Introducing Assignment 1: Image Processing - Morphing

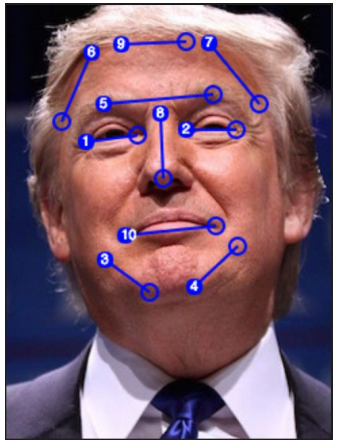
COS 426: Computer Graphics (Spring 2019)

Jiaqi Su & Carlo Rosati

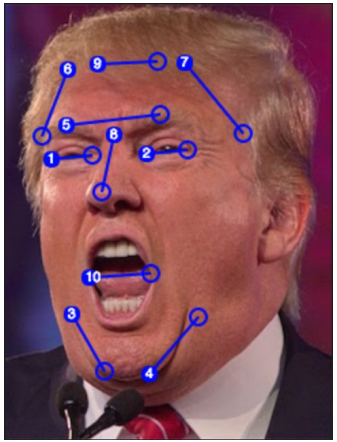
Morph

- Basic concepts
  - transform the background image to the foreground image
  - alpha = 0: show background
  - alpha = 1: show foreground
  - alpha is the blending factor / timestamp
- General approach
  - specify correspondences (morphLines.html)
  - create an intermediate image with interpolated correspondences (alpha)
  - warp the background image to the intermediate image
  - warp the foreground image to the intermediate image
  - blend using alpha

Interpolate Morph Lines



Background Image



Foreground Image

Interpolated Correspondances:

current_line[i] = (1 – alpha) * background_lines[i] + alpha * foreground_lines[i]

Warp Image

Projection of PX onto PQ

scalar
- $u = \frac{(X-P)\cdot(Q-P)}{||Q-P||^2} = \frac{(\mathbf{X}-\mathbf{P})\cdot(\mathbf{Q}-\mathbf{P})}{||\mathbf{Q}-\mathbf{P}||} * \frac{1}{||\mathbf{Q}-\mathbf{P}||}$

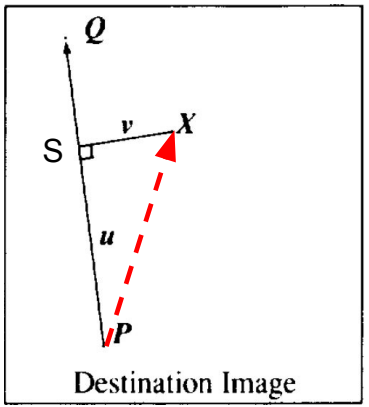
scalar
- $v = \frac{(X-P)\cdot Perpendicular(Q-P)}{||Q-P||}$ unit vector

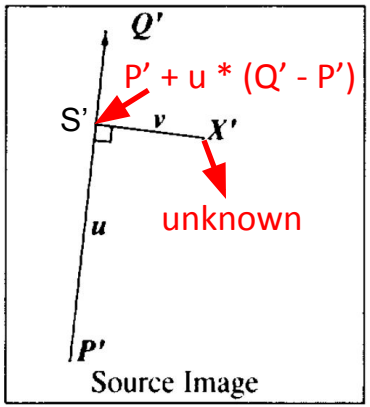
If Q - P = (x, y),
Perpendicular(Q – P) = (y, -x)

- $X' = P' + u \cdot (Q' - P') + \frac{v\cdot Perpendicular(Q'-P')}{||Q'-P'||}$ unit vector

- $dist = shortest\ distance\ from\ X\ to\ PQ$
  - 0 <= u <= 1: dist = |v|
  - u < 0: dist = ||X – P||
  - u > 1: dist = ||X – Q||

- $weight = (\frac{length^p}{a+dist})^b$
  - we use p = 0.5, a = 0.01, b = 2

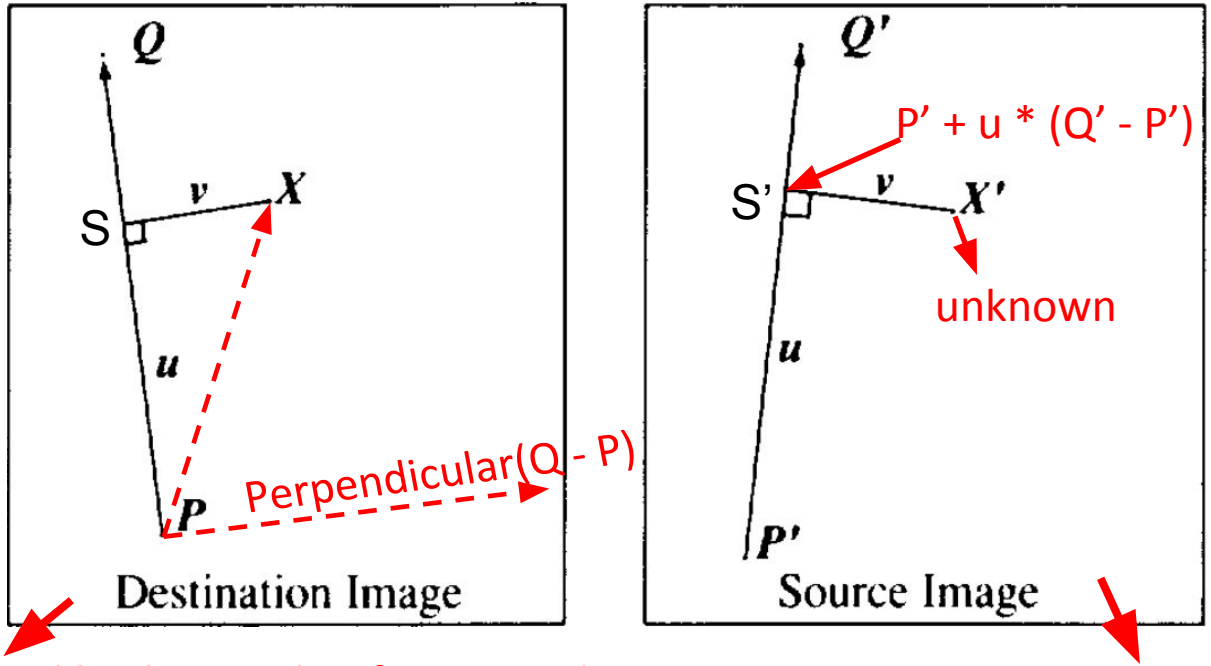




Since many morph lines, we do a weighted average of computed X'(s) for X.

Warp Image

Inverse Mapping



Let S be the projection point of X onto PQ

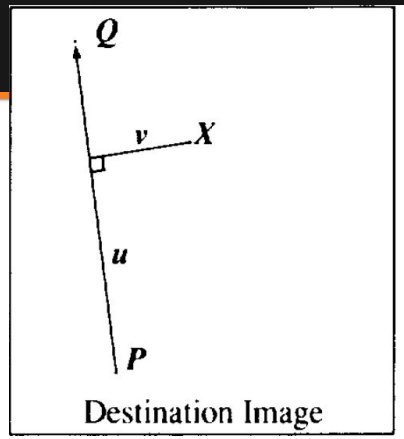
u = fraction of SP's signed length over PQ's absolute length

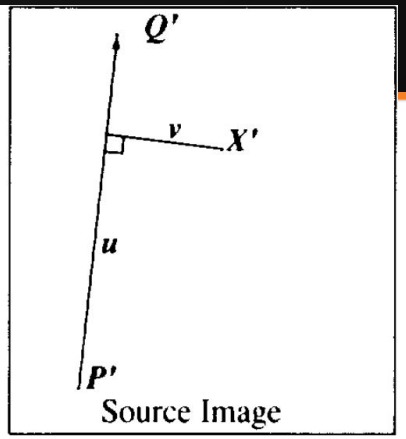
v = X's signed distance to PQ, or to say, signed length of SX

Warped background or foreground (currently black)

Pixel source (background or foreground)

Warp Image





For each pixel X in the destination
  $DSUM = (0,0)$
  $weightsum = 0$
  For each line $P_i Q_i$
    calculate u,v based on $P_i Q_i$
    calculate X'_i based on u,v and $P_i' Q_i'$
    calculate displacement $D_i = X_i' - X_i$ for this line
    $dist$ = shortest distance from X to $P_i Q_i$
    $weight = (length^p / (a + dist))^b$
    $DSUM \mathrel{+}= D_i * weight$
    $weightsum \mathrel{+}= weight$
  $X' = X + DSUM / weightsum$
  destinationImage(X) = sourceImage(X')

Blending

Background Image

WarpImage()

+

WarpImage()

Foreground Image

alpha = 0.5 (also the blending factor)

Blending

alpha = 0.5 (also the blending factor)

Background Image

Foreground Image

Morph

```
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
```

Q&A