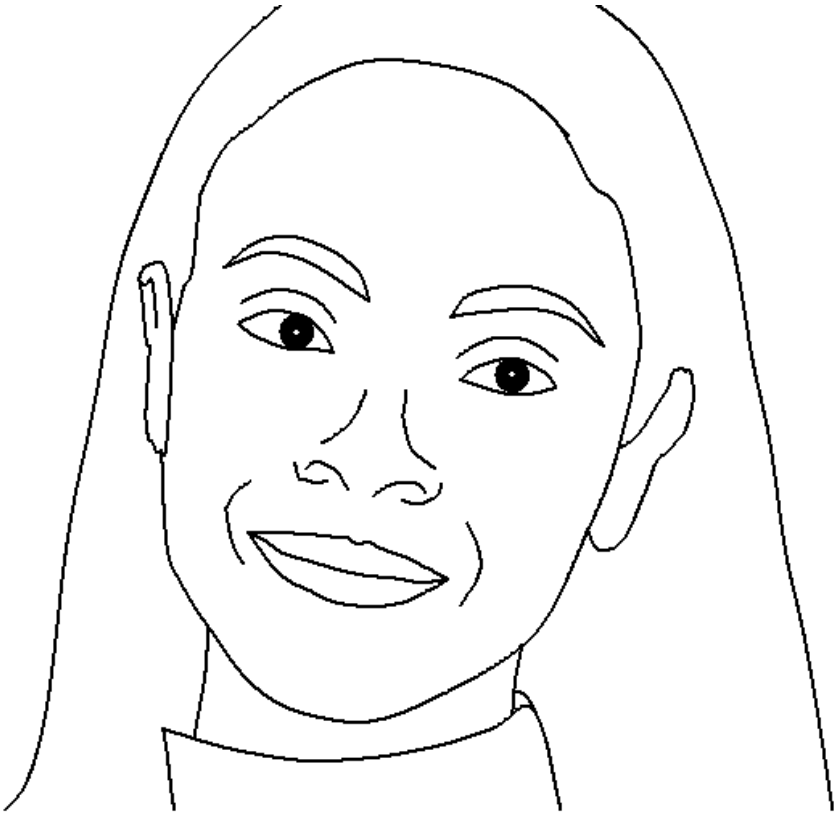
Constructing Profile from Landmarks
Keyvan Yaghmayi

Assume that we have a collection of points along the profile of a face and we want to (approximate and) draw the whole profile. Because Spline offers a flexible and precise approximation for a curve passing through given points; we can use it to connect our data points and construct the outline curve.

Example 1: The following figures show 14 points along the border of upper lip and the approximation of the outline curve by *Natural Cubic Spline*.

Example 2: Now lets consider the whole face with some landmark points. We manually chose a set of 349 data points and saved their coordinates in a *.txt* file.

The output of our program is:



Comment: Other model, which is commonly used in computer graphics, is *B-Spline*, in particular, *Non-Uniform Rational Basis Spline* (NURBS). It is easy, at least in our code, to replace our natural cubic spline with NURBS and one should consider this model especially if they want to build 3D modeling software.

The structure of our program is as the following:

- We read landmark points from a *.txt* file and save them in an array by calling method *readPoints()* from class *ReadData*.
- For each collection of points, we call method *CubicSpline()* and compute the coefficients of piece-wise defined polynomial. We write these coefficients into an another file by calling method *writeCoefficients()* from class *WriteData*.
- Class *DrawFace* extends *JPanel* and has access to these coefficients through its constructor. This class draws the profile using *paintComponent()* method.
- We create an object of class *DrawFace* and add it to our *JFrame*. Finally, class *SaveImage* saves our image.