Fall 2014

CSCI 420: **Computer Graphics**

Homework #2 Tips

Hao Li

http://cs420.hao-li.com

Common Questions

- How do I construct a Catmull-Rom curve?
- What is NTB Space? How do I calculate NTB?
- How do I move the camera?
- How do I draw coaster tracks?

Catmull-Rom Construction

- A simple curve construction is given here: http://www.mvps.org/directx/articles/catmull/
- This link contains ~11000 C++ code snipers for Matrix Multiplication: http://code.ohloh.net/search?s=matrix%20multiplication&browser=Default&pp=0&fl=C%2B%2B&mp=1&ml=1&me=1&md=1&ff=1&filterChecked=true
- BUT, DO NOT BORROW ANY CODE :-)

NTB Space

- 'NTB' is an 'on curve' coordinate system, consisting of orthogonal vectors T (curve tangent), N (curve normal) and B (curve "binormal", obtained via TxN). Creating an NTB coordinate 'frame' at each timestep of your animation is the best way to orient your car along the coaster track (and have it tilt and turn correctly as the track winds and loops around). Such an NTB frame is called a 'Frenet frame'.

- Here is a visualization: http://faculty.evansville.edu/ck6/GalleryTwo/CK_Frenet_Osculating_A.gif

Calculating NTB

- Calculating T is rather easy (via the curve's derivative). N is trickier - but we can start with an initial guess, and use it to get a good starting point. B is simply the cross product of T and N. Only for the very first point in the calculated chain of points, instead of starting with T and N to calculate B, it seems better to start with T and B, and derive N from those (again, via cross product). Once we have a starter set (ie. our first NTB triplet), we could calculate further NTB sets incrementally. Such incremental propagation guarantees that the NTB frame won't wildly 'flip' around at regions of severe normal change (not calculating NTB incrementally, eg. computing it explicitly for each point, could result in this hard-to-fix problem).

- This paper mentions NTB flipping: http://www.dhal.com/downloads/PathsForVolumeSlices.pdf

Moving the Camera

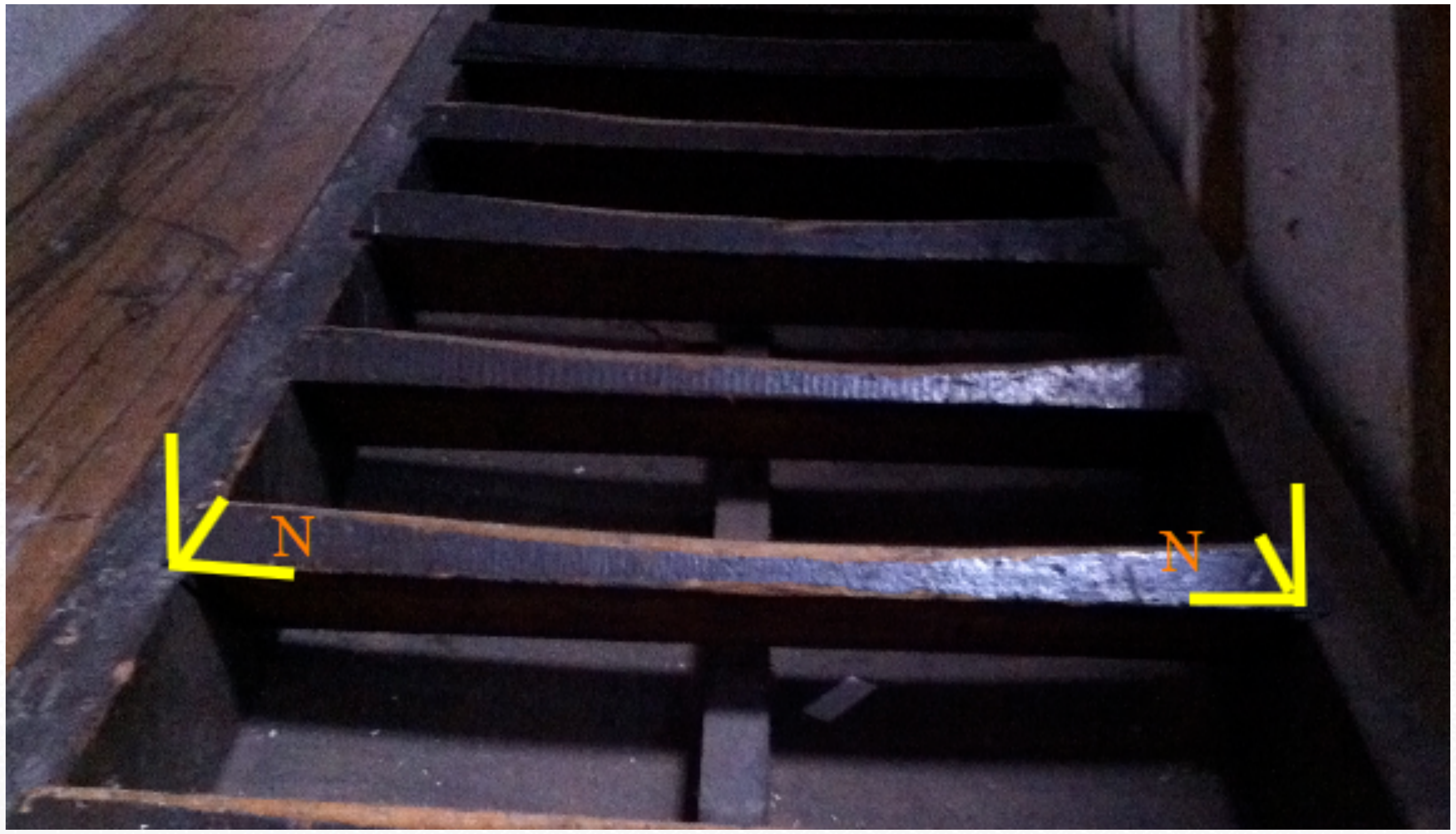
- Like an FPS, the goal is to place the camera on the moving car to get a first-person rider POV. Use these two calls:

gluLookAt - inputs change each frame, based on (animating) NTB

gluPerspective/glFrustum - one-time setup, to specify the lens FOV

Drawing Coaster Tracks

- You need eight vertices to draw a cuboid that spans the two end ("rail") Catmull-Rom curves - four verts (forming a SQUARE cross section) on the left (using the yellow-colored BTN (specifically, B,T) axes, centered at a curve point), and four corresponding verts on the right. Using the 8 verts you can draw 6 quads (2 end squares, 4 middle rectangles) to produce a single 'slat'.



Drawing Coaster Tracks

- How would you form a square at a curve point on either track? Easy:

```
vert0: curvePoint
vert1: curvePoint+T
vert2: curvePoint+T+B
vert3: curvePoint+B
```

- To get the other square, repeat the above on the other side,with the corresponding curvePoint that's across.

Drawing Coaster Tracks

- Given the close spacing of the curve points, it's likely that the slat cuboids would OVERLAP. To fix the problem (ie. to create thin slats that have gaps between them), just scale the squares down appropriately, by identically scaling the T and B vectors using a <1 scale factor (you'll need to find a good value by trial and error):

```
// might need to set 's' bigger/smaller than 0.01..
s = 0.01
vert0: curvePoint
vert1: curvePoint+(s*T)
vert2: curvePoint+(s*T)+(s*B)
vert3: curvePoint+(s*B)
```

- Another problem that can occur (depending on track configuration) is this: in high curvature areas (eg. where drops are situated), your second (offset) track might loop back on itself, ie. create an unwanted pinch/bubble/loop. How would you fix it? Simple - instead of using the normalized N to locate the track (which creates an inter-track spacing of 1 by definition), use a scaled down value, eg. 0.5N.

- Form 8 verts as above, draw 6 squares+rects using them to get a cuboid, do this for all curve point pairs

Texturing

- Each polygon (square or rect.) could have the following texture coords, in the order specified below:

(0,0)
(1,0)
(1,1)
(0,1)

More Resources

- http://www.rcdb.com/ - a big database!!
- http://www.geaugalaketoday.com/Roller%20Coasters/ - tons of photos!
- http://www.visitkingsisland.com/online-fun/webcams - cams near Kings Island coasters