

# Lecture 8

## More IMAGING! AND COMPOSITING

What is a pixel:

1. a little region of the image? (square, circle)
2. a sample location?

#1 is an easy way to think about #2

line drawing



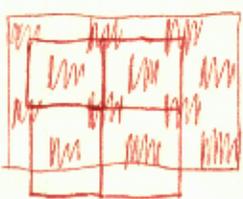
partially filled squares

or



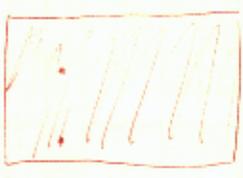
correctly filtered, then point sampled

size reduction



new pixels cover larger area

or



point sample properly filtered

1 2 1  
 2 4 2  
 1 2 1

## The Binomial Filter

1 1  
 1 2 1 ← 1 3 3 1  
 1 4 6 4 1  
 1 5 10 10 5 1  
 1

1 1      1 2 1  
 1 1      2 4 2  
           1 2 1

1 3 3 1  
 3 9 9 3  
 3 9 9 3  
 1 3 3 1

↑  
 1 1 1 \* 1  
 |

Compositing - layering images on top of one another

What happens to the unfilled area of each pixel?

half filled with white vs all filled w/ gray  
opacity / transparency

### Compositing Pre-Computer

Photographic Printing - MATTE

Optical Printing

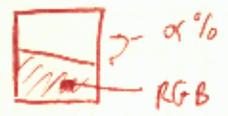
holdout matte (garbage matte)  
↳ traveling matte

How to get Matte?

- a story for later

### WITH A COMPUTER

Each Pixel has : Color (RGB) + MATTE (A)



pixel appears  $\alpha \cdot C$  —  $\alpha C$  non-premult  
or  $\alpha, \alpha C$  premult

### Composite

2 half full pixels



Can't know!

threw away info

or, ditch "geometry"

Assumption: uncorrelated areas

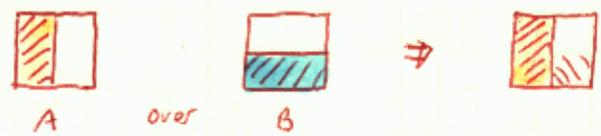
proportion of A in any sub area is the same

so  $\alpha$  of a in pixel =  $\alpha$  of A in B

random scatter or



### Over operation



all of A  $\alpha_A C_A$

B, except where covered by A  $\alpha_B C_B (1 - \alpha_A)$

NOTICE: Formulas simpler if pre-multiplied

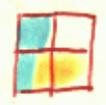
NOTICE 2: Result is pre-multiplied

What is new  $\alpha$

$\alpha_A + \alpha_B - \alpha_A \alpha_B$   
└ don't count this twice

### Other Operators (Porter-Duff Algebra)

4 regions



- O - O
- A - O or A
- B - O or B
- AB - O, A or B

2 · 2 · 3 = 12 possibilities

Clear	0, 0, 0, 0	O	use pre-mult case "blending function"	0	0
A	0, 1, 0, 1	$\alpha_A A$		1	0
B	0, 0, 1, 1	$\alpha_B B$		0	1
A over B in out atop	0, 1, 1, A	$\alpha_A A + (1 - \alpha_A) \alpha_B B$		1	$(1 - \alpha_A)$

What's with Alvy Ray's wierd approximations?

ignore them - do floats?

really worried about drops of performance - millions of pixels

divide by 255 is worrisome (slow on many computers)

divide by 256 is wrong

ANTI-ALIASING AS COMPOSITING