CS7491 Notes: January 6, 2004

Justin Jang and Daniel Sternberg

Administrivia:

Lecture segments will be 3 weeks each.

There will be two scribes to take notes for each class. The scribes will E-mail a PDF file to the whole class and to Jarek within 48 hours of the class. The scribes will collect feedback and E-mail a final PDF file to Jarek within 7 days of the class acknowledging contributions where appropriate.

The class web-page can be found at http://www.gvu.gatech.edu/~jarek/courses/7491

Any E-mails to Prof. Rossignac should have 7491 in the subject line.

Project 0:

Due January 13, 2004.

Personal Project Page (PPP)

The PPP will be a web-page of the following form: The title will be CS 7491. The page will contain your name, a link to your home page (if one exists), a picture ID, your status, advisor and links to four project pages.

E-mail to jarek@cc.gatech.edu your name, student ID#, the URL for the PPP and a password should one be required.

Future projects will be in two pieces. The first is a PDF file detailing the work. The second is a web-page with the details of the work.

Curve Compression:

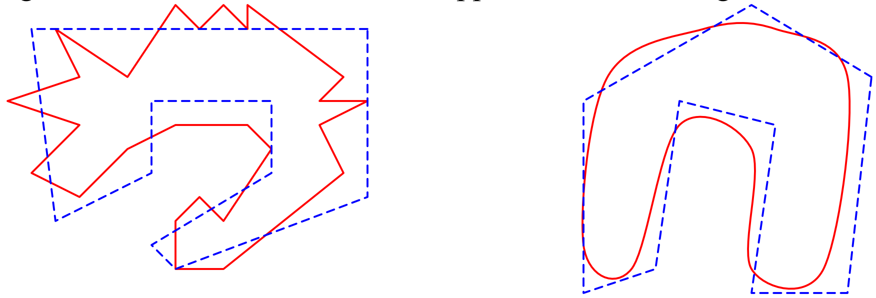
Domain: Types of curves include curves that are represented as scalar functions of time (joint angle evolution for character animation), trajectories over time (motion paths of birds, bees, etc) and point sets (drawings of maps, rivers, pipes, etc).

Motivation: Compress a set of curves to improve the speed of an algorithm or accelerate transmission. When we have too much data, we need compression. We also need efficient “out-of-core” algorithms.

Issues: We need to understand how to measure accuracy and how to trade accuracy for conciseness. We need to be able to measure compression, especially for lossy compression. We need to be able to measure the “complexity” of the curves. This last issue is sometimes ignored through the use of a standard set of curves.

One proposed tool for curve compression was to identify features.

We will focus on a single closed-loop polygonal curve. In the figure, the red curve is the original and the blue dashes are the approximation. How good are these approximations?



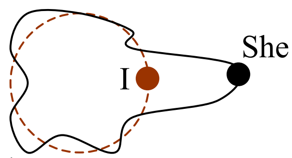
Hausdorff Error:
We want to measure the "maximum deviation" (i.e. Hausdorff distance) between two curves: A and B.

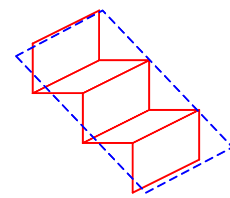
A possible definition:

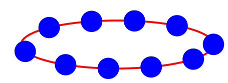
Look for a on A and its closest point b on B and find the pair (a, b) such that d = ||ab|| is biggest. (Note that this is a one-sided/one-way definition. One must find this reversing A and B and take the bigger of the two.)

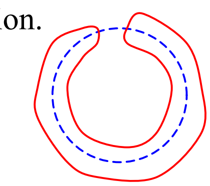
A metaphor:

I love her
She hates me
She picks the set
I must stay on the other one









Note that the Hausdorff distance is a ***bad*** measure of similarity.
There are no concepts of smoothness, continuity, topology, or orientation.

It does have a good property in that
$H(A, B) = 0 \Leftrightarrow A = B$

Another idea:
Use the area of $(R(A) \otimes R(B))$, where $R(A) \otimes R(B) = (R(A) \cup R(B)) - (R(A) \cap R(B))$ where R(A) is the region covered by A.

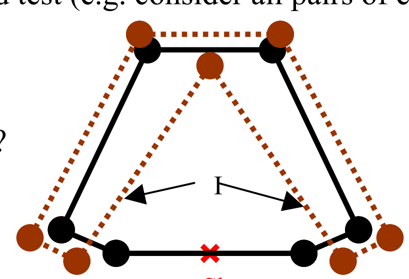
What is a point? Is it a vertex? It is if we quantize the curves.

How are the curves represented? It is a data-structure (table of vertices) with semantics added to it (edges joining consecutive vertices).

The Algorithm:
Generate (a sufficient set of candidates) and test (e.g. consider all pairs of edges).

Conjecture:
She <u>must</u> be on a vertex
Is there a counterexample? Is there a proof?

Counterexample:
She wears a skirt and I wear pants



1. Assume she's on a vertex – Do vertex-edge tests and report results.
2. Assume she's on an edge – Do edge-vertex-vertex, edge-vertex-edge, edge-edge-edge tests and report results.