

CPSC 314

Computer Graphics

Dinesh K. Pai
Hierarchies, A1 Spotlight

1

Announcements

- My office hour will move (after this week) to Thursday morning 10-11am. Reason: I have another repeating Faculty of Science meeting scheduled for Thursday afternoons ☹
- Quiz 1 will be handed back later this week. Will discuss solutions in class after handback
- Assignment 2 available soon (probably by tomorrow)

2

Today

- Assignment 1 Spotlight
- Wrap up of transformations

3

A1 Spotlight

- Purpose: to share some interesting student work for Part 2 (“Creative License”)
- This is not meant to represent a “best of” .. i.e., this is not a competition. There are no extra marks for this. It’s just a sample of the great work submitted.
- If your assignment was picked it does not necessarily mean that it wasn’t as interesting. We only have a limited amount of time.

4

Dynamic Deformation with DyRT [James & Pai 02]



Dynamic

physically-based modal deformation

Response

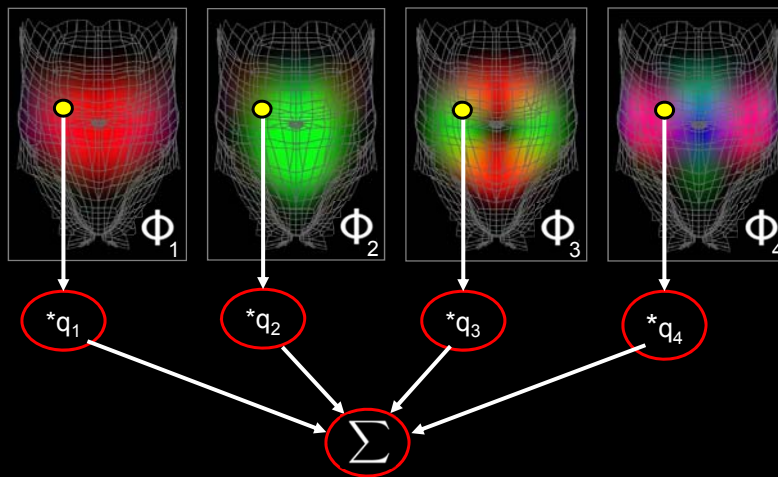
to bone-based animation

Textures

precomputed, sampled, and rendered almost entirely on graphics hardware



DyRT Vertex Program: Displacement Example



DyRT movie

Laparoscopic example



[James & Pai 02]

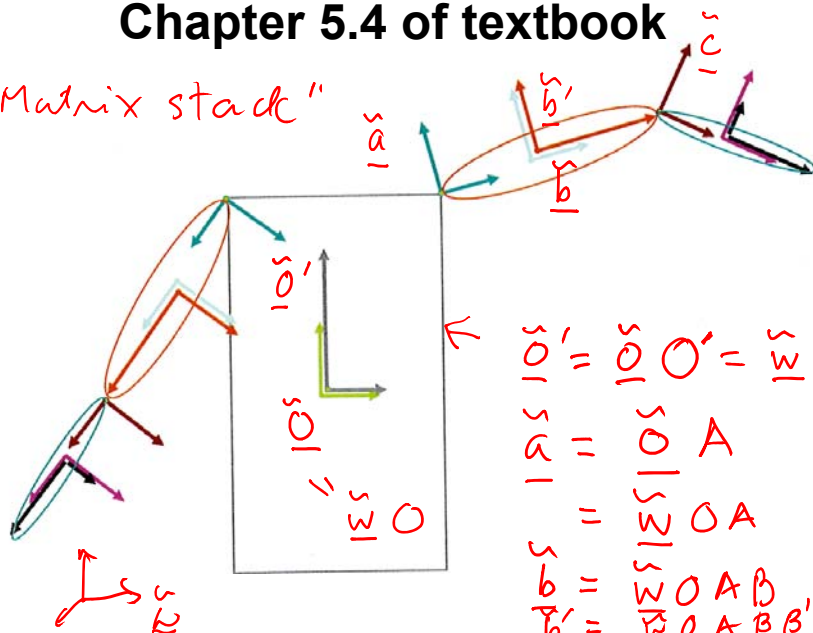
Transformations, wrap up

- Homogenous coordinates and transformations are fundamental to computer graphics.
- Scene graphs and hierarchies (brief intro)
- Three.js support for hierarchies
- Assignment 2

9

E.g. a robot hierarchy Chapter 5.4 of textbook

"Matrix stack"



$$\begin{aligned}
 \tilde{O}' &= \tilde{O} \quad \tilde{O}' = \tilde{W} \tilde{O} \tilde{O}' \\
 \tilde{a} &= \tilde{O} \quad \tilde{a} = \tilde{W} \tilde{O} \tilde{a} \\
 \tilde{b} &= \tilde{W} \tilde{O} \tilde{a} \tilde{b} \\
 \tilde{b}' &= \tilde{W} \tilde{O} \tilde{a} \tilde{b} \tilde{b}'
 \end{aligned}$$

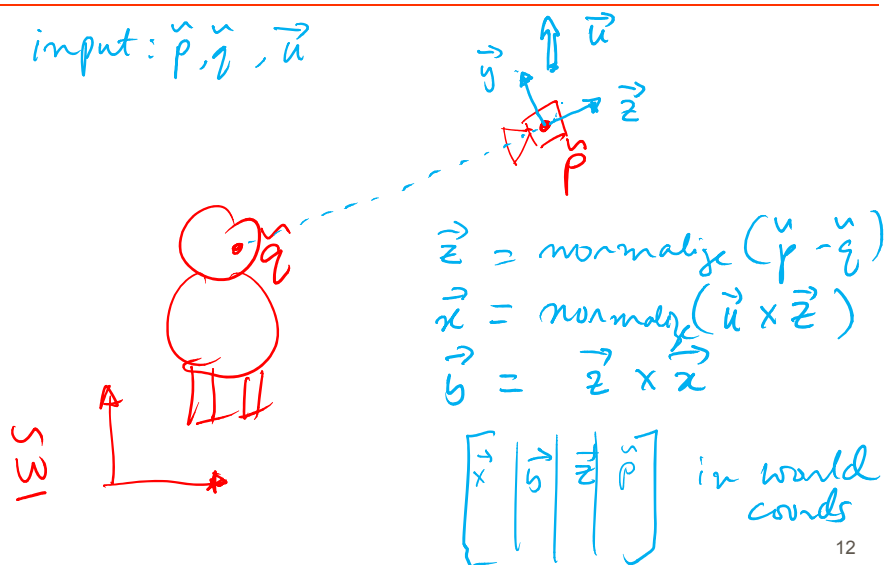
Three.js support

- Reviewed documentation at <http://threejs.org/> especially <http://threejs.org/docs/#Reference/Core/Object3D>
- Object3D is the basic scene graph node

11

A closer look at "lookAt"

input: $\tilde{p}, \tilde{q}, \vec{u}$



12

A closer look at “lookAt”

- Book description in 5.2.3 has a bug, fixed in online Errata (make this and other corrections in your textbook copy)
 - $z = \text{normalize}(p - q)$
 - $x = \text{normalize}(u \times z)$
 - $y = (z \times x)$
- The book’s “lookAt” is the inverse of Three.js’s `camera.lookAt()` method
- The author is aware of these issues, will fix it in future editions

13