Face Swapping between Videos

Caren Jerome
MSE CIS, University of Pennsylvania
Pennsylvania, USA
carenj@seas.upenn.edu

Malathy Nagalakshmi
MSE CIS, University of Pennsylvania
Pennsylvania, USA
malathyn@seas.upenn.edu

Neha Dohare
MSE CIS, University of Pennsylvania
Pennsylvania, USA
neha75@seas.upenn.edu

Pooja Dattatri
MSE CIS, University of Pennsylvania
Pennsylvania, USA
poojadat@seas.upenn.edu

Revathi Vijayaraghavan
MSE CIS, University of Pennsylvania
Pennsylvania, USA
revathiv@seas.upenn.edu

Sharanya Venkat
MSE CIS, University of Pennsylvania
Pennsylvania, USA
sharan25@seas.upenn.edu

***Abstract*—Face swapping refers to the task of transferring the source face to the target while maintaining the target's expressions and environmental context. Through this project, we implement an efficient framework for face swapping while handling the challenges related to preservation of emotions, varying exposure, lighting and shadows, and keeping track of the number of frames of the two videos. We achieve a good level of video-realism on a variety of human faces with different identities, ethnicities, skin colors, and expressions.**

***Index Terms*—video swapping, python, cv2, dlib, sklearn, affine transformation, convex hull, delaunay triangulation, optical flow, seamless clone.**

I. Problem Formulation and Objectives

The goal of this project is to automatically detect and swap faces between two videos while preserving the source face's emotion. To obtain this goal, the following objectives are defined: Initially, the faces are detected and the facial landmarks are obtained in the source and target frame. Stabilization is performed to account for the variances that may arise due to pixel movement that happens when there is continuous motion across frames of the video. Following this, a convex hull of the detected faces is created for the purpose of feature extraction. The transformation between the two faces is then estimated to perform the swapping. After successful swapping, blending techniques are incorporated to produce refined, seamless results.

II. Related Works

There has been a lot of work done in the domain of face swapping. Several approaches have been used to achieve promising results.

J. Naruniec et al [1] present a way to generate high-resolution, photo-realistic, and temporally stable face swaps by introducing a progressively trained, multi-way comb network that embeds input faces in a shared latent space and decodes them as any of the selected identities while maintaining the input face expression. The authors have proposed a full face-swapping pipeline including a contrast- and light-preserving compositing step and a landmark stabilization procedure that allows for generating temporally stable video sequences.

L. Ma et al [2] present a method that takes a single source portrait image and a target video clip as inputs, and outputs a video-realistic video clip with the swapped source face. From the input source image and target video the system captures fine-scale 3D facial performance The appearance of the source face is harmonized to match the target video. A novel face is rendered with the source identity, harmonized appearance under the target conditions. The rendered face is blended into the warped target frame.

R. Chen et al [3] have proposed "SimSwap" - a framework for high fidelity face swapping using an autoencoder approach. Their methodology first uses an Encoder and extracts features from the target image. Then, the ID Injection Module transfers the identity information from the source image into the extracted features. The Decoder then restores the modified features to the result image.

K. Dale et al [4] have proposed a method to swap faces in two videos. They first model and track facial performances of both the source and target videos with a multi-linear method, following which they optionally re-time the videos to temporally align the performances, to synchronize the videos coarsely. The next step is to spatially align the source in the target video. An optional seam is computed through the target video to minimize blending artifacts, and the final composite video is created with gradient-domain blending.

Our approach follows the following basic steps - We first break down our videos into frames and detect the facial landmarks, we then perform frame-wise stabilization using 68-point detector. We then swap faces between the frames of the two videos using Delaunay triangulation. We then perform blending to blend the swapped face with the rest of the video and finally we stitch the frames to get the resultant video.

III. Proposed Approach

This section provides a detailed elaboration of the proposed method to efficiently implement the task of swapping faces in videos. We address the challenges of exposure, lighting, and shadows to obtain visually aesthetic results. Fig. 1. shows the steps followed to perform face swapping.

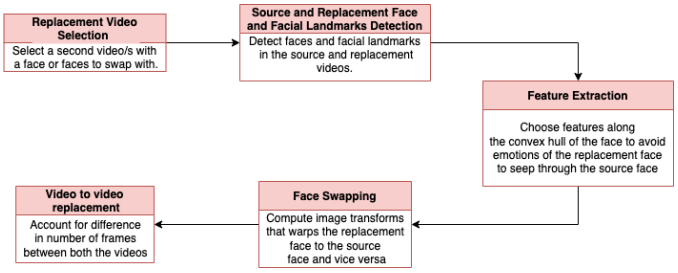

Fig. 1. Project Pipeline

A. Face and Facial Landmarks Detection

The first important step is to accurately detect faces within a given frame of the video and further proceed to identify the facial landmarks within the detected face. Facial landmark points are precisely extracted from the face images to allow for efficient swapping of faces while maintaining realism. A frontal face detector from the Dlib package is used to initially obtain the faces in an image. This is followed by a pre-trained predictor model, namely `shape_predictor_68_face_landmarks.dat`, that will extract the facial landmarks within the detected face localizing the region around the eyes, eyebrows, nose, mouth, chin and jaw to identify 68 points.

B. Stabilization

Due to continuous motion of pixels in videos, the translational and rotational variances may cause facial landmarks to be unstable. Few of the points can go missing from the features of the face giving rise to problems in feature estimation, which in turn affects the efficiency of the model and the results. To avoid this problem, a stabilization process is executed. Histogram Equalization is initially performed to adjust the global contrast by updating the pixel intensity distribution of the image's histogram. Doing so enables areas of low contrast to obtain higher contrast in the output image. The inter-eye distance is then estimated to account for facial contortions. The iterative Lucas-Kanade method with pyramids is also used as an approximation of the motion between pixels. This optical flow technique assumes that the flow is essentially constant in a local neighbourhood of the pixel under consideration, and solves the basic optical flow equations for all the pixels in that neighbourhood, by the least squares criterion.

C. Feature Extraction

Using the landmarks previously obtained, a convex hull is created that will further serve as indices for both the frames in consideration for swapping. A Delaunay Triangulation is then generated for the convex hull. The resultant mesh that is formed is then utilised during the warping and transformation processes that are executed during the final step of face swapping.

D. Face Swapping

For each of the triangles obtained on Delaunay Triangulation, the vertices are to be warped between the source and target images. A bounding rectangle is created for the triangle in the source. A mask is also defined to reflect the bounding rectangle for the triangle in the target image. Finally, the affine transformation is applied to the source image based on the affine transform calculated from the target. The triangular region from this transformed image may then be utilised for the final swapped result. Once this is done, the mask of the hull is utilised to obtain the center of the face which is required for the final refinement procedure of seamless cloning. This allows for the composed face to be seamless and natural looking. Thus, for the given ROI, distortions in color, intensity corrections, filters and slight deformations are handled effortlessly.

E. Video to Video Replacement

The procedure described thus far explains the steps to be executed for face swapping between two given frames. However, to successfully execute this task across an entire video, the pipeline is sequentially executed for all frames from the given source and target videos. To account for varying lengths between the source and target videos, the last frame from the shorter video will be utilised to populate the frames until the last frame from the longer video is reached. Finally, the individual frames are stitched back to generate the resultant videos after swapping.

IV. Experiments and Results

The following images show the results obtained with our implementation. Fig. 2. shows the facial landmarks detected for successive frames in a video. Note that despite the motion of the man in the frame, the landmarks are successfully detected along his face.

Fig. 2. Facial Landmark Detection

Using the landmarks detected, a convex hull is created and delaunay triangles are formed. The delaunay triangles formed for the source and the target images would be the same to maintain consistency as shown in Fig. 3. and Fig. 4.

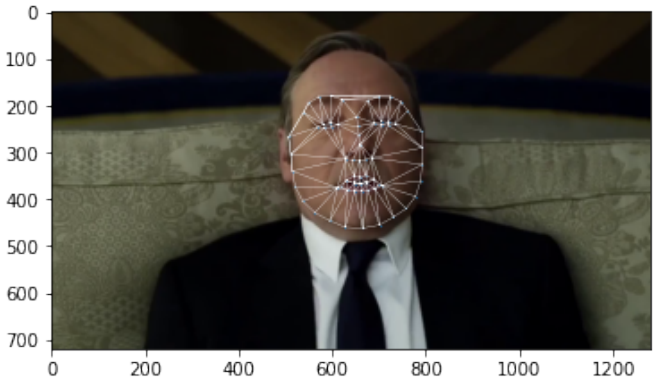

Fig. 3. Delaunay Triangulation on source

Fig. 4. Delaunay Triangulation on target

Fig. 5. shows the result of face swapping without seamless blending. We can clearly make out the difference between the source and the target face which makes the result undesirable.

Fig. 5. Result of face swap of source on target without Seamless blending

Fig. 6. shows the result of face swapping after we perform seamless blending. This makes the swapped faces look real with their new bodies.

Fig. 6. Result of face swap of source on target after Seamless blending

Fig. 7. is another example of face swapping without and with seamless blending.

Fig. 7. Result of face swap of source on target after Seamless blending

We also successfully performed Face Swapping within a single video as shown in Fig. 8. and Fig. 9. The basic flow of the algorithm remains the same as discussed before. We run the facial landmark detection algorithm, and if two faces are detected in a single frame, we know we have to swap two faces in the same video. Although the entire procedure remains the same, there were modifications to be made in the triangulation and blending steps such that one single frame with the swapped faces was returned.

Fig. 8. Two faces in the same video before swapping

Fig. 9. Two faces in the same video after swapping

V. Conclusion and Discussion

We are successfully able to swap faces between the source and the target videos. Our proposed approach can handle challenges related to preservation of emotions, varying exposure, lighting and shadows, and keeping track of difference in the number of frames of the two videos - the last frame from the shorter video is utilised to populate the frames until the last frame from the longer video is reached. We achieve a good level of video-realism on a variety of human faces with different identities, ethnicities, skin colors, and expressions. We have also successfully implemented face swapping between two faces in the same video.

Even though the results obtained for most cases are decent, we observe that for videos where the full frontal face is not visible the results obtained are not very satisfactory. Also, the code doesn't handle the case when more than one face is present in either the source or the target video.

We identified certain areas of future work. One of them is that for videos in which faces move out of view, we could track features more robustly using history of frames. We could also try swapping multiple faces in videos, and handle cases like unequal number of faces between source and target videos.

References

[1] Jacek Naruniec, Leonhard Helminger, Christopher Schroers, and Romann M Weber. High-resolution neural face swapping for visual effects. In *Computer Graphics Forum*, volume 39, pages 173–184. Wiley Online Library, 2020.

[2] Luming Ma and Zhigang Deng. Real-time face video swapping from a single portrait. In *Symposium on Interactive 3D Graphics and Games*, pages 1–10, 2020.

[3] Renwang Chen, Xuanhong Chen, Bingbing Ni, and Yanhao Ge. Simswap: An efficient framework for high fidelity face swapping. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2003–2011, 2020.

[4] Kevin Dale, Kalyan Sunkavalli, Micah K Johnson, Daniel Vlasic, Wojciech Matusik, and Hanspeter Pfister. Video face replacement. In *Proceedings of the 2011 SIGGRAPH Asia conference*, pages 1–10, 2011.