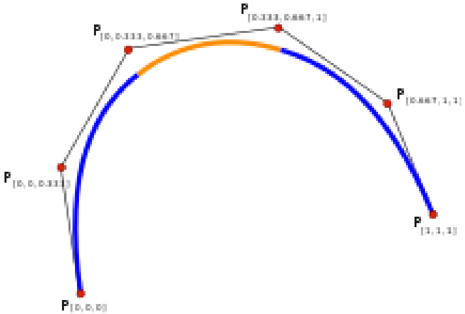
- Laws of motion: physical or procedural animation



Example

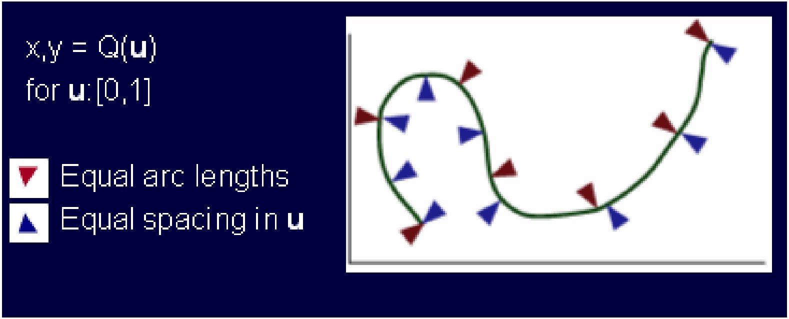
Spline Driven Animation

- Spline: Continuous piece-wise polynomial
- The spline is expected to be as smooth as possible at the joints(knots or boundary of the intervals)
- Splines are used to define/ represent curves



Spline Driven Animation (cont.)

- Suppose the key frames are not equally spaced over time(parameter u), but we wish to to make uniform motion in the animation
-



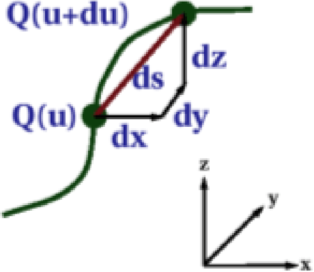
Spline Driven Animation (cont.)

Given

$$Q(u) = au^3 + bu^2 + cu + d$$

$$ds = \sqrt{dx^2 + dy^2 + dz^2}$$

$$= \sqrt{\left(\frac{dx}{du}\right)^2 + \left(\frac{dy}{du}\right)^2 + \left(\frac{dz}{du}\right)^2}$$



Spline Driven Animation (cont.)

$s = A(u) = \int_{u_0}^{u} [(\frac{dx}{du})^2 + (\frac{dy}{du})^2 + (\frac{dz}{du})^2]^{1/2} \mathrm{du}$

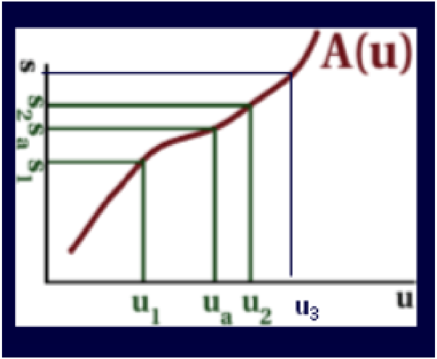
$Q(u) = Q(A^{-1}(s))$

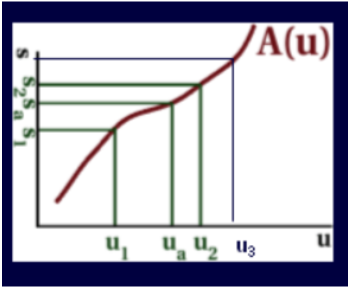
Reparameterization

Find $u = A^{-1}(s)$

Find $u = A^{-1}(s)$

A: monotonically increasing i.e $u_1 < u_2$ and $A(u_1) < A(u_2)$





Bisection method to find u_a such that $A(u_a) = s_a$, given s_a

- Start with full interval$(u_1 ... u_3)$
- Get u_2 in the middle, find s_2
- if $s_a < s_2 : u_a \in [u_1, u_2]$
- if $s_a > s_2 : u_a \in [u_2, u_3]$
- Continue till within a threshold

How to find arc length s

$s = \int_{u_0}^{u} [(\frac{dx}{du})^2 + (\frac{dy}{du})^2 + (\frac{dz}{du})^2]^{1/2} \mathrm{du}$

Since

$Q(u) = au^3 + bu^2 + cu + d$

$s = \int_{u_0}^{u} [Au^4 + Bu^3 + Cu^2 + du + E]^{1/2} \mathrm{du}$ ——(2)

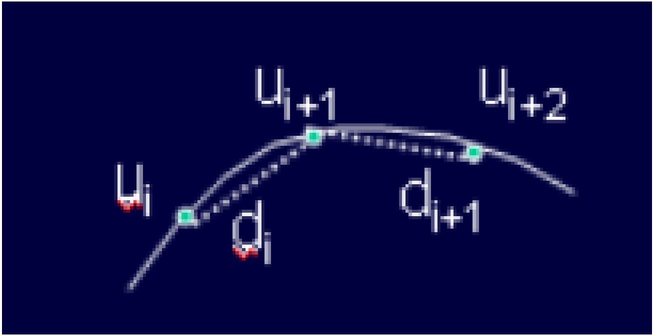
Apply Numerical method(numerical integration) to find s

Summary

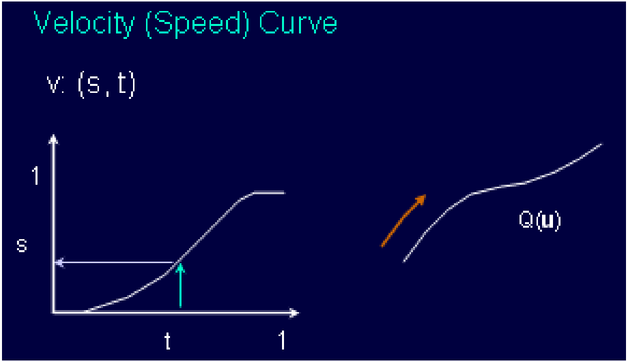
- Using equation (2) find arc length for $s = 1, 2, 3, 4....$
- Using bisection method, find the value of u for each of those vlues of s
- Compute the frame at u using spline interpolation

Alternatively

Chord length parameterization

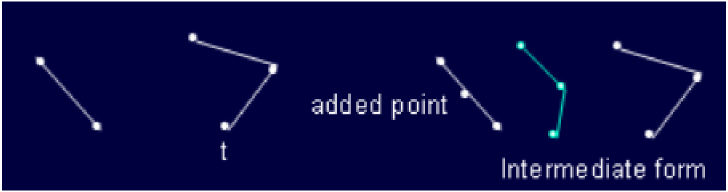


Spline Driven Animation (cont.)



Transformation of object shapes from one form to another

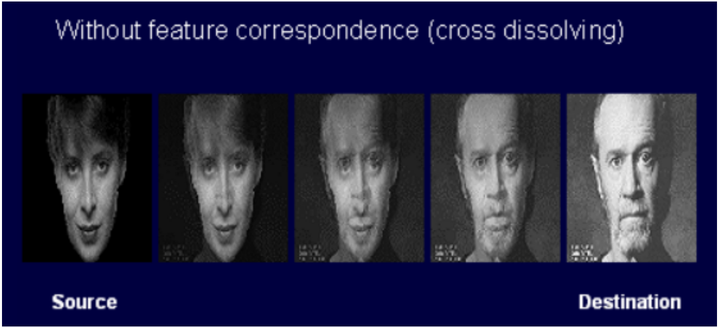
- Each form may be considered as a key frame
- Establish common topology for the Etc key frames
- Interpolate the intermediate frames



Morphing (cont.)

Transformaton of one Image (source) to another Image (target)

- Normalization of both the images
- Feature correspondence
- Warping of the two images(spatial deformation)
- Colour blending



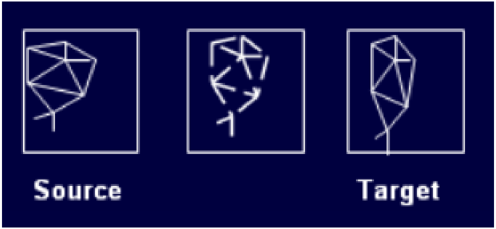
Issue with Interpolation without feature correspondence: the interpolated images have two noses and two eyes

Morphing (cont.)

Interpolation of features will result in one appearance of the feature Triangle Method

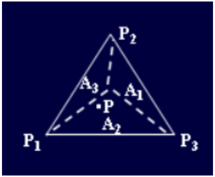
Morphing (cont.)

- 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.



Triangle Method

Interpolation in triangular domain



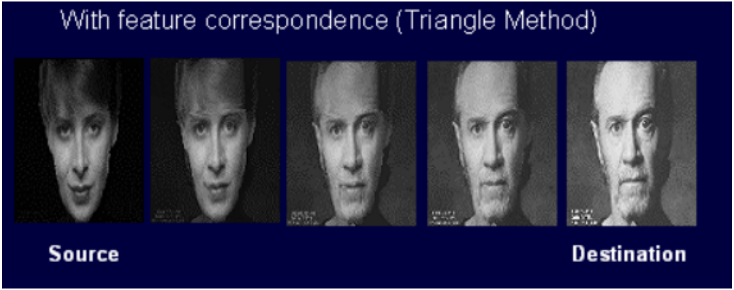
How is P related to P_1, P_2 and P_3?

P $= uP_1 + vP_2 + wP_3$

u = A1/A, v = A2/A, w=A3/A

u,v,w are Barycentric coordinates

Triangle method



Triangle method

Particle Systems

Williams T Reaves (1983) SIGGRAPH

Particle Systems "A Technique for Modeling a Class of Fuzzy Objects"

- 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

Particle Systems (cont.)

In a typical particle system

- Generate new particles with initial attributes
- Particles have lifespan: Killott dead particles
- Modify particle attributes: color, position
- Render particles

Particle Systems (cont.)

Particle Generation

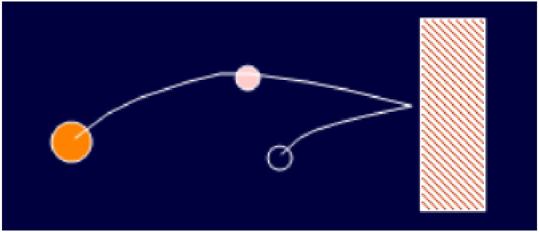
- Stochastic
- N = average + rand()*var

Particle Attributes

- Determine motion status, appearance, and its life in the partcle system (position, color, opacity, size, speed, life-span etc.)
- Initialized at the time of creation

Particle Termination

- For each new frame, particle's life time is decremented by one
- When life time = zero, the particle is removed



Particle Systems (cont.)

Particle dynamics

- From force find accelerator, velocity, position
- Other attributes (color, opacity, etc.) may also change with time

Particle Systems (cont.)

Particle dynamics

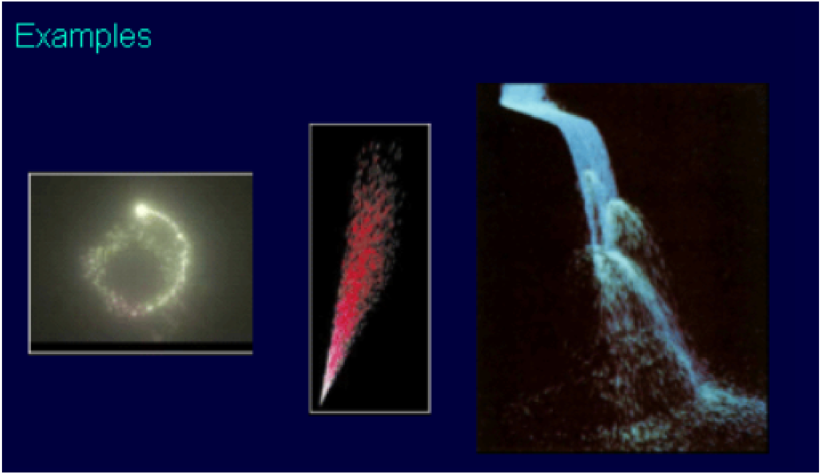
- $v_i^{new} = v_i + \Delta t \frac{F(x_i, v_i, t_i)}{m_i}$
- $x_i^{new} = x_i + \Delta t v_i^{new}$

Particle Systems (cont.)

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

Particle Systems (cont.)



Wrath of Khan

Particle Systems (cont.)

Particle Systems (cont.)

- A grass clump is a particle system
- A particle is a blade of grass
- Draw parabolic streak over entire life time

Particle Systems (cont.)

Behavioral Animation

Flocking of birds

Deformable Objects

Mesh of springs: Cloth

Lines of springs: Hair

Acknowledgements

- The slides have been adopted from NPTEL Lectures by Prof. Prem Kalra The due credits are acknowledged.

Thank You! :)