

Face Replacement

by Mike Woods and
Michael O'Meara



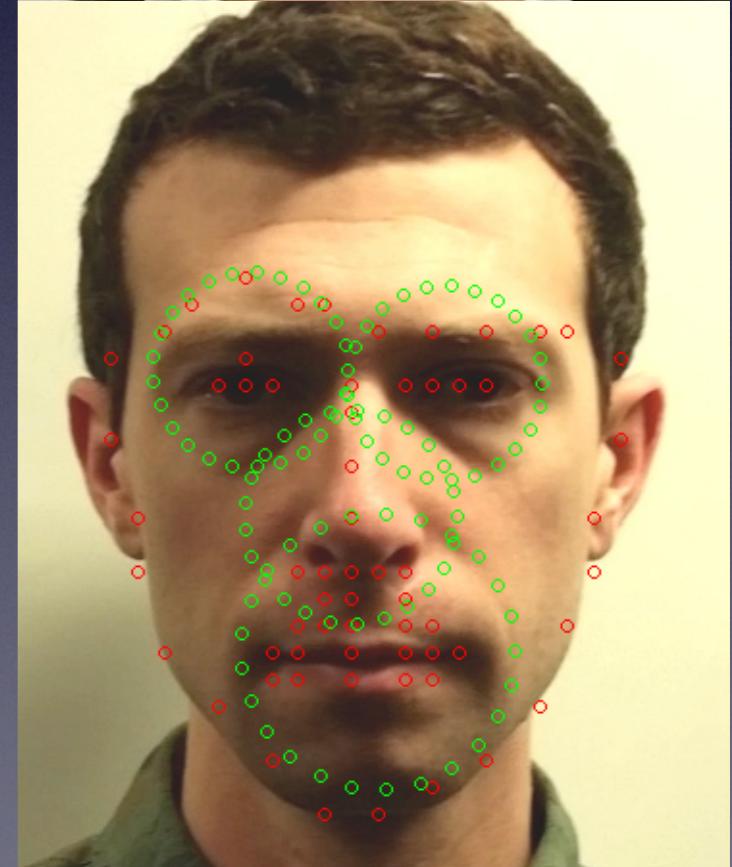
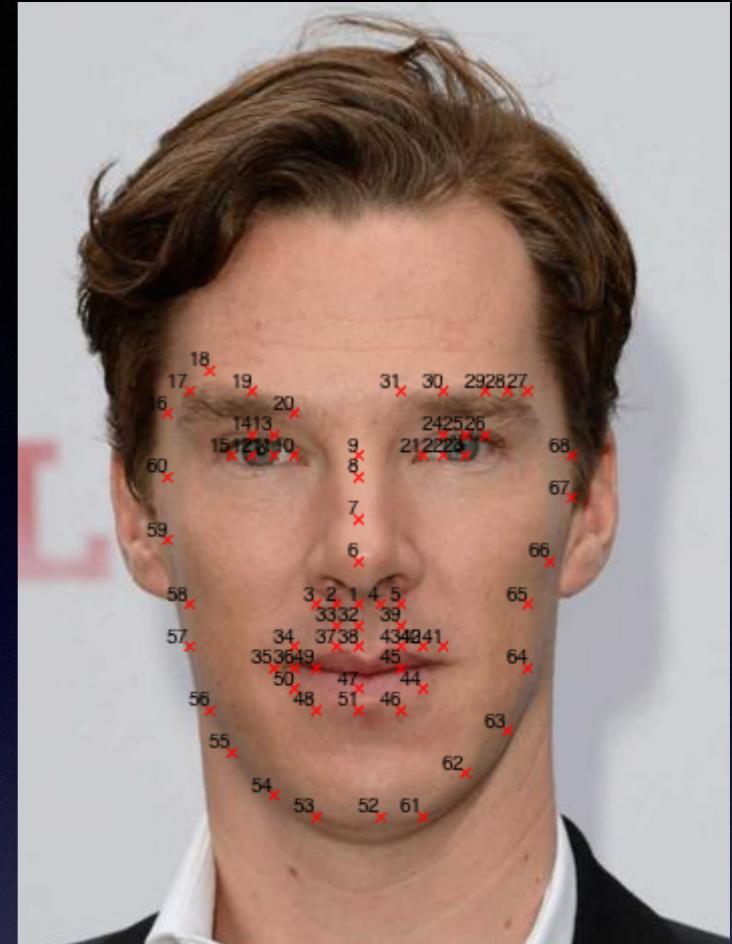
Our general approach

1. Initially we used the code published by Zhu and Ramanan based on a paper called “Face Detection, Pose Estimation and Landmark Localization in the Wild” to perform landmark detection on both the input and target faces. This approach uses a deformable template to find key features in the face.
2. We thought we’d need to use Thin-Plate Spline warping but, that turned out not to be necessary.
3. Instead, we do an affine transform to adjust the angle of our face to match the target face(s).
4. We also snapped points to “edge detected” sides of the face for better results
5. Lastly, we decided to do two different types of blending to match the surrounding target face. (feathering and poisson)

How we use deformable template code?

1. Using the 68 feature points generated by Zhu and Ramanan's software, we locate the point clusters for the eyes, nose, and mouth.
2. Using these points, we dilate the centers of those points into circles, determined by the relative distances of features on the face. Then using the convex hull, we perform point-wise matching with outliers rejected by RANSAC.
3. Then, with those circles, we calculate a mask of both the reference images and the target images using `imfill` and `poly2mask`.
4. Using the points, we create an affine transformation using `fitgeotrans` and warp the mask and the source image.

NOTE: Points the each face always have the same index assigned to them, namely those from the deformable template.



Initial Results: Not so good



Final Results: Much better

