

## Notes

---

- New course URL:  
<http://www.cs.ugrad.ubc.ca/~cs426>
- Small typo in assignment o (fixed now)
  - If you can't hand in Monday in class or office hours, slip under my office door before **Tuesday** morning
- Next week, I will be out Thursday and Friday (September 16,17)
  - No office hours
  - No lecture
- Nobody come near the laptop! :-)

## Classic Principles

---

- Squash and Stretch
- Timing
- Anticipation
- Staging
- Follow-Through and Secondary Motion
- Overlapping Action and Asymmetry
- Slow In and Slow Out
- Arcs
- Exaggeration
- Appeal
- Straight-Ahead and Pose-to-Pose

## Animation Principles

---

- Disney and co. developed certain principles (starting in the 1930's) for making good animation
  - Fluid, natural, realistic motion
  - Effective in telling the story
- Developed for traditional 2d cel animation, but equally applicable to all sorts of animation
- This course is mostly about the underlying technology for computer animation, but these are still important to have in mind

## Squash and Stretch

---

- Rigid objects look robotic---let them deform to make the motion more natural and fluid
- Accounts for physics of deformation
  - Think tennis ball...
  - Communicates to viewer what the object is made of, how heavy it is, ...
  - Usually large deformations conserve volume: if you squash one dimension, stretch in another to keep mass constant
- Also accounts for persistence of vision
  - Fast moving objects leave an elongated streak on our retinas



## (squash and stretch cont'd)

---

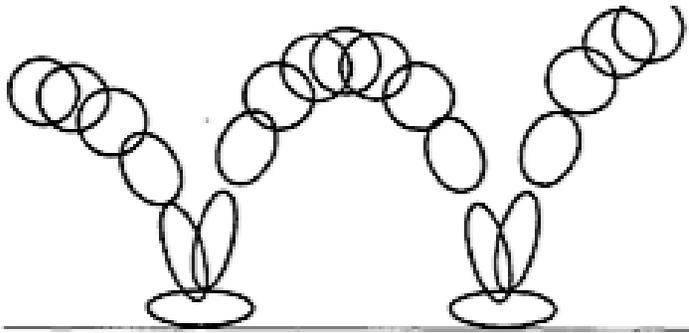


FIGURE 2. Squash & stretch in bouncing ball.

## Timing

---

- Pay careful attention to how long an action takes -- how many frames
- How something moves --- not how it looks --- defines its weight and mood to the audience
- Also think dramatically: give the audience time to understand one event before going to the next, but don't bore them

## Anticipation

---

- The preparation before a motion
  - E.g. crouching before jumping, pitcher winding up to throw a ball
- Often physically necessary, and indicates how much effort a character is making
- Also essential for controlling the audience's attention, to make sure they don't miss the action
  - Signals something is about to happen, and where it is going to happen

## Staging

---

- Make the action clear
- Avoid confusing the audience by having two or more things happen at the same time
- Select a camera viewpoint, and pose the characters, so that visually you can't mistake what is going on
  - Clear enough you can tell what's happening just from the silhouettes (highest contrast)

## Follow-Through and Secondary Motion

---

- Again, physics demands follow-through -- the inertia that's carried over after an action
  - E.g. knees bending after a jump
  - Also helps define weight, rigidity, etc.
- Secondary motion is movement that's not part of the main action, but is physically necessary to support it
  - E.g. arms swinging in jump
- Just about everything should always be in motion - "moving hold"
- Animator has to give the audience an impression of reality, or things look stilted and rigid

## Slow In and Out

---

- Also called "easing in" and "easing out"
- More physics: objects generally smoothly accelerate and decelerate, depending on mass and forces
- Just how gradual it is helps define weight, mood, etc.
- Also helpful in emphasizing the key frames, the most important or "extreme" poses
  - Character spends more time near those poses, and less time in the transition
  - Audience gets better understanding of what's going on

## Overlapping Action and Asymmetry

---

- Overlapping action: start the next action before the current one finishes
  - Otherwise looks scripted and robotic instead of natural and fluid
- Asymmetry: natural motion is rarely exactly the same on both sides of the body, or for 2+ characters
  - People very good at spotting "twins", synchronization, etc.
  - Break up symmetries to avoid scripted or robotic feel

## Arcs

---

- Natural motions tend not to be in straight lines, instead should be curved arcs
  - Just doing straight-line interpolation gives robotic, weird movement
- Also part of physics
  - gravity causes parabolic trajectories
  - joints cause circular motions
  - etc.
- Keep motion smooth and interesting

## Exaggeration

---

- Obvious in the old Loony Tunes cartoons
- Not so obvious but necessary ingredient in photo-realistic special effects
- If you're too subtle, even if that is accurate, the audience will miss it: confusing and boring
- Think of stage make-up, movie lighting, and other "photo surrealistic" techniques
- Don't worry about being physically accurate: convey the correct psychological impression as effectively as possible

## Straight Ahead vs. Pose-to-Pose

---

- The two basic methods for animating
- Straight Ahead means making one frame after the other
  - Especially suited for rapid, unpredictable motion
- Pose-to-Pose means planning it out, making "key frames" of the most important poses, then interpolating the frames in between later
  - The typical approach for most scenes

## Appeal

---

- Make animations that people enjoy watching
- Appealing characters aren't necessarily attractive, just well designed and rendered
  - All the principles of art still apply to each still frame
  - E.g. controlling symmetry - avoid "twins", avoid needless complexity
- Present scenes that are clear and communicate the story effectively

## Extremes

---

- Keyframes are also called extremes, since they usually define the extreme positions of a character
  - E.g. for a jump:
    - the start
    - the lowest crouch
    - the lift-off
    - the highest part
    - the touch-down
    - the lowest follow-through
  - The frames in between ("inbetweens") introduce nothing new---watching the keyframes shows it all
  - May add additional keyframes to add some interest, better control the interpolated motion

## Computer Animation

---

- The task boils down to setting various animation parameters (e.g. positions, angles, sizes, ...) in each frame
- Straight-ahead: set all variables in frame 0, then frame 1, frame 2, ... in order
- Pose-to-pose: set the variables at keyframes, let the computer smoothly interpolate values for frames in between
- Can mix the methods:
  - Keyframe some variables (maybe at different frames), do others straight-ahead

## Splines and Motion Curves

---

## Layering

---

- Work out the big picture first
  - E.g. where the characters need to be when
- Then layer by layer add more details
  - Which way the characters face
  - Move their limbs and head
  - Move their fingers and face
  - Add small details like wrinkles in clothing, hair, ...

## Motion Curves

---

- The most basic capability of an animation package is to let the user set animation variables in each frame
  - Not so easy --- major HCI challenges for designing an effective user interface
  - We'll ignore these issues though
- The next is to support keyframing: computer automatically interpolates in-between frames
- A motion curve is what you get when you plot an animation variable against time
  - Computer has to come up with motion curves that interpolate your keyframe values

## Splines

---

- Splines are the standard way to generate a smooth curve which interpolates given values
- A spline curve (sometimes just called spline) is just a piecewise-polynomial function
  - Split up the real line into intervals
  - Over each interval, pick a different polynomial
- If the polynomials are small degree (typically at most cubics) it's very fast and easy to compute with

## Knots and Control Points

---

- The ends of the intervals, where one polynomial ends and another one starts, are called “knots”
- A control point is a knot together with a value
- The spline is supposed to either interpolate (go through) or approximate (go near) the control points

## Hermite Splines

---

- Hermite splines have even richer control points: as well as a function value, a slope (derivative) is specified
  - So the Hermite spline interpolates the control values and must match the control slopes at the knots
- Particularly useful for animation---more control over slow in/out, etc.