CS6491-2015 P2: Swirl

Dingtian Zhang, Clement Julliard

Sep 18, 2015

1 Objective

We want to compute the mathematical formulation and the implementation of Steady Interpolating Similarity $SIS(F_a, t, F_b)$. and validate our solution by starting with user controlled frames F_0 and F_1, by extrapolating a steady pattern of frames F_i such that the transformation T between F_i and F_{i+1} is independent of i, and by showing (visually) that $F_i = SIS(F_0, i/k, F_k)$ for all i.

2 Definitions of Entities

A 3D frame $\{I, J, K, O\}$ is a similarity frame if all its factors have the same magnitude, if each is orthogonal to the other two, and if $(I \times J) \cdot K > 0$.

A Similarity Motion is an similarity frame $F(t)$ parameterize by time.

The Steady Interpolating Similarity $SIS(F_a, t, F_b)$ is a Similarity Motion, such that:

- SIS(Fa,0,Fb)= Fa,
- SIS(Fa,1,Fb)= Fb,
- And the similarity transformation from SIS(Fa,t,Fb) to SIS(Fa,t+u,Fb) is independent of t,

Figure 1 shows some of the 3D swirls in nature and man-made structures.

3 Approach

Steady interpolating similarity motion is a combination of rotation around a fixed axis, translation along that axis, and scaling. There are many kinds of SIS motions, where rotation, translation, and scaling are different combinations. For example we have,

(a) 

Figure 1: (a) Swirl in nature

3.1 Screw Motion

Screw motion is the combination of rotation and translation. Assume the rotation is around axis (F, N), initial point P_0, unit time rotation angle α, speed along the axis is constant d, we can describe the screw motion as:

$$P(t) = F + FP_0^o(t\alpha) + dtN \tag{1}$$

Figure 2: Art of Screw

3.2 Spiral Motion

Spiral motion is the combination of rotation and scaling. One common spiral motion is log spiral. Assume the the rotation is around axis (F, N), initial point P_0, unit time rotation angle α, scaling factor is exponential of m, we can describe the log spiral motion as:

$$P(t) = F + m^t F P_0^o(t\alpha) \tag{2}$$

Figure 3: Art of Spiral

3.3 Swirl Motion

Swirl motion is the combination ("commutatie composition") of log spiral motion around an axis and exponential translation along the axis. Assume the the rotation is around axis (F, N), initial point P_0, unit time rotation angle α, scaling factor is exponential of m, velocity along the axis is exponential of d, we can describe the log spiral motion as:

$$P(t) = F + m^t F P_0^o(t\alpha, N) + d^t N \tag{3}$$

Figure 4: Art of Swirl

4 Implementation

4.1 Computing N

For computing the N from similarity frames $F_0(O_0, I_0, J_0, K_0)$ and $F_1(O_1, I_1, J_1, K_1)$, we can derive N as

$$N = \frac{\Delta I \times \Delta J + \Delta J \times \Delta K + \Delta K \times \Delta I}{3} \tag{4}$$

where $\Delta I = I_1 - I_0$, $\Delta J = J_1 - J_0$, $\Delta K = K_1 - K_0$.

Note that we only derived the vector ot N, without knowing which point the axis N pass through. We can project F_0 and F_1 on any plane perpenticular to N, thus the question becomes to solve the center F' of a spiral motion, which N will pass through.

4.2 Computing F

For computing the F from F_0 and F_1, suppose we are given the input the same as above, we get

$$A = F + FA \tag{5}$$

$$D = F + FA^o(t\alpha, N)m^t + d^t N \tag{6}$$

Therefore we can derive F by

$$A^o(t\alpha, N)m^t + d^t N - D = F^o(t\alpha, N)m^t - F \tag{7}$$

where

$$d^t = FD \cdot N - FA \cdot N \tag{8}$$

$$m^t = \frac{FD - (FD \cdot N)N}{FA - (FA \cdot N)N} \tag{9}$$

And rotating an arbitary point around an axis can be done by ([1]):

(1) Translate space so that the rotation axis passes through the origin.
(2) Rotate space about the z axis so that the rotation axis lies in the xz plane.
(3) Rotate space about the y axis so that the rotation axis lies along the z axis.
(4) Perform the desired rotation by about the z axis.
(5) Apply the inverse of step (3).
(6) Apply the inverse of step (2).
(7) Apply the inverse of step (1).

4.3 Computing F_i from F_0 and F_1 in 3D

If we know F, F_0, and F_1, we know the swirl motion description $P(t) = F + m^t F P_0^o(t\alpha, N) + d^t N$, so we can interpolate each frame simply by changing t to $2t$, $3t$, …, kt.

5 Analysis

Our program has not been able to function correctly before the deadline. We merged Frame and Swirl into one program and implemented finding axis and the fixed point, but were not successful in animating it yet.

References

[1] Glenn Murray, *Rotation About an Arbitrary Axis in 3 Dimensions*, http://inside.mines.edu/fs_home/gmurray/ArbitraryAxisRotation/.