

# Compositing and the alpha channel

Dinesh K. Pai

Textbook Chapter 16

Several slides courtesy of M. Kim

1

## Today

---

- Announcements
  - Q3 has been graded and is available from handback
- Multisample anti-aliasing
- Compositing

2

## Q3 solutions

---

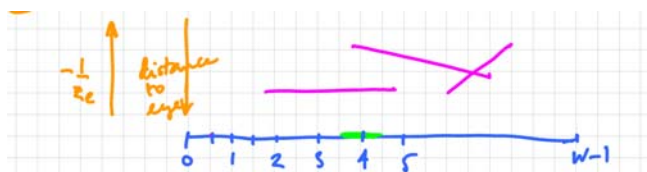
- Q1: 9,7,4,18,10,15,14,3
- Q2: T, F, T, T
- Will discuss other questions on Wednesday

3

## Recap: Coverage

---

- Texture => Pre-filtered textures, "mip mapping"
- Shading => generally changes slowly, except at edges of triangles
- Depth discontinuities => check if discontinuity passes through pixel



- Estimate partial **coverage** of pixel by triangle fragment
- Fraction of pixel covered is denoted alpha ( $\alpha$ ).

4

## Compositing?

---

- Example of demo reel  
<http://vimeo.com/72617082>

5

## Overview of Compositing

---

- Generalize idea of anti-aliasing to representing the “coverage” of each pixel by an object
- Essential for multi-pass rendering, requiring combination of images
- Historically, related to “matte”s in film, now done using the “alpha” channel in RGBA color images
- Importance increasing due to increasing availability of digital imagery
- Widely used: Visual Effects, “Sprites” in games, etc. Natively supported in most OS’s for GUI

6

## Image compositing

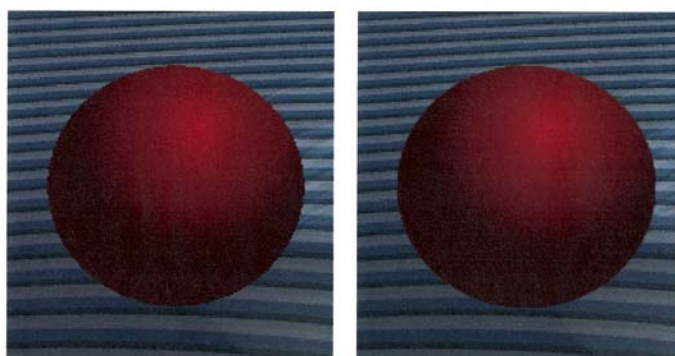
---

- Given two discrete images, a foreground,  $I^f$ , and background,  $I^b$ , that we want to combine into one image  $I^c$ .
- Simple: in composite, use foreground pixels where they are defined. Else use background pixels.
- This will give us a jagged boundary.
- Real image would have “boundary” pixels with blended colors.
- But this requires using “sub-pixel” information.

7

## Image compositing

---



8

## Alpha blending

---

- Associate with each pixel in each image layer, a value,  $\alpha[i][j]$ , that describes the overall opacity or coverage of the image layer at that pixel.
  - An alpha value of 1 represents a fully opaque/occupied pixel, while a value of 0 represents a fully transparent/empty one.
  - A fractional value represents a partially transparent (partially occupied) pixel.
- Alpha will be used during compositing.

9

## Alpha definition

---

- More specifically, let  $I(x, y)$  be a continuous image, and let  $C(x, y)$  be a binary valued  $(x, y)$  *coverage function* over the continuous domain, with a value of 1 at any point where the image is “occupied” and 0 where it is not.
- Let us store in our discrete image the values:

$$I[i][j] \leftarrow \iint_{\Omega_{i,j}} I(x, y)C(x, y)dx dy$$

$$\alpha[i][j] \leftarrow \iint_{\Omega_{i,j}} C(x, y)dx dy$$

10

## Over operation

*Note: a technical term*

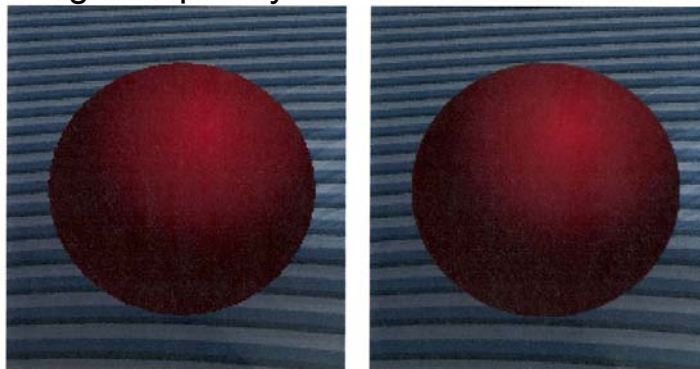
- To compose  $I^f[i][j]$  over  $I^b[i][j]$ , we compute the composite image colors,  $I^c[i][j]$ , using
 
$$I^c[i][j] \leftarrow I^f[i][j] + I^b[i][j] (1 - \alpha^f[i][j])$$
 That is, the amount of observed background color at a pixel is proportional to the transparency of the foreground layer at that pixel.

- Likewise, alpha for the composite image can be computed as:

$$\alpha^c[i][j] \leftarrow \alpha^f[i][j] + \alpha^b[i][j] (1 - \alpha^f[i][j])$$

## Over operation

- If background is opaque, so the composite pixel is opaque.
- But we can model more general case as part of blending multiple layers.



12

## Over properties

---

- This provides a reasonable approximation to the correctly rendered image.

- One can easily verify that the over operation is associative but not commutative. That is,

$$I^a \text{ over } (I^b \text{ over } I^c) = (I^a \text{ over } I^b) \text{ over } I^c$$

$$I^a \text{ over } I^b \neq I^b \text{ over } I^a$$

13

## Another example of using depth and compositing

---

- Depth cameras are now becoming available in mobile phones. Can use with compositing
- See <http://www.engadget.com/2014/03/25/htc-announces-the-new-one/> video around 0:50



14