Animation

Animation is the process of giving motion to your geometric models. Before animating, there are questions you need to ask first:

- Why am I doing this?
- Do I want the animation to obey the real laws of physics? Partially? Which elements?
- Am I willing to “fake” the physics to get the objects to move in a way that I tell it?
- Do I have specific key positions I want the objects to pass through no matter what?
- Do I want to simply record the motion of a real person, animal, etc., and then play it back?

Keyframe Animation

These icons refer to explanatory videos on the class web site.
Here's Some Code that Lets You Create DIY Keyframe Animations

Instead of Key-frames, I like specifying Key Times better. And, so, I created a C++ class to do it for you.

class Keytimes:

  void AddTimeValue(float time, float value);
  float GetFirstTime();
  float GetLastTime();
  int GetNumKeytimes();
  float GetValue(float time);
  void PrintTimeValues();

Instead of Key Frames, I Like Specifying Key Times Better

Instead of Key-frames, I like specifying Key Times better. Here it wants to interpolate and animate the x-location of something.

int main(int argc, char *argv[])
{
  Keytimes Xpos;
  int msec = glutGet(GLUT_ELAPSED_TIME)
  % MSEC;
  float nowTime = (float)msec / 1000.;
  glPushMatrix();
  glTranslatef(Xpos.GetValue(nowTime),
  Ypos.GetValue(nowTime),
  Zpos.GetValue(nowTime));
  glRotatef(ThetaX.GetValue(nowTime), 1., 0., 0.);
  glRotatef(ThetaY.GetValue(nowTime), 0., 1., 0.);
  glRotatef(ThetaZ.GetValue(nowTime), 0., 0., 1.);
  // draw the object
  glPopMatrix();
}

Using the System Clock in Display( ) for Timing

#define MSEC 10000 // i.e., 10 seconds
Keytimes Xpos, Ypos, Zpos,
Keytimes ThetaX, ThetaY, ThetaZ;

if(AnimationIsOn )
{
  if( # msec into the cycle (0 - MSEC-1 ) )
  // turn that into a time in seconds:
  float nowTime = (float)msec / 1000.;
  ...
Forward Kinematics:
Change Parameters – Connected Things Move
(All children understand this)

Inverse Kinematics (IK):
Things Need to Move to a Particular Location – What Parameters Will Make Them Do That?

Of course, there will always be target locations that can never be reached.
Think about that spot in the middle of your back that you can never scratch! 😊

Forward Kinematics solves the problem “If I know the link transformation parameters, where are the links?”.
Inverse Kinematics (IK) solves the problem “If I know where I want the end of the chain to be \((X^*, Y^*)\), what transformation parameters will put it there?”.
Particle Systems: A Cross Between Modeling and Animation?

The basic process is:

- Emit
- Random Number Generator
- Display
- Update
Particle Systems Examples

The Lion King (2019) — Disney

A Particle System to Simulate Colliding Galaxies in Cosmic Voyage

Particles Don’t Actually Have to Be “Particles”
Newton's second law:

\[ \text{force} = \text{mass} \times \text{acceleration} \]

or

\[ \ddot{x} = \frac{\text{force}}{\text{mass}} \]

In order to make this work, you need to supply physical properties such as mass, center of mass, moment of inertia, coefficients of friction, coefficients of restitution, etc.

\[ x(T) = \int_{t=0}^{T} \ddot{x} \, dt \approx \sum \ddot{x} \Delta t \]
Functional Animation: Make the Object Want to Move Towards a Goal Position

\[ m\ddot{x} + c\dot{x} + kx = 0 \]
**Functional Animation:**
While Making it Want to Move Away from all other Objects

\[ m\ddot{x} = \sum F_{\text{repulsive}} \]

- **Repulsion Coefficient**
- **Distance between the boundaries of the 2 bodies**
- **Repulsion Exponent**

**Total Goal** – Make the Free Body Move Towards its Final Position
While Being Repelled by the Other Bodies

\[ m\ddot{x} + c\dot{x} + kx = \sum F \]

**Increasing the Stiffness**
Stiffness = 3, Stiffness = 6, Stiffness = 9

**Increasing the Repulsion Coefficient**
Repulse = 10, Repulse = 30
Functional Animation

Motion Capture ("MoCap") as an Input for Animation

Motion Capture is for Faces Too

Even Animals can be MoCapped

My cat would never have put up with this…

https://www.youtube.com/watch?v=zyg_LQrHpoo
Tron I –
They probably should have used physics, but didn’t

Card Trick

Pixar Animated Shorts