CS7491 3D COMPLEXITY LECTURE NOTES FOR 1-27-04

Gordon Brown & Tazama St. Julien

Pre-lecture Discussions

1. Compression vs. Simplification

Simplification can be a form of compression, but not vice versa. For example, in 3D a convex hull can be simpler (removed holes), but geometrically it can be more complex.

Lecture

Since the bulk of this lecture was delivered with the aid of a PowerPoint slide show, we decided to scribe only the information which supplements the content of the slides. Many of the slides are self-explanatory, so we have supplemental material below on only a subset of the slides.

Supplemental Material for the First Set of Slides

The slides ("*Compression, simplification, and progressive transmission of 3D models and animations*") are located at http://www.gvu.gatech.edu/~jarek/courses/7491/01-Intro.ppt.

Slide 4: *How should one measure shape **complexity**?*

- Stabbing Number: Number of times a ray (selected from a space of rays) will stab a surface.
- $Area^3/Volume^2$: Measure of how "convoluted" an object is.

Slide 5: ***Storage size** depends on*

- There exists a trade-off between error and storage size.

Slide 7: *Focus on **explicit** representation (T-mesh)*

- Connectivity:
  - Who are you neighbors?
  - How are the organized?
- V(3B+k) = the cost of storage, where V is the number of vertices and B is the number of bits.

Slide 15: *Storage size depends on **accuracy***

- E_B: Quantization Error
- E_T: Simplification Error = K/T (as defined in the slides)
  - Distance between sub-sampled and original shape (Hausdorff distance, for example)

Slide 16: *Different **Error** Measures*

- The measure of error should depend on the problem at hand.
- Sometimes error is best measured in screen space, sometimes in model space.

Slide 18: ***Complexity** of a shape = Storage/Error curve*

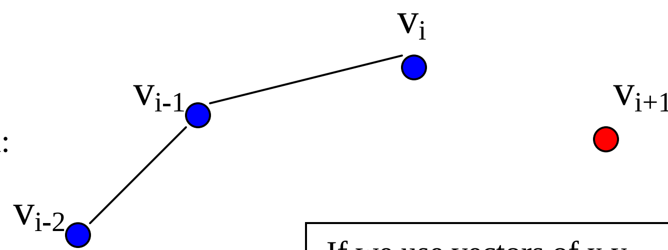
- In order to compare simplification schemes, one must keep in mind the Storage/Error Curve (Simplification and Compression).
- Curve depends on the representation and compression scheme used.

Proposed Prediction Techniques

After the slides were presented, we had two ideas of how to predict the positions of subsequent vertices on a curve.

1) *James' Scheme*

The goal is to predict v_{i+1}.



Remember that 3 points define a parabola:

$y_1 = a\, t_1^2 + bt_1 + c$
$y_2 = a\, t_2^2 + bt_2 + c$
$y_3 = a\, t_3^2 + bt_3 + c$

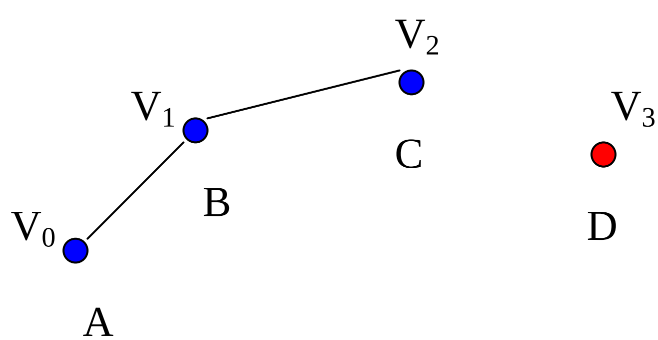
Solve these equations to find values for coefficients:

a= ((y3 – y2)(t2-t1) – (y2-y1)(t3-t2)) / (t3-t2)(t2-t1)(t3-t1)
a= ((y3-y2) – (y2 – y1)) / 2

> If we use vectors of x,y components of points.
> With v_i = C, v_{i-1} = B, v_{i-2} = A.
>
> a = (BC – AB) / 2

2) *Physics scheme*

$v_1 – v_0 = v_2 – v_1$
$(v_2-v_1) – (v_1-v_0) = (v_3–v_2) – (v_2-v_1)$
…simplifies to…
$v_3 = 3(v_2-v_1) + v_0$
…or…
D = 3BC + A



Supplemental Material for the Second Set of Slides

The slides ("*Simple Meshes*") are located at http://www.gvu.gatech.edu/%7Ejarek/courses/7491/02-Meshes.ppt.

Slide 6: *Dual graphs and **spanning trees***

- Think of a triangle spanning tree as a vertex spanning tree of the dual graph.

Slide 7: ***Euler formula** for Simple Meshes*

- The number of triangles in a simple mesh is T = 2V - 4