Retargetting Motion to New Characters

Project Report
Xin Li
Recall that Motion retargetting is to adapt an animated motion from one character to another, independently of how the motion was created.

Example of retargetting

- A constrained optimization problem
- Spacetime constraints: Meet all the requirements over frames
- Objective function
- Minimized energy

1). Examples from the paper of Michael Gleicher: Retargetting Motion to New Characters (1998)
The goal of the project is to get the matrix data transformed into quaternion and doing various manipulation on it.

- **Rotation matrices** => **Quaternion**
  - **Interpolation** to slow the motion
    - Using slerp interpolation
    - Motion are slowed down as the frames are interpolated
- **Change the Motion**
  - Limb length
  - Motion parameters: strike length etc
- **Quaternion** => rotation matrices

<table>
<thead>
<tr>
<th>Formula</th>
<th>Formula</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2<em>qy²-2</em>qz²</td>
<td>2<em>qx</em>qy-2<em>qz</em>qw</td>
<td>2<em>qx</em>qz+2<em>qy</em>qw</td>
</tr>
<tr>
<td>2<em>qx</em>qy+2<em>qz</em>qw</td>
<td>1-2<em>qx²-2</em>qz²</td>
<td>2<em>qy</em>qz-2<em>qx</em>qw</td>
</tr>
<tr>
<td>2<em>qx</em>qz-2<em>qy</em>qw</td>
<td>2<em>qy</em>qz+2<em>qx</em>qw</td>
<td>1-2<em>qx²-2</em>qy²</td>
</tr>
</tbody>
</table>
The transforms and changes of the data are done in MATLAB, by manipulating the quaternion.

### What is quaternion

- Quaternion \( q = w + xi + yj + zk \)
- Represents a point on a hypersphere in 4-D space
  - Exists a unique shortest path from one point \( q_a \) to point \( q_b \)
  - Given a rotation matrix \( m \), the transfer function to a quaternion \( q = [w \ x \ y \ z] \) is:
    \[
    qw = \sqrt{\frac{1+m_{00}+m_{11}+m_{22}}{2}}
    \]
    \[
    qx = \frac{(m_{21} - m_{12})}{4 * qw}
    \]
    \[
    qy = \frac{(m_{02} - m_{20})}{4 * qw}
    \]
    \[
    qz = \frac{(m_{10} - m_{01})}{4 * qw}
    \]
Slerp is shorthand for spherical linear interpolation, introduced by Ken Shoemake in the context of quaternion interpolation for the purpose of animating 3D rotation.

Slerp Interpolation

- The certain type of interpolation using on Quaternion
- Constant speed motion along a unit radius great circle arc, given the ends and an interpolation parameter between 0 and 1.

For two quaternion $q_0$ and $q_1$, $\cos \Omega = p_0 \cdot p_1$, $0 \leq t \leq 1$.

$$\text{Slerp}(p_0, p_1; t) = \frac{\sin[(1-t)\Omega]}{\sin \Omega} p_0 + \frac{\sin[t\Omega]}{\sin \Omega} p_1$$
The first step Result is as the following, motion is interpolated from N frames to 2N frames, with the motion changes smoothly.

Motion interpolation

- One frame is add between every two frames and simply perform “Half” of the motion, so it looks like a ½ speed slow motion.
Then is to change limb length, which means motions have to be changed accordingly to fit the new segment lengths.

Motion interpolation

Original Motion of one huge step

Adding one frame and change the step length

- One frame is breaking into two frames so that the step length is shorter, the objective of this process is to retarget the motion to a smaller character with shorter leg, who can not make such a huge step as the original one.
To change the step length

Interpolate the leg quaternion \( q_l \) with the Identity-rotation quaternion

\[ q = [1 \ 0 \ 0 \ 0] \iff \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \]

Matrices are coded by absolute values

Find the relative change matrix \( dR \)

Apply \( dR \) to the motion changes

For two joint segments \( J_1 \) and \( J_2 \), their rotation matrices are \( R_1 \) and \( R_2 \):

\[ R_2R_1^T = dR \]

Xin LI - Motion Retargeting
What this project is hopefully to get done is to set up the constraints according to the change of the limbs

- Quaternion is a convenient format of modifying motions in 3-D space
- When changing limb lengths
  - When the lengths of the segments are changed, the motion may looks weird like a short person walking in steps that are too large
  - Need to change motion parameters accordingly
  - Set up constraints to make sure feet are on the ground!
- Others …

http://www.youtube.com/watch?v=whNGvrctxwE