Image Compositing & Blending

© NASA
Image Compositing
Compositing Procedure

1. Extract Sprites (e.g. using *Intelligent Scissors* in Photoshop)

2. Blend them into the composite (in the right order)
Need blending
Need blending
Alpha Blending / Feathering

\[ I_{\text{blend}} = \alpha I_{\text{left}} + (1-\alpha) I_{\text{right}} \]
Affect of Window Size

Large windows -> lots of ghosting
Affect of Window Size

Small window → seams are too obvious
Good Window Size

“Optimal” Window: smooth but not ghosted

Medium window -> just right!
What is the Optimal Window?

To avoid seams

• window = size of largest prominent feature
What is the Optimal Window?

To avoid seams
- window = size of largest prominent feature

To avoid ghosting
- window <= 2*size of smallest prominent feature
What is the Optimal Window?

To avoid seams
- window = size of largest prominent feature

To avoid ghosting
- window $\leq 2$ * size of smallest prominent feature

Natural to cast this in the *Fourier domain*
Fourier domain blending
Fourier domain blending

Idea (Burt and Adelson)

- Compute $F_{\text{left}} = \text{FFT}(I_{\text{left}})$, $F_{\text{right}} = \text{FFT}(I_{\text{right}})$
Fourier domain blending

Idea (Burt and Adelson)

• Compute $F_{\text{left}} = \text{FFT}(I_{\text{left}})$, $F_{\text{right}} = \text{FFT}(I_{\text{right}})$
• Decompose Fourier image into octaves (bands)
  $F_{\text{left}} = F_{\text{left}}^1 + F_{\text{left}}^2 + \ldots$
Fourier domain blending

Idea (Burt and Adelson)

• Compute $F_{\text{left}} = \text{FFT}(I_{\text{left}})$, $F_{\text{right}} = \text{FFT}(I_{\text{right}})$
• Decompose Fourier image into octaves (bands)
  – $F_{\text{left}} = F_{\text{left}}^1 + F_{\text{left}}^2 + \ldots$
• Feather corresponding octaves $F_{\text{left}}^i$ with $F_{\text{right}}^i$
Fourier domain blending

Idea (Burt and Adelson)

- Compute $F_{\text{left}} = \text{FFT}(I_{\text{left}})$, $F_{\text{right}} = \text{FFT}(I_{\text{right}})$
- Decompose Fourier image into octaves (bands)
  - $F_{\text{left}} = F_{\text{left}}^1 + F_{\text{left}}^2 + \ldots$
- Feather corresponding octaves $F_{\text{left}}^i$ with $F_{\text{right}}^i$
- Sum feathered octave images in frequency domain
Fourier domain blending

Idea (Burt and Adelson)

- Compute $F_{\text{left}} = \text{FFT}(I_{\text{left}})$, $F_{\text{right}} = \text{FFT}(I_{\text{right}})$
- Decompose Fourier image into octaves (bands)
  - $F_{\text{left}} = F_{\text{left}}^1 + F_{\text{left}}^2 + \ldots$
- Feather corresponding octaves $F_{\text{left}}^i$ with $F_{\text{right}}^i$
- Sum feathered octave images in frequency domain
- Compute IFFT to get final blended result.
Fourier domain blending

Idea (Burt and Adelson)

- Compute $F_{\text{left}} = \text{FFT}(I_{\text{left}})$, $F_{\text{right}} = \text{FFT}(I_{\text{right}})$
- Decompose Fourier image into octaves (bands)
  - $F_{\text{left}} = F_{\text{left}}^1 + F_{\text{left}}^2 + \ldots$
- Feather corresponding octaves $F_{\text{left}}^i$ with $F_{\text{right}}^i$
- Sum feathered octave images in frequency domain
- Compute IFFT to get final blended result.

Actually better implemented in the spatial domain
Image Pyramids

Known as a **Gaussian Pyramid** [Burt and Adelson, 1983]

- In computer graphics, a *mip map* [Williams, 1983]
A bar in the big images is a hair on the zebra’s nose; in smaller images, a stripe; in the smallest, the animal’s nose.

Gaussian Pyramid

Figure from David Forsyth
Image sub-sampling

Throw away every other row and column to create a 1/2 size image - called *image sub-sampling*
Image sub-sampling

1/2

1/4 (2x zoom)

1/8 (4x zoom)

Why does this look so bad?
Gaussian pre-filtering

Solution: filter the image, *then* subsample
Subsampling with Gaussian pre-filtering

Solution: filter the image, *then* subsample
Compare with...

1/2

1/4 (2x zoom)

1/8 (4x zoom)
A bar in the big images is a hair on the zebra’s nose; in smaller images, a stripe; in the smallest, the animal’s nose.

Gaussian Pyramid
Laplacian Pyramid

Lowpass Images
Laplacian Pyramid

Lowpass Images

[Images of lowpass images]
Laplacian Pyramid

Lowpass Images
Laplacian Pyramid

Lowpass Images
Laplacian Pyramid

Lowpass Images

Bandpass Images
Pyramid Blending

Left pyramid | blend | Right pyramid
laplacian
level
4

left pyramid
right pyramid
blended pyramid
Pyramid Blending
Blending Regions
Horror Photo

© david dmartin (Boston College)
Season Blending (St. Petersburg)
Season Blending (St. Petersburg)
Simplification: Two-band Blending

Brown & Lowe, 2003

- Only use two bands: high freq. and low freq.
- Blends low freq. smoothly
- Blend high freq. with no smoothing: use binary alpha
2-band Blending

Low frequency ($\lambda > 2$ pixels)

High frequency ($\lambda < 2$ pixels)
Linear Blending
2-band Blending
Don’t blend, CUT!

Moving objects become ghosts

So far we only tried to blend between two images. What about finding an optimal seam?
Segment the mosaic

- Single source image per segment
- Pick a good seam (cut) so that when you merge the segments it looks ok.
Graph cuts
(simple example à la Boykov&Jolly, ICCV’01)

Minimum cost cut can be computed in polynomial time
(max-flow/min-cut algorithms)
Graph Cuts

In graph theory, a cut is a partition of the vertices of a graph into two sets. More formally, let $G(V, E)$ denote a graph. A cut is a partition of the vertices $V$ into two sets $S$ and $T$. Any edge $(u,v)$ in $E$ with $u$ in $S$ and $v$ in $T$ is said to be crossing the cut and is a cut edge.

Cost of a cut is the sum of edges crossing the cut.
Lazy Snapping

Interactive segmentation using graphcuts

Here what should the cut do?
Putting it all together

Compositing images

• Have a clever blending function
  – Feathering
  – Center-weighted
  – blend different frequencies differently

• Choose the right pixels from each image
  – Graph-cuts
Interactive Digital Photomontage

Aseem Agarwala, Mira Dontcheva, Maneesh Agrawala, Steven Drucker, Alex Colburn, Brian Curless, David Salesin, Michael Cohen
Challenges

Find good seams between parts of images so they can be joined with few visible artifacts

Blend along seams to reduce or remove any artifacts remaining after joining
Seam Objectives

Measures suitability of a seam between two image regions

- Match colors across seams
- Match colors and color gradients across seams
- Match colors across seams, but prefer seams that lie along edges
User Selection
Result of Pasting
Gradient Smoothing

Remove any remaining artifacts after merging source image parts

View input images as sources of gradient information rather than color.

Using graph cut labeling, form composite whose gradients best match source gradient vectors.
Final Result after smoothing
Extended Depth of Field

Input Images
Source Map
Extended Depth of Field
Relighting

Input Images
Source Map
Relighted Result
Visualizing Movement
Visualizing Movement
Selective Composites
Selective Composites
Selective Composites
Background reconstruction
Background reconstruction
Relighting
Compositing and Obstruction removal
Fictional Composites
Interactive Digital Photomontage

Aseem Agarwala, Mira Dontcheva
Maneesh Agrawala, Steven Drucker, Alex Colburn
Brian Curless, David Salesin, Michael Cohen