Deepfakes

How easy are they to make and detect?

ISACA Redstone Arsenal
May 16, 2022

Catherine Bernaciak, PhD

Shannon Gallagher, PhD

Dominic Ross

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM22-0322

What are Deepfakes

- A deepfake is a piece of media—either image, video, or speech, typically representing a human subject—which has been altered to be deceptive using deep neural networks (DNNs) and involves the substituting of identities between a source and a target subject.

ToDo: Put in pics of public deepfakes:
Cruise,
Cage/Lawrence,
Zelensky ?

Main Type and Tools

Replacement

- aka ‘faceswap’
- identity of source is transferred onto destination subject.
- destination’s facial expressions and head movements remain the same

https://faceswap.dev

https://github.com/iperov/DeepFaceLab

Re-enactment

- aka ‘puppet-master’
- identity of destination is transferred onto source subject.

FirstOrderMotion
https://github.com/AliaksandrSiarohin/first-order-model

Evolution of Deepfakes

- Originated in 2017 on r/deepfakes Reddit forum
  - Original and current usage mostly for pornography
- Many novelty YouTube channels exist demonstrating swaps of celebrities (Tom Cruise, Nicholas Cage, etc.)
- Academic Computer Vision research with Machine Learning progressed throughout 90's - 2000's
  - Video Rewrite 1997 (traditional ML – new video from audio)
- Until Deep Neural Networks: Convolutional Neural Networks & Generative Adverserial Networks
  - ***2014*** first Deep Neural Networks: *DeepFace* [Taigman 2014].
  - ***2014-2015*** *DeepId* [Sun 2014a, 2014b, 2014c, 2015]
  - ***2017*** *pix2pix* using GAN's [Isola 2017a]
  - ***2019*** CycleGAN introduced [Zhou 2019]

Creating a Deepfake is a 5-step process

1. Gather source & destination video (CPU)

- high-quality (4K), voluminous (> 10 minutes) source footage and destination footage of subject with similar appearance

2. Extraction (CPU/GPU))

- Faces isolated from each frame using DNN & ML based facial recognition models (S3FD)
    a. faces detected in frame
    b. faces aligned, facial landmarks identified
    c. mask of face segmented from frame

Face after detection and alignment step showing bounding box (green) and facial landmarks (yellow dots) [Source: Reprinted with permission from Faceswap 2017c]

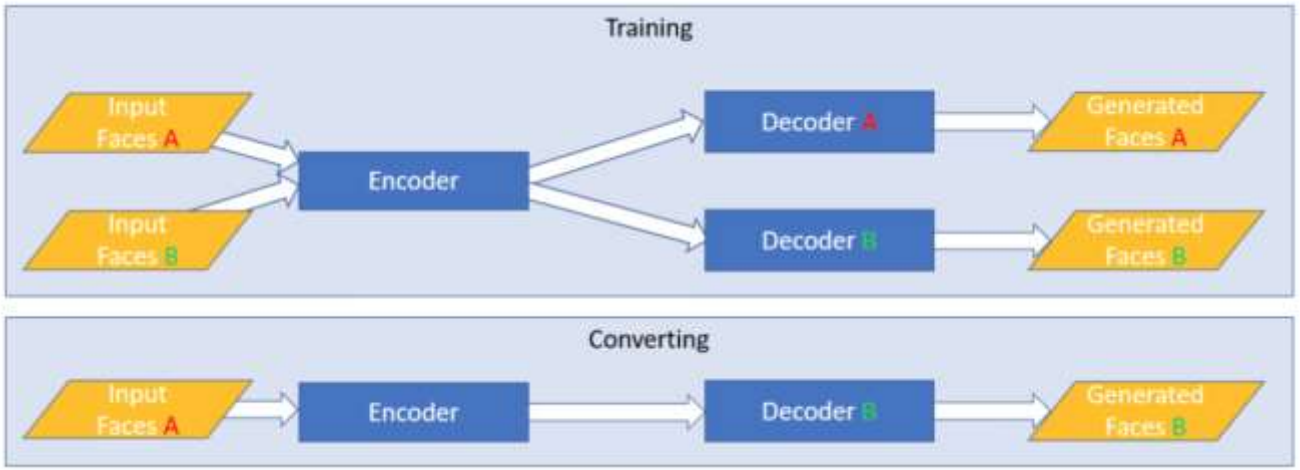
Creating a Deepfake is a 5-step process

3. Training (GPU)

- DFL and FS mostly use autoencoder-decoder networks with CNN layers
- FS and DFL ~ O(10) models each, some very similar

4. Conversion (CPU/GPU)

5. Post-processing (CPU)



General structure of a DNN architecture for face swapping in DFL and FS [Source: Re-printed with permission from Faceswap 2017d]

Some Hype Unwarranted

- Open source materials are not ‘set it and forget it’
  - Requires data science knowledge to make operational (can use pre-trained models, but need to know how software works)
  - Requires skilled editing
  - Extensive compute power & time (GPU’s, training time)
- Ease (computation power, clock time, skill) of creating realistic fakes is hyped in popular articles
- Many visual features can make fake generation difficult, e.g., beards, baldness, relative sizes of subjects
- ***However….*** A single gaming GPU O($1K) can give great results with significant time and effort.

Post-processing

- ***Dom – we have room for several slides on this topic***

Deepfake Detection

- Deepfake detectors *discriminate* between real and deepfake images
  - Detector = discriminator
- To develop models that automatically discriminate, we can:
  - **We** feed the model with features associated with differences
  - **Model** 'learns' useful features on its own
  - Combine the above two approaches

Feeding the model real features

What *describable* features make deepfakes different?

- 'Obvious' errors (e.g. two heads)
- Facial boundaries blurring into background
- Asymmetries (e.g. earrings not matching)
- Inconsistent light sources
- Odd color frequencies
- Irregular 'heartbeats'
- Discontinuity between frames
- Lack of variation

Obvious features: conjoined heads

Image from thispersondoesnotexist.com

Obvious features: assymmetries

Image from
thispersondoesnotexist.com

Advantages of feeding the model ourselves

- Fortunately, photo forensics and image analysis have been active fields long before deepfakes came along
  - Hany Farid: physical objects follow physical laws
    - Light, shadow, weight, specularities, lenses, etc.
- Computer vision –huge strides in detecting human facial features
- Explainability
- Generalizability

Disadvantages of feeding the model ourselves

- Intuition can be difficult to transform into digital representation
  - ‘affine transformation’ problems: scaling, rotating, sliding, mirroring
  - Unclear boundaries: e.g. what if a hand is covering the face?
- Hard to capture all describable differences in one model

Computer extracted features: Neural nets

Advantages of computer extracted features

- Work well in practice (e.g. image detection)
- (Relatively) easy to code
- Can find real, *latent* features that are associated with differences that humans cannot easily find

Disadvantages of computer extracted features

- Find lots of *spurious* features that *seem* to be associated with differences but are just random noise
- Hard to interpret
- Can be difficult to implement and train
- Generalizability

The generator-detector game

1. Adversary generates makes fake images
2. We make detector with high accuracy to discriminate real/fake
3. Adversary introduces improved fake images
4. We make improved detector
5. …

Normally, this game takes a huge amount of effort and time by both the adversary and us.

But Generative Adversarial Networks (GANs) automate the game

GANs: a brief primer

GAN endgame??

Open questions

A. Can generated images be completely indistinguishable from real images? (i.e. are we (detectors) destined to lose this game?)

B. Can we protect our detectors from GANs?

    A. What if we (detectors) release blackbox results but not model parameters?

C. Human-eye detection is a blackbox we cannot currently model nor automate (to scale). Can this be used to our advantage?

Conclusions

- ***Once all the slides are together we can make some conclusions here***

SEI CMU Team

Dr. Catherine Bernaciak Ph.D
Research Scientist
cabernaciak@cert.org

Dominic Ross
Multimedia Design Team Lead
daross@cert.org

Dr. Shannon Gallagher Ph.D.
Data Scientist
skgallagher@cert.org

Jeff Mellon
Machine Learning Research Scientist
jlmellon@cert.org