Sangeun Lee
Thy (Tea) Tran
Team 8
CIS 581 - Computer Vision

Final project: Face Swapping

Introduction

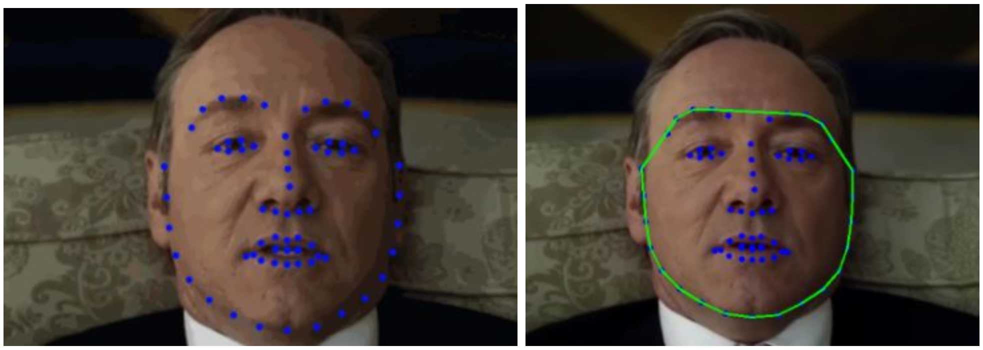
The goal for this project is to detect a face in one source video and put it on another video seamlessly. This is achieved by using facial landmarks detection, delaunay triangulation and blending.

Methodology

Our procedure for this project is divided into the following steps.

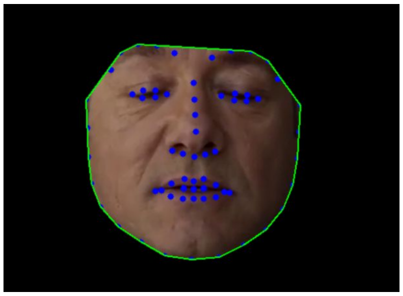
1. **Facial landmarks detection**

   Detect faces and facial landmarks in the source and replacement videos using openCV dlib library



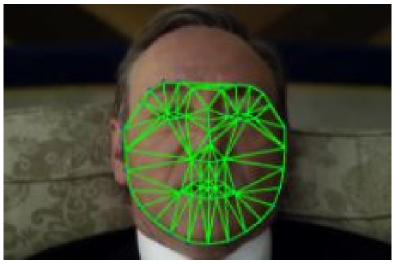
2. **Feature extraction**

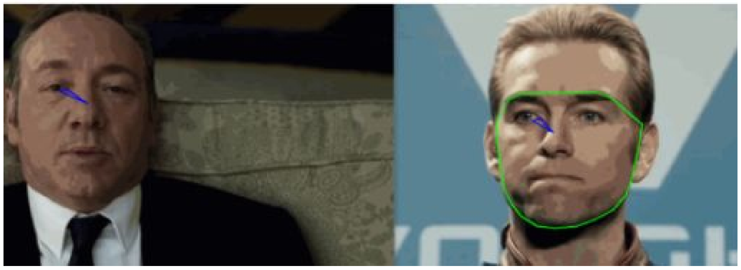
   Extracting face from the first video to later apply on the second video by using a mask.



3. **Delaunay triangulation**

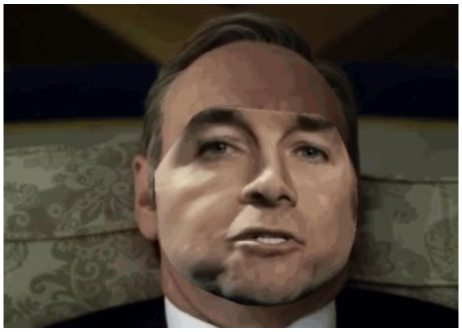
   Because the first face might not have the same dimension, orientation or resolution as the second face, we need to perform Delaunay triangulation on the first face and second face. Then, we find and transform corresponding triangles on the first face on the second face.





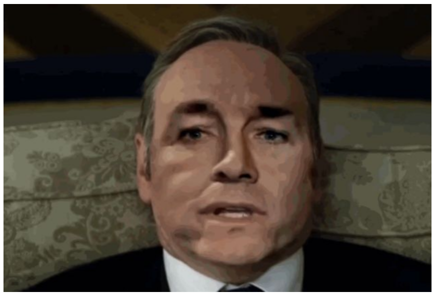
4. **Face swapping**

   After assembling all of the transformed triangles into one convexhull, we can put it onto the second video (frame by frame).

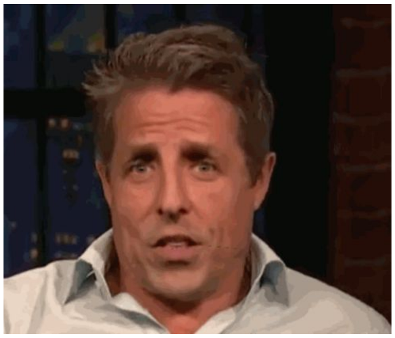


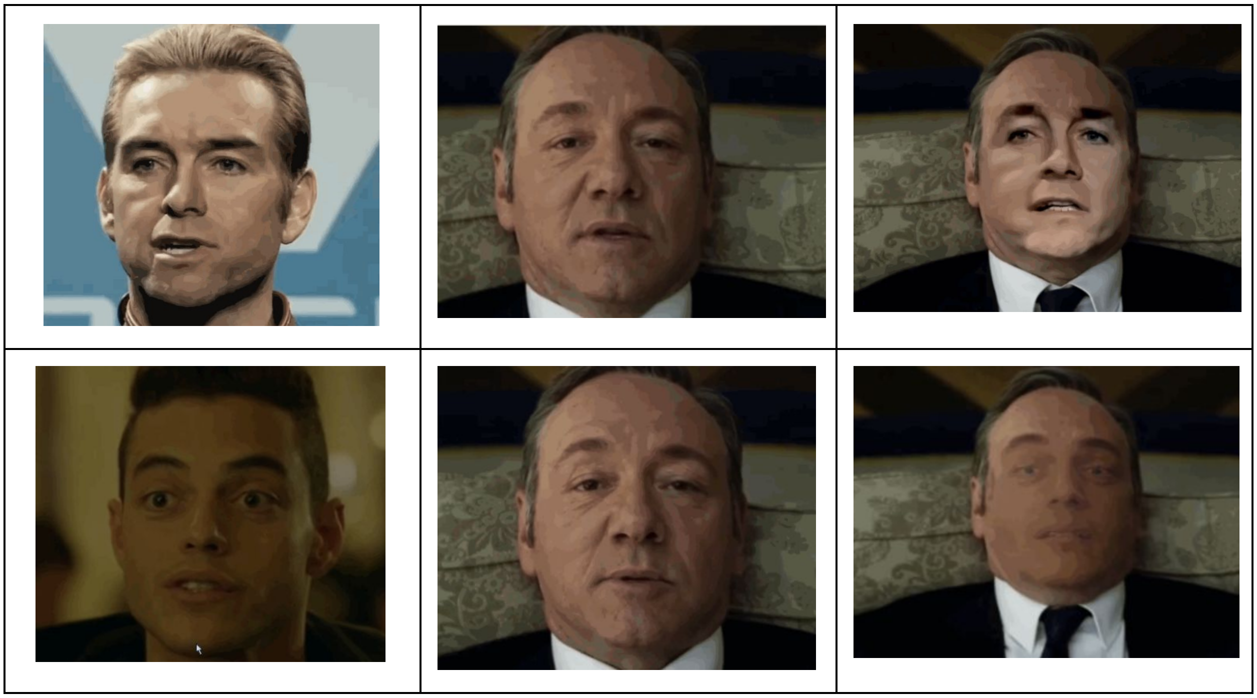
5. **Blending**

   We also perform seamless cloning to blend the first face to the head of the other video to make the face swapping look more natural.



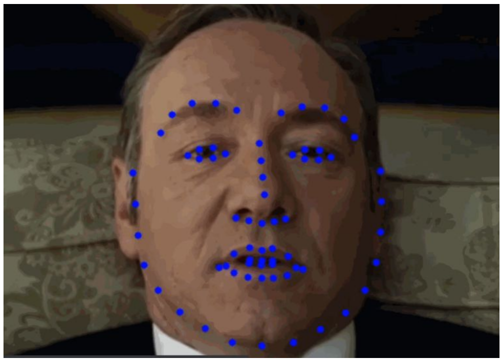
Results

Original video 1	Original video 2	Result
		

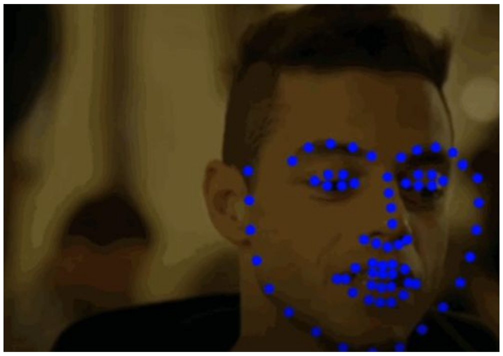


Future work

While our implementation works well with most of the tested videos, it has a hard time detecting features correctly in scenes where there is little difference in colors between non-face and face. This might be improved with a better method to find edges of face and detect features.



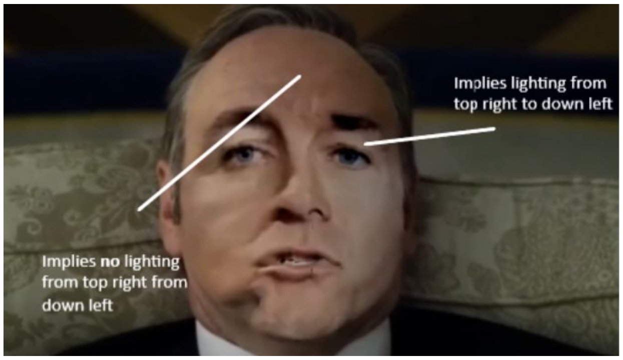
Good landmark detection (distinctive background.)

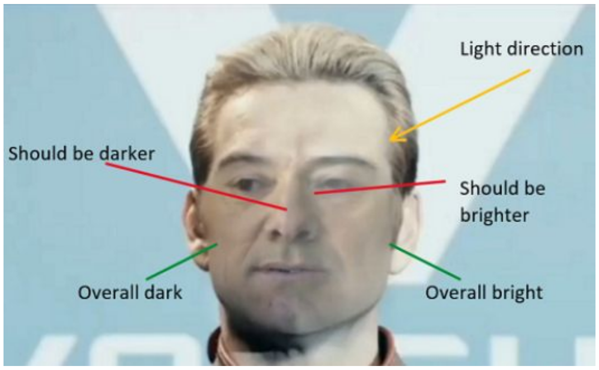


Bad landmark direction (background having similar colors to face.)

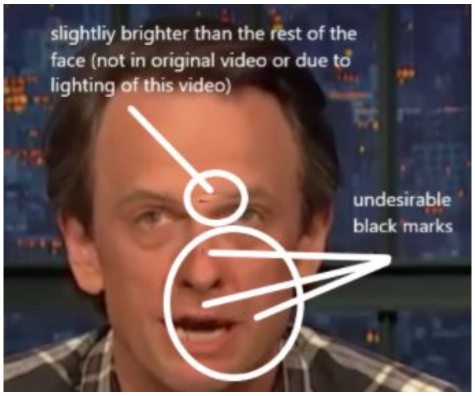
Our current implementation does not perform well on swapping faces with different skin tones. Especially in the case of pasting face with a darker skin on a head of lighter skin, this is much more obvious. This is an area that could be improved.

Our current implementation also does not handle changing light environments extremely well. If the pasted face has changing light colors, it is likely not matching with the rest of the body in the video pasted on. If the pasted face does not have changing light colors, the face overall still matches with the rest of the lighting in the scene pretty well. However, there are details on the face that should have been brighter or darker, but our implementation fails to do so. This is an area that could be improved as well with a better blending method.


Implies lighting from top right to down left
Implies no lighting from top right from down left



Some other minor details include the blending between triangles mapped from one face to another. There are at times small black marks in the output. In addition, occasionally the blending results in the center top of the swapped face has brighter lighting than the rest of the face despite not being a feature in its original video. Again, a better method for blending or masking might help to remove this.


slightliy brighter than the rest of the face (not in original video or due to lighting of this video)
undesirable black marks

References

OpenCV

Dlib