Face Similarity Metrics via Face Morphing

Ryan Baker
rtb2@illinois.edu

I. Introduction and Motivation

It can be very helpful to be able to identify the similarity or difference between two given images. One particular application that is interesting is being able to identify the similarity of two individuals (e.g. parent and child, brother and sister). This has fun implications for things like determining if a child looks more like the father or the mother, and more practical applications like family history work. So, this project aims to produce metrics that can help identify similar human faces. Along the way, this work includes an implementation of image morphing, which is a method for transforming an image of one object into an image of another object and forms a foundation for evaluating the similarity of two objects by understanding how morphing can relate one object to another using affine transformations.

II. Approach

A. Face Morphing

The first key piece of this project is implementing image morphing, specifically for two different faces, starting with gathering, cropping, and resizing images of faces. The main idea of image morphing is to find an image of the average object, not an average image. This can be done by first dividing up each image into equivalent regions. Then, a new region can be computed that is the average of two corresponding regions in two images by averaging the region vertex coordinates, thus producing an average shape of the objects. This can be done for every pair of corresponding regions. Finding equivalent regions of the face is accomplished by identifying equivalent landmarks and then creating a triangulation of these points. Examples of landmarks are the tip of the nose, corners of the mouth, or bottom of the chin. See Figure 1 for an example result of landmarks and triangulation. With an average object defined after triangulation, an affine transformation is used in order to warp each original image toward the average object. An affine transformation is able to control changes of translation, rotation, scale and sheer. This is sufficient for transforming triangles to triangles and can be easily solved for when the vertices of the two triangles are known, as in the case of our triangulation. Once the transformations are complete, the final average object can be computed by averaging, or cross-dissolving, the colors of the images. This process can be done using weighted averages to produce an intermediate object of any arbitrary balance of the two original objects. Doing this for many different weights produces a sequence of images that can create a smooth movie of morphing objects, such as faces.

B. Face Similarity

Recall that for morphing one face into another face, affine transformations are used to transition from one face to another. Also recall that these transformations control translation, scale, rotation, and sheer. So, for the purposes of morphing, a piece of two faces are the same when the transformation between the two is an identity transformation, and two faces are the same when all pieces of the face are the same. So, two faces are very similar when the transformation is not identity but does minimal work. This information can be exploited to identify metrics that can tell us about the similarity between two faces.

To begin, note that there are a number of triangles in a triangulation of an image that are not actually part of the face. These triangles are ones that rely on landmarks along the edge of the image. For the purposes of computing similarity of two faces, we will only use triangles formed exclusively by interior landmarks. It is assumed that the face in the image is square to the camera. Note, however, that faces in different images don't necessarily occupy the same pixels, nor do they occupy the same total number of pixels.

Given the above assumptions, it is possible to identify metrics based upon an understanding of affine transformations that will help compute the similarity of two faces. The first metric used is based on the angles of corresponding triangles on the face that helps encode similarity in shape of each part of the face. Let $\theta_{max,i,j}$ be the maximum angle of a triangle j of face i and $\theta_{min,i,j}$ be the minimum angle of the same triangle and face and let n be the number of triangles. Given

$$m_{max} = \frac{\sum_j \frac{|\theta_{max,1,j} - \theta_{max,2,j}|}{\theta_{max,2,j}}}{n} \tag{1}$$

$$m_{min} = \frac{\sum_j \frac{|\theta_{min,1,j} - \theta_{min,2,j}|}{\theta_{min,2,j}}}{n} \tag{2}$$

then we can compute a metric m_1 of similarity between two faces with the following:

$$m_1 = \frac{m_{max} + m_{min}}{2} \tag{3}$$

This quantity measures an average relative change of shape of each part of the face transitioning from one face to the second. A similar quantity can be computed to encode similarity in relative size of each part of the face. Recall that each face doesn't necessarily occupy the same amount of pixels in the image, so this metric is normalized according to the total pixel area of the face. Let $A_{total,i}$ be the total pixel area of the face

i and $A_{i,j}$ be the total pixel area of the face and triangle j. Given

$$A_{p,i,j} = \frac{A_{i,j}}{A_{total,i}} \tag{4}$$

then we can compute a metric m_2 of similarity between two faces with the following:

$$m_2 = \frac{\sum_j \frac{|A_{p,1,j} - A_{p,2,j}|}{A_{p,2,j}}}{n} \tag{5}$$

This quantity measures an average relative change of size of each part of the face transitioning from one face to the second. Because these two metrics are both on the same scale of percentages, these two can be combined to a summary third metric

$$m_3 = \frac{m_1 + m_2}{2} \tag{6}$$

representing an overall relative percent change in shape and size. Note that for each of these metrics, lower scores are better with 0 indicating identical faces.

C. Implementation Details

The approach is accomplished via Python3 code in both Jupyter Notebooks and python scripts. Processing of images and data is aided by basic Python packages, including scipy, numpy, opencv, and matplotlib. The identification of landmarks on faces is handled by the dlib package and a pre-trained model [1] published by the authors of that package. The triangulation of images is handled by the Delaunay method of the scipy package. Everything else including image processing, finding average objects in a morphing sequence, computing and applying affine transformations, and identifying and computing similarity metrics is original work. One unique addition not mentioned as a primary part of the approach is that the triangulation leaves out some pixels. These remain black in the final product such as with salt and pepper noise. This is solved by applying a very small median filter to fill in these missing pixels. A few methods, used for naively combining two images, recording clicks on an image, and creating a video from images are utilized from code provided by staff from this course during earlier projects [2] (see source code README for details).

III. Results

To evaluate the accuracy of the face morphing algorithm, several sequences of face morphing videos were created. See Figure 2 for a limited example of face morphing. For a fun demonstration of this part of the project, real faces with auto-detected landmarks are morphed into Miis with manually specified landmarks (and vice versa). The Miis were created via Nintendo's Mii Studio [3]. Full sequences of face morphing images can be found in the source code [4] and full videos are accessible on YouTube [5].

To evaluate the face similarity metrics, all of the metrics were computed in two different styles:

1) Individual face compared with individual face
2) Individual face compared with average face of a group

This was done for 14 different faces with 4 different groups: males, females, and two family groups. For each type of comparison, scores were averaged for each group and the corresponding opposite group. For example, for the first family group, an average similarity score (for each metric) was computed with all pairs of faces where both faces belong to the family group, and an average score was computed for pairs where at least one face didn't belong to that family group. The test is considered passing if the score of pairs exclusively inside the group is lower (better) than the score of pairs not exclusively inside the group. The full results are shown in Tables I and II. One test was executed per group for each type of comparison per metric. Tests based on type 1 comparisons passed for 2 of 4 groups for each metric. Tests based on type 2 comparisons passed for 3 of 4 groups for each metric, and the groups that failed were much closer to passing than for type 1 comparisons.

The test results show that the metrics are useful in determining the similarity of faces to a limited extent. That is, these metrics could be used for casually playing with face similarity, but certainly would not be reliable in more impactful cases, for example in legal situations. There seem to still be a variety of other factors that are significant in determining similarity of faces. Recall that these metrics focus only on the affine aspects of the face itself, and don't rely on features such as ear shape, hair length, or skin color. Including these other factors, perhaps also combined with information about genetic impacts on face similarity, could produce better results. One additional key takeaway from these results is that individual face comparisons tend to produce more outliers than average group face comparisons. This is an expected result given that averaging the faces allows for more consistent comparisons by reducing noise from outlier differences between two individual faces.

IV. Challenge and Innovation

The main two challenges associated with this project are

1) Setting up systems of equations, solving for, and applying affine transformations to faces
2) Translating knowledge of morphing, in particular affine transformations, into computable metrics

and the primary innovation lies within challenge 2. Succeeding in the challenges required understanding and applying the fundamentals of affine transformations, such as what image changes are possible in affine space. The main element of innovation is identifying ways to quantify the understanding of affine transformations given picture content and assumptions about face images. For this portion of the project, I expect to receive between 15 and 20 points out of 20 possible points. The project successfully comes up with a new idea for face similarity based on fundamental principles. The proposal definitely had risks, and the new metrics are moderately successful, not as good as hoped for in computing face similarity, but nevertheless the metrics are complete.

V. Appendix

A. Figures and Tables

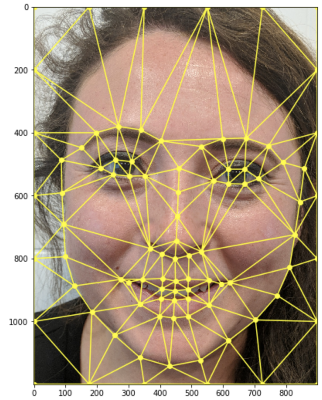

Fig. 1: Triangulation Example

(a) Face 1 Original

(b) 1/4 Morph

(c) 1/2 Morph (Average Face)

(d) 3/4 Morph

(e) Face 2 Original

Fig. 2: 5-Stage Morphing Example

Group	Metric 1	Metric 2	Metric 3
Family 1	0.113	0.174	0.144
Non-Family 1	0.240	0.400	0.320
Family 2	0.356	0.473	0.414
Non-Family 2	0.192	0.332	0.262
Male	0.118	0.216	0.167
Non-Male	0.238	0.388	0.313
Female	0.293	0.517	0.405
Non-Female	0.186	0.297	0.242

TABLE I: Type 1 (Individual) Comparison Tests

Group	Metric 1	Metric 2	Metric 3
Family 1	0.106	0.157	0.131
Non-Family 1	0.143	0.214	0.178
Family 2	0.173	0.314	0.244
Non-Family 2	0.294	0.718	0.507
Male	0.110	0.212	0.161
Non-Male	0.124	0.256	0.190
Female	0.183	0.441	0.312
Non-Female	0.170	0.348	0.259

TABLE II: Type 2 (Group) Comparison Tests

B. Self-Grading

In addition to the points specified in the Challenge and Innovation section, I expect to receive 50 out of 50 points for the completeness element of this project. The fundamental technique of morphing is fully implemented with successful results and the innovative and new metrics are fully implemented, although not as successful as hoped for. I expect to receive between 25 and 30 points for the paper completeness and clarity, including motivation, approach, results, and challenge discussions. All sections are present and concentrated effort is given to make things as clear as possible given length constraints.

References

[1] "Pre-trained face landmark model." [Online]. Available: http://dlib.net/files/shape_predictor_68_face_landmarks.dat.bz2
[2] "Cs 445 spring 2020 course." [Online]. Available: https://courses.engr.illinois.edu/cs445/sp2020/
[3] "Nintendo mii studio." [Online]. Available: https://studio.mii.nintendo.com/
[4] "Final project source code." [Online]. Available: https://github.com/rtb-illinois/rtb2-cs445-sp20-final-project
[5] "Morphing videos." [Online]. Available: https://www.youtube.com/playlist?list=PLbYxLwBJLS8cj2zsNp1fGOFNklfjE0njQ