CIS581: Computer Vision and Computational Photography.
Project4, OptionB :
FACE REPLACEMENT

Pruthvi H R, Urja Nadibail (Group26)

December 2016

1 Objective

The current project attempts to replace one or more faces in a given video. The faces in a video frame are detected, the keypoints are then determined, Thin Plate Spline warping is done on the polygon hull. The contrasts of the target face with the source face are adjusted by histogram equalization and finally a Laplacian or Gaussian blend is performed for a seamless boundary. This report aims at providing an overview of the approaches and algorithms used and their analysis.

2 Introduction

Face replacement is one of those interesting and intriguing areas of vision with various applications ranging from social media to crime detection. The ways and means that one could go about to achieve this is an interesting exercise by itself. The various factors like face orientation, alignment, warping, blending et al that needed to be taken care of to successfully replace a face helped us demonstrate the concepts and skills acquired through out the course.

3 Algorithms and implementation

An overview of the approach followed by us is illustrated in figure1. The implementation was done in MATLAB with the aid of a few external libraries and mostly recycling the code from the previous projects done in the course.

* Face detection and keypoint localization:

We used the face++ library [1], which does an excellent job of detecting faces and locating the feducial features. In the given test videos, face++ API helped

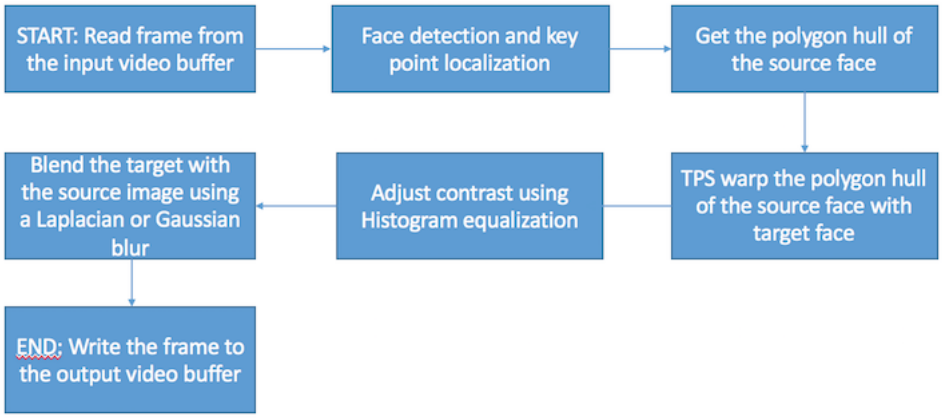

Figure 1: Pipeline of the approach

us detect faces as well as the key point features which are the feducial features like eyes, mouth, nose et al, to an average accuracy of around 90% for all the easy videos, and around 70% for most of the medium and hard videos.

* Get the polygon hulk of the source image:
The polygon hulk of the image was obtained by seam carving. A mask was created by making use of the detected face and the a mask was carved out using the lowest energy seam.

* Warping:
The target image was warped into the source image by using a Thin Plate Spline warping mechanism.

* Adjusting contrast:
Each of the Red, Green and Blue streams were considered separately and by using the mechanism of histogram equalization, the contrast of the replaced area was adjusted.

* Blending:
In order to have a seamless boundary, the replaced region was blended with a Laplacian kernel or the Gaussian kernel.

4 Analysis

Face detection: One of the key challenges was to have a highly efficient face detector, else all the other mechanisms performed after it would go for a toss.

Initially, we tried making use of the Viola Jones face detector by utilizing the library functions available on MATLAB. Even though, this worked efficiently with the individual test images, it failed to work on many frames when run on a video. On the other hand, face++ API not only was more accurate in giving the boundary box of the face, but was also better at detecting the eyes, nose and mouth. This is illustrated in figure2 and figure3

Another issue was with the orientation and the similarity of the source and target image. We observed that the replacement was more realistic when the facial features were very similar between both the images. So care was taken to ensure that the two images were similar in aspects like orientation, contrast and shape.

Blending was another aspect that added to the quality of the replaced video. It was observed that Gaussian blend performed poorly with respect to the seamless blending but was more resistant to the smaller motions in the video. Whereas Laplacian blending performed better with respect to creating a seamless blend, but was sensitive to the motion in the video. This motion sensitivity could be overcome by motion compensation. This is illustrated in figures (3) and figure (4).

One observation made with respect to the replacement target was that when there are any cases where there are people opening their mouth and talking, and thus have frames with visible teeth, it is important to have a target image with visible teeth too. Otherwise, the frame failed to look realistic to the lack of teeth when the mouth is opened. This is illustrated in figures (6) and figure (7).

5 Future work

We observed that the project was lacking in the following features and implementing them should increase the quality of the results obtained.
1. Motion compensation: This could be done using RANSAC.
2. Pose estimation and alignment: The orientation of the face could be determined and depending on that a corresponding target image would be chosen. This could be done using a set of images in different poses and using a classifier like SVM to determine the replacement pose for the given pose orientation in the target face.

6 Acknowledgement

The authors would like to thank Prf. Jianbo Shi and the TAs Jizhou Yan, Fanghan Yang and Luke Delucca for their continuing guidance and support during the course.

7 References

[1] Face++ MATLAB API *https* : *//github.com/FacePlusPlus/facepp−matlab−sdk*

K Dale, K Sunkavalli, M K Johnson, D Vlasic, W Matusik, H Pfister, Video Face replacement - *http* : *//www.eecs.harvard.edu/ kalyans/research/facereplace/VideoFaceReplace*$_S$*IGA*11.*pd*

D Bitouk, N Kumar, S Dhillon, P Belhumeur, S K Nayar, Face swappingh: Automatically replacing faces in photographs; *http* : *//www*1.*cs.columbia.edu/CAVE/publications/pdfs/Bitouk*

Overall flow: *https* : *//github.com/aniruddha*199100

Face detection: *https* : *//www.mathworks.com/matlabcentral/fileexchange/*11073*−face − detection − system*

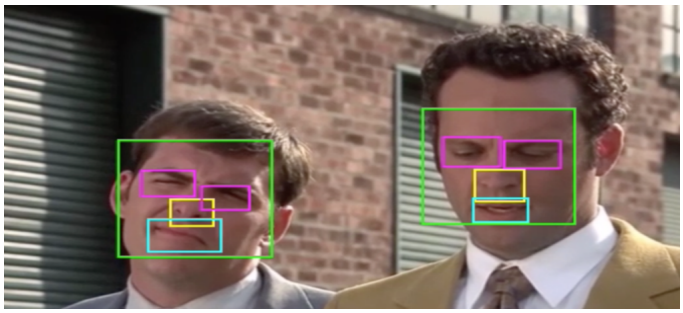

Figure 2: Face detection using Viola Jones

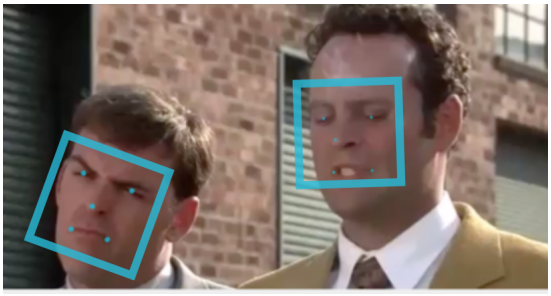

Figure 3: Face detection using Face plus plus

Figure 4: Seamless Laplacian blend

Figure 5: Gaussian blend with poor result

Figure 6: Replaced face with teeth

Figure 7: Poor Result: Replaced face without teeth