Computer Animation

Lecture 6.

Motion Editing

Taku Komura
Overview

• Motion Warping
• Motion Blending
• Foot sliding
• Inverse Kinematics
• Spacetime constraints
• Motion Retargeting
• Motion Graphs
In the previous lectures...

- We learnt about motion capture
- It can provide realistic movements for characters
- However,...
Animators are not so happy with the raw motion capture data

• The style of motions are not stylised
  – Want to exaggerate some expressions

• The character’s body is different from those of humans
  – Penetrations, foot sliding
  – Need to “retarget” the motion to the character size

• There might be obstacles in the scene – the character needs to avoid them while walking
Editing motions

–So need an efficient method to edit the trajectories of the body
–So that they satisfy constraints
–Appear more stylized
Motion Warping

• Edit the captured motion a little bit so that it satisfies the requirements
  – Effective for changing the location the hand or the foot passes
Basic idea

- Adding offset to the data so the constraint is satisfied

Warped Motion = original motion + offset

Offset can be a simple 1D motion
Motion Warping

• Add an offset to the trajectories to edit the motion
  1. Determine the duration the motion should be edited
  2. Insert a keyframe so that the motion satisfies the requirements
Motion Warping

- Very simple
- Good for avoiding obstacles
- Not so good for constraining body segments
  - Fixing the position of the support foot
  - Specifying the 3D trajectory of the end effectors

http://www.youtube.com/watch?v=BzfxGwO7swg
http://www.youtube.com/watch?v=Ro30uWsWI_s
Motion Blending

- Given two different motions A and B
  Blended Motion = \( A \ast (1-s) + B \ast s \) \((0<s<1)\)
- How can we use this?
  - Generate a motion inbetween
  - Exaggerating the motions by extrapolation \((s>1, s<0)\)
  - Gradually shift from motion A to motion B
  - Concatenating two motions – blending the two ends by gradually shifting \(s\) from 0 to 1

http://www.youtube.com/watch?v=Uha81cm5fRw
The problems when blending motions

- The durations of motions are different
- The timing of some events are different
  - Suppose we blend two walking motions
  - The timing of steps are different
  - Need to synchronize the motions
Dynamic Time Warping (DTW)

- Synchronizing the motions by finding the most matching frames
- Compute the distance of every frame in motion 1 and motion 2 (using the equation used in motion graph)
- Create a similarity matrix of all frames
- Find the optimal time alignment that minimizes the total distance between matched frames

http://www.youtube.com/watch?v=EL7ARaH5jHU
Creating a motion inbetween / exaggerating the motion by extrapolation

- Generating a new motion by interpolating multiple motions
  - A tired walking motion \( s \times (1-s) \) ordinary walking motion = A little bit tired motion \( (0 < s < 1) \)
  - A very tired motion by setting \( s > 1.0 \)
  - Cheerful motion by setting \( s < 0 \)

http://www.youtube.com/watch?v=yQIvwWQTr-I
http://www.youtube.com/watch?v=PkrCP68TI4k
Foot Sliding

• We usually blend the motions in the level of generalized coordinates (root position, root orientation and joint angles)
  – No information about contacts
• The supporting foot/feet slides when the motion is simply blended

http://www.youtube.com/watch?v=KumgRt895EU
Inverse Kinematics

• Editing the postures/motions by specifying the 3D location of the avatars
• We can use IK to solve the problem of foot sliding
• Edit the postures, frame by frame, and drag the foot to the original position
Problems with IK for motion editing

• IK is a frame-based method
• Say we want the avatar to hold the door knob at frame $i$
• If we just edit the posture at frame $i$, there will be discontinuity in the motion
• We don’t want the avatar to suddenly move the hand to the door knob at time $t$, but prepare in advance
• There might be other constraints such as
  • putting the hand in a pocket to look for the key before opening the door
  • dragging the hand of the child while approaching the door
  • Different constraints at different timing
• Any method to handle all these constraints together?
Spacetime constraints

- Instead of editing the motion frame-by-frame, here we edit the whole motion at once.
- We add displacement to the whole motion so that the constraints are satisfied while the motion is smooth.
- We want the displacement to be small, so minimize the integral of displacement.
- We may reduce the acceleration to avoid abrupt changes.

\[
\begin{aligned}
\text{a given motion} & \quad m_0(t) \\
\text{an unknown motion} & \quad m(t) = m_0(t) + d_{\text{dis}}(t) \\
\text{a set of constraints} & \quad f_i(q^{fi}) = c_i \quad i = 1...k \\
\text{a function to be minimized} & \quad g(m) = \int_t d_{\text{dis}}(t)^2
\end{aligned}
\]
Motion Retargeting

• Use the captured motion for a character with a different body size
• Captured motion (adult) -> apply it for a child character
• Need to edit the motion while satisfying the constraints
  – the support must be landed on the ground,
  – The segments must not penetrate the other parts of the body

We can use IK/spacetime constraints to solve this problem

http://www.youtube.com/watch?v=x4Dl6xJs-FY
Data driven Inverse Kinematics

- Finding similar sample postures in the database and apply DTW
  - K-nearest neighbors \((p_1,p_2,\ldots,p_k)\) : parameters \((q_1,q_2,\ldots,q_k)\) : generalized coordinates
- Blend these postures to generate the posture \(q\) that brings the end effector to the desired position \(p\)
- The weights are inverse-proportional to the distance between the sample and the desired position
  \[ q = w_1 q_1 + w_2 q_2 + \ldots + w_k q_k \]

- Kovar and Gleicher SIGGRAPH ‘04

http://www.youtube.com/watch?v=EL7ARaH5jHU
Summary of Motion Editing

• Animators cannot immediately use captured motions
• Need techniques to edit the motion
  – Keyframe animation
  – Motion warping
  – Motion Blending
  – Inverse kinematics
  – Spacetime Constraints
  – Motion Retargeting
Interactive Character Control?

- We want to enable users to interactively control characters
- We want the movements to be continuous
  - Even in modern computer games, the motions are sometimes not continuous
  - Sliding feet, sudden fast movements
  - We want to observe a natural transition from one motion to another
Motion Graphs

- A data-driven approach that enables interactive control of the avatar
- A directed graph structure whose
  - nodes represent the postures, and
  - edges represent short motion clips
Demo movie of Motion Graph

http://www.youtube.com/watch?v=5i8UhW_ExoQ
The procedure is fully automatic.

1. Capture the motion data - The actor moves his/her body in arbitrary way.
2. Similar postures are searched in the data, and if they are similar, they are connected by an edge.
Unstructured Input Data

A number of motion clips
• Each clip contains many frames
• Each frame represents a pose
Unstructured Input Data

Connecting transition
• Between similar frames
Similarity of Postures

• There can be two ways
  – Comparing the joint angles (generalized coordinates) and angular velocities
  – Comparing the 3D location of the joints and their velocities

If the difference is small, the postures are similar
For example, if we are using generalized coordinates

\[ D(i, j) = d(p_i, p_j) + \alpha d(v_i, v_j) \]
Pruning the Transitions

- Some parts of the graph may contain dead-ends
- Some parts of the graph might be isolated from the rest of the graph
  - Connectivity is important
  - We can remove nodes which have low connectivity
Moving along the Motion Graph

- Once the graph is built, continuous nice movements can be obtained when moving along the edges of the graph.
Problems with Motion Graph

Because the system is fully automatic …

• Much duplicated data
• Many postures and motion clips
  – difficult to control the body when the graph is large
• The data is not annotated, so cannot directly specify what the avatar should do at each moment
• Can only replay the captured data
  – No motion synthesis
Adding a High Level Structure to Motion Graphs

- The nodes and edges are labelled and organized
  - left turn, right turn, walk straight, etc.
- The user can select the appropriate edge among those directed towards the same node
- Mainly used for locomotion
  - Walking, running
- Shin et al, Lau et al (SCA 05)

http://youtu.be/V4yRkSvL1Cc
Parametric Motion Graphs

• If there are many similar motions, new motions can be generated by blending them
• For example, a motion to
  – swing a racket in the middle
  can be generated from a motion to
  – Swing the racket to the left and
  – Swing the racket to the right
• Prepare a parameter to blend the motions and produce smooth transitions

http://youtu.be/cExdl2X_ThI
Summary

• Motion graph is a data structure automatically generated from the captured motion data
• Problems with the motion graph
• Hierarchical motion graphs
• Annotated Motion Graphs
• Parametric Motion Graphs
Readings:

• Witkin and Popovic “Motion Warping”, SIGGRAPH 95
• Unuma, Anjyo, Takeuchi “Fourier Principles for Emotion-based Human Figure Animation”, SIGGRAPH ’95
• Kovar, Gleicher, Pighin, “Motion Graphs”, SIGGRAPH ’02
• O Arikan, D Forsyth, “Interactive motion generation from examples”, SIGGRAPH02
• Heck, Gleicher “Parametetric Motion Graphs”, I3D 2007
• **Automated Extraction and Parameterization of Motions in Large Data Sets** Lucas Kovar, M. Gleicher SIGGRAPH 2004