Image Analogies

Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless, and David H. Salesin
What is an image analogy?
What is an image analogy?
What is an image analogy?
What is an image analogy?
What is an image analogy?
Assumptions

• Markov random field model
  – Conditional distribution
    • Pixels relate to one another in a neighborhood system
  – $P(x_{ij}|x_{(i-1)j}, x_{i(j-1)}, \ldots)$

• What properties should we track?
Computational Complexity

- Track features
  - RGB
  - YIQ
  - Filter responses
- Index and search over A, A’, and B
  - Uses a similarity metric
Avoiding Complexity

- Images
  - A unfiltered source image
  - A' filtered source image
  - B unfiltered target image
  - B' outputted filtered target image

- Feature vector for each pixel
  - Note: Features used for A and B need not be the same for A' and B'

- For source pixel \( p \) and target pixel \( q \), \( s(q) = p \)
- Multiscale representations at each resolution
Creating the Analogy

- Compute Gaussian pyramid representations
- Compute feature vectors
- Find best match
  - At each level, compare each target pixel $q$ to every source pixel $p$
- Set feature vector in target filtered image to feature vector of best-matched $p$ in source filtered image
- Record source of target pixel [$p=s(q)$]
Finding the Best Match

- **Approximate-Nearest Neighbor Search**
  - Tries to find closest matching pixel according to $p$, $q$, and their neighborhoods

- **Coherence Search**
  - Based on Ashikhmin’s approach
  - Tries to preserve coherence with neighboring synthesized pixels

- **Coherence parameter ($\kappa$)**
  - Larger values favor coherence over accuracy
Mapping the pixels

\[ A_{l,1} \rightarrow A_1 \]
\[ B_{l,1} \rightarrow B_1' \]

\[ p \]
\[ q \]
Feature Vectors

- RGB: too many dimensions
- Alternative: YIQ
  - Use luminosity (Y) only
  - Recover color by copying I and Q channels of B to B'
- Alternative: Steerable filters
  - Detects similar orientations
Luminance Remapping

- Challenge: light A and dark B
- Solution: Luminance transformation

\[ Y(p) \leftarrow \frac{\sigma_B}{\sigma_A} (Y(p) - \mu_A) + \mu_B \]

- \( \mu_A \) and \( \mu_B \) are mean luminances
- \( \sigma_A \) and \( \sigma_B \) are standard deviations of the luminances
Traditional Image Filters
Improved Texture Synthesis

Input image
Wei-Levoy
Ashikhmin
Hertzmann et al
Super-resolution:
Training data
Super-resolution: Results
Texture Transfer
Artistic Filters
Texture-by-numbers
Texture-by-numbers
Interactive Editing
Future work

• Speed
• Estimating other statistics
• Better features
• Better color processing
• Automatic texture segmentation
• 3D models and animation
• Filter learning problems