Image Face Swapping Algorithm Exploration

516030910494

June 21, 2020

1 Introduction

Face swapping is the task of transferring a face from source to target image, so that it seamlessly replaces a face appearing in the target and produces a realistic result.Since this technology is synthesized in movies, There are wide application prospects in computer games and privacy protection. This technology has caused widespread visual The attention of the image field. At the same time, the privacy and security of personal photos on the Internet are closely integrated with their related technologies.

However, the face exchange in movie videos is very complicated, and professional video editors and CGI experts need to spend a lot of money It takes time and effort to complete the face exchange in the video. So I chose to experiment with some face-changing algorithms for static pictures.

2 Method

2.1 Python

This is a easy method to make face swapping, I tried it simply to start learn the face swapping algorithm. The main experimental steps are shown below.

2.1.1 dlib

First, use the DLIB library to detect the face area of the image and extract feature points. The feature extractor (predictor) requires a rough bounding box For algorithm input, a traditional can return a rectangular list Provided by a face detector (detector), each rectangular list in There is a face in the image.

2.1.2 Procrustes

Second, rotate, zoom, pan and the second picture to match the first step. Procrustes are used to analyze and adjust the face. Two marker matrices, each row has a set of coordinates corresponding to a specific facial feature. To solve how to rotate, translate and scale the first vector, to make them fit the points of the second vector as much as possible. One idea is to overlay the second image on the first image with the same transformation. Mathematically solve this problem and find T, s and R, so that the following expression:

$$\sum_{i=1}^{68} ||sRp_i^T + T - q_i^T||^2$$

The result is the smallest, where R is a 2*2 orthogonal matrix, s is a scalar, T is a two-dimensional vector, and pi and q_i are the rows of the labeled matrix above.

2.1.3 color adjustment

Third, adjust the color balance of the second picture to fit the first picture. The Gaussian blur value by dividing im2 by im2 and then multiplying by im1 Gaussian blur value. The idea here is to use RGB scaling for color correction, but not Instead of using the overall constant scale factor of all images, each pixel has its own local scale factor. However, this method is relatively simple.

2.1.4 mix

Finally, mix the characteristics of the second picture in the first picture. Use a mask to choose which parts of Image2 and Image1 should be the final displayed image. The place where the value is 1 (displayed as white) is the area where image2 should be displayed, and the place where the value is 0 (displayed as black) is the area where image1 should be displayed. The value between 0 and 1 is the mixed area of image1 and image2.

2.1.5 defect

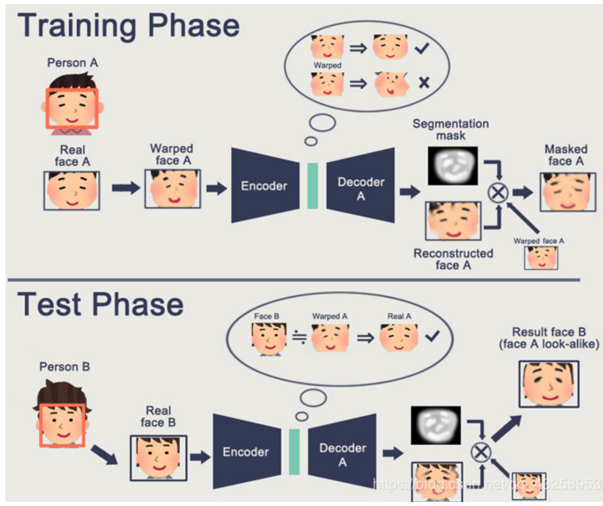
1) The inconsistency of the size of the face area. For example, it is obviously inappropriate for a fat and thin person to directly replace the operation. We need to unify the face in advance to have subsequent operations.

2) After face replacement, there will be a clear color difference between the face area and the surrounding skin tissue. Problems such as lighting make the replacement gap more abrupt, and the final processed image will be very "false"

2.2 FSGAN

2.2.1 theory

There is the basic theory of face swapping. For training phase: Face A encoding, and then restore through A decoding. face B encoding, and then restore through B decoding A/B is an encoder, and A/B uses different decoders. It can be understood that coding is to extract common features of the face. Decoding is to restore the face Personality characteristics, so A/B uses different decoders. For test phase: After B's face passes through the encoder, it is restored by A's decoder. The result is that B's face looks up Come like A, realize the face-changing action.



Encoder: Facial feature extraction realized by the convolution part five times, and a deconvolution is performed after the fully connected layer to realize the recovery of common features.

Decoder: Four consecutive deconvolutions to restore individual characteristics, and then through the residual network to achieve feature fusion, the final output.

Discriminator: The input is the original image and the decoded image, and the confrontation is achieved by resolving the network. Improve encoding and decoding capabilities.

2.2.2 FSGAN model

FSGAN consists of three main components which is shown below:

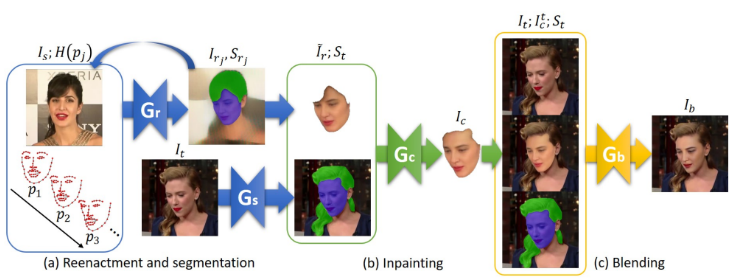


Figure 2: Overview of the proposed FSGAN approach. (a) The recurrent reenactment generator G_r and the segmentation generator G_s. G_r estimates the reenacted face F_r and its segmentation S_r, while G_s estimates the face and hair segmentation mask S_t of the target image I_t. (b) The inpainting generator G_c inpaints the missing parts of $\tilde{F}_r$ based on S_t to estimate the complete reenacted face F_c. (c) The blending generator G_b blends F_c and F_t, using the segmentation mask S_t.

1) First, based on the source image I_s and the target target image I_t, based on the 3d face landmark detector, extract the original image and the target image face frame, landmark, Euler angle, these three indicators.

2) Use a reenactment to reproduce the face pose network, and use the source image to generate the image Ir with the Euler angle of the face of the target image.

3) Separate the generated image I_r and the target image It through a segmentation model. Get the mask area of the face and hair.

4) Through an inpainting repair network, repair the face area to get I_c.

5) Finally, based on a blending fusion network, based on the target image, the restoration, and the mask image, fusion is performed to obtain the final output image I_b after face change.

2.2.3 result analysis

There are two result images.

There are some improvement in this method:

1) It is proposed to solve the interpolation problem of the same face in multi-view based on triangulation (Delaunay Triangulation) and barycentric coordinates (barycentric coordinates).

2) Stepwise consistency loss is proposed to train face reproduction, and Poisson blending loss training is proposed to merge the generated image with the original image.

3 Difficulty

1) The installation. Use Linux for better performance. CUDA Toolkit, CUDNN, the latest NVIDIA driver and PyTorch are needed.

2) Learning many kinds of Loss functions. This is hard for me to real understand it.

3) Quantitative comparison should be used more scientifically in the verification and comparison of results, that is, feature vectors are extracted, and cosine similarity is used to measure feature gaps.

4 Relation

Due to various privacy and security issues caused by face-changing technology, academia and industry began to study how to use AI technology to reversely identify the authenticity of images and videos. It has impact on studying the security and privacy of users and the degree of correlation between users in the mobile Internet .

5 Conclusion

In this major assignment, I learned some algorithm implementations related to face swapping. In fact, there are many studies on face swapping, and new methods are emerging. Correspondingly, the security and privacy detection of whether the pictures are intelligently recognizable or not is also a hot topic. I hope to follow up the study and inquiry further.

References

Nirkin Y, Keller Y, Hassner T. FSGAN: Subject Agnostic Face Swapping and Reenactment[C]//Proceedings of the IEEE International Conference on Computer Vision. 2019: 7184-7193.

Vahid Kazemi and Josephine Sullivan KTH, Royal Institute of Technology Computer Vision and Active Perception Lab Teknikringen 14, Stockholm, Sweden vahidk,sullivan@csc.kth.se

Rössler, Andreas, et al. Faceforensics++: Learning to detect manipulated facial images, 2019

Nirkin Y, Masi I, Tuan A T, et al. On face segmentation, face swapping, and face perception[C]//2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018). IEEE, 2018: 98-105.