

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN



1st Project Seminar

DeepFake: an introduction and state-of-the-art framework

Lecturer: Vo Hoai Viet

Class: Computer Vision - CSC16004_20TGMT

Contents

1. Group members.....	3
2. Overview.....	3
3. Related work.....	4
4. Application.....	5
5. Other solutions: pros and cons.....	6
6. Dataset: Celebrity Facesets by DeepFakeVFX.com.....	6
7. Code.....	8
8. References.....	9

1. Group members

Name	Student ID
Pham Nguyen Tam	20127620
Le Quoc Trung	20127369

2. Overview

- Deepfake refers to a type of synthetic media that is created using deep learning algorithms, particularly artificial neural networks. It involves the use of algorithms to manipulate or superimpose images, videos, and audio recordings in a way that makes it difficult to distinguish the fake from the original. Deepfake technology can be used to create realistic-looking videos of people saying or doing things they never actually did. The process involves training an artificial intelligence model on a large dataset of real images, videos, and audio recordings of a particular person. The model then uses this data to generate a new video or audio clip that appears to be authentic.
- **DeepFaceLab** is a powerful deep learning software application used for face swapping and face restoration. It allows users to swap faces in videos and images with remarkable accuracy and is widely used by video editors, content creators, and special effects artists. The software utilizes deep learning techniques, particularly the Generative Adversarial Network (GAN), to create high-quality face swaps that look realistic and seamless. It can detect and track faces in videos, extract facial landmarks, and replace them with another person's face while preserving the original expressions and movements.
- Minimum requirements for making very basic and low quality/resolution deep fakes: modern 4 core CPU supporting AVX and SSE instructions - 16GB of RAM - modern Nvidia or AMD GPU with 6GB of VRAM (good for up to 192 resolution models) - plenty of storage space and pagefile set to 4 x of RAM size minimum.
- **Input:** that can be photos or videos, videos are preferred due to variety of expressions and angles that are needed to cover all possible appearances of the face so that model can learn it correct, photos on the other hand often offer excellent detail and are perfect for simple frontal scenes and will provide much sharper results. You can also combine videos and photos. Below are some things that you need to ensure so that your source dataset is as good as it can be
- **Input requirement, constraints:**
 - **Videos/photos should cover all or at least most of possible face/head angles:** looking up, down, left, right, straight at camera and everything in between, the best way to achieve it is to use more than one interview and many movies instead of relying on single video (which will mostly feature one angle and some small variations and one lighting type).

- o **Videos/photos should cover all different facial expressions:** that includes open/closed mouths, open/closed eyes, smiles, frowns, eyes looking in different directions - the more variety in expressions you can get the better results will be.
 - o **Source content should be consistent (high resolution, consistent in lighting):** you don't want blurry, low resolution and heavily compressed faces next to crisp, sharp and high quality ones so you should only use the best quality videos and photos you can find, if however you can't or certain angles/expressions are present only in lower quality/blurry video/photo then you should keep those and attempt to upscale them. Upscaling can be done directly on frames or video using software like Topaz or on faces (after extraction) like DFDNet, DFL Enhance, Remini, GPEN and many more (new upscaling methods are created all the time, machine learning is constantly evolving).
 - o Keep the total amount of faces in your source dataset around 3.000 - 8.000 of image.
- **Output:**
 - o Video/image with the same resolution of input video/image.
 - o Swap face between source video/image and destination video/image.
 - o Output video/image should keep the same motion, face expression, direction with source video/image.

3. Input and output for each step

Prior to DeepFaceLab and other popular deepfake tools, existing methods for face swapping were limited in their accuracy and required significant manual effort, making the process time-consuming and often producing unsatisfactory results. DeepFaceLab was developed to address these issues and provide a more effective and automated solution for creating high-quality face-swapping videos.

Input: A large dataset of images and videos used to train the deep learning model for a specific task; settings used to configure the training process, such as the number of epochs, batch size, and learning rate. Input must be of human, has high quantity, quality, variety (poses, expressions,...)

Output: The output of the training process is a trained deep learning model that can be used for a specific face manipulation task.

- Step by step:

Data collection and preparation:

Input: Source data - a large dataset of images and videos used to train the deep learning model for a specific task.

Output: Cleaned and preprocessed dataset with corresponding landmarks and labels.

Landmark detection:

Input: Cleaned and preprocessed dataset.

Output: Facial landmarks that are used to align the faces in the source and target videos or images.

Face encoding:

Input: Aligned faces from the source and target videos or images.

Output: Encoded faces in a feature space, which captures the unique characteristics of each face.

Model training:

Input: Encoded faces and corresponding labels.

Output: A trained deep learning model that can be used for a specific face manipulation task.

Model optimization:

Input: Trained deep learning model.

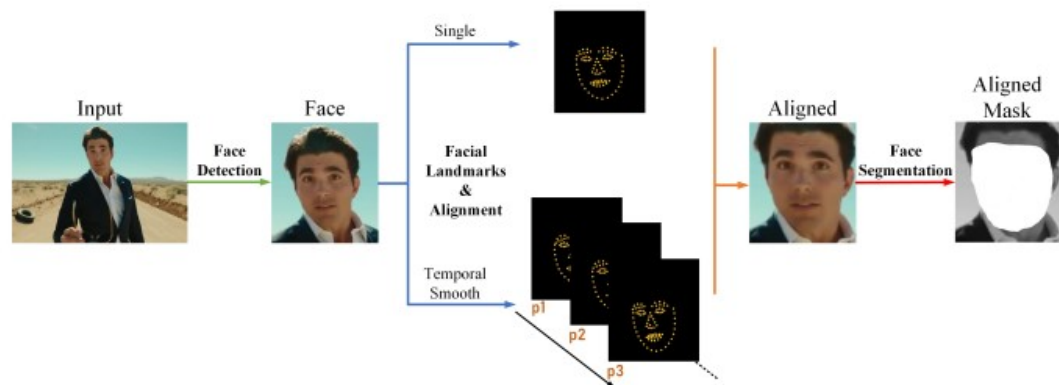
Output: An optimized model that produces more accurate and efficient results.

Testing and validation:

Input: Test dataset and trained model.

Output: Evaluation metrics, such as accuracy and loss, to assess the performance of the trained model.

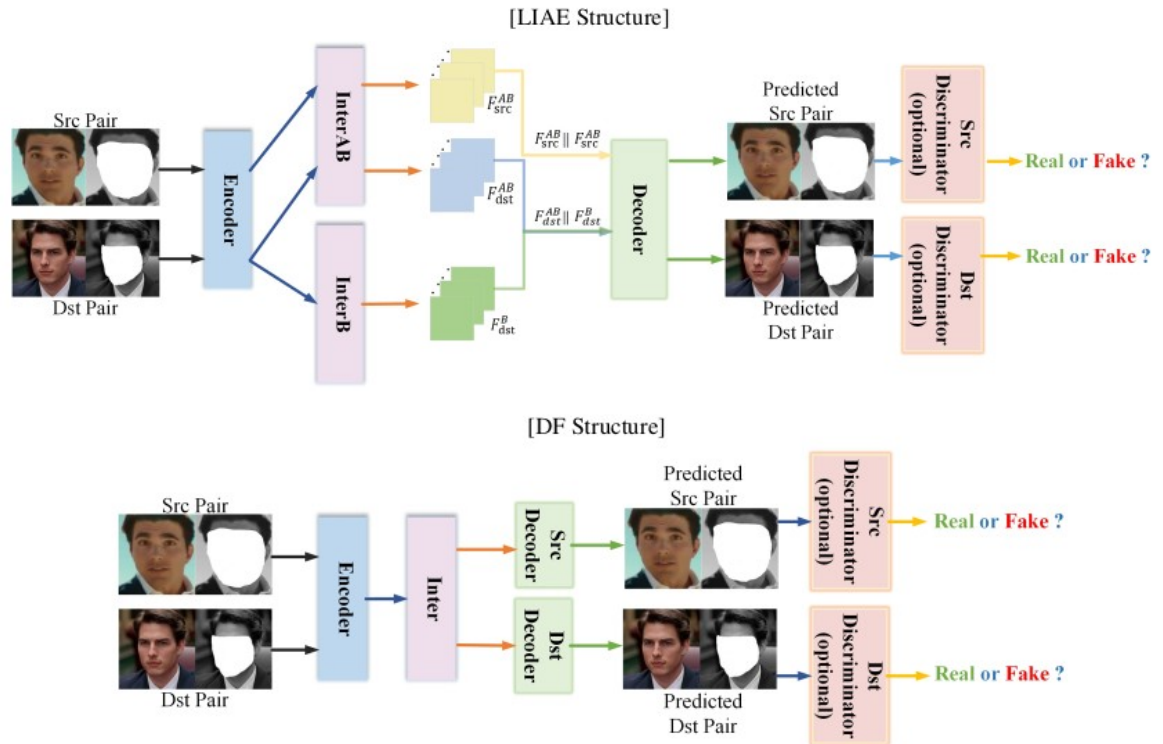
4. Related work



Pipeline:

- Face detection: The first step in extraction phase is to find the target face in the given data source and destination data
- Face alignment: The second step is face alignment. After numerous experiments and failures, we realized that facial landmarks are the key to maintaining stability over time. We need to find an effective facial landmarks algorithm essential in producing an excellent successive footage shot and film.
- Face segmentation: After face alignment, a data folder with faces of standard front/side-view (aligned src or aligned dst) is obtained. We employ a fine-grained

Face Segmentation network on top of (aligned src or aligned dst), through which a face with either hair, fingers, or glasses could be segmented exactly



5. Application

Good Side:

- Deepfakes can be used in the entertainment industry to create realistic special effects in movies or video games.
- Engaging with viewers or customers
- The technology can be used to generate realistic simulations for scientific research, such as modeling the behavior of materials or predicting the spread of diseases.
- Deepfakes can also be used for educational and training purposes, such as simulating medical procedures or training pilots, educating people in a more interactive way
- For many decades, Hollywood has used high-end CGI, VFX, and SFX technologies to create artificial but believable worlds for compelling storytelling. In the 1994 movie, Forrest Gump, the protagonist meets JFK and other historical figures. The creation of the scenario and effect was accomplished using CGI and different techniques with millions of dollars. These days sophisticated CGI and VFX technologies are used in movies to generate synthetic media for telling a captivating story.

Bad Side:

- One of the most concerning aspects of deepfakes is their potential to be used for malicious purposes, such as spreading false information, propaganda, or discrediting public figures.
- Deepfakes can be used for cyberbullying, harassment, or blackmail, where the person in the fake video or audio is portrayed doing or saying things they did not do.
- The technology can also be used to create convincing identity theft, where an individual's likeness is used to commit fraud or other criminal activities.

6. Other solutions: pros and cons

DeepFaceLab, being open-source, is still being continuously maintained and improved ever since its release in 2018. FaceSwap (CNN-based architecture with an autoencoder approach), a similar open-source software to DeepFaceLab was also released in 2018, followed by other similar face-swapping softwares: Avatarify (GAN architecture with pre-trained facial landmark detection network) being released in 2020, DeepArt (VGG-19 architecture) in 2019, OpenFaceSwap (GAN-based architecture with a generator network and a discriminator network) in 2020,...

With all these competitors, DeepFaceLab still remains one of the most popular deep fake softwares. Due to the early release compared to other alternatives, it has gained a large community of users, which leads to the robustness of guides and tutorials available online to help users get started and troubleshoot issues. It also offers a wide range of features and tools compared to others, such as the ability to edit and adjust the alignment of facial landmarks and control the degree of blending between the source and target faces, and also has a more advanced and customizable user interface compared to some of the alternatives, which may be appealing to users with more technical expertise. On the other hand, DeepFaceLab can be more difficult to set up and use compared to some of the alternatives, as it requires a more powerful computer and more technical knowledge. Moreover, the steep learning curve, and resource-intensive, time-consuming nature of training DeepFaceLab may render it impractical for some users.

7. Dataset: [Celebrity Facesets by DeepFakeVFX.com](https://www.deepfakevfx.com)

This dataset consists of batches of images from many celebrities submitted by the members of the community. Each image batch of a certain celebrity contains thousands of images in varied angles, poses and expressions, which will serve best for the result of training deep fake models.

We will choose about eight celebrities, which will total to approximately 20 thousand pictures. We will split the dataset afterwards as follows: 70% for training, 10% for validation and 20% for testing.

Filters

Search Facesets

Search ...

Sorting

Newest

Face Type

☒ Head
 ☒ WF
 ☒ F

Resolution


256 - 2048

XSeg


☒ None
 ☒ Generic
 ☒ Custom

Submit


Reset




Kim Min-Jeong (Winter)
 Face: WF / Res: 1024 / XSeg:
 Generic / Qty: 7,122




Im Na-Yeon (Nayeon)
 Face: WF / Res: 1024 / XSeg:
 Generic / Qty: 4,613




Chou Tzuyu (Tzuyu)
 Face: WF / Res: 1024 / XSeg:
 Generic / Qty: 2,563




Hwang Ye-Ji (Yeji)
 Face: WF / Res: 1024 / XSeg:
 Generic / Qty: 3,587




Go Yoon-Jung
 Face: WF / Res: 512 / XSeg:
 Generic / Qty: 10,278



Shin Yu-Na (Yuna)
 Face: WF / Res: 1024 / XSeg:
 Generic / Qty: 2,104



Rowan Atkinson (Mr. Bean)
 Face: WF / Res: 512 / XSeg:
 Custom / Qty: 26,823



Jake Gyllenhaal
 Face: WF / Res: 512 / XSeg:
 Generic / Qty: 36,654

« Previous Page

1 2 3 4 5 ... 20

Next Page »

Some facesets from <https://www.deepfakevfx.com>



Part of Gal Gadot's faceset on <https://www.deepfakevfx.com>

8. Code

The code base of DeepFaceLab can be divided into several modules, including:

- **Data Loader:** This module is responsible for loading and preprocessing facial images for training or inference. It includes functions for data augmentation, alignment, and cropping.
- **Model Architecture:** This module defines the deep neural network architecture used for facial image manipulation tasks. DeepFaceLab provides various pre-defined architectures such as SAE, H64, H128, and H256, and also allows users to define their own custom architectures.
- **Loss Functions:** This module defines the loss functions used during the training of the deep neural network. DeepFaceLab provides a variety of loss functions such as L1, L2, VGG, and IDLoss, which can be used for different facial image manipulation tasks.
- **Training Pipeline:** This module includes functions for training the deep neural network on a given dataset using a specific architecture and loss function. It includes features such as checkpointing, learning rate scheduling, and early stopping.
- **Inference Pipeline:** This module includes functions for deploying the trained deep neural network for facial image manipulation tasks. It includes features such as batch processing, face detection, and seamless blending.

9. References

<https://mrdeepfakes.com/forums/threads/guide-deepfacelab-2-0-guide.3886/>

<https://mrdeepfakes.com/forums/threads/guide-deepfacelab-2-0-google-colab-guide.1340/>