

Face Morph

Degree programme : BSc in Computer Science | Specialisation : Computer Perception and Virtual Reality
Thesis advisor : Prof. Marcus Hudritsch
Expert : Prof. Harald Studer (Optimo Medical AG)

It's always fascinating to see how peoples faces change over the years. This project was to made to visualize these changes with a video. The Face Morph program is a desktop application written in C# that automates the morphing process.

Implementation

To get a nice looking morph working we need to find corresponding points between the two images. Since we are working with faces we use facial feature points. To get those feature points, we first need to locate the face inside the image. This was done using the Haar-Cascade Classifier. The next step is to detect the facial feature points. These facial feature points are then triangulated using the Delaunay triangulation. With this triangulation we get sequence of triangles which correspond with each other. Every one of those triangles will get morphed together with its counterpart. To do this the corresponding triangles need to align perfectly, meaning that the propor-

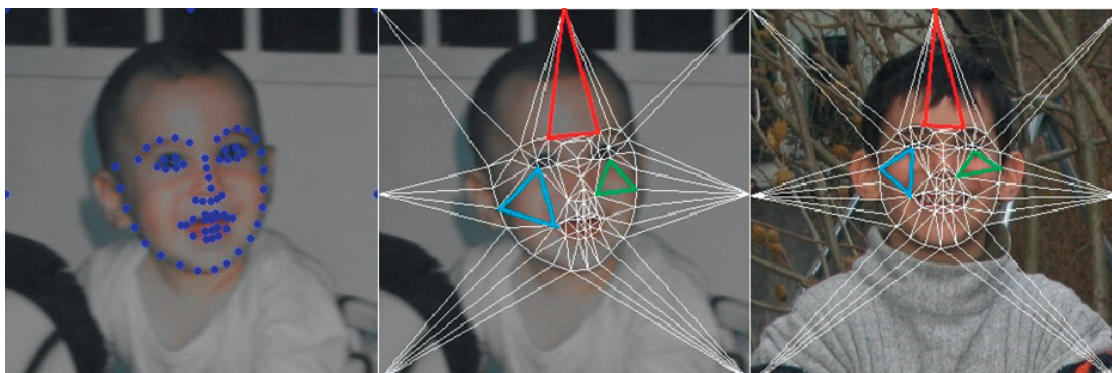
tions and sizes of those triangles need to match. We achieve this using affine transformations. In the last step the transition from the starting image to the next image is made. For this we need to calculate the pixel values of the newly created morphed image. This is done using alpha-blending.

Goal

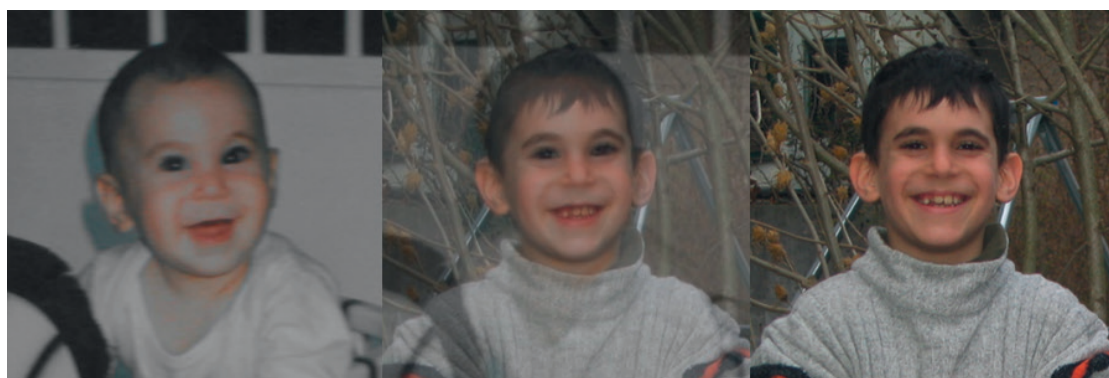
This project was created to generate a video that shows images with a face on them sequentially morph into the next one. The aim of this work was to have a program that nicely shows how people's faces change over time. I also wanted to learn how to use the OpenCV library (in this case it's C# wrapper called



Jonathan Eric Hyams



Left: Facial feature points, centre and right: Corresponding triangles to be warped together



Two faces morphed together