

Final Project Proposal - Image Morphing

Eric Lee

Due: 2024-05-13

Proposal

The central focus of the project is to perform image morphing, where one image will be seamlessly transitioned into another. In the context of this project, the goal will be to create a video from images of faces such that each face will smoothly morph into each other in sequence. Our project will build up several methods over time in order to explore all the different algorithms that exist.

Given two images, the simplest way to blend two images would be to cross-dissolve them. We can use this as a baseline to compare out image blending methods. We can improve on this method by linearly transforming one image to a location that has the highest correlation, resulting in the best position to blend images. We can test if this will improve images that are similarly oriented (such as forward looking faces), and if differences in offset or rotation can be mitigated.

However our goal is to create warping that may be local, not just global, leading to our main focus: [Inverse Warping on Triangular Meshes](#). The core idea is that two images will be labeled manually with points. For example, two faces will have points corresponding to their lips, eyes, hair, and facial structure. Then, we can create a triangular mesh (using [Delaunay Triangulation](#), dual to Voroni Diagrams) between the points, and perform a linear transformation on each triangle to the other triangle and meet halfway. This would perform more local image blending. Additionally, if we save all steps of this linear transformation, we could create a video that slowly transitions between the faces.

Once we have this project, we can extend the transformations to easily transform the mesh of a face without transitioning, resulting in facing being morphed by shape. This could create caricatures of people by mapping just their mesh to existing caricatures. We could also extend this to simpler photos like flowers being arbitrarily morphed in any mesh transformation we want, simply by defining the beginning and end states.

The project would be strictly software based, requiring only images of faces (possibly from a [labeled dataset](#)), and outputting images and videos. We would not require any hardware aside from a device that can run the code.

Future Considerations / Extra Topics

The following topics are topics I would like to get to, but are most likely out of scope.

I would like to create some interface that would allow people to quickly draw meshes on images, identifying key features in order to make the software easier to use for this purpose. For the project, we may just keep it to predefined meshes stored in csv files.

The [Beier-Neely Algorithm](#) applies the same concept of “morphing meshes”, but is given a set of feature *lines* rather than points. This process is more computationally expensive, but lends to interesting topics of its own. For example, edge detection with the Laplacian method or Sobel kernels can lead to automatic generation of these lines, automating the entire process. By testing different algorithms to identify feature lines, we can look into the Beier-Neely algorithm in depth.

We can also research if there are any facial recognition tools such as machine learning models or simpler ones through [OpenCV](#) that can allow us to automatically select points in images of faces to avoid the step of manually labeling faces.