



# Image Compositing & Morphing

Adam Finkelstein

COS 426, Spring 2016

Princeton University

# Image Processing Operations I



- Luminance
  - Brightness
  - Contrast.
  - Gamma
  - Histogram equalization
- Color
  - Black & white
  - Saturation
  - White balance
- Linear filtering
  - Blur & sharpen
  - Edge detect
  - Convolution
- Non-linear filtering
  - Median
  - Bilateral filter
- Dithering
  - Quantization
  - Ordered dither
  - Floyd-Steinberg

# Image Processing Operations II



- Transformation

- Scale
- Rotate
- Warp



Last time

- Combining images

- Composite
- Morph
- Computational photography



Today

# Image Processing Operations II



- Transformation
  - Scale
  - Rotate
  - Warp



Last time

## Combining images

- Composite
- Morph
- Computational photography



Today

# Image Composition



Jurassic Park

# Image Composition



- Issues:
  - Segmentation of image into layers/regions
  - Blend into single image seamlessly



# Image Composition



- Issues:
  - Segmentation of image into layers/regions
  - Blend into single image seamlessly

# Image Segmentation



- Chroma keying (blue- or green-screen)
  - Photograph object in front of screen with known color



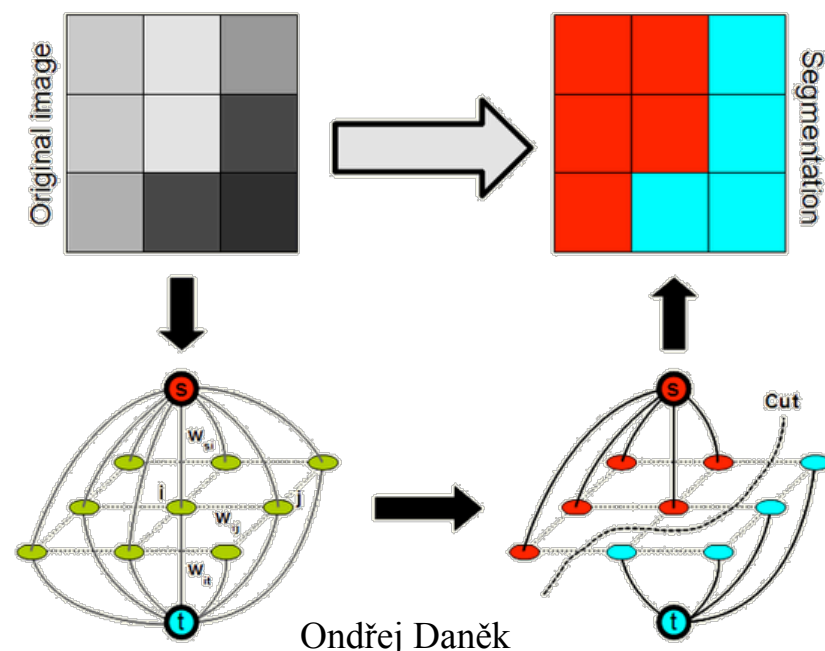
Rosco Spectrum



# Image Segmentation



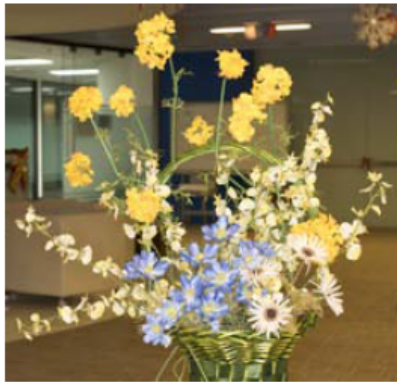
- Specify segmentation by hand
    - Purely manual: rotoscoping (draw matte, every frame)
    - Semi-automatic: graph-cut (draw a few strokes)
- Separate image regions along minimal cuts (where edges measure differences between adjacent pixels)



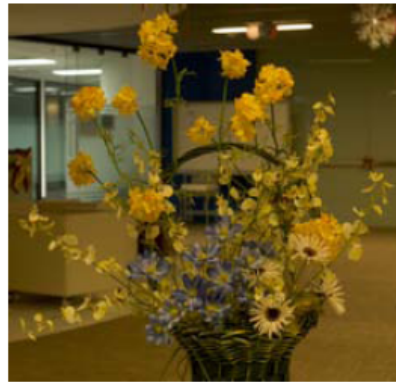
# Image Segmentation



- Novel methods, e.g. flash matting



flash



no flash



matte



composite

# Image Composition

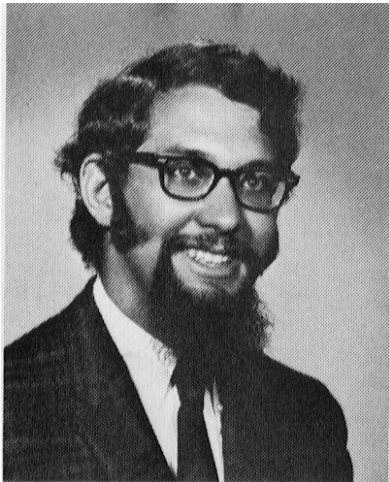


- Issues:
  - Segmentation of image into layers/regions
  - Blend into single image seamlessly



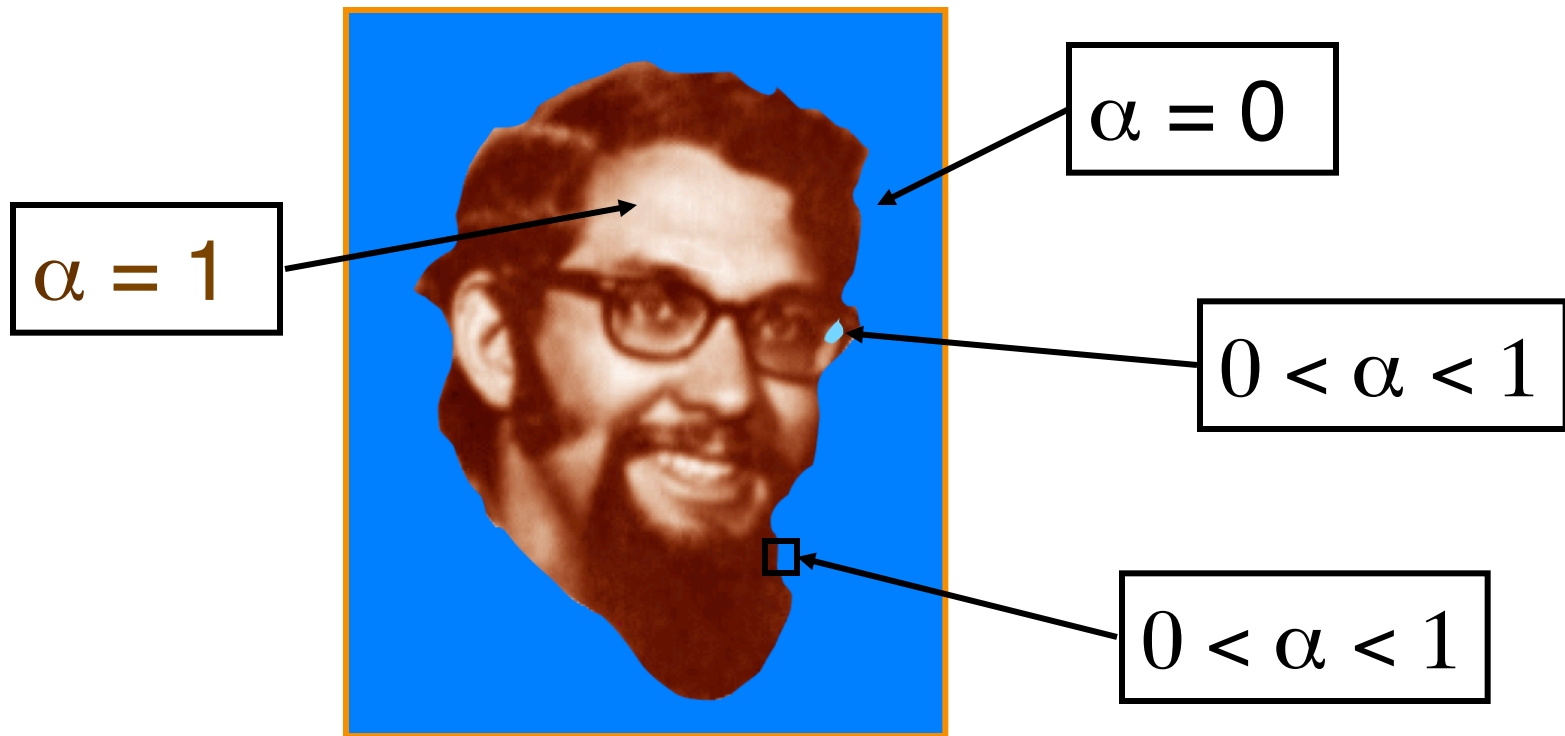
# Image Blending

- Ingredients
  - Background image
  - Foreground image
- Goal
  - Put foreground over background seamlessly



# Blending with Alpha Channel

Alpha controls the linear interpolation of foreground and background pixels when elements are composited.





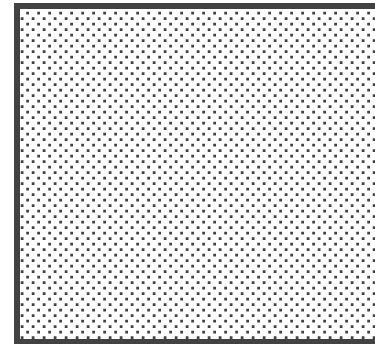
# Blending with Alpha Channel

- Alpha encodes pixel coverage information
  - $\alpha = 0$ : no coverage (or transparent)
  - $\alpha = 1$ : full coverage (or opaque)
  - $0 < \alpha < 1$ : partial coverage (or semi-transparent)
- Example:  $\alpha = 0.3$



Partial  
Coverage

or

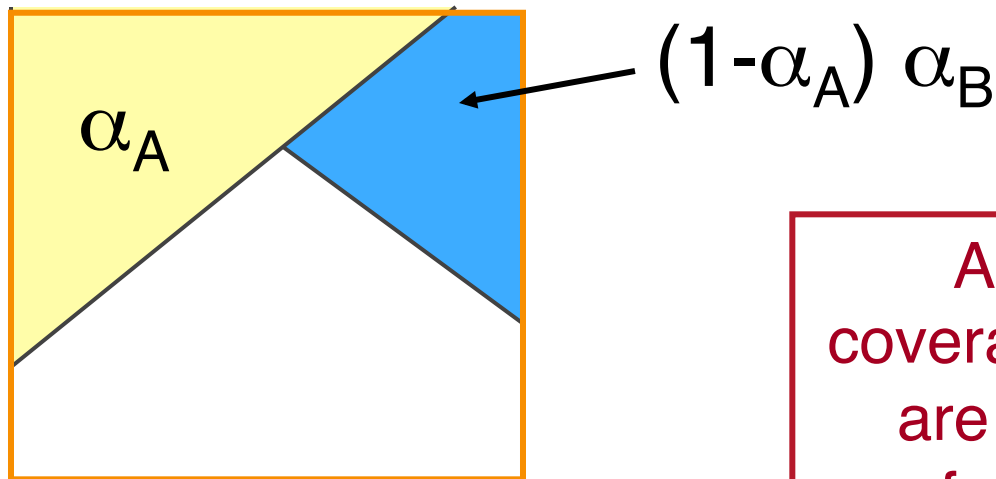


Semi-  
Transparent



# Blending with Alpha Channel

- Example:  $C = A \text{ Over } B$ 
  - $C' = \alpha_A A + (1 - \alpha_A) \alpha_B B$
  - $\alpha = \alpha_A + (1 - \alpha_A) \alpha_B$

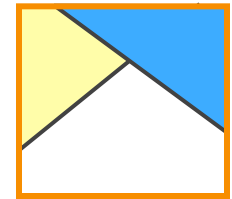
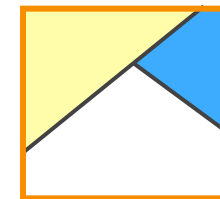
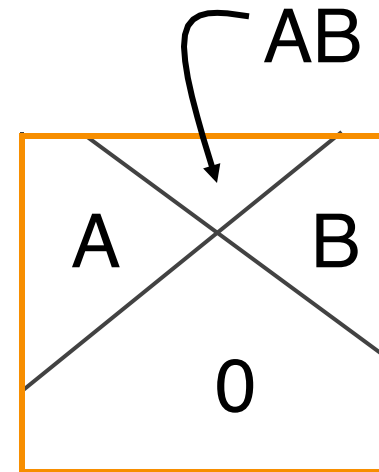
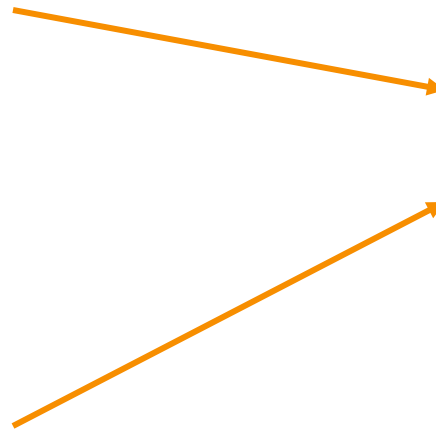
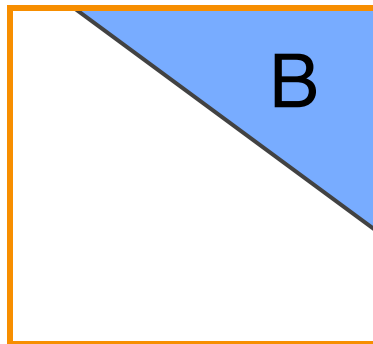
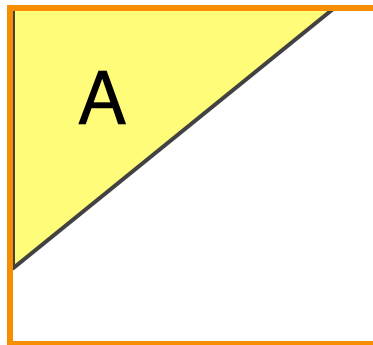


A over B

Assumption:  
coverages of A and B  
are uncorrelated  
for each pixel

# Other Composition Operations

- Ways to combine 2 partially covered pixels?
  - 3 possible colors (0, A, B)
  - 4 regions (0, A, B, AB)



???

???





# Other Composition Operations

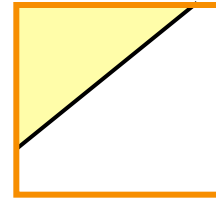
Composition algebra – 12 combinations

$$C' = F_A \alpha_A A + F_B \alpha_B B$$

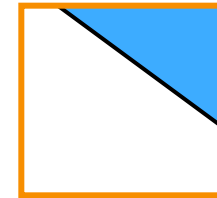
Operation	$F_A$	$F_B$
Clear	0	0
A	1	0
B	0	1
A over B	1	$1 - \alpha_A$
B over A	$1 - \alpha_B$	1
A in B	$\alpha_B$	0
B in A	0	$\alpha_A$
A out B	$1 - \alpha_B$	0
B out A	0	$1 - \alpha_A$
A atop B	$\alpha_B$	$1 - \alpha_A$
B atop A	$1 - \alpha_B$	$\alpha_A$
A xor B	$1 - \alpha_B$	$1 - \alpha_A$



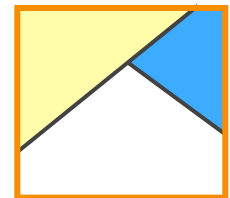
clear



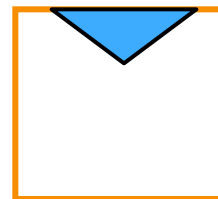
A



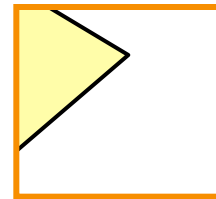
B



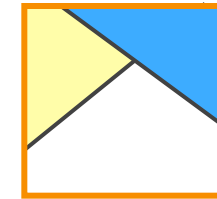
A over B



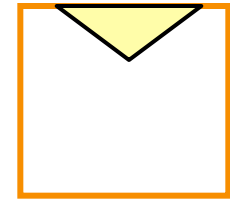
B in A



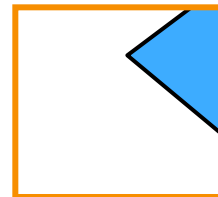
A out B



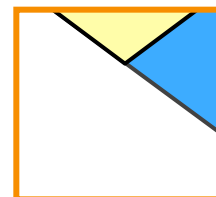
B over A



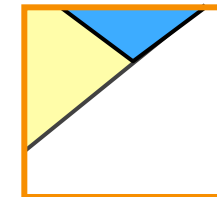
A in B



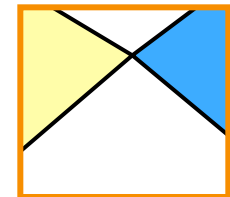
B out A



A atop B



B atop A

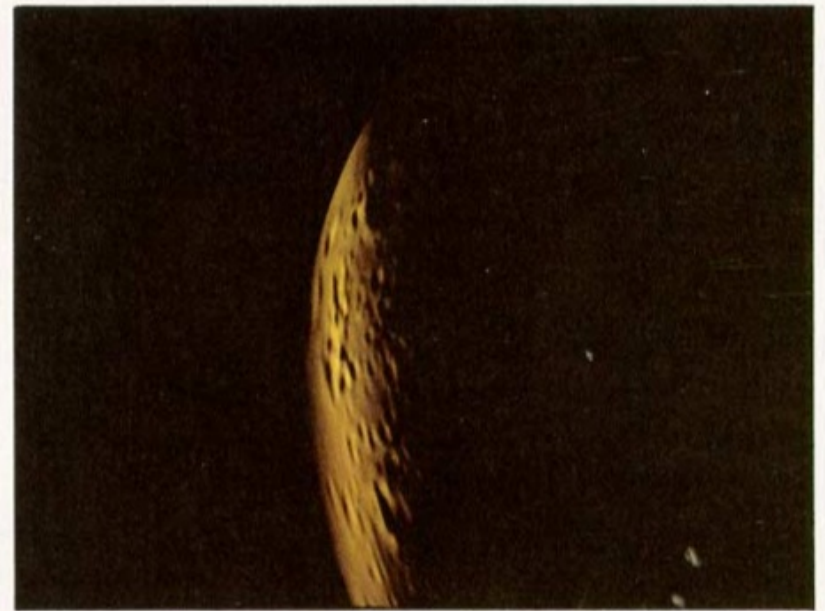


A xor b

# Image Composition Example



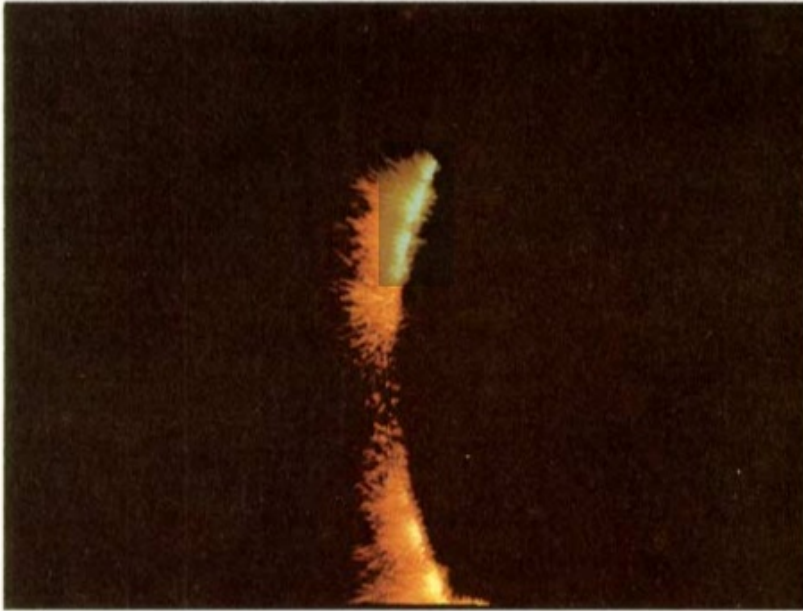
Stars



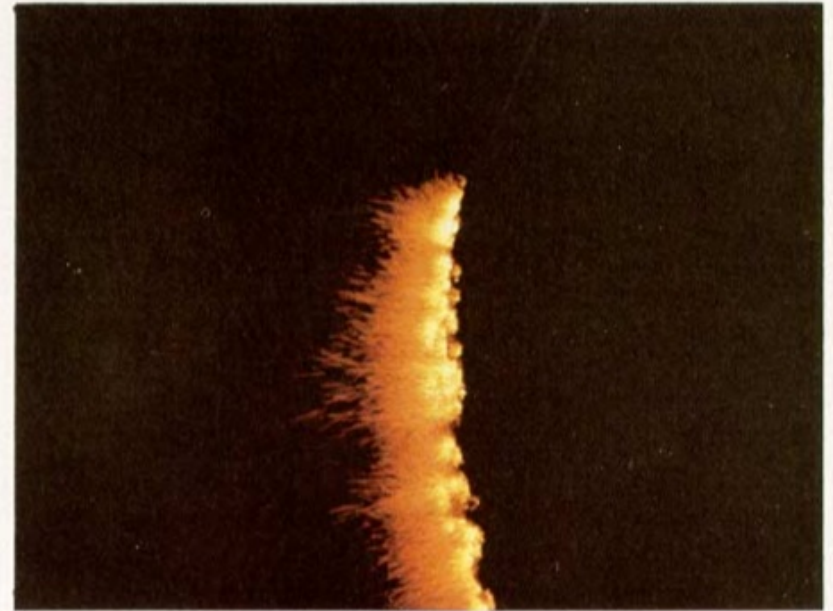
Planet

[Porter&Duff *Computer Graphics* 18:3 1984]

# Image Composition Example



BFire



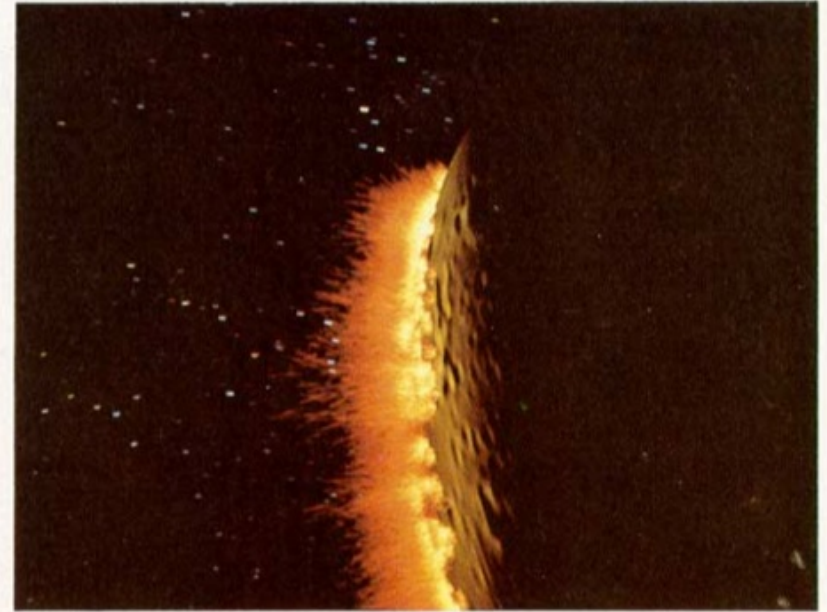
FFire

[Porter&Duff *Computer Graphics* 18:3 1984]

# Image Composition Example



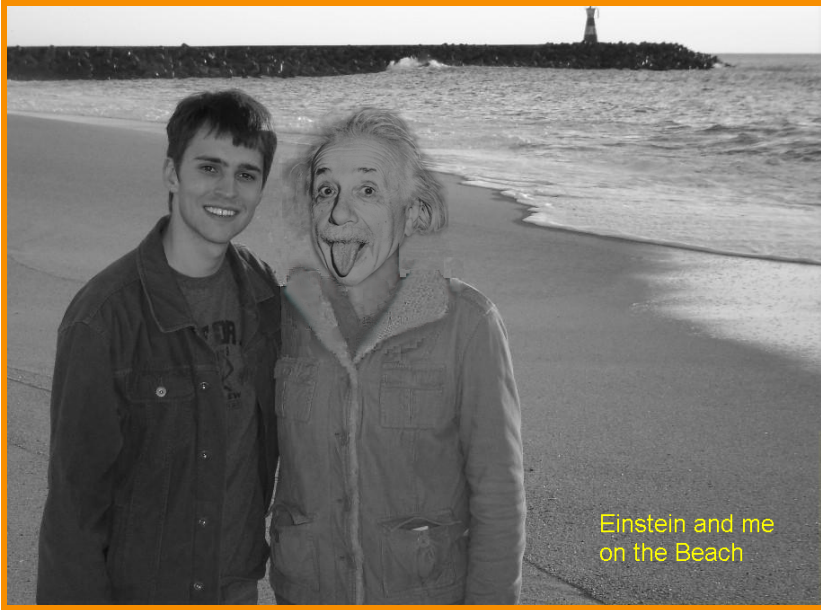
BFire out Planet



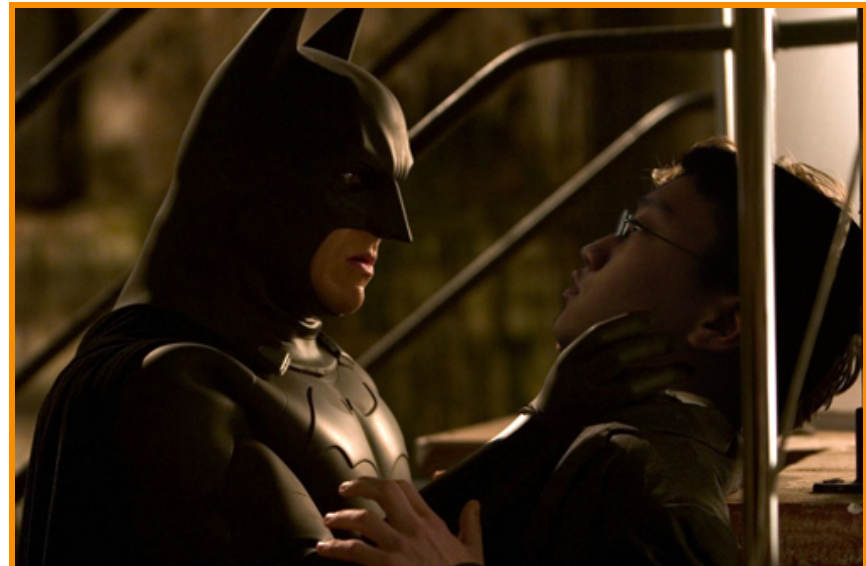
Composite

[Porter&Duff *Computer Graphics* 18:3 1984]

# COS426 Examples



Darin Sleiter

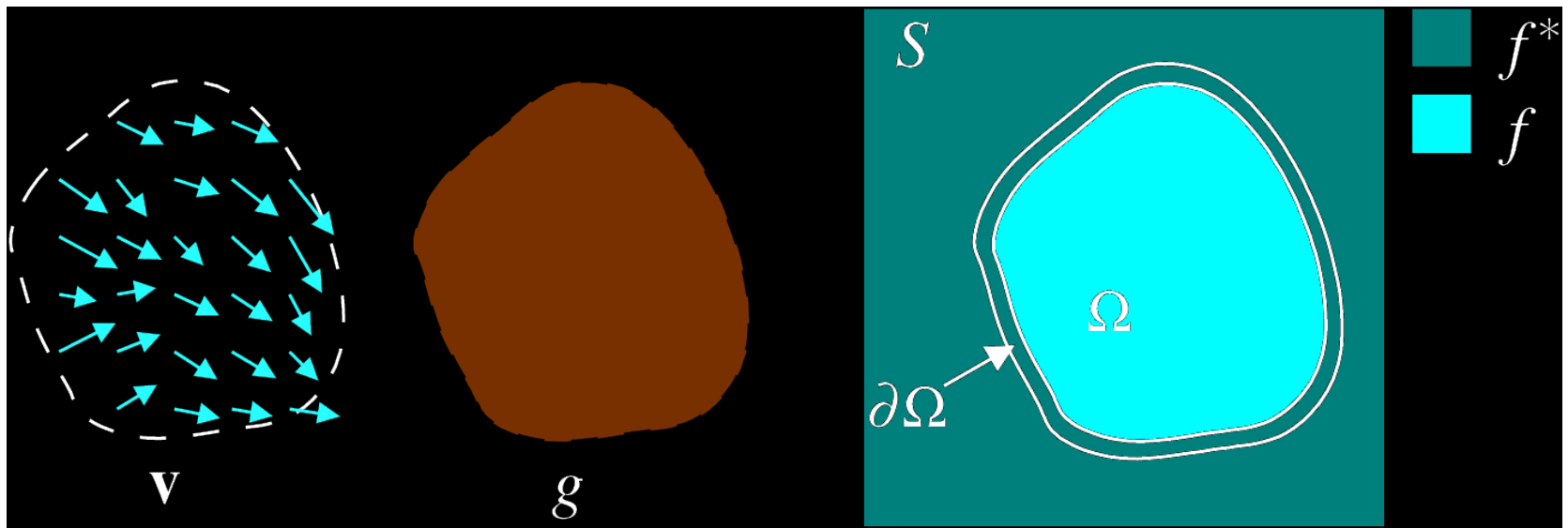


Kenrick Kin

# Poisson Image Blending

Beyond simple compositing

- Solve for image samples that follow gradients of source subject to boundary conditions imposed by dest

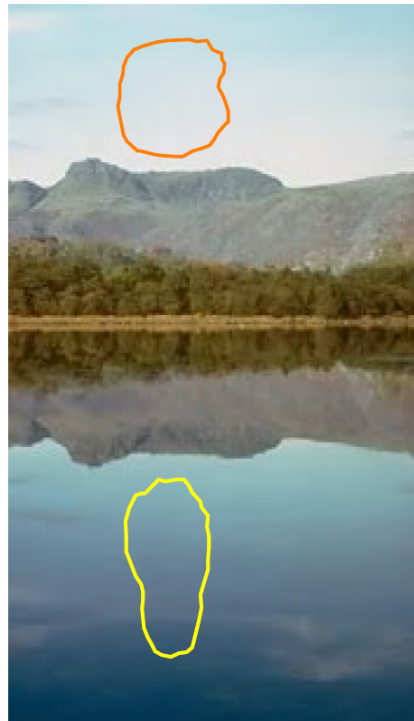


$$\begin{cases} \nabla^2 f = \nabla \cdot \mathbf{v} \\ f|_{\partial\Omega} = f^*|_{\partial\Omega} \end{cases}$$

# Poisson Image Blending



sources



destinations



cloning



seamless cloning

# Poisson Image Blending



source/destination



cloning



seamless cloning



# Poisson Image Blending



Stefan Chmiela & Andreas Langenhagen



# Digital Image Processing

- Changing intensity/color
  - Linear: scale, offset, etc.
  - Nonlinear: gamma, saturation, etc.
  - Add random noise
- Filtering over neighborhoods
  - Blur
  - Detect edges
  - Sharpen
  - Emboss
  - Median
- Moving image locations
  - Scale
  - Rotate
  - Warp
- Combining images
  - Composite
  - Morph
- Quantization
- Spatial / intensity tradeoff
  - Dithering

# Image Morphing



- Animate transition between two images



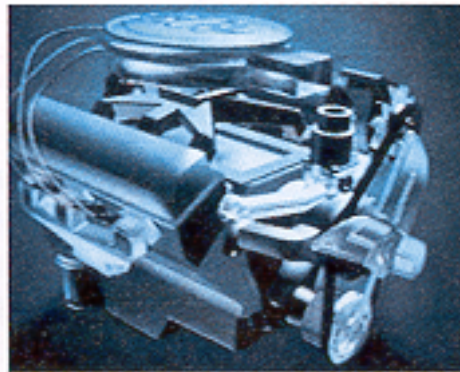
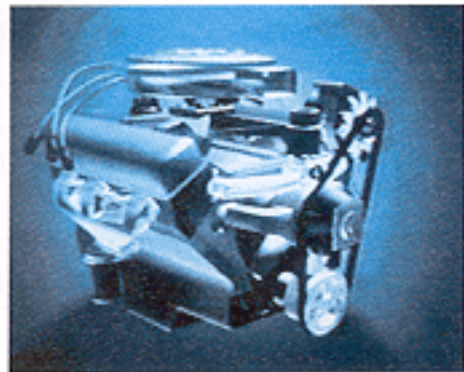
(a)



(b)



(c)



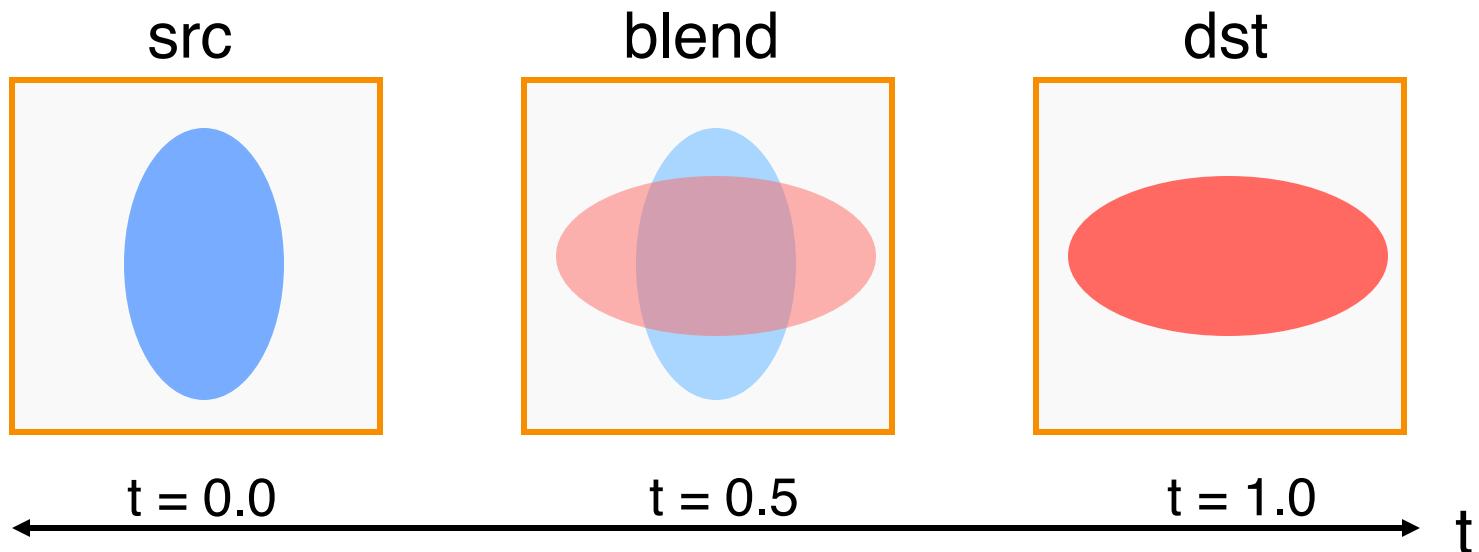
*Figure 16-9*

Transformation of an STP oil can into an engine block. (Courtesy of Silicon Graphics, Inc.)

# Cross-Dissolving

- Blend images with “over” operator
  - alpha of bottom image is 1.0
  - alpha of top image varies from 0.0 to 1.0

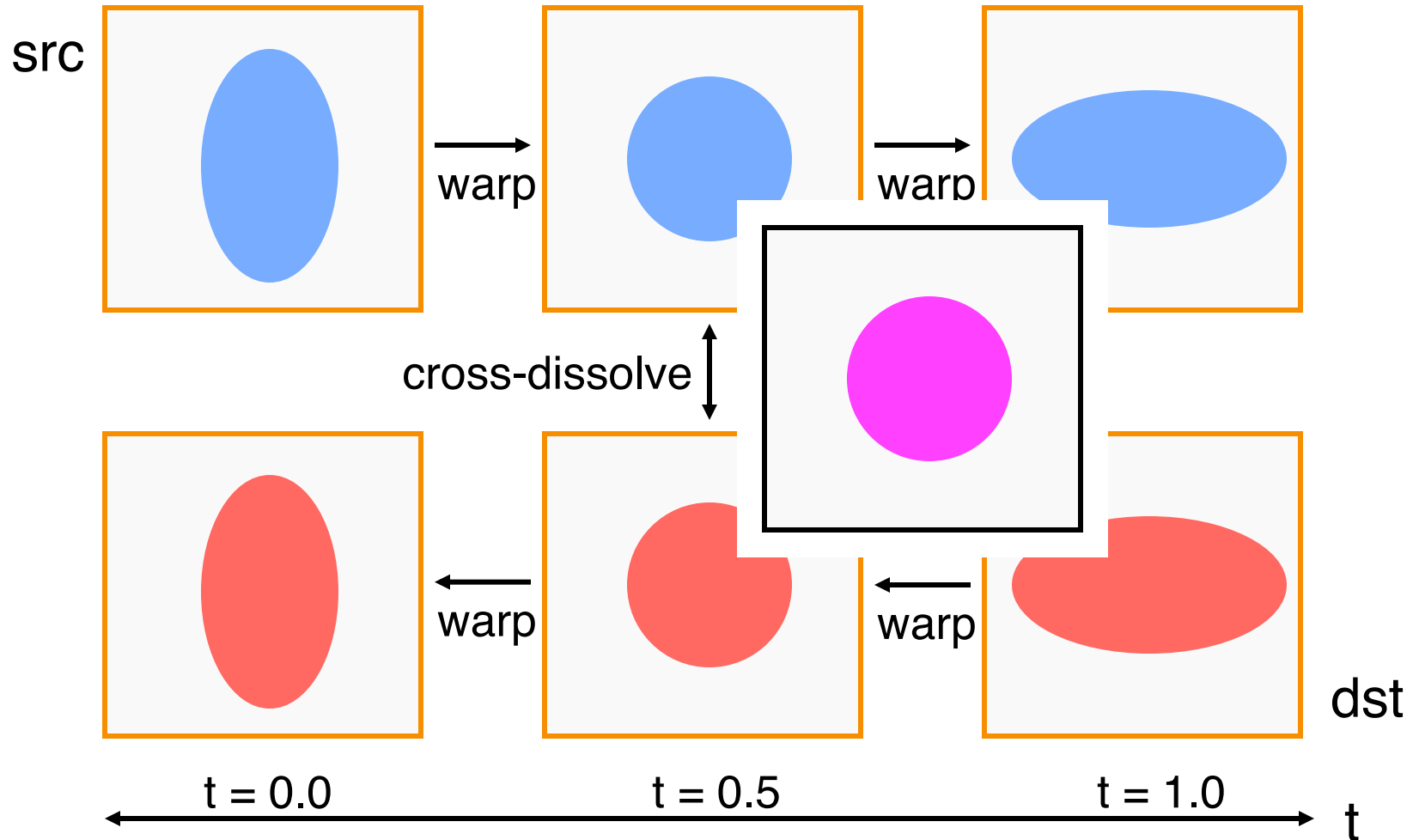
$$\text{blend}(i,j) = (1-t) \text{src}(i,j) + t \text{dst}(i,j) \quad (0 \leq t \leq 1)$$



# Image Morphing



- Combines warping and cross-dissolving



# Beier & Neeley Example



Image<sub>0</sub>

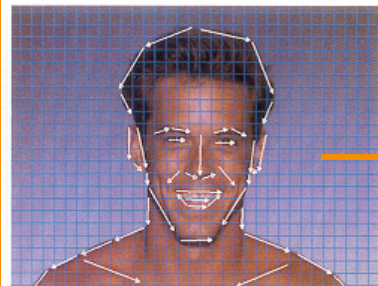


Figure 7

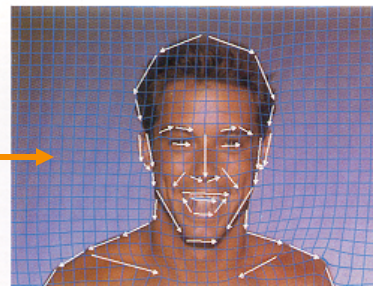


Figure 10

Warp<sub>0</sub>

Result

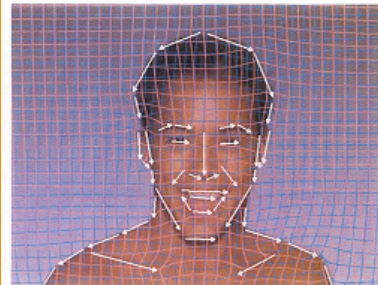


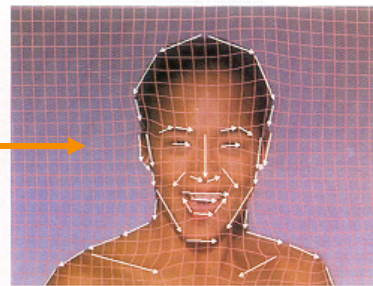
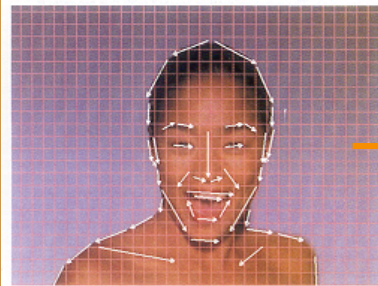
Figure 8

Figure 7 shows the lines drawn over the a face, figure 9 shows the lines drawn over a second face. Figure 8 shows the morphed image, with the interpolated lines drawn over it.

Figure 10 shows the first face with the lines and a grid, showing it is distorted to the position of the lines in the intermediate frame. Figure 11 shows the second face distorted to the same intermediate position. The lines in the top and bottom picture are in the same position. We have distorted the two images to the same "shape".

Note that outside the outline of the faces, the grids are warped very differently in the two images, but because this is the background, it is not important. If there were background features that needed to be matched, lines could have been drawn over them as well.

Image<sub>1</sub>

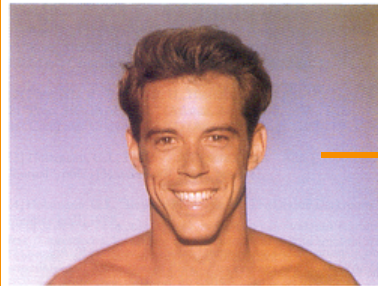


Warp<sub>1</sub>

# Beier & Neeley Example



Image<sub>0</sub>

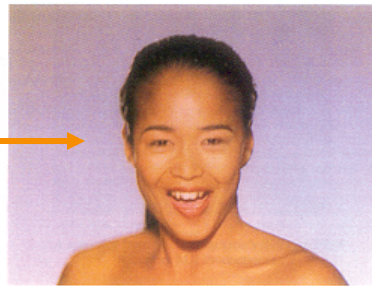


Warp<sub>0</sub>

Result



Image<sub>1</sub>



Warp<sub>1</sub>

The final sequence is figures 14, 15, and 16.  
Figure 15 is the first face distorted to the intermediate position without the grid. Figure 16 is the second face distorted toward that same position. Note that the grid between the two distorted images is much more like than the lines of the distorted images themselves. We have noticed this happens very frequently.

# Beier & Neeley Example



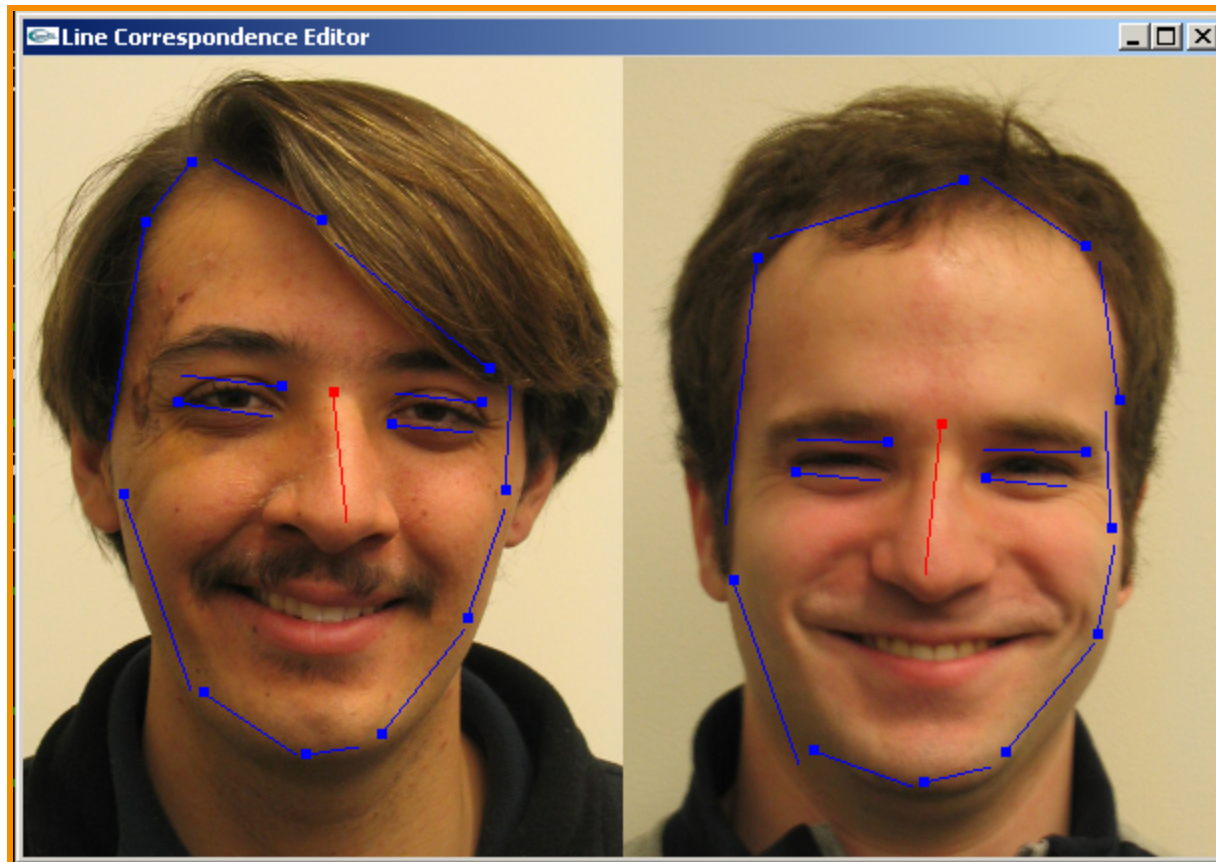
Black or White, Michael Jackson



# Line Correspondence Mappings

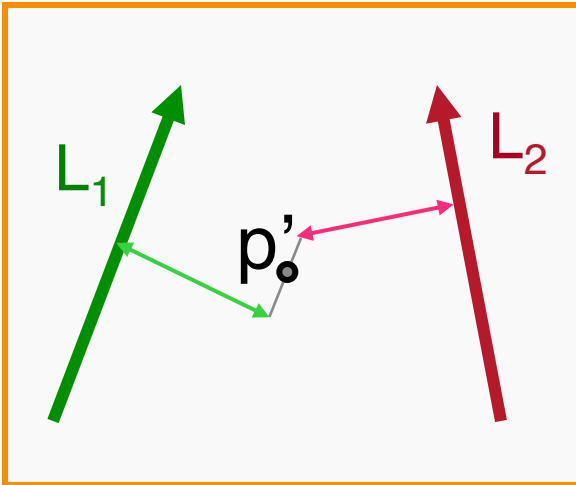


- Beier & Neeley use pairs of lines to specify warp



# Warping Pseudocode

```
WarpImage(Image, L' [...], L [...])  
begin  
  foreach destination pixel p do  
    psum = (0,0)  
    wsum = 0  
    foreach line L[i] in destination do  
      p'[i] = p transformed by (L[i],L'[i])  
      psum = psum + p'[i] * weight[i]  
      wsum += weight[i]  
    end  
    p' = psum / wsum  
    Result(p) = Resample(p')  
  end  
end
```



# Morphing Pseudocode



```
GenerateAnimation(Image0, L0[...], Image1, L1[...])
begin
  foreach intermediate frame time t do
    for i = 1 to number of line pairs do
      L[i] = line t-th of the way from L0 [i] to L1 [i]
    end
    Warp0 = WarpImage(Image0, L0, L)
    Warp1 = WarpImage(Image1, L1, L)
    foreach pixel p in FinalImage do
      Result(p) = (1-t) Warp0 + t Warp1
    end
  end
end
```

# COS426 Example



Amy Ousterhout

# COS426 Examples



ckctwo



Jon Beyer

# COS426 Examples from Last Year

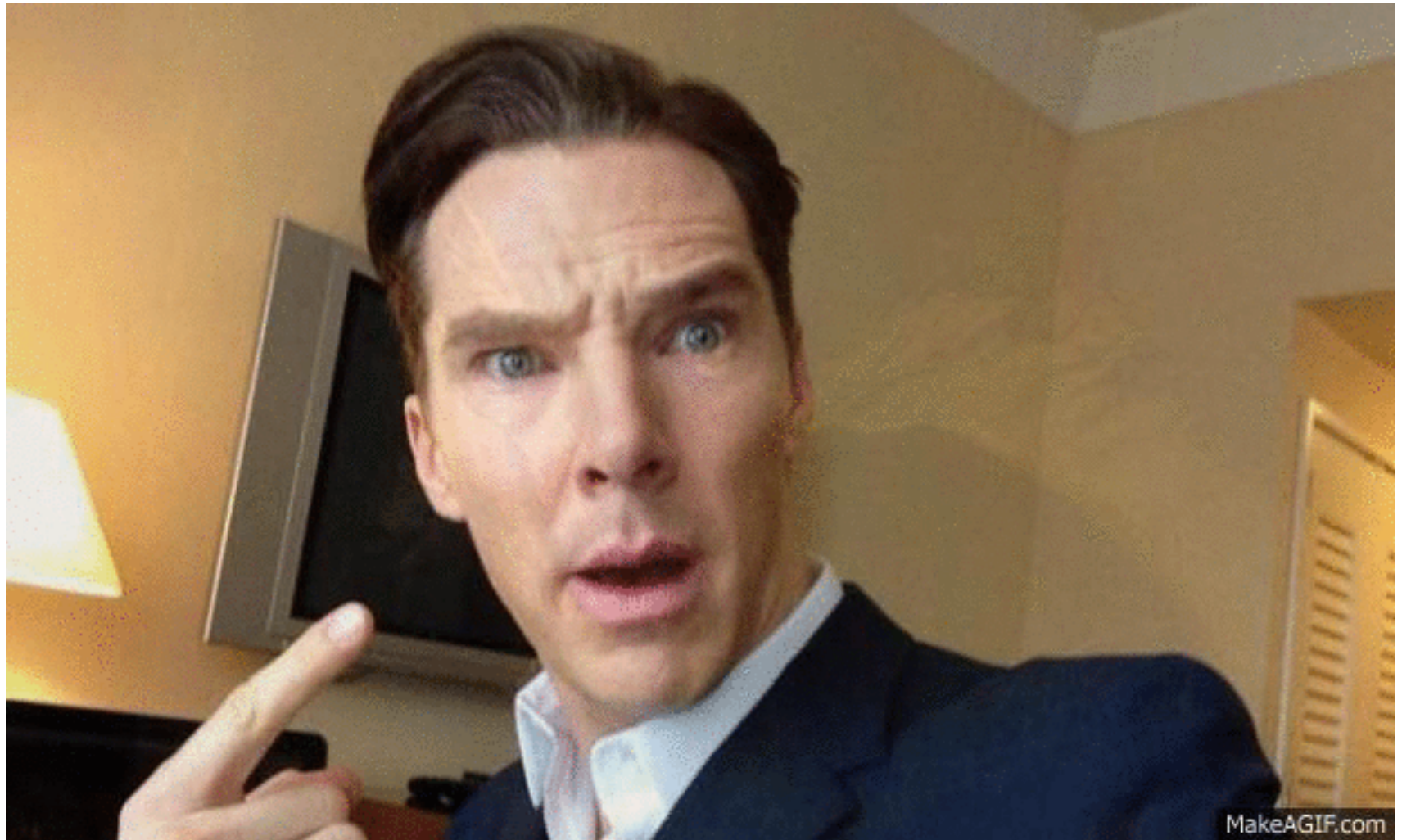


Sam Payne



Matt Matl

# COS426 Examples from Last Year



MakeAGIF.com

atran

# Image Composition Applications



- *Computational photography:* enable new photographic effects that inherently use multiple images + computation + composition





# Image Composition Applications



- Extended depth-of-field

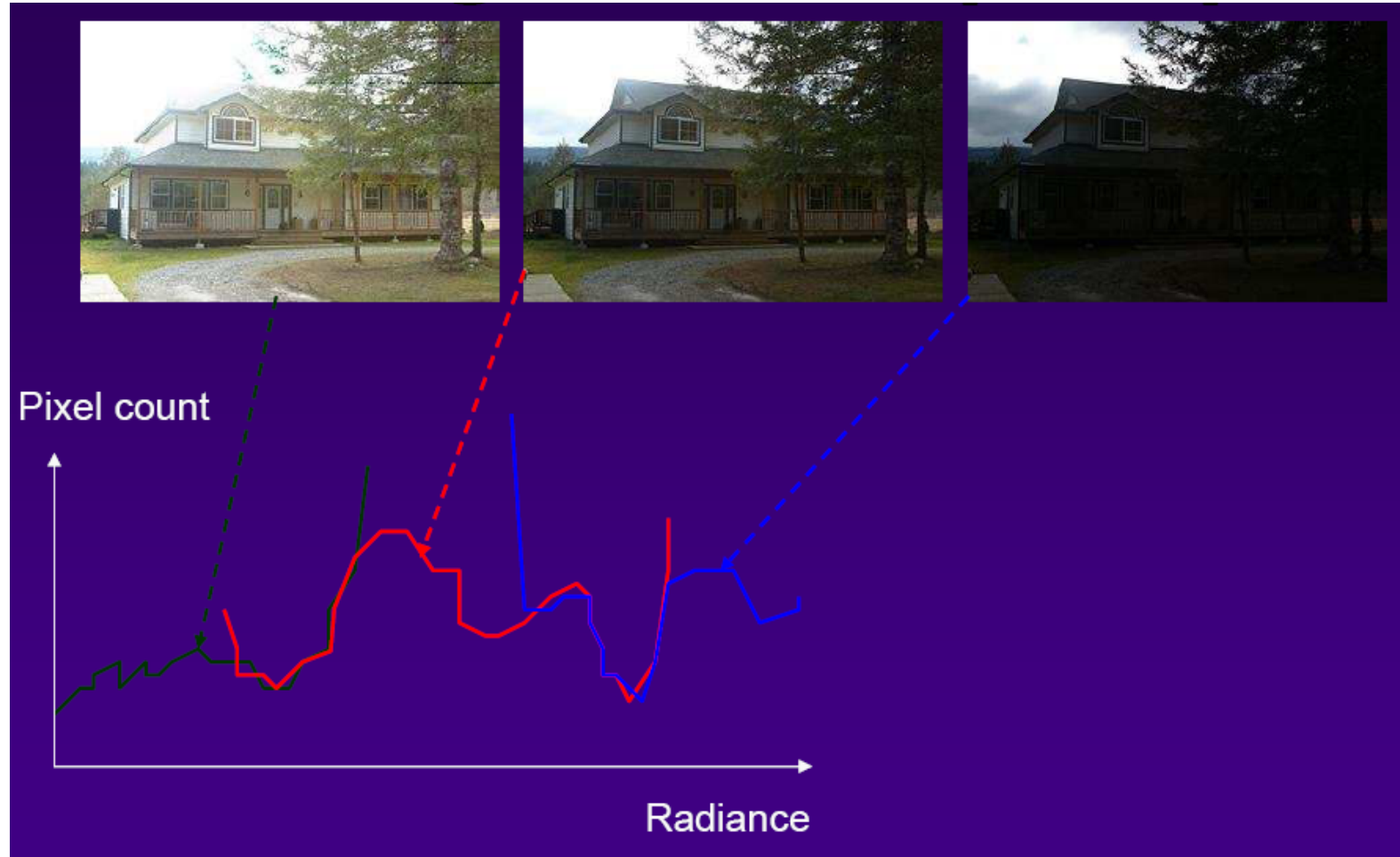


[Michael Cohen]

# Image Composition Applications



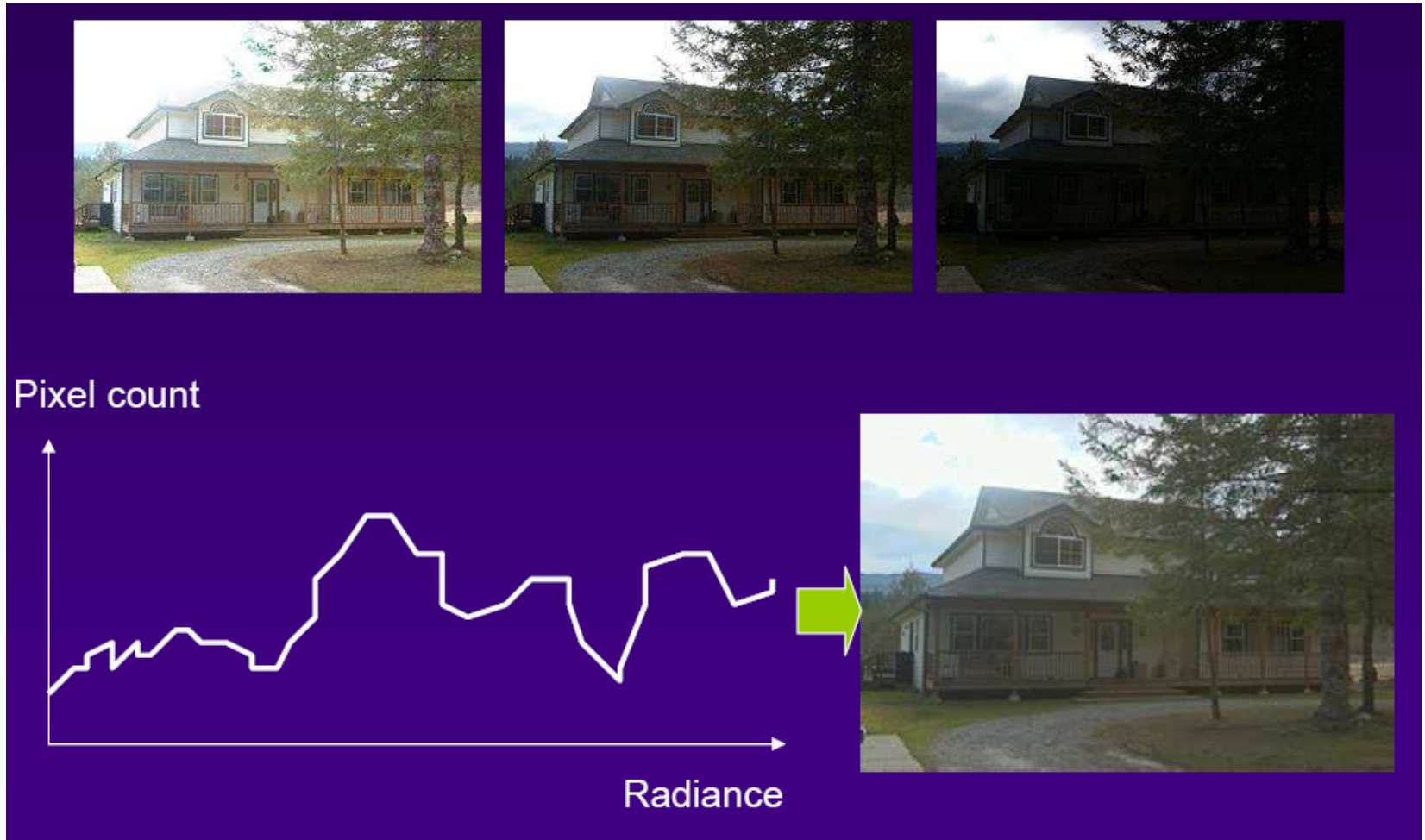
- High dynamic range images



# Image Composition Applications



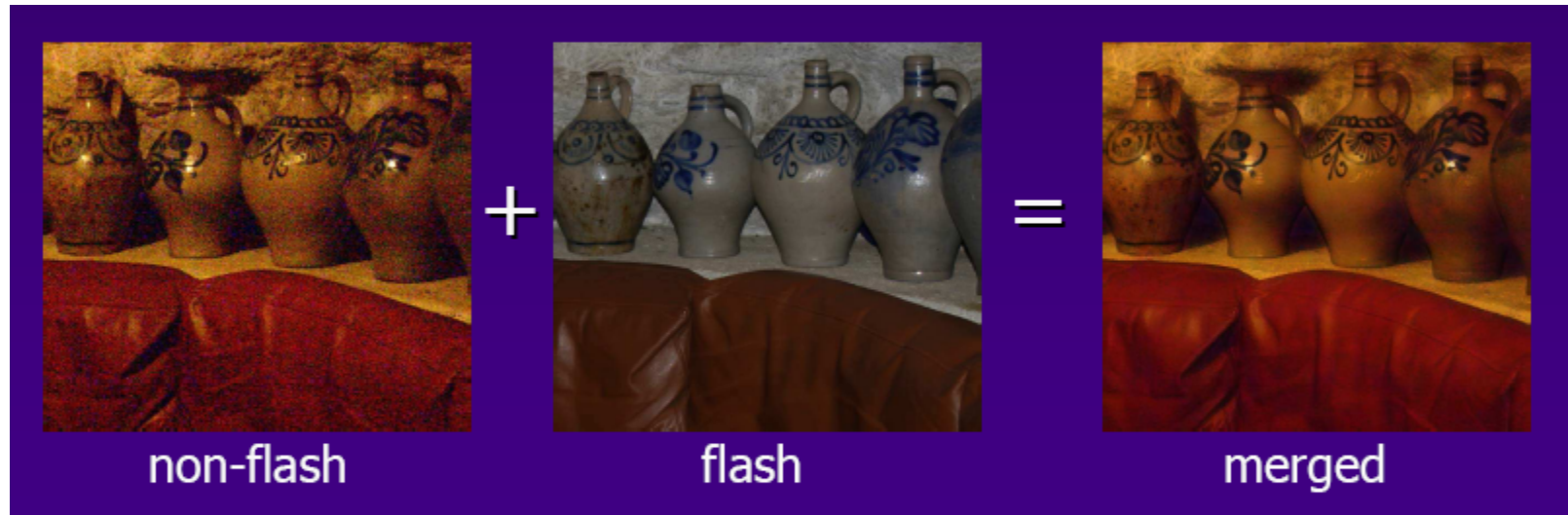
- High dynamic range images



# Image Composition Applications



- Flash / No flash



# Image Composition Applications



- Stoboscopic images

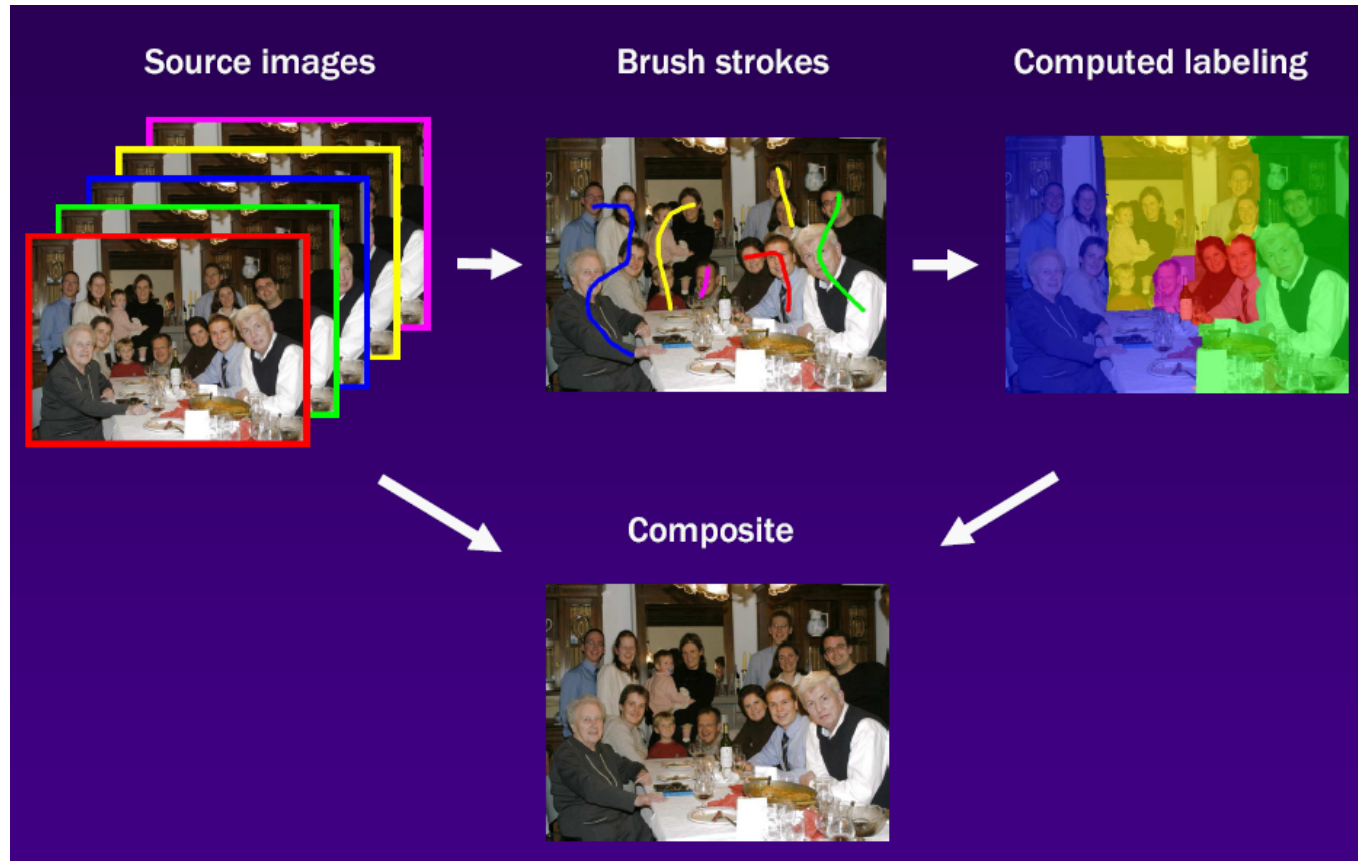


[Michael Cohen]

# Image Composition Applications



- Photo montage



# Image Composition Applications



- Photo montage

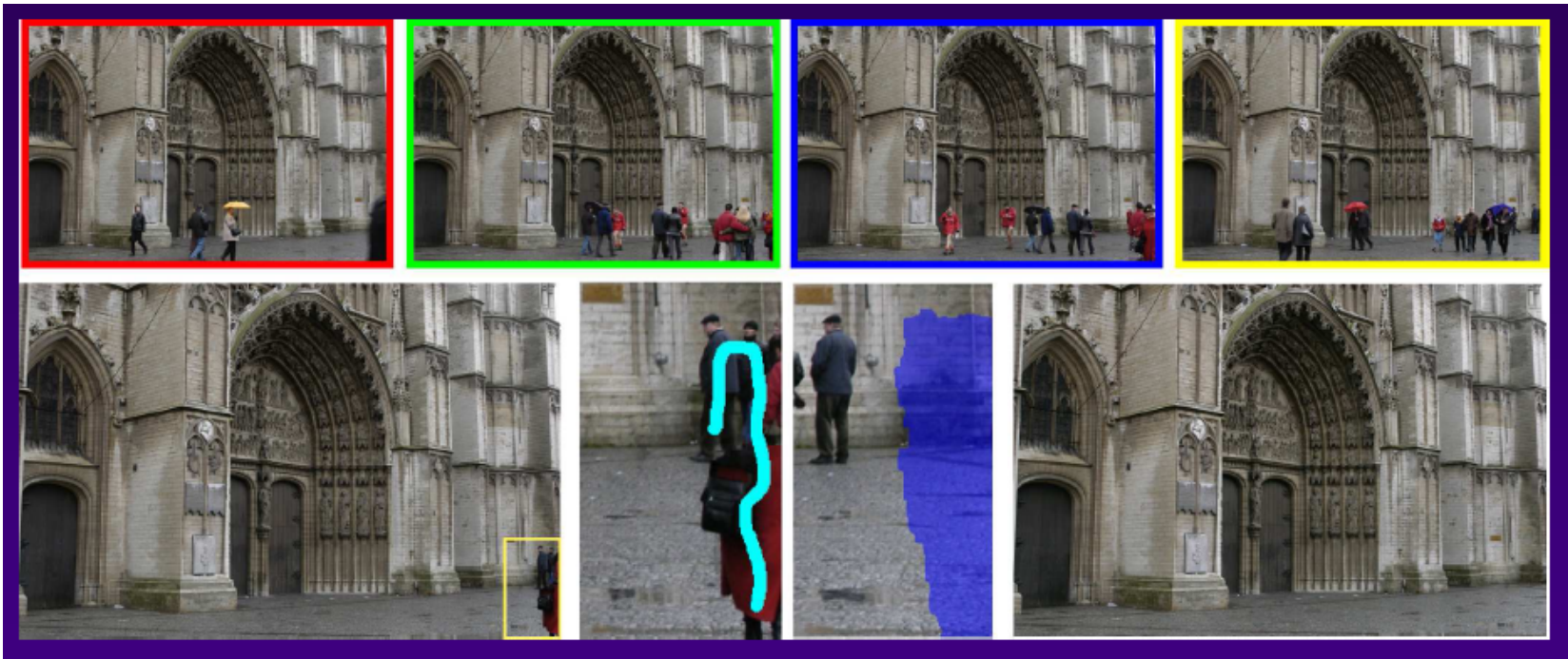


[Michael Cohen]

# Image Composition Applications



- Removing people





# Scene Completion Using Millions of Photographs

James Hays and Alexei A. Efros

SIGGRAPH 2007

Slides by J. Hays and A. Efros







# Image Completion



# Image Completion

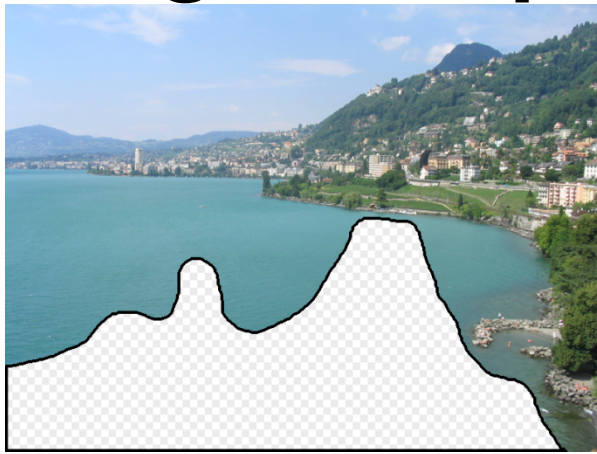
2.3 Million unique images from Flickr



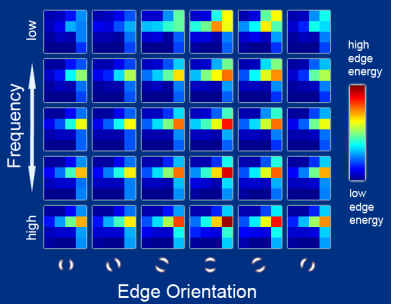


Scene Completion Result

# Image Completion Algorithm



Input image



Scene Descriptor

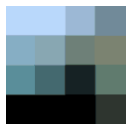
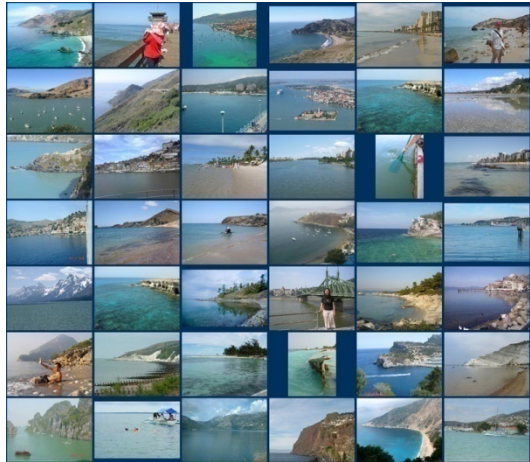


Image Collection



200 matches



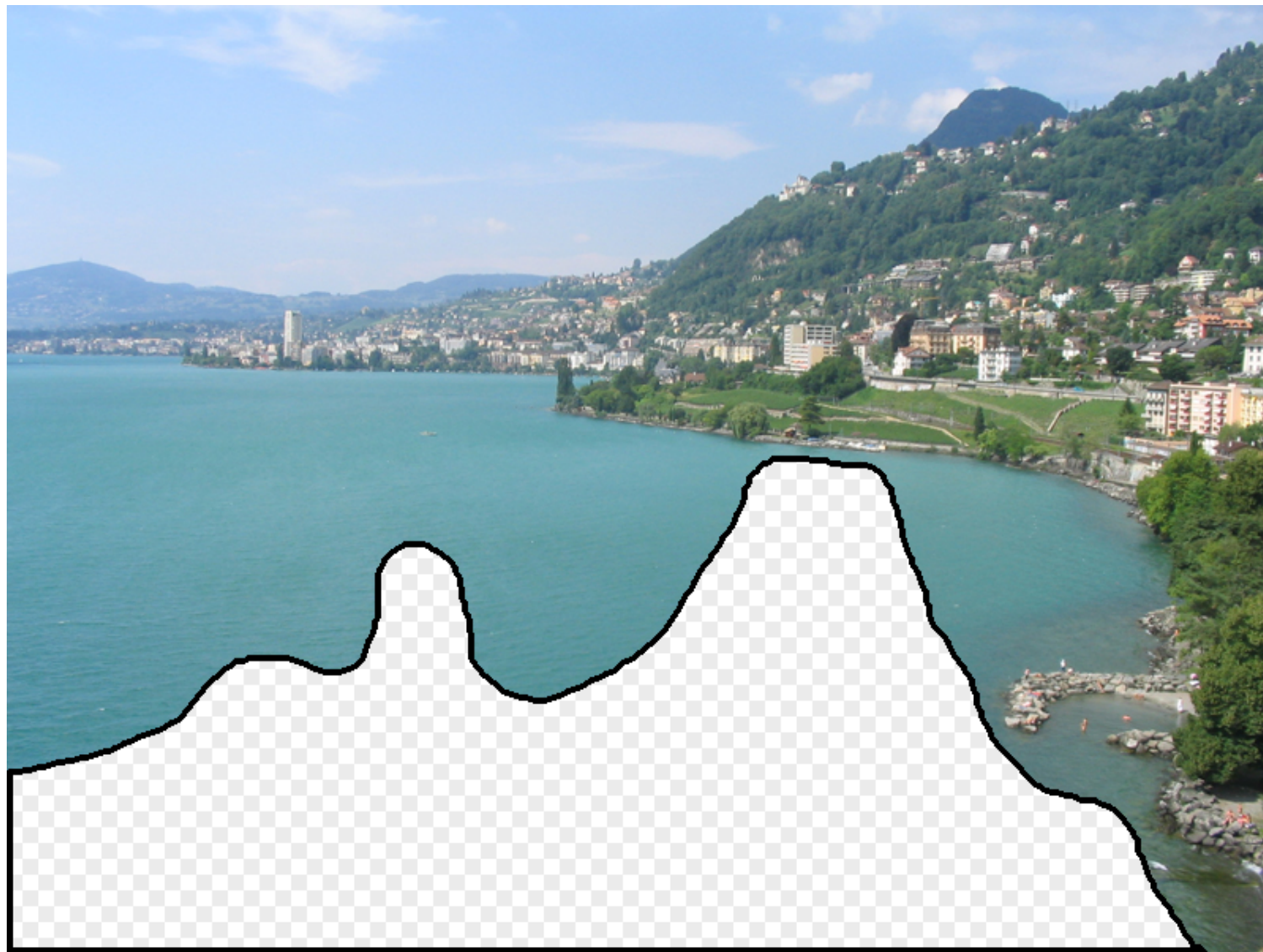
Mosaicing



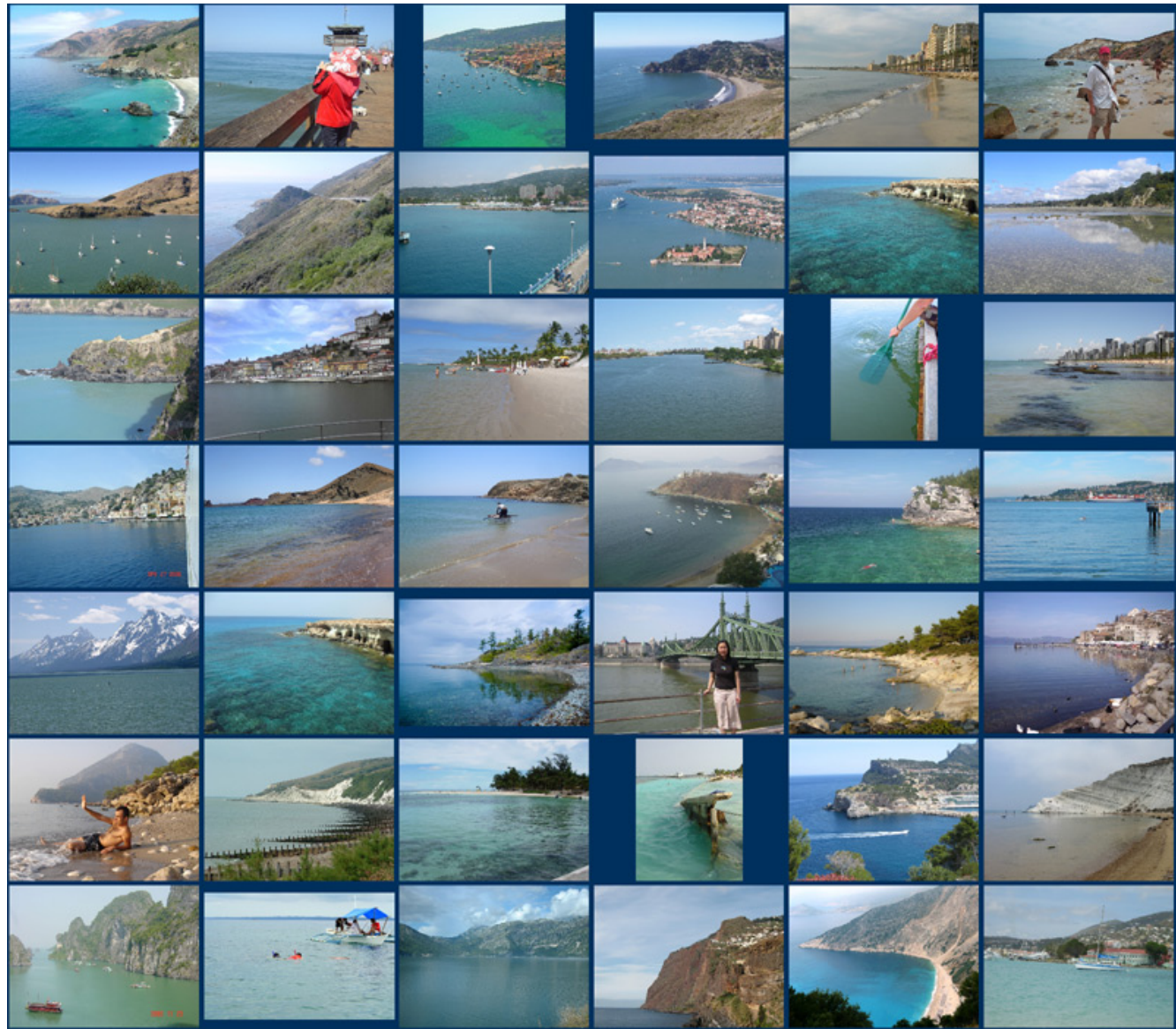
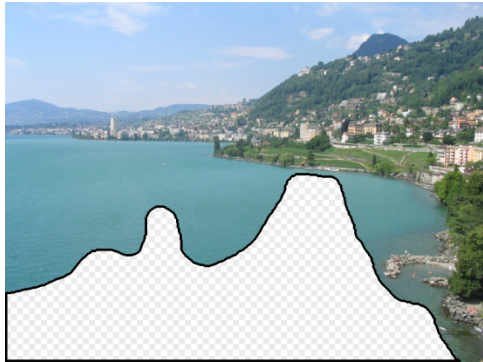
20 completions



# Image Completion



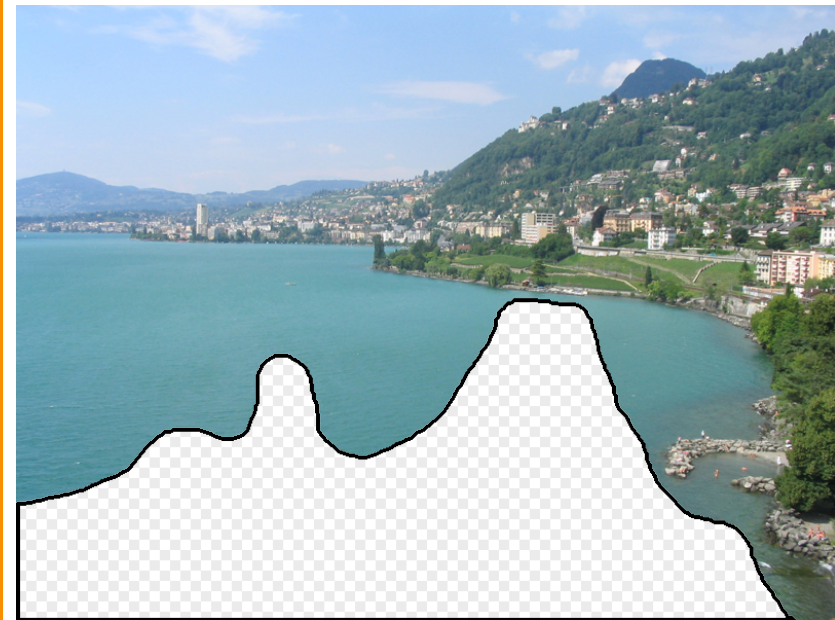




... 200 best matches

Hays et al. SIGGRAPH 07

# Image Completion

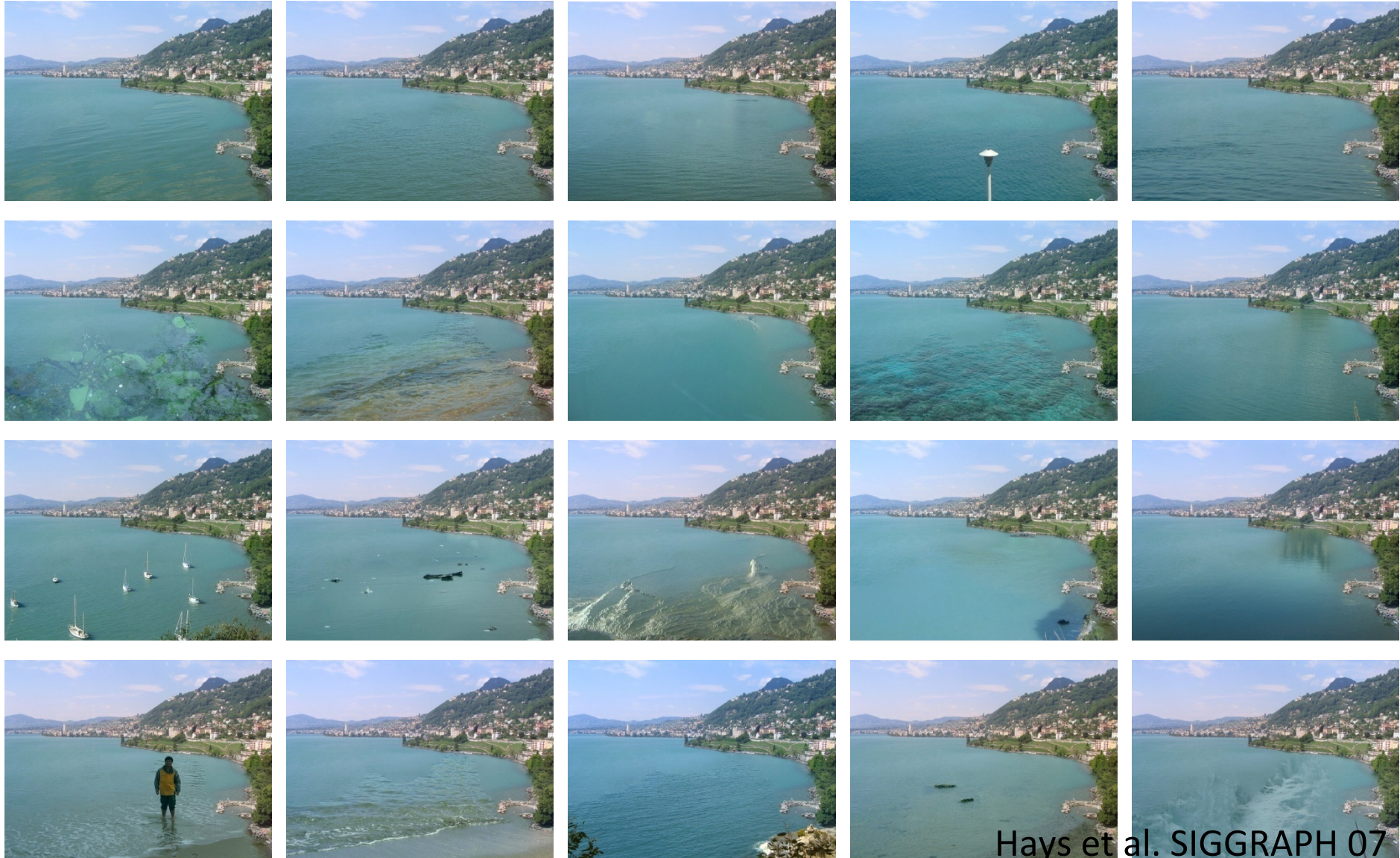




# Image Completion Result



# Image Completion Results













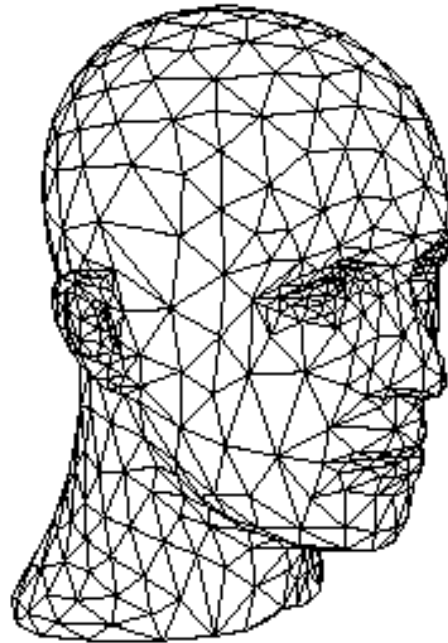


# Summary



- Image compositing
  - Alpha channel
  - Porter-Duff compositing algebra
- Image morphing
  - Warping
  - Compositing
- Computational photography

# Next Time: 3D Modeling



Denis Zorin