Lecture 8 - Compositing

Michael Gleicher
September, 2007
Notes – not for display

Outline - Compositing
• What is compositing?
• Why? (really defer till next week)
• What is digital compositing?
• How does it work?
• (briefly) where do mattes come from?

Compositing
• How do we combine pictures seemlessly?
  • Weatherman in front of weather map?
  • Flying saucer/monster in 1950s movie?
  • CG element in a live action movie?
  • Multiple live action elements?

How would you do it by hand?
• Cut out picture and paste it
  • Picture has an irregular shape
    – Not the same as the original image
  • Want to keep images all the same size
    – Film: keep it all the same size
    – Digital – hard to have non-rectangles

Mattes
• Make the images the same size, but have transparent regions

• Optical Printing (old movie production)
  – Expose the film multiple times
  – Have an image that is solid where the image should not go
  – Mask or “matte”

Digital Mattes
• An image the same size as the “regular” image
• For each pixel, store whether or not you want that pixel
• When assemble the images – pick the “top” image when matte has a value
Matte as Opacity

- Value 0 = transparent
- Value 100% (255?) = solid
- Can have semi-transparent (50% opacity)

Matte as filling pixels

- Little square model
- Opacity says how full each pixel is
  - 100% - totally full, nothing gets through
  - 0% - totally empty, everything gets through
  - 50% - half full – on average, half the stuff gets through
- Can’t see the details of which 50% (since its just a pixel, and has a single value)
- Get accurate edges (halfway = half full)

Alpha channel

- Store opacity as a 4th “Color”
- Each pixel has RGB, A
- A=100% (255?) then solid

Compositing

- Assume 1 image “over” the other
- Image A=(r,g,b,a) and B=(r,g,b,a)
- Done per pixel (assume samples in the same place)
- What is the resulting color
  - C (r,g,b,a) = A(r,g,b,a) OVER B(r,g,b,a)
  - There are other operators too

How over works

- Use the little square model for intuition
- Suppose half full for the moment
- Can’t know which half – assume statistically uncorrelated.
- We’ll draw pictures so that it works out.

The 4 possibilities:

- Alpha says probability a location is “covered”
  - Aa means in A
  - Ba means in B
- 4 possibilities
  - In neither (1-Aa)(1-Ba)
  - In A but not B Aa (1-Ba)
  - In B but not A Ba (1-Aa)
  - In both Aa Ba

1 = neither
2 = A only
3 = B only
4 = A and B
**Compositing Operator**

- Decide what should happen in each region
- Weight result by the area of each region
  - Areas = probabilities, percentage of the whole

**A Over B**

- Neither = nothing
- A only = A
- B only = B
- A and B = A (since over)

\[ \text{Aa} (1 - \text{Ba}) + \text{Aa} \text{ Ba} \text{ Acolor} + \text{Ba} (1 - \text{Aa}) \text{ Bcolor} = \text{Aa Acolor} + (1 - \text{Aa}) \text{ Ba Bcolor} \]

**What is the new alpha?**

- When is there “something”

\[ \text{Aa} + \text{Ba} - \text{Aa Ba} \text{ (don’t count the both case twice)} \]

- Add up the areas of the regions that are filled
  - \(\text{Aa} (1 - \text{Ba}) + \text{Ba} (1 - \text{Aa}) + \text{Aa Ba}\)
  - \(\text{Aa} - \text{Aa} \text{ Ba} + \text{Ba} - \text{Aa} \text{ Ba} + \text{Aa Ba}\)
  - \(\text{Aa} + \text{Ba} - \text{Aa} \text{ Ba}\)

**Other operators**

- Specify what happens in each “quadrant”
  - 1 = neither (always 0)
  - 2 = A only (choices 0 or A)
  - 3 = B only (choices 0 or B)
  - 4 = A and B (choices 0, A or B)

- For each of 2-4 pick a different result to get a different operation
- 12 possible operators

**A held-inside B**

- Give color A only when it is inside of B

\[ \text{Ca} = \text{Aa Ba} \]

**A atop B**

- Give color A when it is on top of B (and give B otherwise)

\[ \text{Ccolor} = \text{Aa Ba Acolor} + (1 - \text{Aa}) \text{ Ba Bcolor} \]
\[ \text{Ca} = \text{Aa Ba} \]
Pre-Multiplied Alpha

• Notice that the color always appears multiplied by alpha
• Can make all this much easier by pre-multiplying (storing Aa Acolor, not Acolor)

• Transparent pixel has no color
  – Conceptually clean
• Popular thing to do
  – Our image library does it (beware)
• Can’t get colors back (alpha=0)

Blending functions

• Premultiplied gives easy common form
  \[ C_{\text{color}} = F \ A_{\text{color}} + G \ B_{\text{color}} \]
  \[ C_a = F_a \ A_a + F_b \ B_a \]

• \( F_a \) and \( F_b \) determined based on operation
  – Over \( F_a = 1, F_b = (1-A_a) \)
  – Atop \( F_a = B_a, F_b = 0 \)

Where does alpha come from?

• Paint it as a separate channel

• Various matting algorithms
  – Chromamatte (equal to a color)
  – Lummamatte (bright/dark pixels)
  – Difference matte (pixels same in 2 image=back)

• Tricky to get partial opacities
  – Is it part red/part blue (screen) or purple?
  – Important: otherwise get fringe effects

Blue Screen / Green Screen

• Naïve version:
  – Assume objects generally have blue in proportion to other colors
  – Grey flying saucers in 1960s movies
  – Alpha = 1 – (B-uG) (where .5 < u < 1.5)
    • Clamp (gives a threshold – must be more blue than green)
    – Alpha = 1 – u1(min(Bf,Bk) + u2 G))

Really hard!

• Background is never really pure color
  – Try to get lighting constant
• Edge pixels, hair, etc. are partially background
• Shadows, spill (blue background reflects onto objects), transparent objects, …

• Modern algorithms use fancy computer vision on neighborhoods of pixels

• Expensive fancy software in effects industry!
Next week…

• I’m at a conference (show stuff later in semester)
• Monday / Wednesday – Li Zhang
  – Talk about warping and morphing (more resampling). Plus vision applications.
  – Part of project. Also, TAs to give project hints
• Fiday – Perry
  – Special Effects!
  – He has had 2 companies – won an Academy Award and an Emmy award!