ANIMATION

Synthesis of motion – creation of "life"
The illusion of motion created by consecutive display of static elements – through Persistence of Vision
- Eye retains an afterimage for a short amount of time
- Persistence of Vision induces the illusion of continuity

Typical Framerates
- CRT's 75 Hz
- LCD's 60 Hz
- Video 50/60 Hz
- Movies 24 Hz
- Cartoons 12 Hz

Insufficient framerates result in flicker

PERSISTENCE OF VISION

Thaumatrope 1824
Phenakistoscope 1832
Zoetrope 1834

Laura Hayes and John Neward Wileman Exhibit of Optical Toys: http://courses.ncssm.edu/gallery/collections/toys/opticaltoys.htm

SONY BRAVIA ZOETROPE

http://www.creativereview.co.uk/cr-blog/2009/february/the-sony-bravia-zoetrope

3D ZOETROPE

Pixar's 3D Zoetrope at Disneyland California
Video: https://www.youtube.com/watch?v=5khDGKGv088

OVERVIEW OF CG ANIMATION

Storyboards + concept art
Recording
Modelling
Shading
Lighting
Images © Pixar: www.pixar.com

© Rick Parent
MODELING VS. ANIMATION

Modeling: What are the parameters?
Rendering: How to display the model?
Animation:
- How do the parameters change? / How does the model change?
- Motion synthesis

TYPES OF ANIMATION

- Scripted
  - Keyframes + interpolation
- Data-driven
  - Performance capture, human input
- Procedural
  - Synthesised, rule-based, automated, behavioural, heuristics, physically based

SCRIPTED ANIMATION

SCRIPTED ANIMATION

Specify key frames
Do "in-betweening"

KEY FRAME ANIMATION

KEY frame’s drawn by lead artist
In-between frames filled in

In Computer Animation:
- automatically in-betweening by various interpolation methods.
- Due to re-usable data, we can interpolate:
  - Frames / Images
  - 3D data / 3D transforms
  - Other Parameters

INTERPOLATION

In CG Animation, in-betweening is done by interpolation

Keyframe 1
Keyframe 2

Given known points (e.g. start and end point in a line), INTERPOLATION is the process of mathematically estimating points that lie in between.
**EXAMPLE: LINEAR INTERPOLATION**

\[ x = (1 - t)x_{\text{START}} + tx_{\text{END}} \quad y = (1 - t)y_{\text{START}} + ty_{\text{END}} \]

This is a parametric equation of points on a line (but we show only use one point at a particular time):

- \( t = 0 \): \( x = (1 - 0)x_{\text{START}} + (0)x_{\text{END}} = x_{\text{START}} \)
- \( t = 0.25 \): \( x = (1 - 0.25)x_{\text{START}} + (0.25)x_{\text{END}} = 0.75x_{\text{START}} + 0.25x_{\text{END}} \)
- \( t = 0.5 \): \( x = (1 - 0.5)x_{\text{START}} + (0.5)x_{\text{END}} = 0.5x_{\text{START}} + 0.5x_{\text{END}} \)
- \( t = 1 \): \( x = (1 - 1)x_{\text{START}} + (1)x_{\text{END}} = 0 \cdot x_{\text{START}} + x_{\text{END}} = x_{\text{END}} \)

**SAMPLE CODE:**

```c
for (t=0; t<1; t=t+0.1) {
    x = (1 - t)*x_start + t*x_end;
    y = (1 - t)*y_start + t*y_end;
    Move_object_to(x, y);
}
```

**INTERPOLATION**

Simplest type of interpolation is Linear Interpolation

\[ x = (1 - t)x_{\text{START}} + tx_{\text{END}} \quad y = (1 - t)y_{\text{START}} + ty_{\text{END}} \]

**INTERPOLATION**

Piece-wise Linear Interpolation

\[ x = (1 - t)x_{\text{START}} + tx_{\text{END}} \quad y = (1 - t)y_{\text{START}} + ty_{\text{END}} \]

**INTERPOLATION**

Piece-wise Linear Interpolation

**INTERPOLATION**

Spline Interpolation
Spline Interpolation

Translations, rotations applied to geometric models... in 3D

Essentially, each point is transformed (move, rotate, etc.) by the same operation by a discrete amount for each in-between frame. Can also apply to Lights, Camera etc.

Rigid Hierarchies Each “part” is a rigid transform (all points are transformed by same amount). Each part has its “parent’s” transforms plus it’s own.

Points on the mesh have different transforms (move differently relative to each other). However each point is still interpolated between start and end pose.

DATA-DRIVEN ANIMATION

Capture each frame of animation.
**INDIRECT INPUT**

Similar to Animatronics / puppetry
- Simple input is abstracted and applied to drive a characteristically different motion.
- E.g. hand motions used to drive eyes and mouth
- Also relevant in Graphics

Other forms include
- Lip-synching applications
- High-level “squad” commands
- Voice-driven interfaces

---

**ROTSOCOPED ANIMATION**

Rotoscopy + interpolation

---

**REAL-TIME VIDEO ABSTRACTION**

Wimmeijer, Olsen and Gooch (2006)

---

**MOTION CAPTURE**

From "Brief History of Motion Capture for Computer Character Animation"

---

**SKINNING**

Underlying "skeletal" mesh is rigid. Skin moves with bones.
WHEN TO USE MOTION CAPTURE?

Good for creating realistic complicated character motion
But some motions are easily done by hand
Artist created motions could be more expressive / have better emotional impact

Drawbacks:
- Some objects are hard to track
- Actors required
- Equipment for professional quality capture is somewhat expensive

PROCEDURAL ANIMATION

“Mathematical” Models of Motion

The change in position and orientation of an object (or parts of an object) determined by a set of functions/rules.

SAMPLE CODE:
Let \( px, py \) be position of the particle
\( vx, vy \) be velocity of the particle

Animation Loop (do this every frame):
Update position using velocity
\( px = px + vx \)
\( py = py + vy \)
Update using velocity using gravity
\( vx = vx; \)
\( vy = vy - 9.81; \)
Collisions (if object falls through the floor, bounce it back up)
\( if (py < 0) \)
\( vy = vy * -1; \)

PARTICLE ANIMATION

Simple rules applied to very simple individual objects

Collectively can result in useful emergent behaviours

SAMPLE CODE:
Let \( px, py \) be position of the particle
\( vx, vy \) be velocity of the particle

Animation Loop (do this every frame):
Update position using velocity
\( px = px + vx \)
\( py = py + vy \)
Update using velocity using gravity
\( vx = vx; \)
\( vy = vy - 9.81; \)
Collisions (if object falls through the floor, bounce it back up)
\( if (py < 0) \)
\( vy = vy * -1; \)

BEHAVIOURAL ANIMATION

Semi-intelligent rules given to control the motion of "agents" e.g.
- Run away from lions
- Avoid obstacles
- Avoid other wildlife

Bird-oids
- Useful for modelling animal behaviour
- Flocks, herds and so on

“BOIDS”

Each boid maneuvers based on the positions and velocities of nearby flock-mates

Three basic types of steering behaviours
- Separation: avoid crowding
- Alignment: steer towards average heading
- Cohesion: move towards average position

Each boid ‘senses’ only within local spherical neighbourhood of itself

PHYSICALLY-BASED ANIMATION

Rules of Physics used to calculate motion
- Collisions
- Forces
- Friction
- Gravity
- Buoyancy
- Elasticity

SOME “COMBO” TECHNIQUES

Motion capture data may sometimes be used to animate a character between two goal states (scripted key-frames)

Physically plausible animation can be procedurally generated between a start and end state

Motion capture data can be re-timed/re-targeted to fit certain prescribed physical conditions.

Available mo-cap data might not exactly suit our needs we need tailor it to specific target frames.

Captured/rotoscoped data can be "stylised"

WHY CG ANIMATION?

One medium of many:
- Live action, classical animation, clay, puppets, ...

3D has aspects of live film
- Camera, cinematography

CG has aspects of classical animation:
- Stylization, fantasy, scripted motion, ...

Accessible
- Computer editing,
- Don’t need to act, don’t need to draw.
- Need to understand acting, need to understand images

REAL-TIME VS. OFFLINE ANIMATION

Real-time
- Interactive

Offline Animation
COMPUTER ANIMATION

- Computer Animation
- Computer Assisted Animation

REFERENCES

RICK PARENT – Computer Animation: Algorithms and Techniques
http://nuclear.dnsalias.com/books/parent-computer_animation.pdf