

Image Morphing

CS-116B: Computer Graphics Algorithms
Spring 2018

Image Morphing

- Image morphing involves fluid animation to transition from one still image into another.
- Movies such as Terminator 2 and music videos such as Michael Jackson's "Black and White" used this visual effect.

Image Morphing: General Technique

- “The idea is to specify a warp that distorts the first image into the second” (p. 36).
- “As the metamorphosis proceeds, the first image is gradually distorted and is faded out, while the second image starts out totally distorted toward the first and is faded in” (p. 36).
- “The morph process consists of warping two images so that they have the same ‘shape’, and then cross dissolving the resulting images. Cross-dissolving is simple: the major problem is how to warp an image” (p. 36).

Warping Images: Methods

- Two methods for warping images (p. 36):
 - Forward mapping.
 - Reverse mapping.
- **Forward mapping:** “scans through the source image pixel by pixel, and copies them to the appropriate place in the destination image” (p. 36).
- **Reverse mapping:** “goes through the destination image pixel by pixel, and samples the correct pixel from the source image” (p. 36).

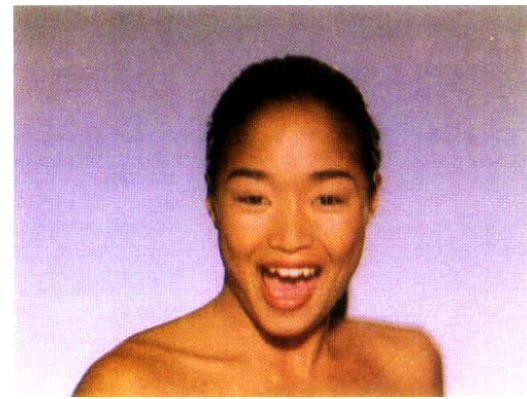
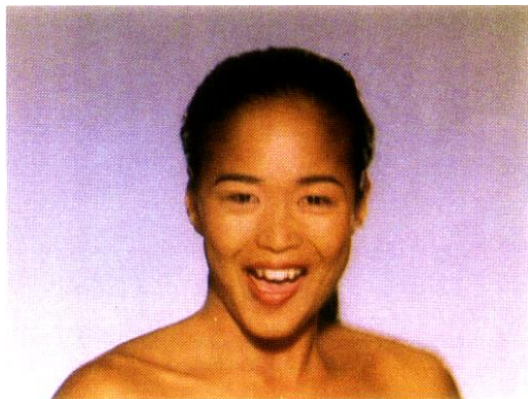
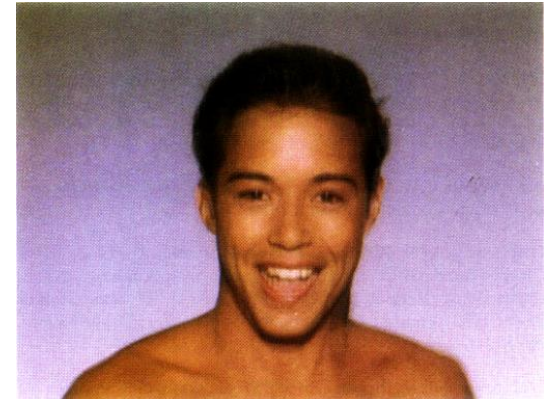
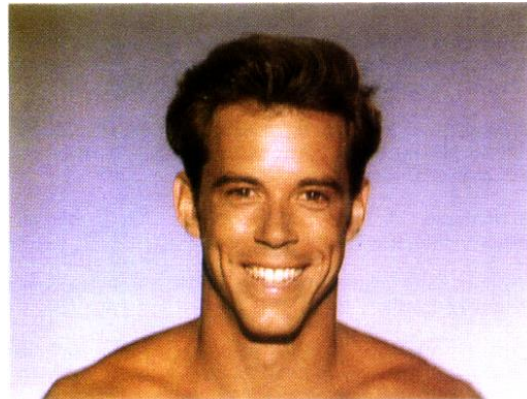
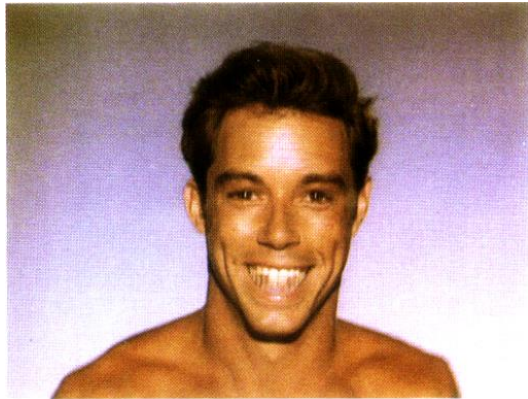
Image Morphing: Example

An excerpt from Michael Jackson's "Black or White" music video showcasing the image morphing technique discussed in today's lecture.

Note: Advance video to 5m 27s:

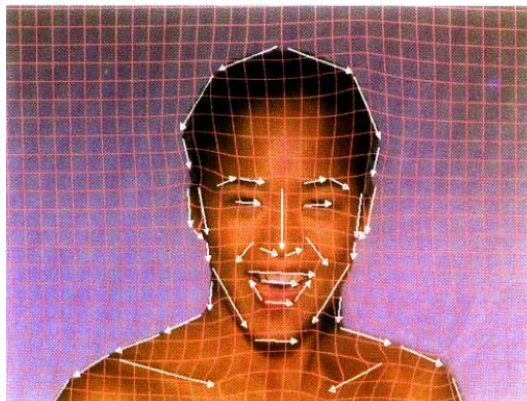
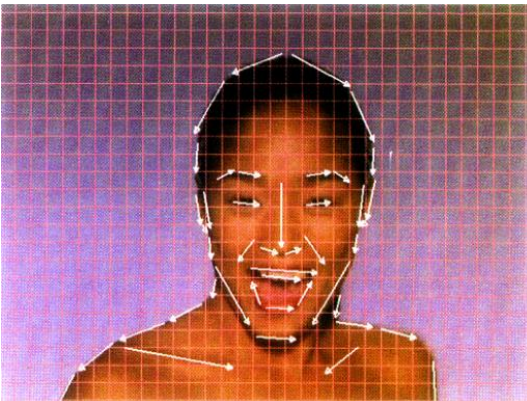
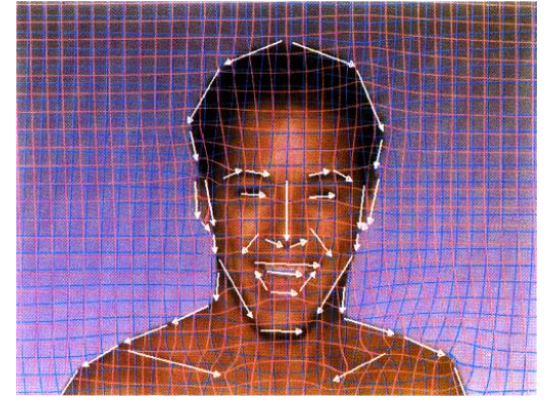
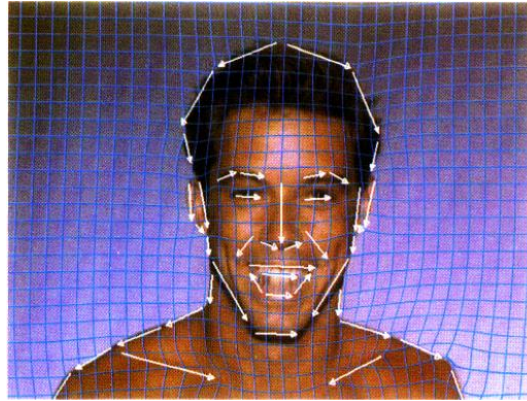
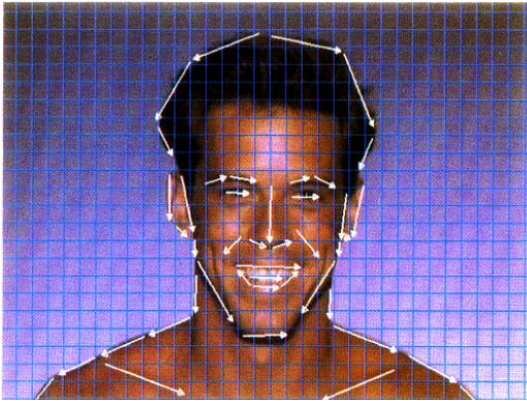
<https://youtu.be/F2AitTPI5U0>

Image Morphing: example



Source: *Feature-Based Image Metamorphosis* by Thaddeus Beier and Shawn Neely, p. 41.

Image Morphing: Reference Grid



Source: *Feature-Based Image Metamorphosis* by Thaddeus Beier and Shawn Neely, p. 40.

The Multiple Line Algorithm

For each pixel X in the destination:

$DSUM = (0,0)$

$weightsum = 0$

For each line P_iQ_i

calculate u, v based on P_iQ_i

calculate X_i' based on u, v and $P_i'Q_i'$

calculate displace $D_i = X_i' - X_i$ for this line

$dist$ = shortest distance from X to P_iQ_i

weight = $(length^P / (a + dist))^b$

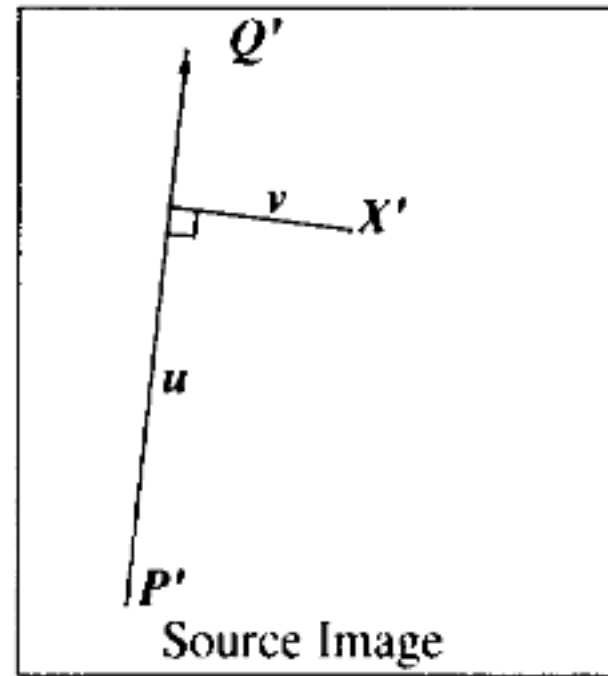
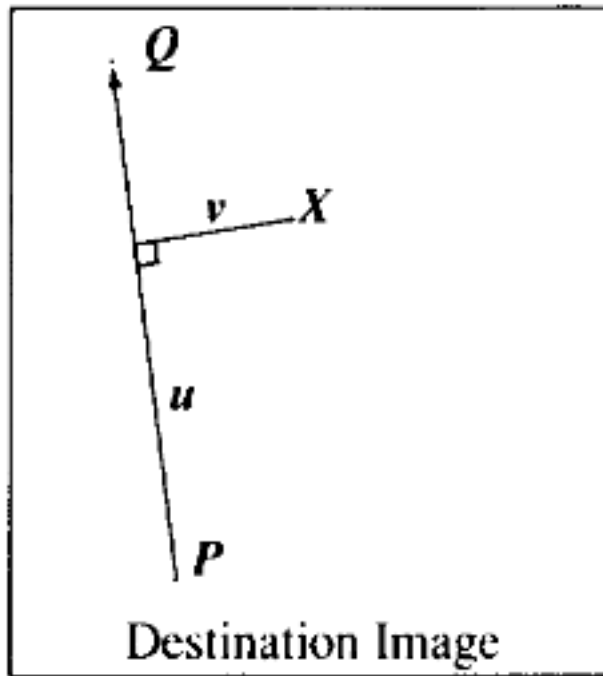
$DSUM += D_i * weight$

$weightsum += weight$

$X' = X + DSUM / weightsum$

destinationImage(X) = sourceImage(X')

The Multiple Line Algorithm



For Further Reading

- *Polymorph: Morphing Among Multiple Images* by Seungyong Lee, George Wolber, and Sung Yong Shin. <http://www-cs.engr.ccnycuny.edu/~wolberg/pub/cga98.pdf>
- *Feature-Based Image Metamorphosis* by Thaddeus Beier and Shawn Neely. <https://www.cs.princeton.edu/courses/archive/fall00/cs426/papers/beier92.pdf>