

# Project3 Face Morphing

---

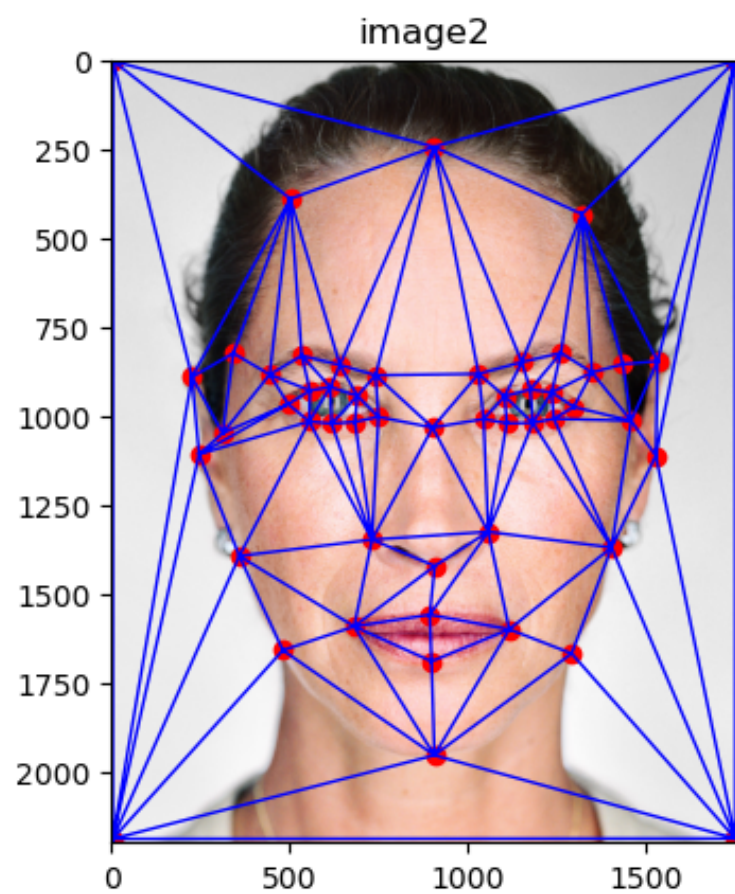
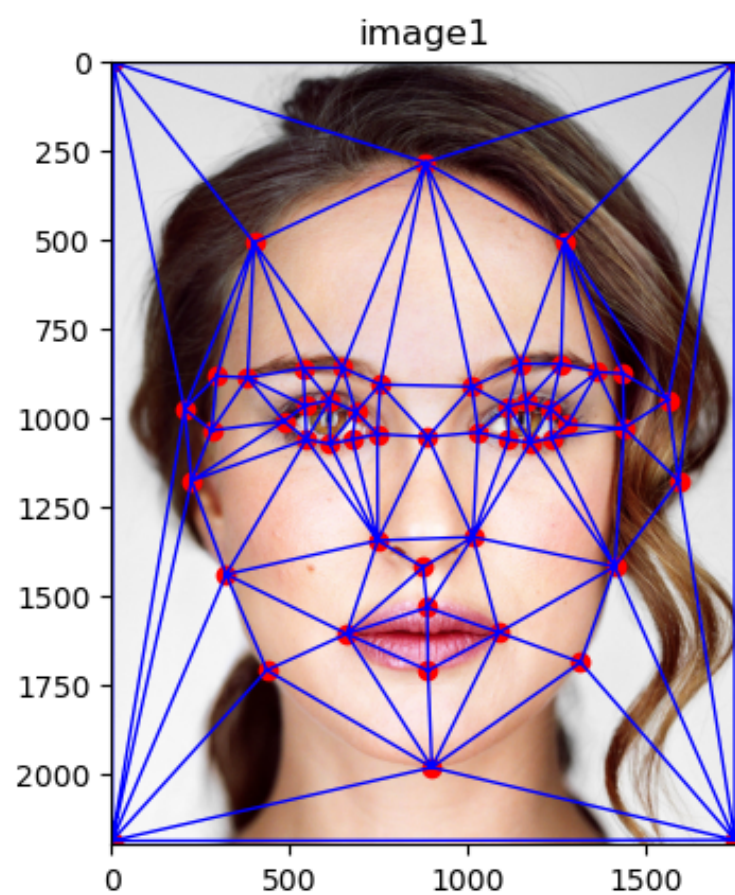
In this project, we will morph a face into someone else's face(shown in a .gif file), compute the mean of a population of faces, and extrapolate from a population mean to create a caricature face.



## Part1. Defining Correspondences

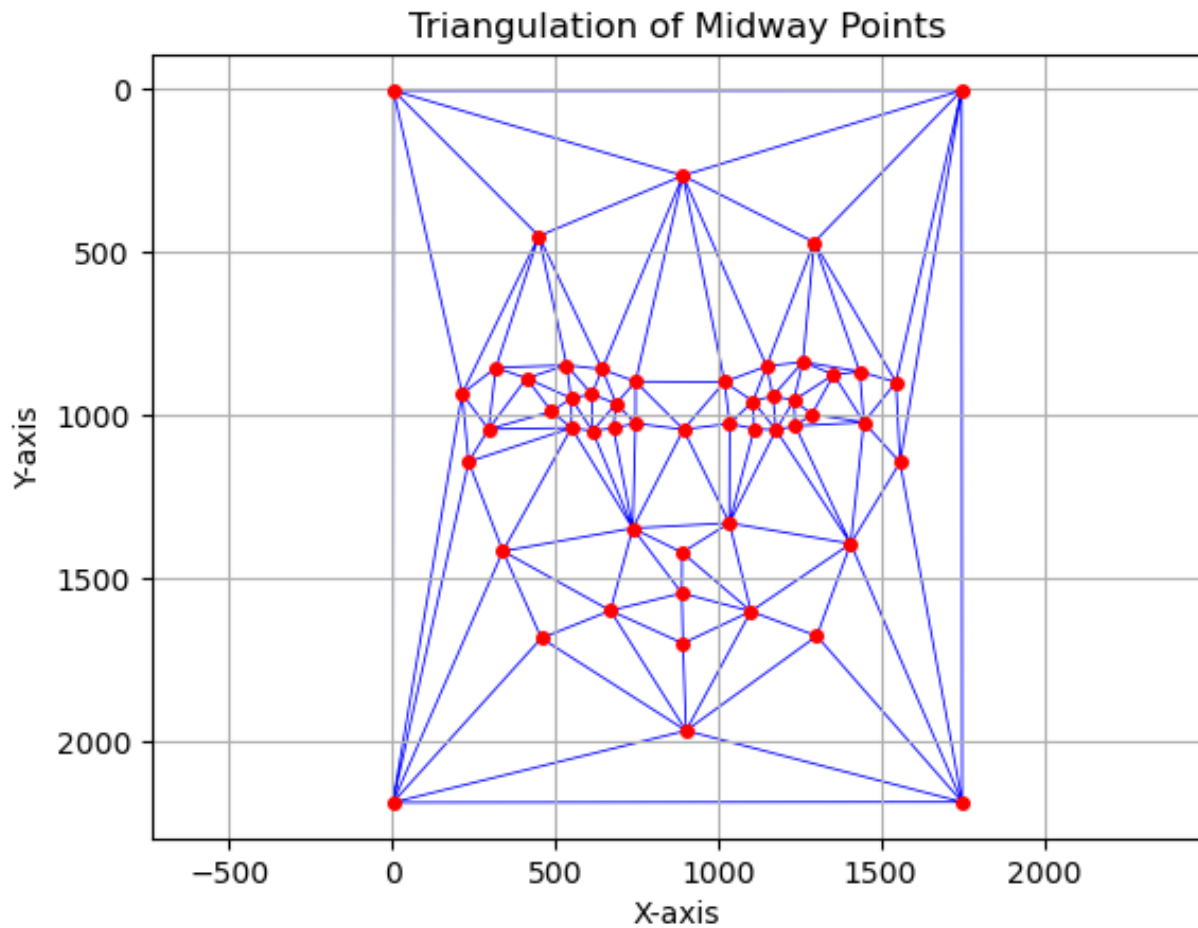
---

We choose facial key points by hand and use Delaunay() to create a triangulation mesh of these points. The figure is shown as below.



## Part2. Computing the "Mid-way Face"

Then, we compute the mid-way face by first averaging the facial keypoints of two images:

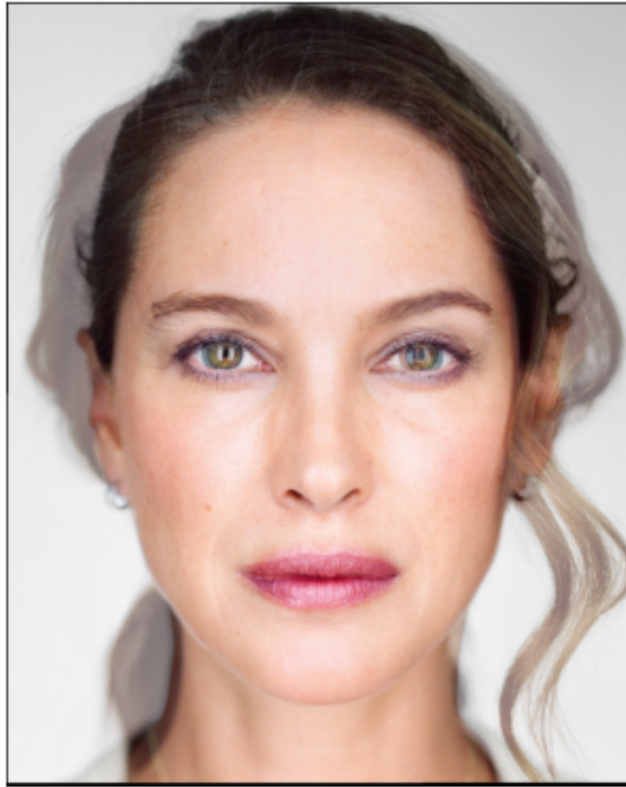


Then, we use the source and destination triangles to calculate a matrix, which is used to warp the axes within the triangle. Here is what `computeAffine(src_pts, dst_pys)` mainly performs:

$$A = \text{dst\_pts} \cdot \text{src\_pts}^{-1} \quad (1)$$

We wrap the triangles one by one.

The color change doesn't require the matrix since the two images are already aligned and can be averaged directly.



## Part 3. The Morph Sequence

---

After generating the mid-face, we could generate the images, which are wrapped from image1 to image2 slowly with 0.1 steps.



Then, use these 11 images to make a gif.



## Part 4. The "Mean face" of a population

---

Next, compute the mean face of a Brazilian population using images and key points given in their data.

Average Face of neutral Faces



Average Face of smiling Faces



Original Image



Morphed Image





Original Image



Morphed Image



Original Image



Morphed Image



Original Image

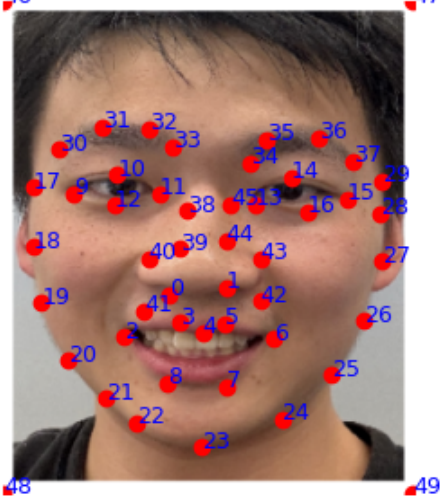


Morphed Image

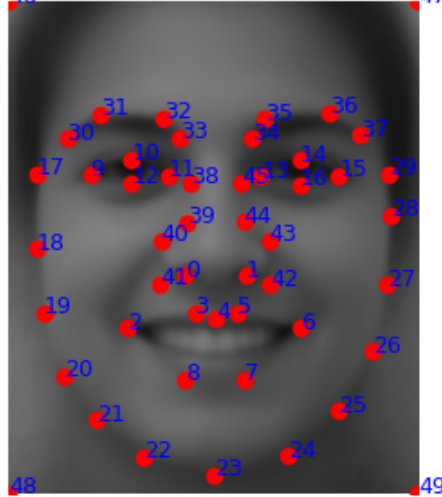


Then, warp my face into the average face's geometry and the average face into my face's geometry.

myface with Keypoints



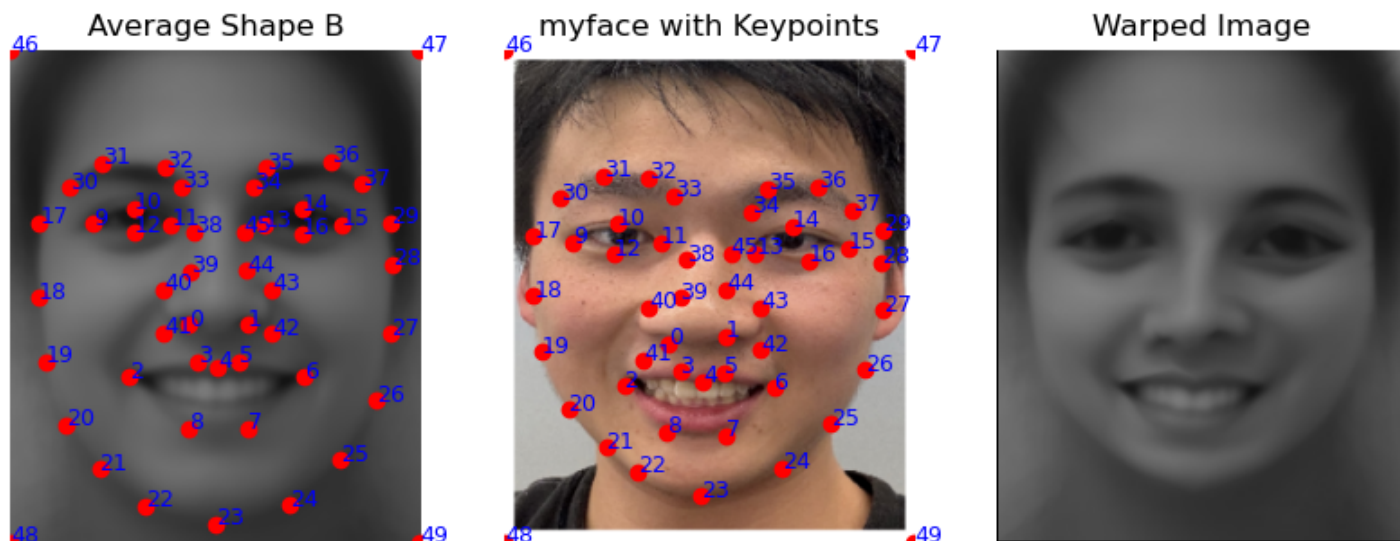
Average Shape B



Warped Image

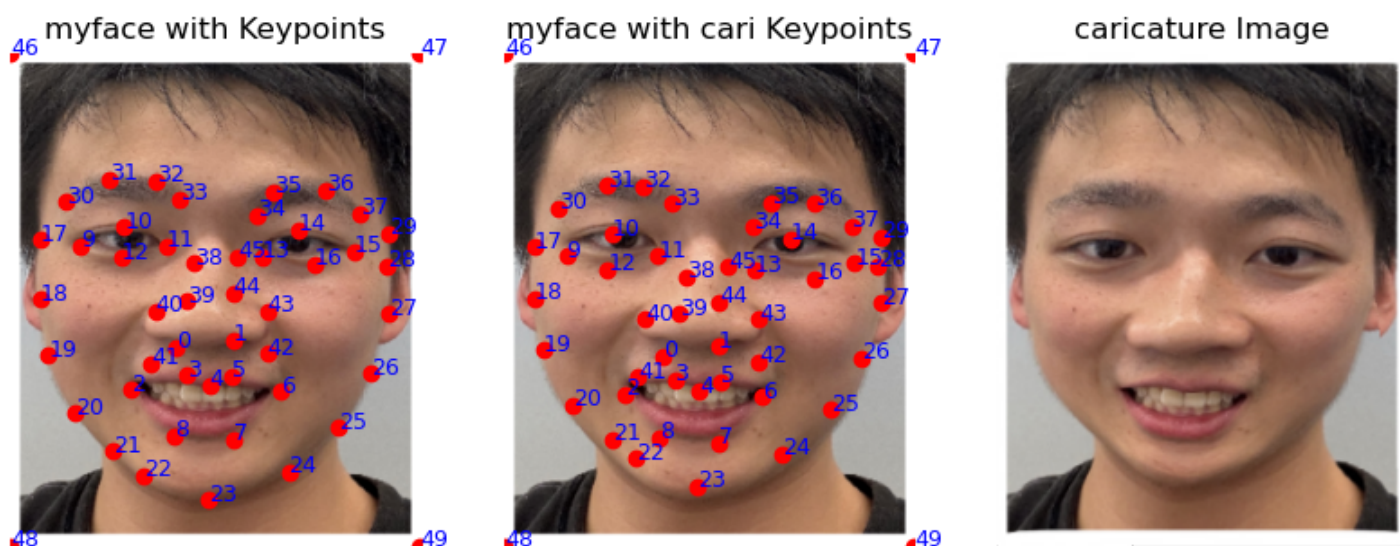






## Part 5. Caricatures: Extrapolating from the mean

Last, I compute the difference between me and Brazil people, and times 1.5 to amplification the difference.



The caricature image looks like an extra-terrestrial.