ZEROFOX®

Playing Offense and Defense with Deepfakes

Mike Price
Chief Technology Officer

Matt Price
Principal Research Engineer

Agenda

1. What a deepfake is
2. Why deepfakes are a problem
3. Public examples of offensive use of deepfakes
4. What our research into offensive use of deepfakes resulted in
5. The solution to offensive use of deepfakes
6. Public examples of solutions to offensive use of deepfakes
7. What our research into solutions to offensive use of deepfakes resulted in
8. Deepstar

What a deepfake is

Deepfake history

June 2016

Face2Face paper released

July 2017

Synthesizing Obama paper released (audio lip syncing)

Winter 2017

r/deepfakes subreddit created

Feb 2018

r/deepfakes subreddit banned

April 2018

Jordan Peele Obama PSA deepfake released

June 2018

In Ictu Oculi: Exposing AI Generated Fake Face Videos by Detecting Eye Blinking paper released

Sept 2018

MesoNet: a Compact Facial Video Forgery Detection Network paper released

April 2019

FaceForensics++ paper and dataset released

May 2019

Few Shot Adversarial Learning of Realistic Neural Talking Heads Model paper released

June 2019

Text-Based Editing of Talking-head Video paper released

June 2019

Mark Zuckerberg deepfake released

What a deepfake is

- A deepfake is generally understood to be a video in which the face of one person has been swapped with the face of another person
- There are variations on this theme (face swap, puppet-master, lip-sync)
- The scope of this presentation is limited to video

Why deepfakes are a problem

Why deepfakes are a problem

- Online content is demonstrably being used for criminal/reputation/influence operations
- It has not been possible to easily fake people in videos
- People therefore trust it more than, say, images, which can be easily faked by anyone with Photoshop
- It is now easier to fake people in videos
- It provides an additional, arguably more effective, means by which to abuse trust
- The common scenario described of a convincing deepfake dropped the night of an election is possible
- The asymmetry of cost to impact could be high

Public examples of offensive use of deepfakes

Public examples of offensive use of deepfakes

- There are thousands of known public deepfakes if we include adult content
- We will be releasing our collection (~300 non-adult content)
- Of all known public deepfakes, none are broadly confirmed to have been used for criminal/reputation/influence operations
- Jordan Peele’s deepfake serves as a solid example of what could be done wrt influence operations
- Many other political figures have been the subject of proof-of-concept deepfakes (e.g. Macri)
- The Nancy Pelosi video was reported by some as being a deepfake but was not
- A leading security vendor recently reported 3 cases of fake audio being used to commit crime (not broadly confirmed and not deepfakes)

Public examples of offensive use of deepfakes

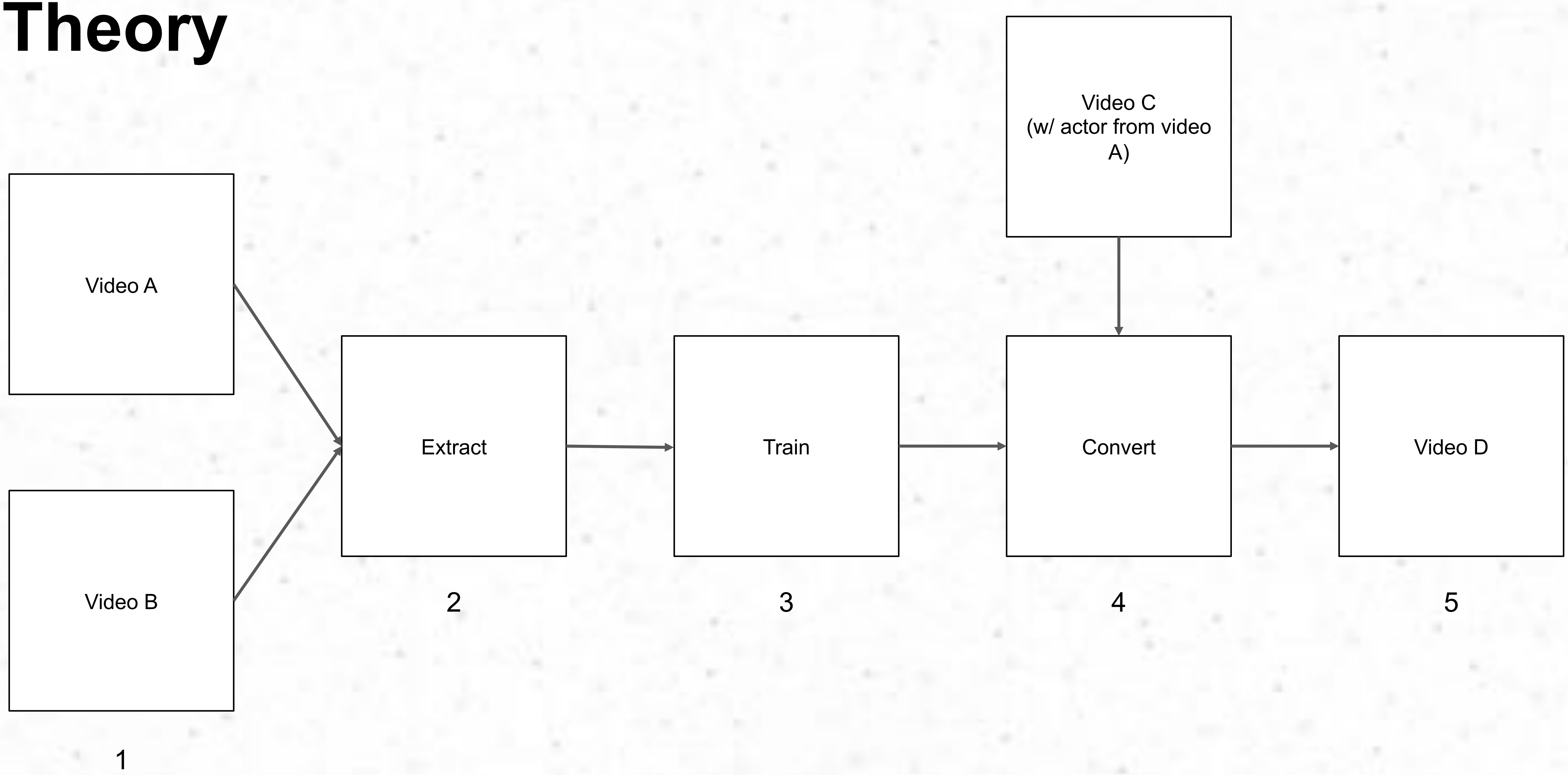
Barack Obama

Mauricio Macri

Nancy Pelosi

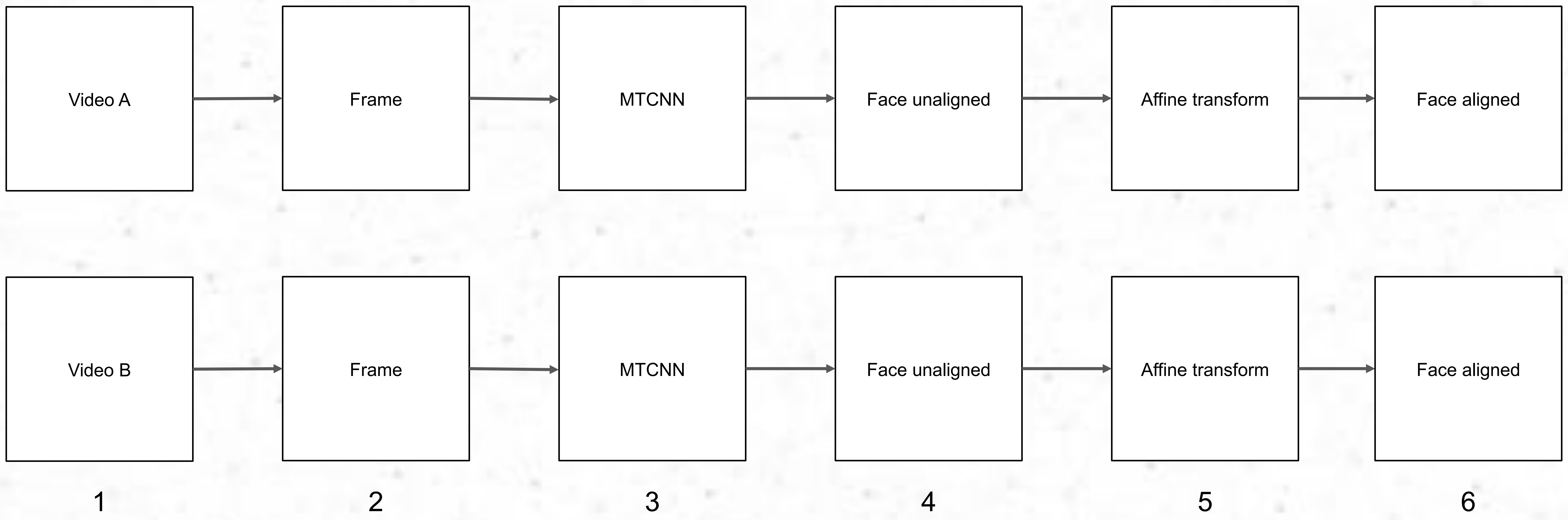
What our research into offensive use of deepfakes resulted in

Theory



Video C
(w/ actor from video A)
Video A
Video B
1
Extract
2
Train
3
Convert
4
Video D
5

Extraction



Video A → Frame → MTCNN → Face unaligned → Affine transform → Face aligned

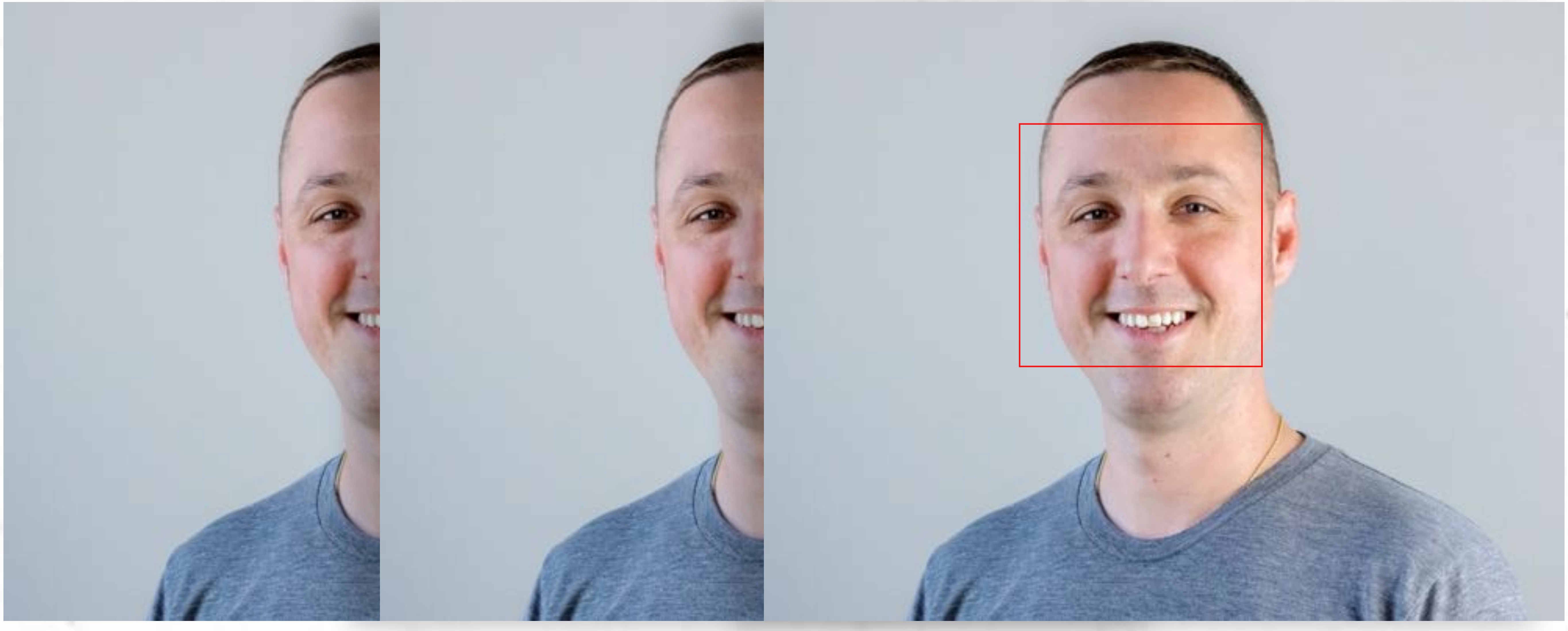
Video B → Frame → MTCNN → Face unaligned → Affine transform → Face aligned

1 2 3 4 5 6

Extraction

Extraction

Extraction

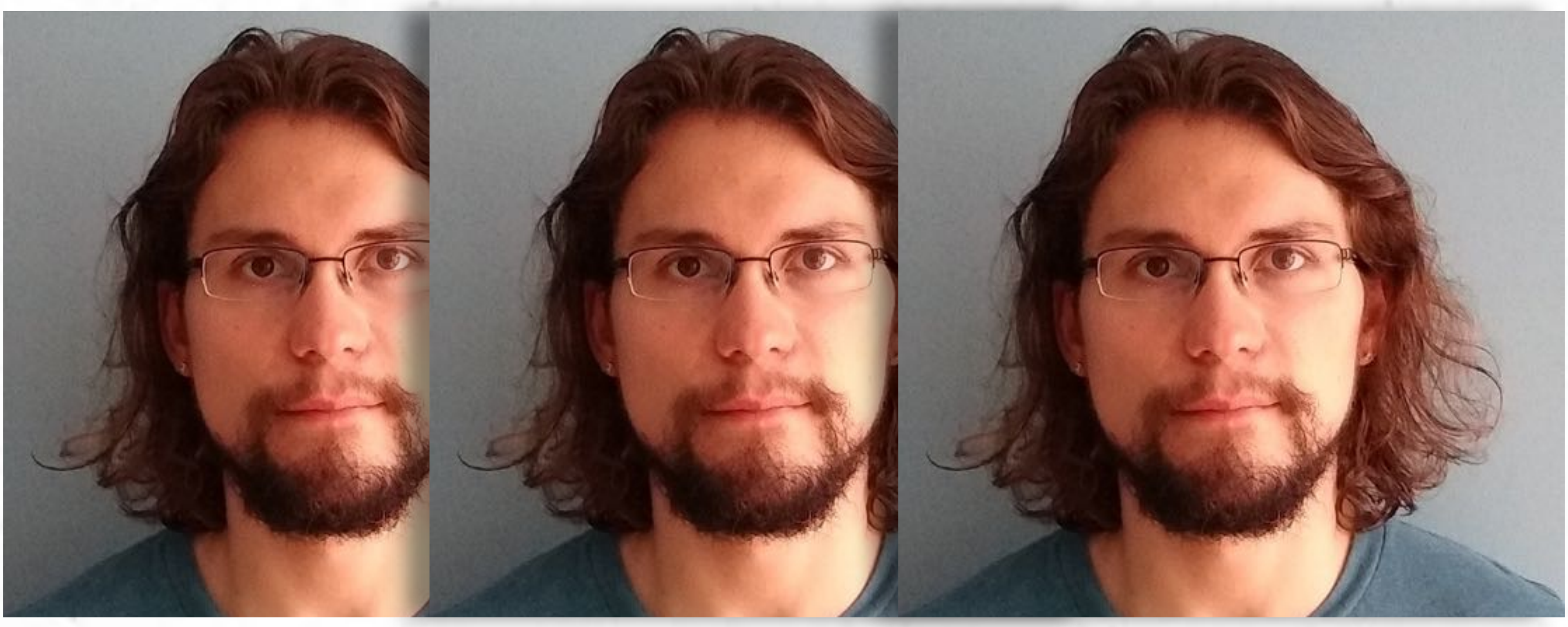


Extraction

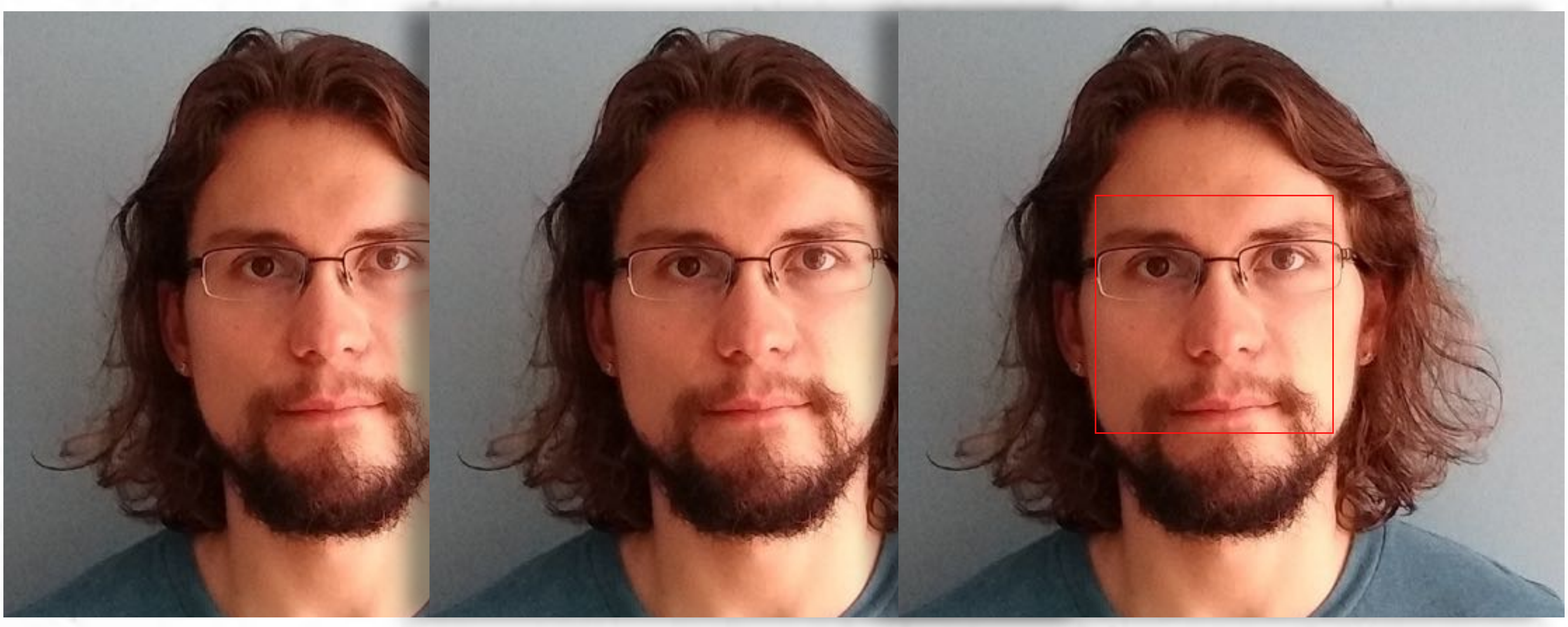
Extraction

Extraction

Extraction

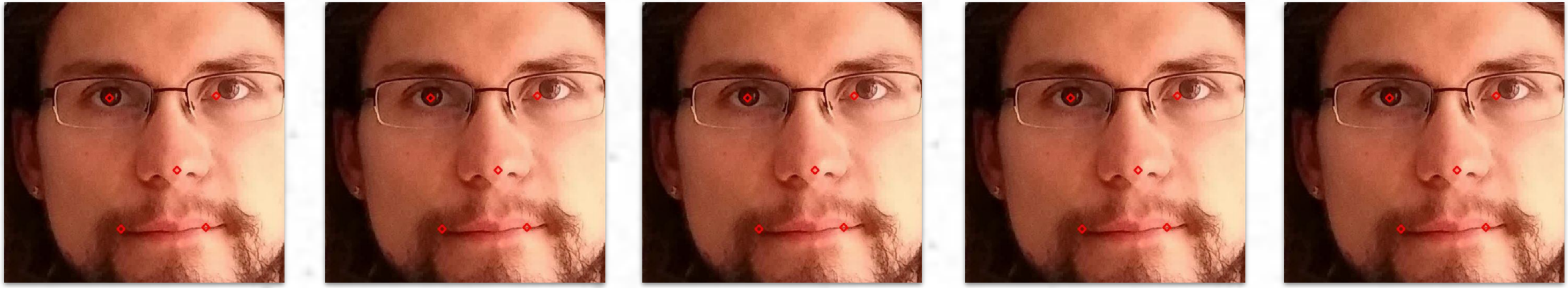


Extraction

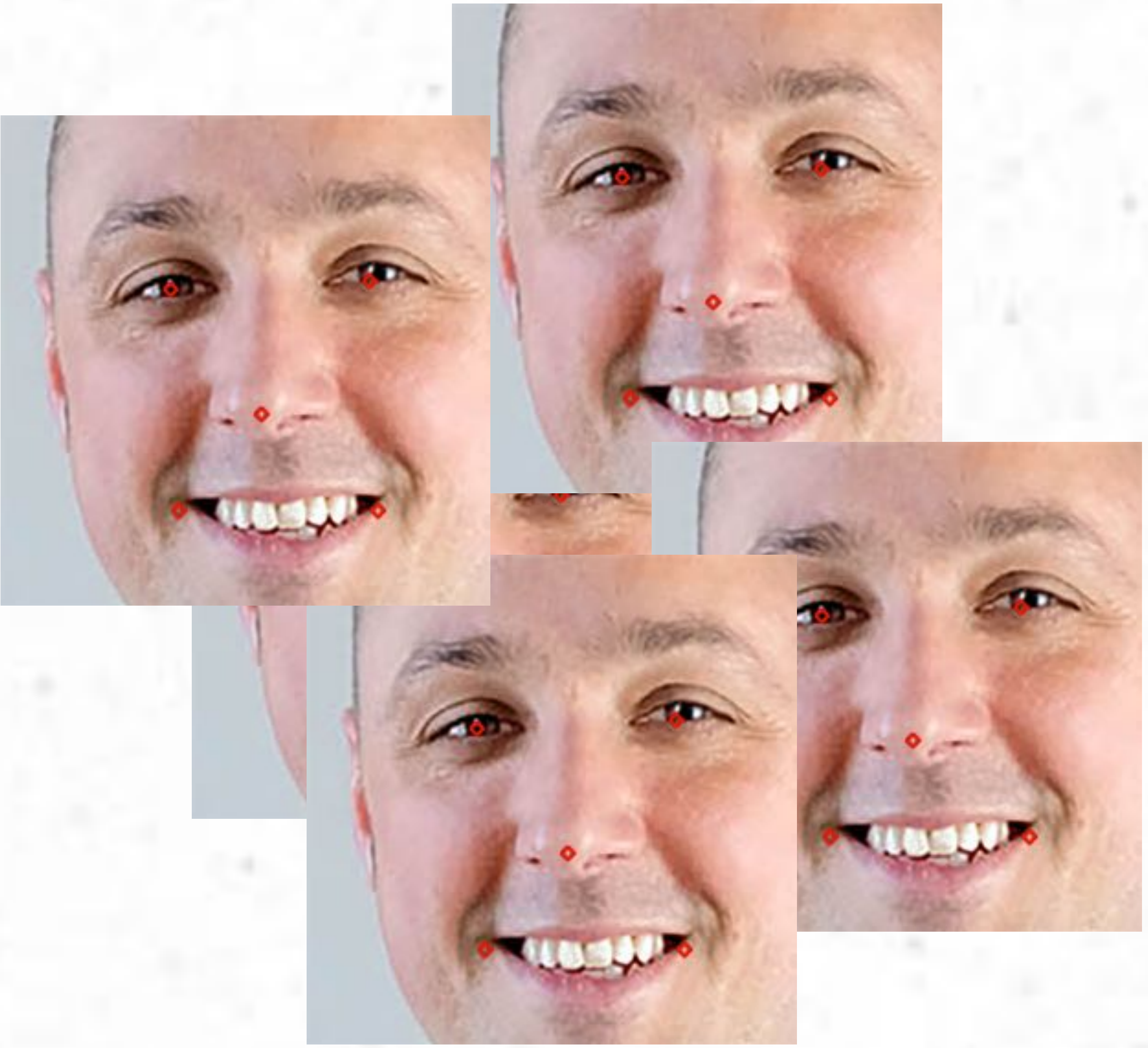


Extraction

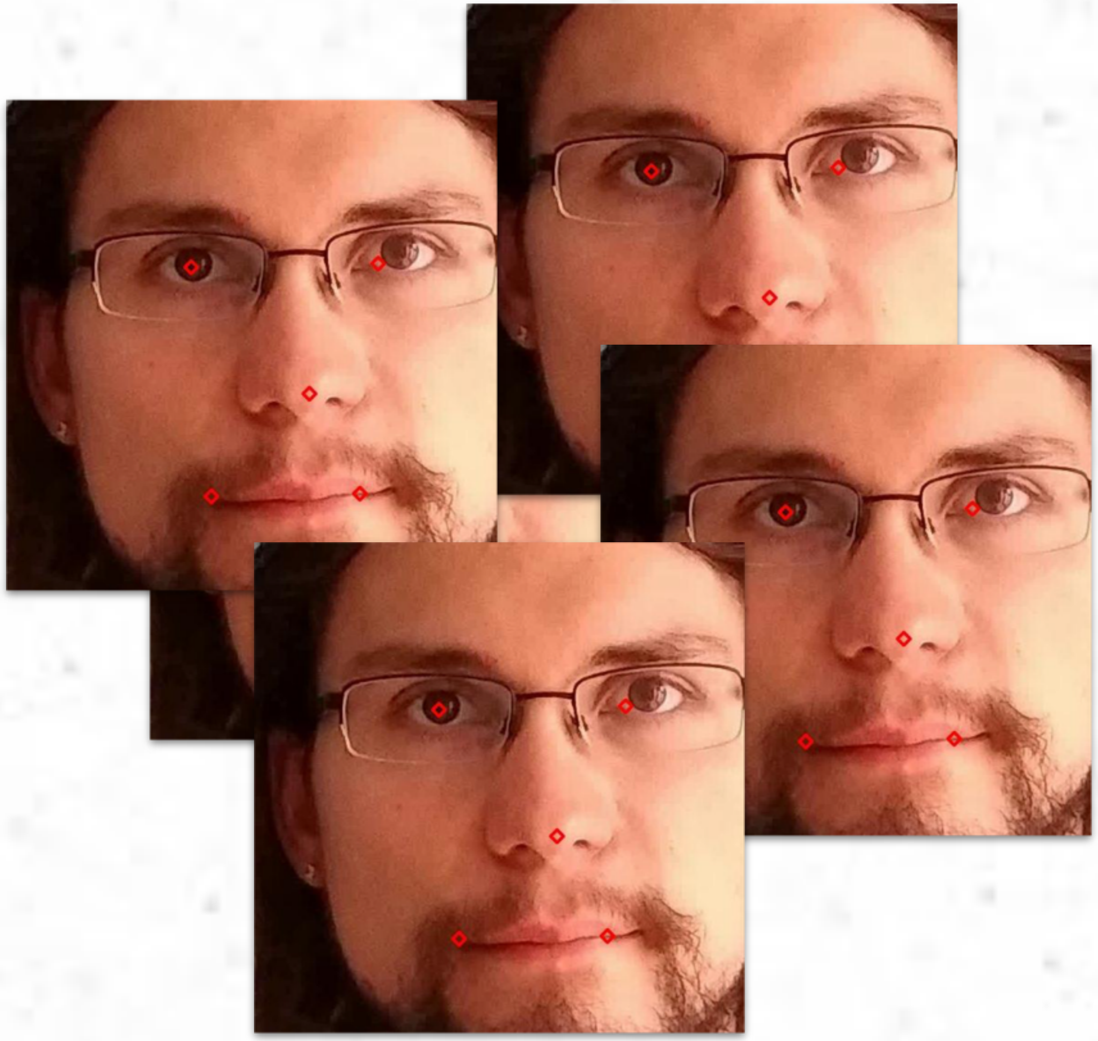
Extraction



Extraction



Group A



Group B

Training

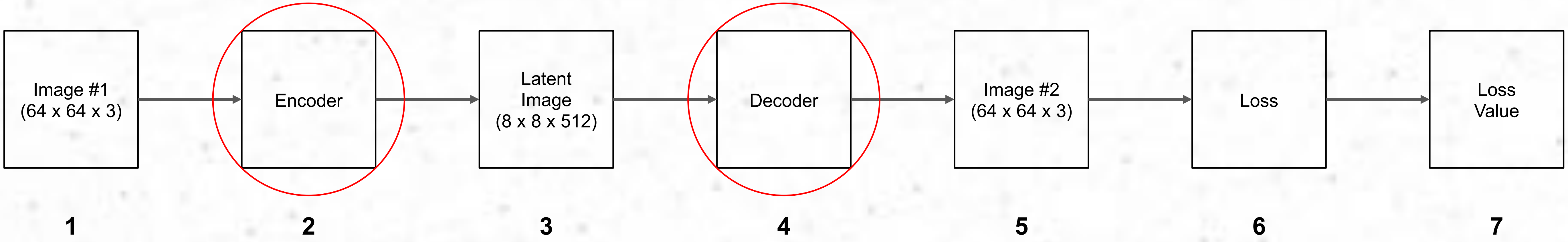
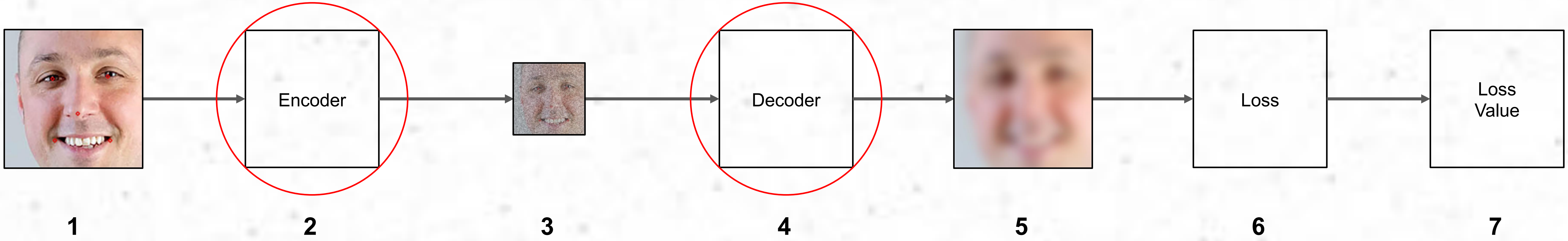


Image #1
(64 x 64 x 3)
Encoder
Latent
Image
(8 x 8 x 512)
Decoder
Image #2
(64 x 64 x 3)
Loss
Loss
Value
1
2
3
4
5
6
7

Training



Training

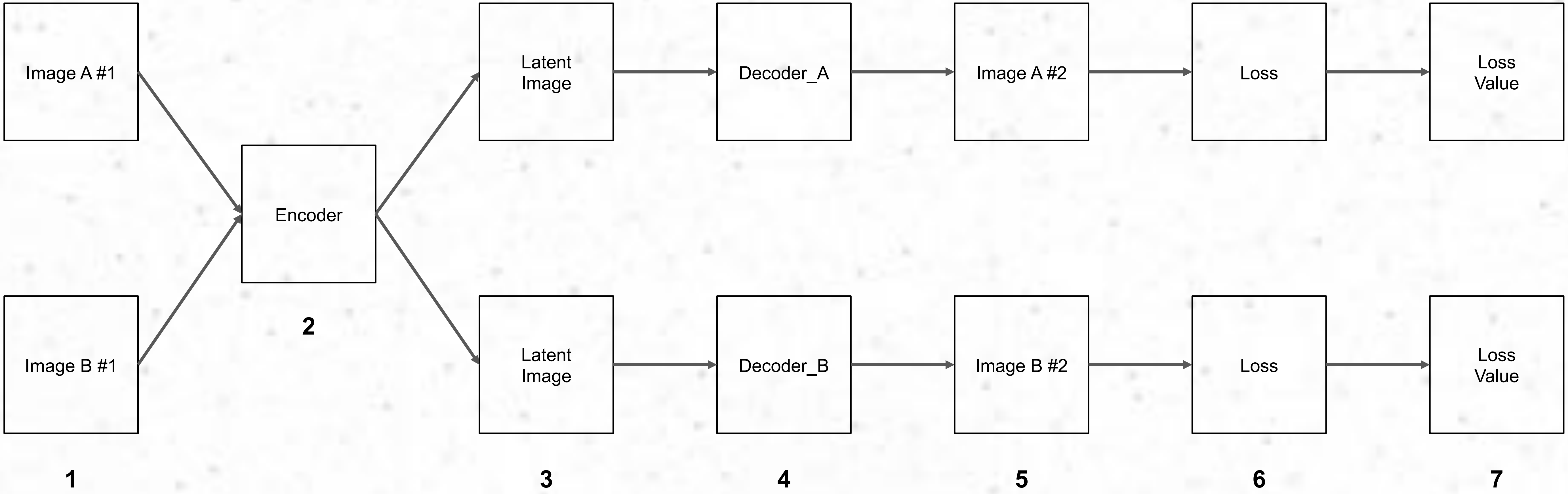
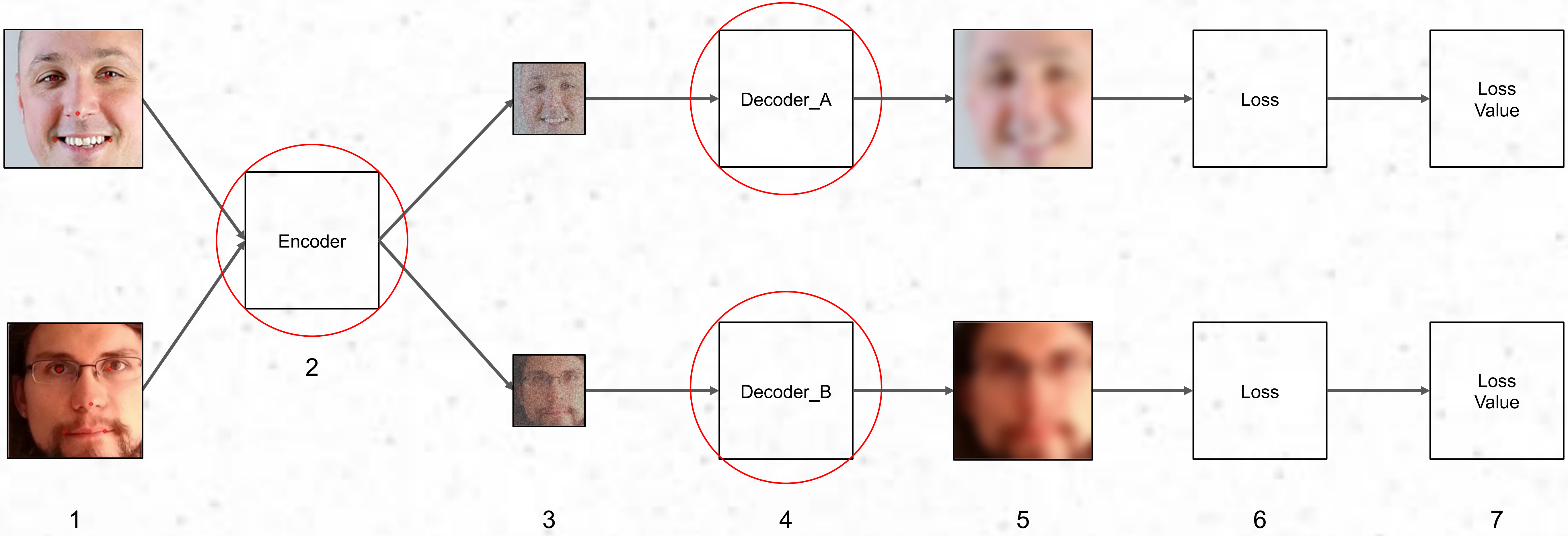
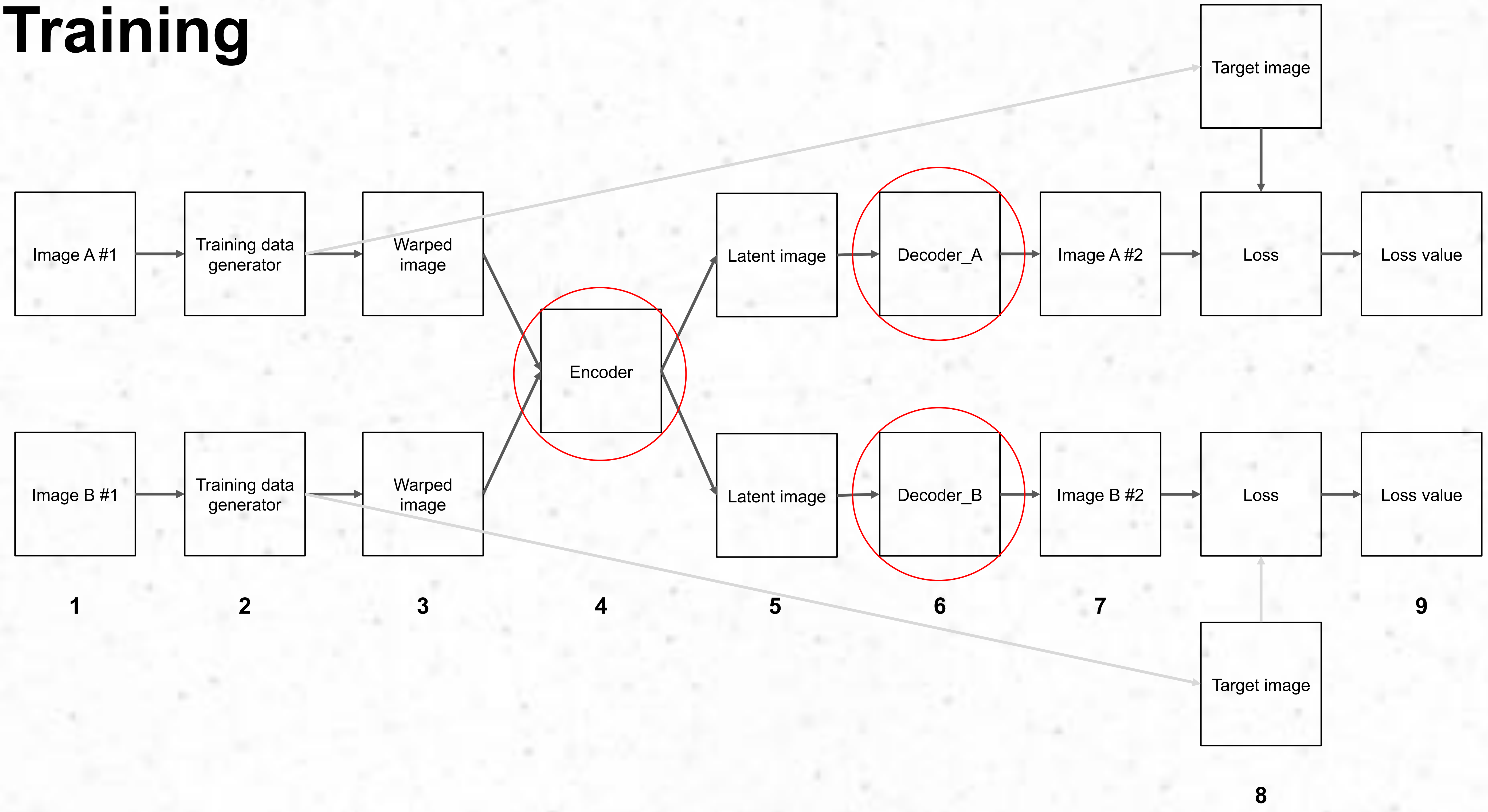


Image A #1
Image B #1
1
Encoder
2
Latent Image
Latent Image
3
Decoder_A
Decoder_B
4
Image A #2
Image B #2
5
Loss
Loss
6
Loss Value
Loss Value
7

Training

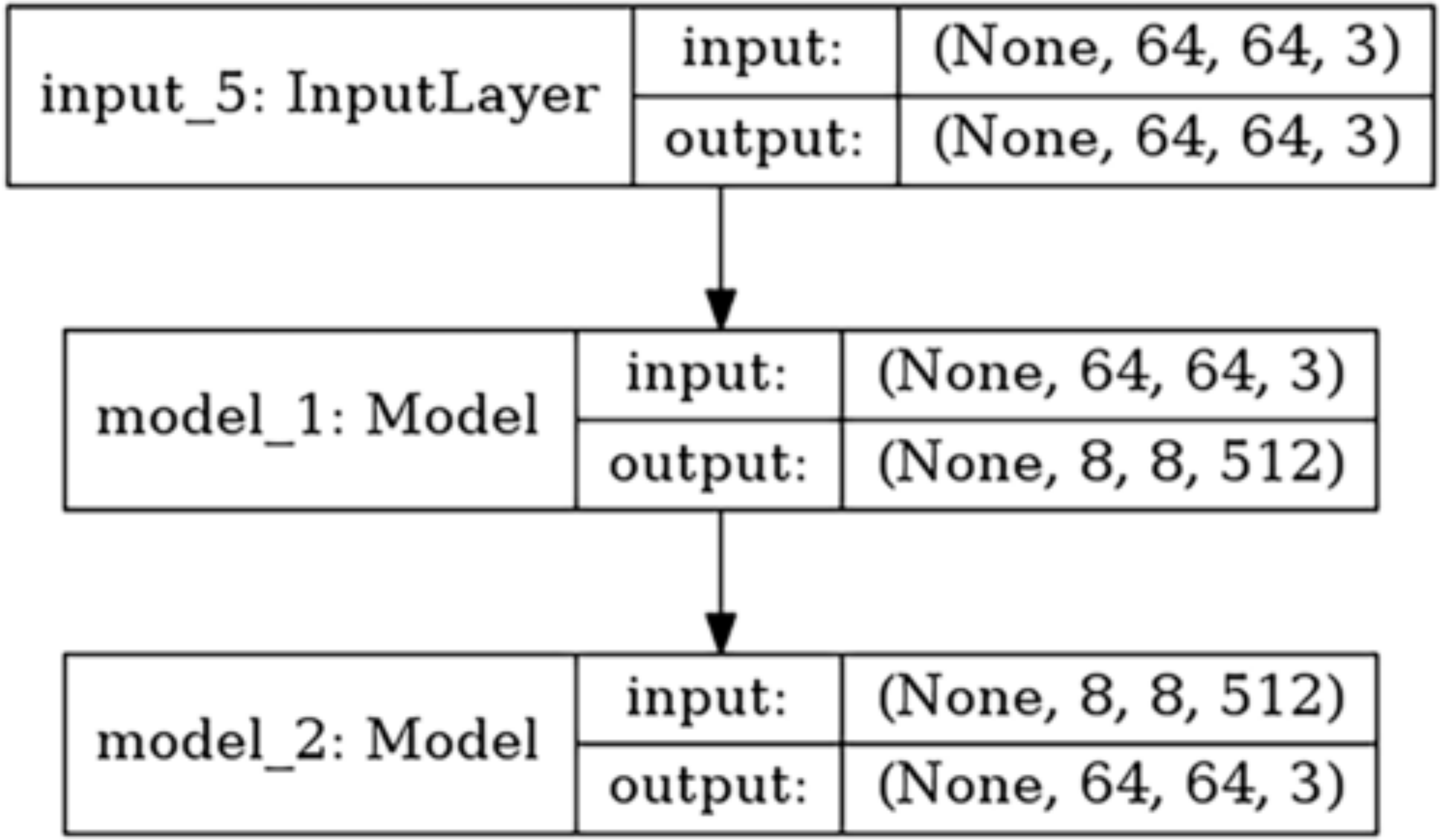

Encoder
Decoder_A
Decoder_B
Loss
Loss
Value
Loss
Loss
Value
1
2
3
4
5
6
7

Training



Target image
Image A #1
Training data generator
Warped image
Latent image
Decoder_A
Image A #2
Loss
Loss value
Encoder
Image B #1
Training data generator
Warped image
Latent image
Decoder_B
Image B #2
Loss
Loss value
1
2
3
4
5
6
7
9
Target image
8

Training

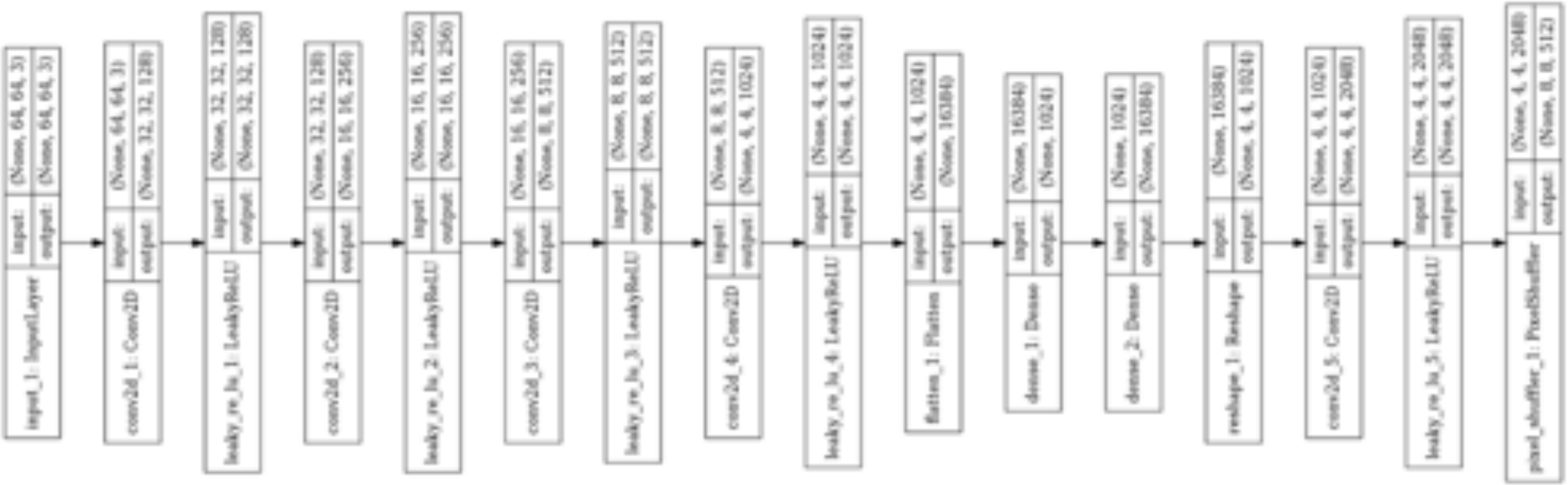


input_5: InputLayer | input: | (None, 64, 64, 3)
output: | (None, 64, 64, 3)

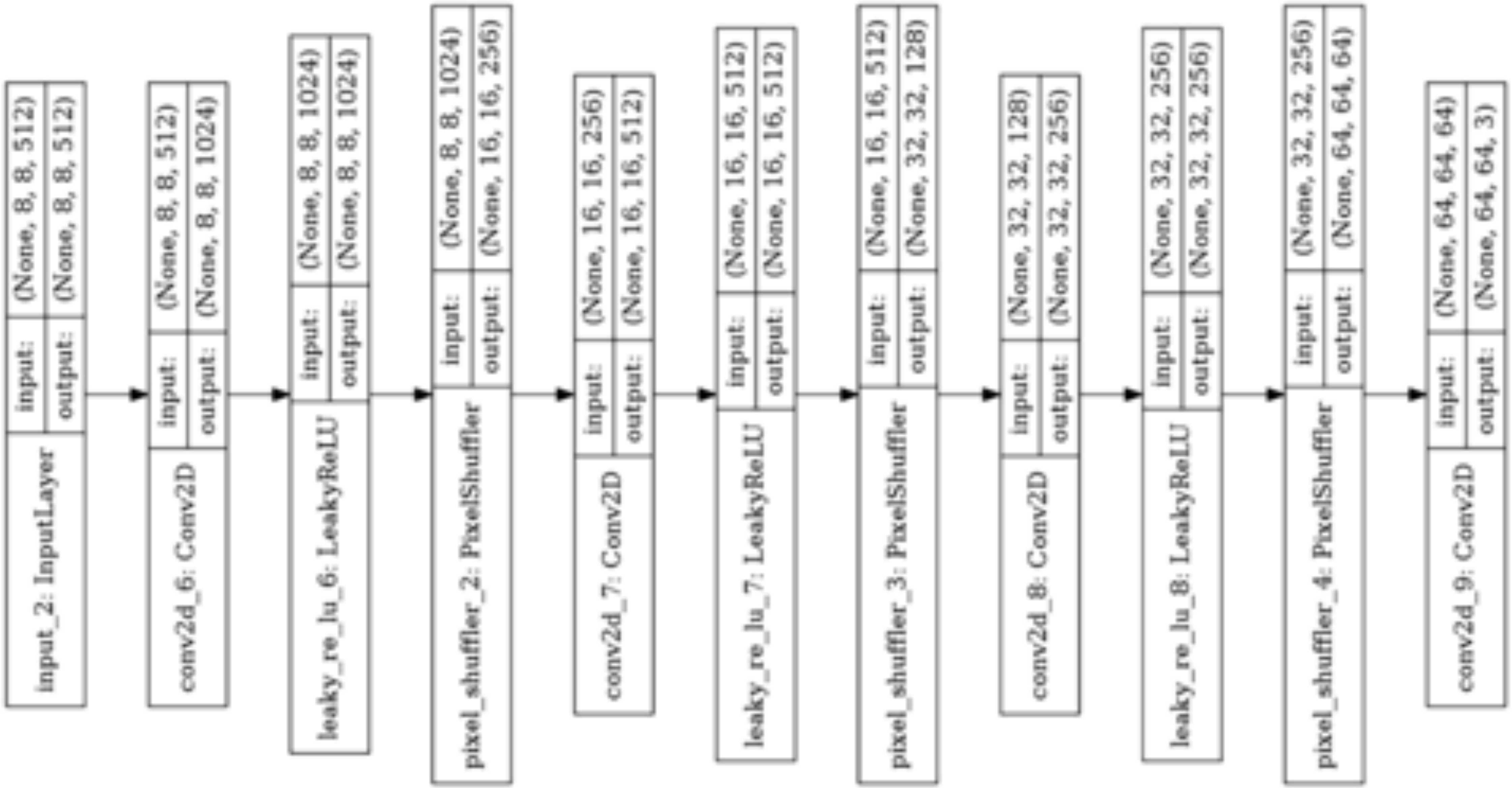
model_1: Model | input: | (None, 64, 64, 3)
output: | (None, 8, 8, 512)

model_2: Model | input: | (None, 8, 8, 512)
output: | (None, 64, 64, 3)

Training


input_1: InputLayer
input: (None, 64, 64, 3)
output: (None, 64, 64, 3)
conv2d_1: Conv2D
input: (None, 64, 64, 3)
output: (None, 32, 32, 128)
leaky_re_lu_1: LeakyReLU
input: (None, 32, 32, 128)
output: (None, 32, 32, 128)
conv2d_2: Conv2D
input: (None, 32, 32, 128)
output: (None, 16, 16, 256)
leaky_re_lu_2: LeakyReLU
input: (None, 16, 16, 256)
output: (None, 16, 16, 256)
conv2d_3: Conv2D
input: (None, 16, 16, 256)
output: (None, 8, 8, 512)
leaky_re_lu_3: LeakyReLU
input: (None, 8, 8, 512)
output: (None, 8, 8, 512)
conv2d_4: Conv2D
input: (None, 8, 8, 512)
output: (None, 4, 4, 1024)
leaky_re_lu_4: LeakyReLU
input: (None, 4, 4, 1024)
output: (None, 4, 4, 1024)
flatten_1: Flatten
input: (None, 4, 4, 1024)
output: (None, 16384)
dense_1: Dense
input: (None, 16384)
output: (None, 1024)
dense_2: Dense
input: (None, 1024)
output: (None, 16384)
reshape_1: Reshape
input: (None, 16384)
output: (None, 4, 4, 1024)
conv2d_5: Conv2D
input: (None, 4, 4, 1024)
output: (None, 4, 4, 2048)
leaky_re_lu_5: LeakyReLU
input: (None, 4, 4, 2048)
output: (None, 4, 4, 2048)
pixel_shuffler_1: PixelShuffler
input: (None, 4, 4, 2048)
output: (None, 8, 8, 512)

Training



input_2: InputLayer
input: (None, 8, 8, 512)
output: (None, 8, 8, 512)

conv2d_6: Conv2D
input: (None, 8, 8, 512)
output: (None, 8, 8, 1024)

leaky_re_lu_6: LeakyReLU
input: (None, 8, 8, 1024)
output: (None, 8, 8, 1024)

pixel_shuffler_2: PixelShuffler
input: (None, 8, 8, 1024)
output: (None, 16, 16, 256)

conv2d_7: Conv2D
input: (None, 16, 16, 256)
output: (None, 16, 16, 512)

leaky_re_lu_7: LeakyReLU
input: (None, 16, 16, 512)
output: (None, 16, 16, 512)

pixel_shuffler_3: PixelShuffler
input: (None, 16, 16, 512)
output: (None, 32, 32, 128)

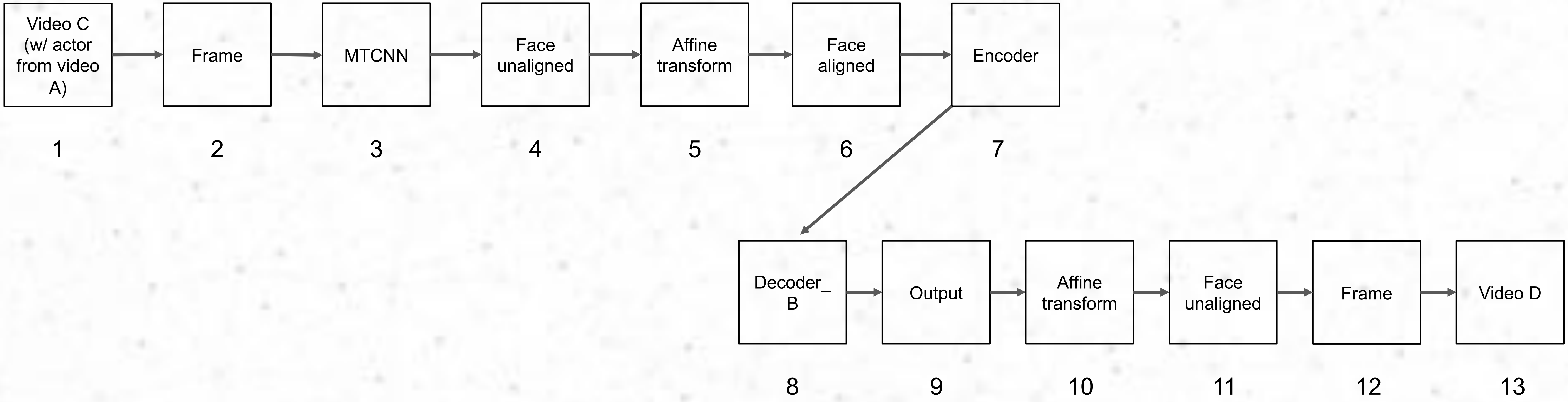
conv2d_8: Conv2D
input: (None, 32, 32, 128)
output: (None, 32, 32, 256)

leaky_re_lu_8: LeakyReLU
input: (None, 32, 32, 256)
output: (None, 32, 32, 256)

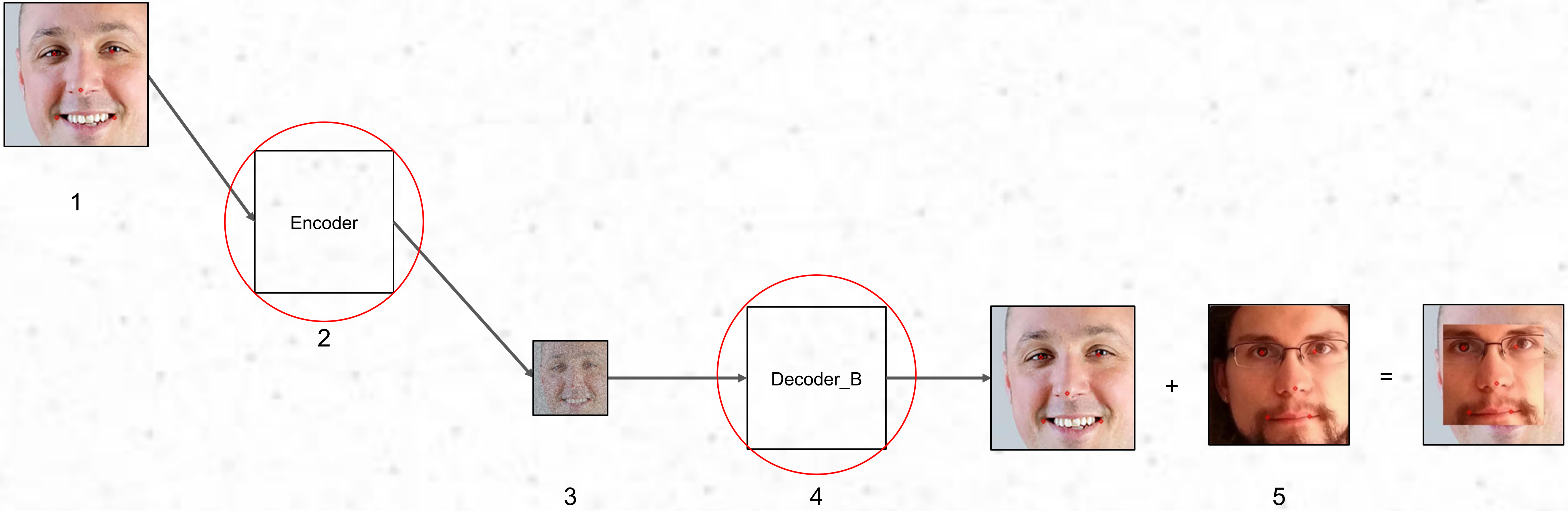
pixel_shuffler_4: PixelShuffler
input: (None, 32, 32, 256)
output: (None, 64, 64, 64)

conv2d_9: Conv2D
input: (None, 64, 64, 64)
output: (None, 64, 64, 3)

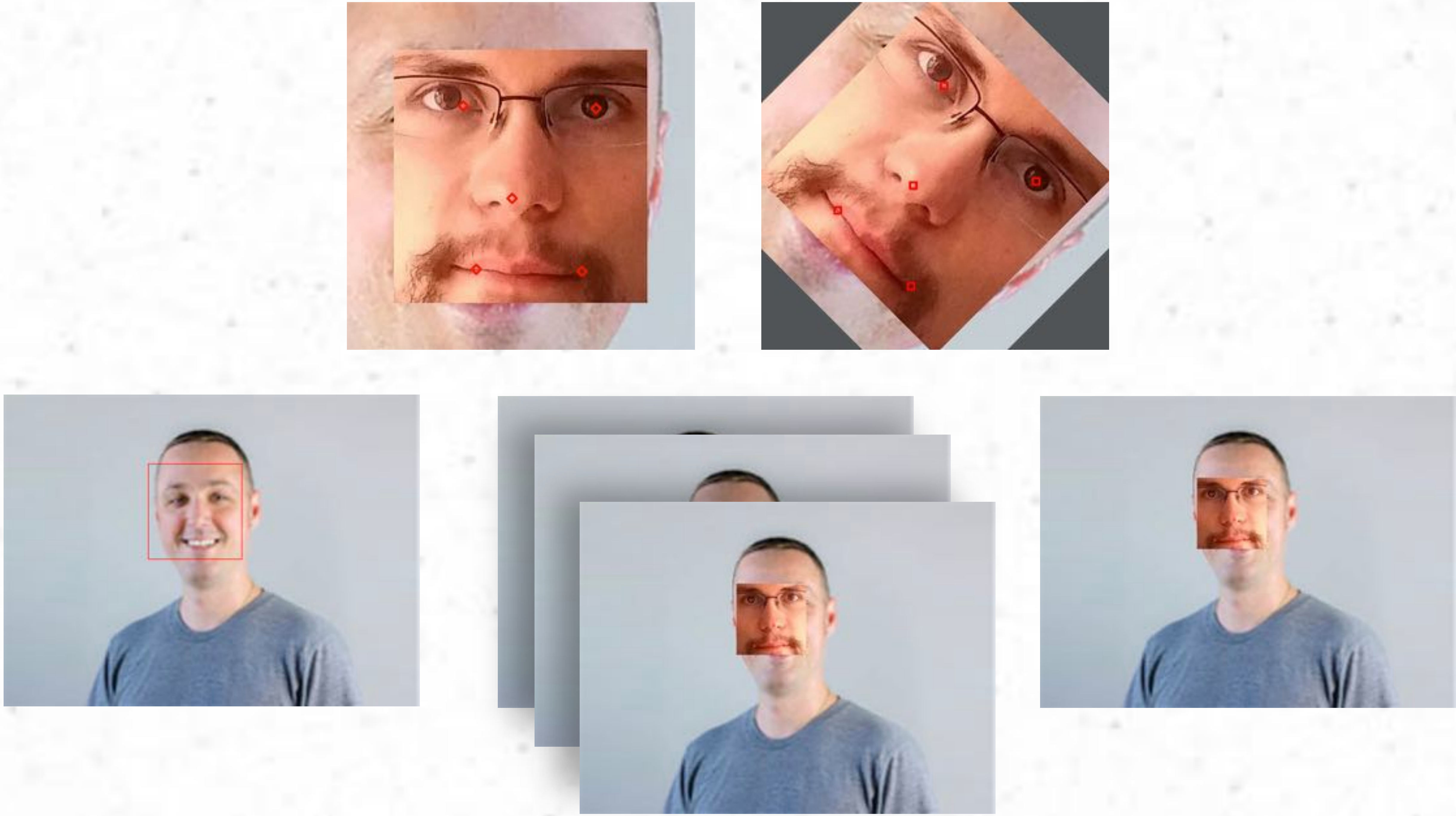
Conversion


Video C (w/ actor from video A)
1
Frame
2
MTCNN
3
Face unaligned
4
Affine transform
5
Face aligned
6
Encoder
7
Decoder_ B
8
Output
9
Affine transform
10
Face unaligned
11
Frame
12
Video D
13

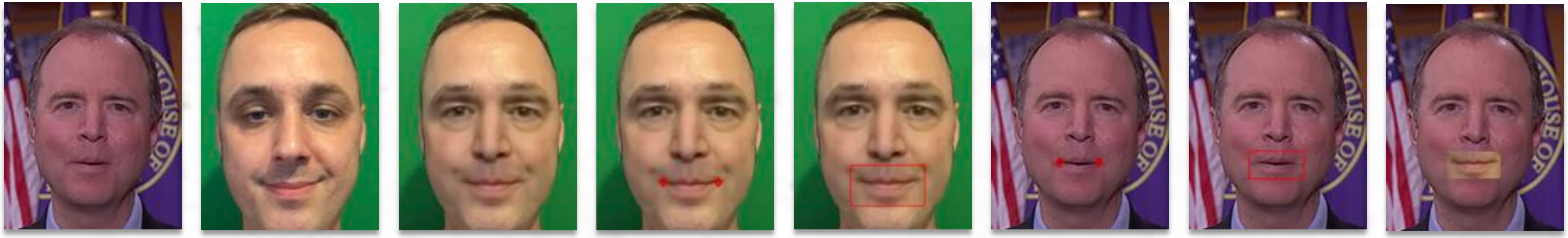
Conversion

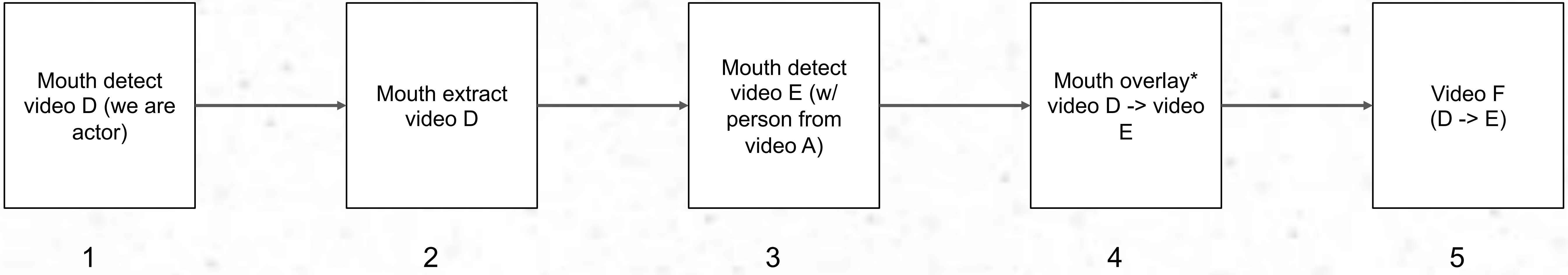


Conversion



Puppet-master





Mouth detect video D (we are actor)

Mouth extract video D

Mouth detect video E (w/ person from video A)

Mouth overlay* video D -> video E

Video F (D -> E)

1

2

3

4

5

E2e deepstar execution to create test videos

- In the following examples, faceswap was used to create the deepfake, and deepstar was used to perform other operations
- Each example is generated with a shell script that roles up a set of commands
- The operations handled by deepstar include: pull video from files/the internet, prepare dataset for faceswap, prepare stock footage, edit video, merge in deepfake, apply puppet-master effect and deploy to email/SMS

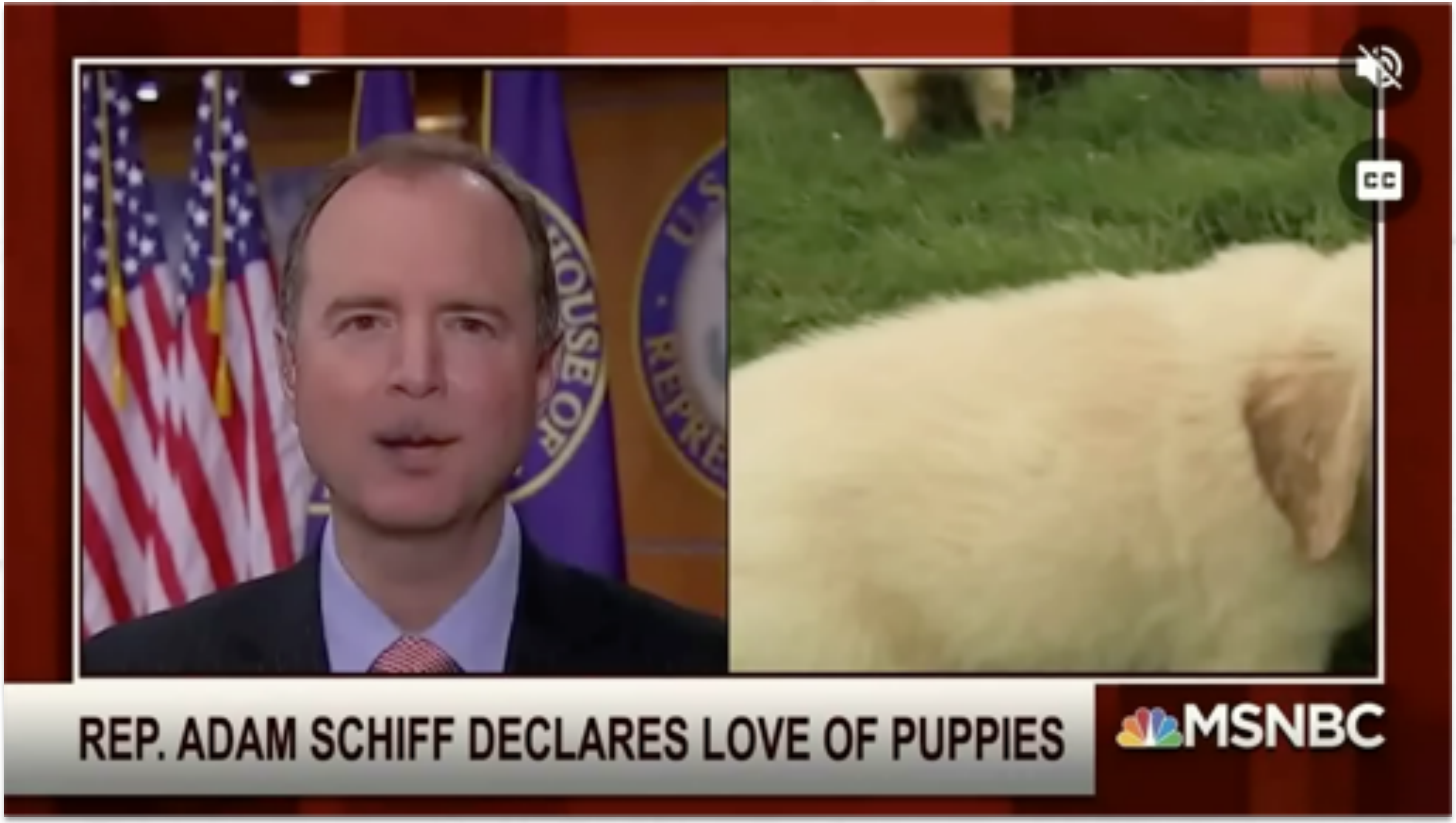
Test video 1

Fox News

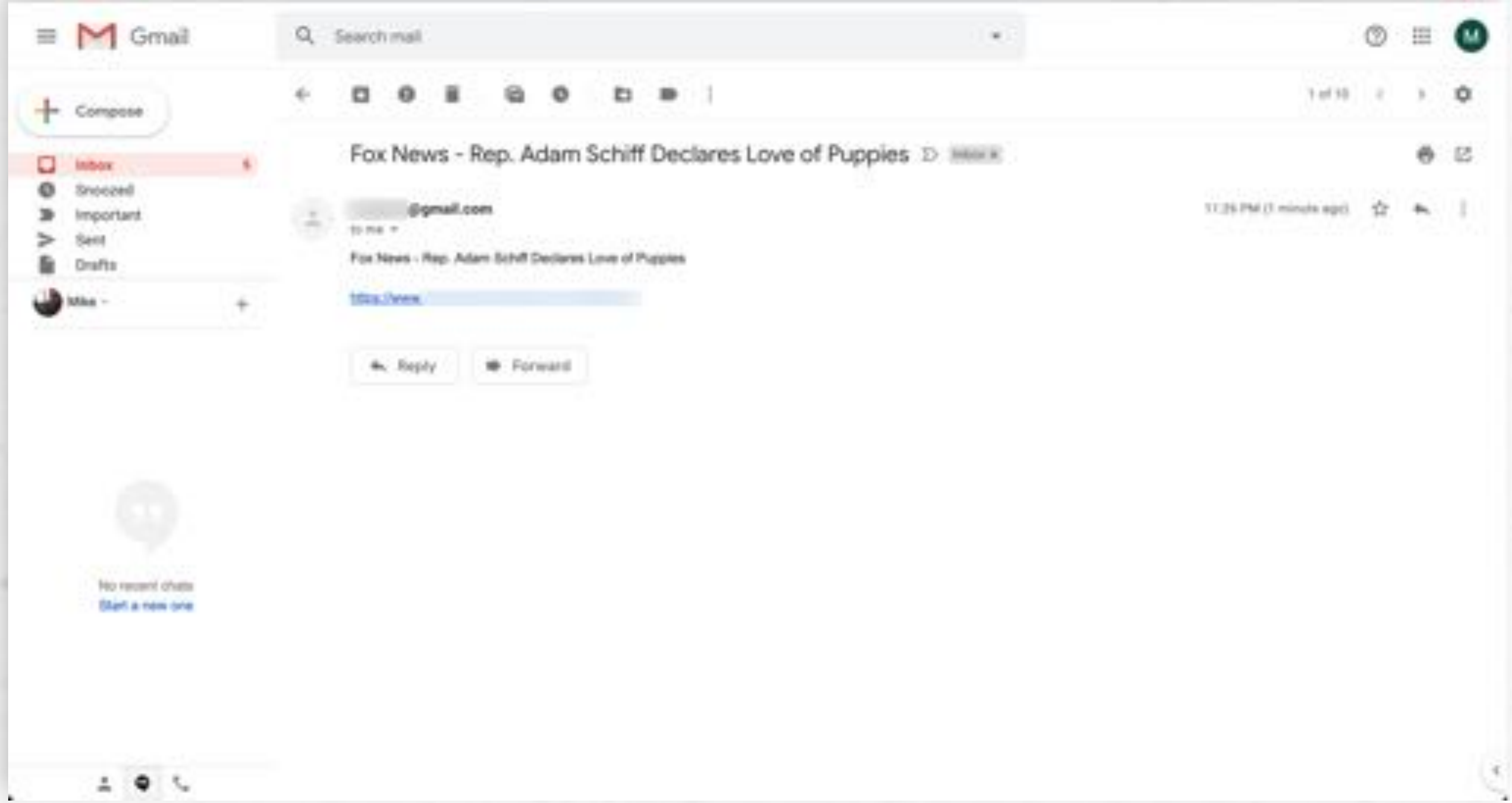
CNN

BREAKING NEWS • BREAKING NEWS • BREAKING NEWS • BREAKING NEWS • BREAKING NEWS • BREAKING NEWS • BREAKING NEWS • BREAKING NEWS
CC
BREAK
NEW

MSNBC



Email screenshots



Fox News - Rep. Adam Schiff Declares Love of Puppies

[redacted]@gmail.com

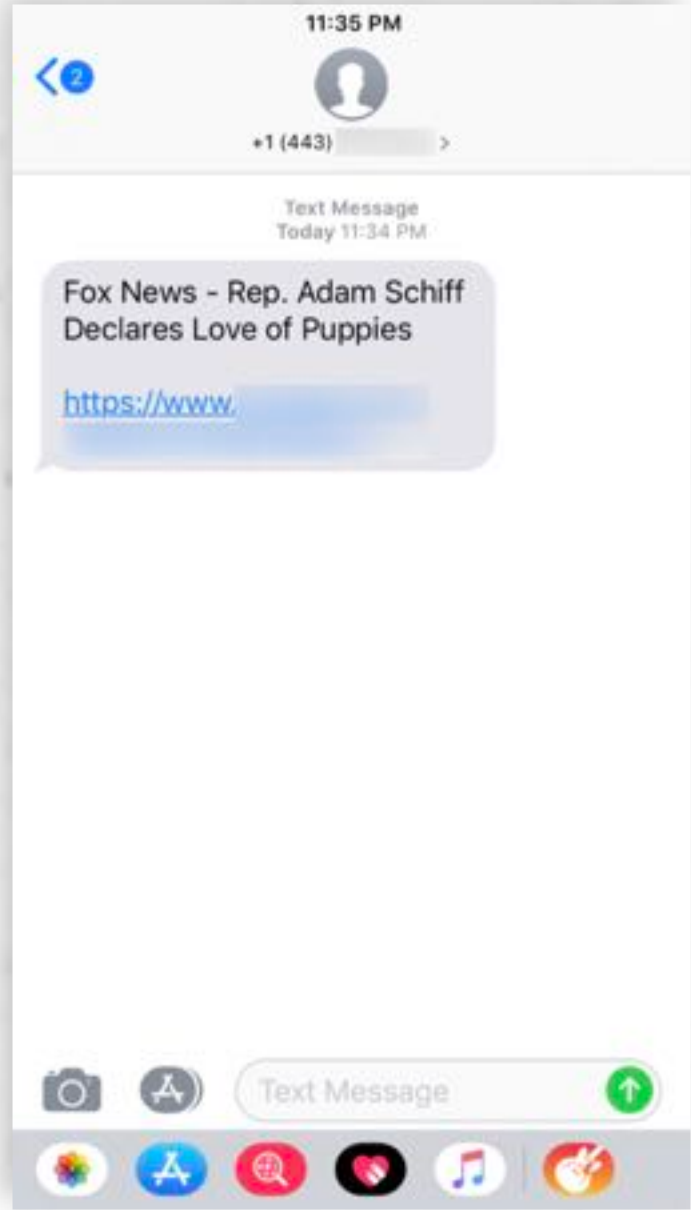
to me

11:26 PM (1 minute ago)

Fox News - Rep. Adam Schiff Declares Love of Puppies

https://www.[redacted]

SMS screenshots



Text Message
Today 11:34 PM

Fox News - Rep. Adam Schiff Declares Love of Puppies

https://www.[redacted]

Test video 2

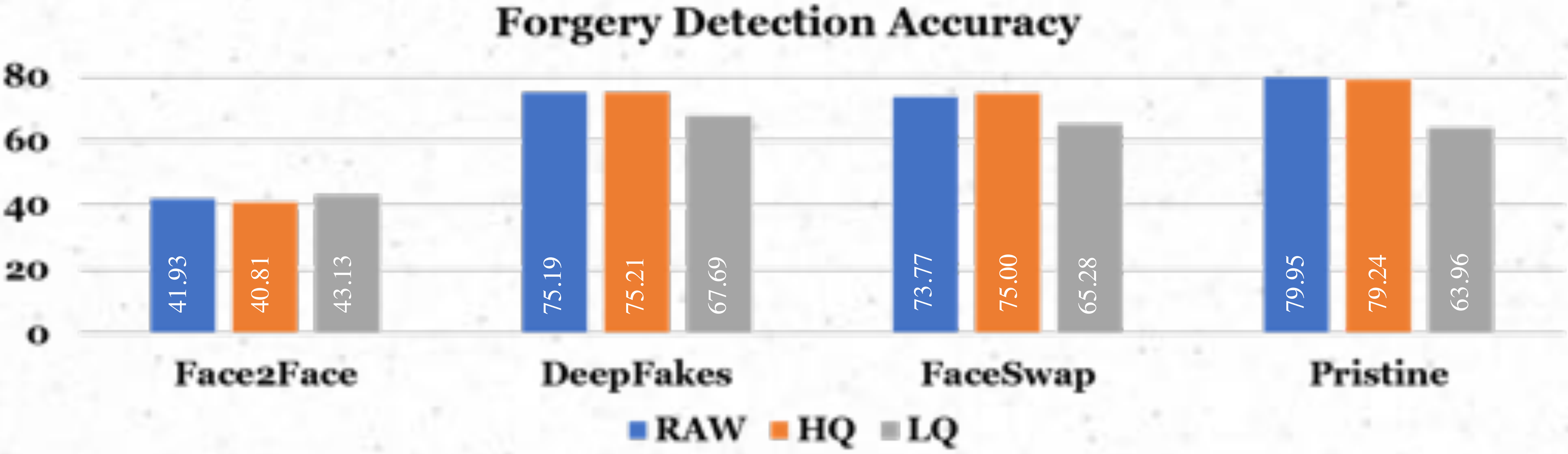
Test video 2

Results

- Thinking about offensive uses there are a few obvious use cases including criminal/reputation and influence
- For this presentation we focused on influence wrt public discourse
- The novel work here is simply to take the particular offensive concern to its next logical step – a kit that rolls up creation and deployment of puppet-mastered deepfake videos in a fake news context
- If a higher quality video were deployed today or on the eve of an election to a large number of people via email or SMS – would we (collectively) be equipped to deal with it?

The solution to offensive use of deepfakes

Humans Can’t Detect Image Manipulation Well



A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, “FaceForensics : Learning to Detect Manipulated Facial Images,” arXiv, Apr. 2019.

Deepfake Detection Methodologies

- Signal level (sensor noise, CFA interpolation, double JPEG compression, etc.)
- Physical level (lighting conditions, shadows, reflections, etc.)
- Semantic level (consistency of meta-data)
- Physiological signals (breathing, pulse, eye blinking, etc.)
  - Individualized or generalized
- Video authentication (ex. blockchain)
- White/black listing

Y. Li, M.-C. Chang, and S. Lyu, “In Ictu Oculi: Exposing AI Generated Fake Face Videos by Detecting Eye Blinking,” arXiv, Jun. 2018.

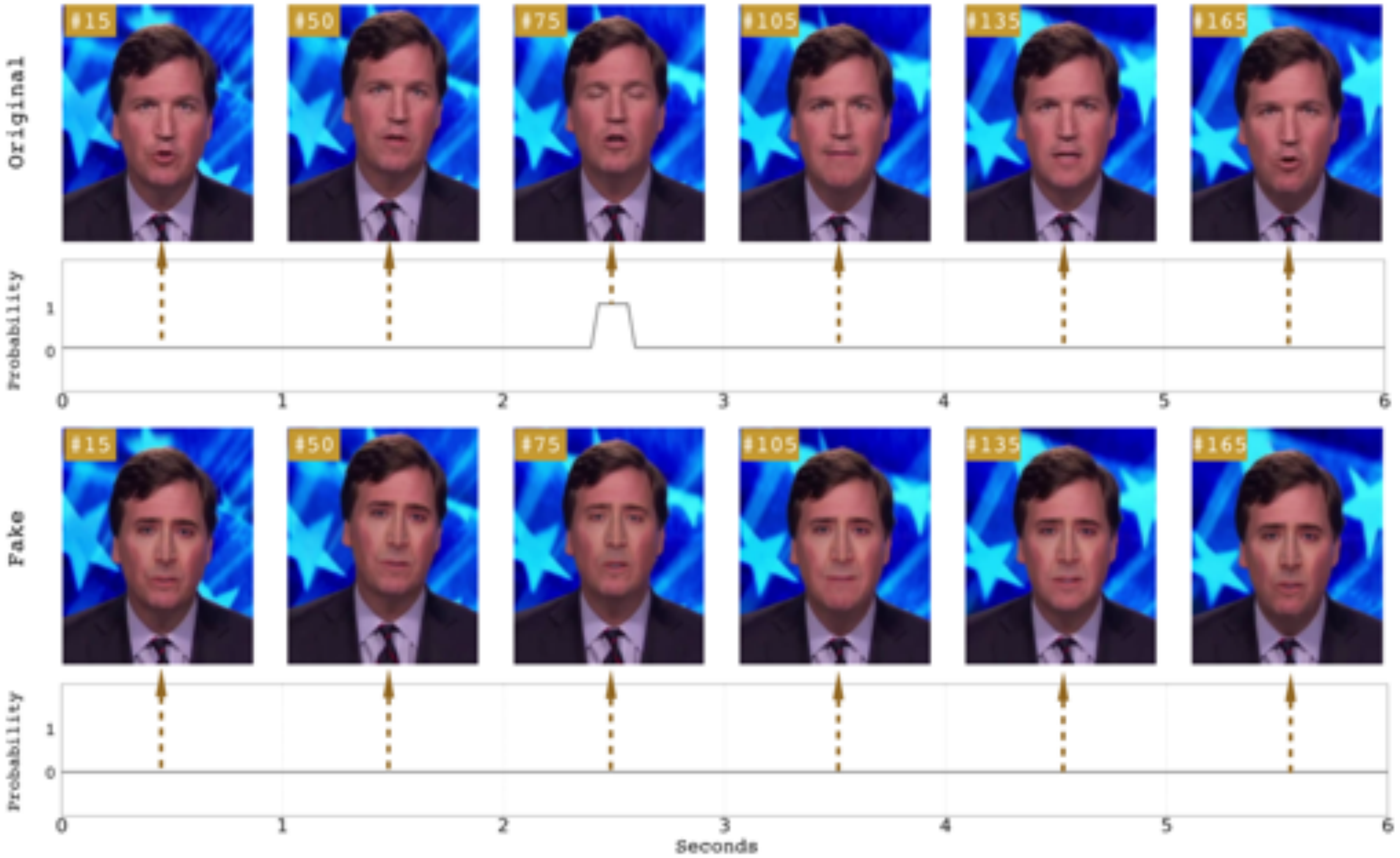
Public examples of solutions to offensive use of deepfakes

Eye Blinking

Observed that in most deepfakes the target was not blinking - human physiological trait that is difficult to capture in synthesized video as it is spontaneous and involuntary

Why does this work?

- Human eye blinking has strong temporal correlation with previous states
- Difficulty in finding images that contain targets blinking (most photos do not capture someone with their eyes closed)



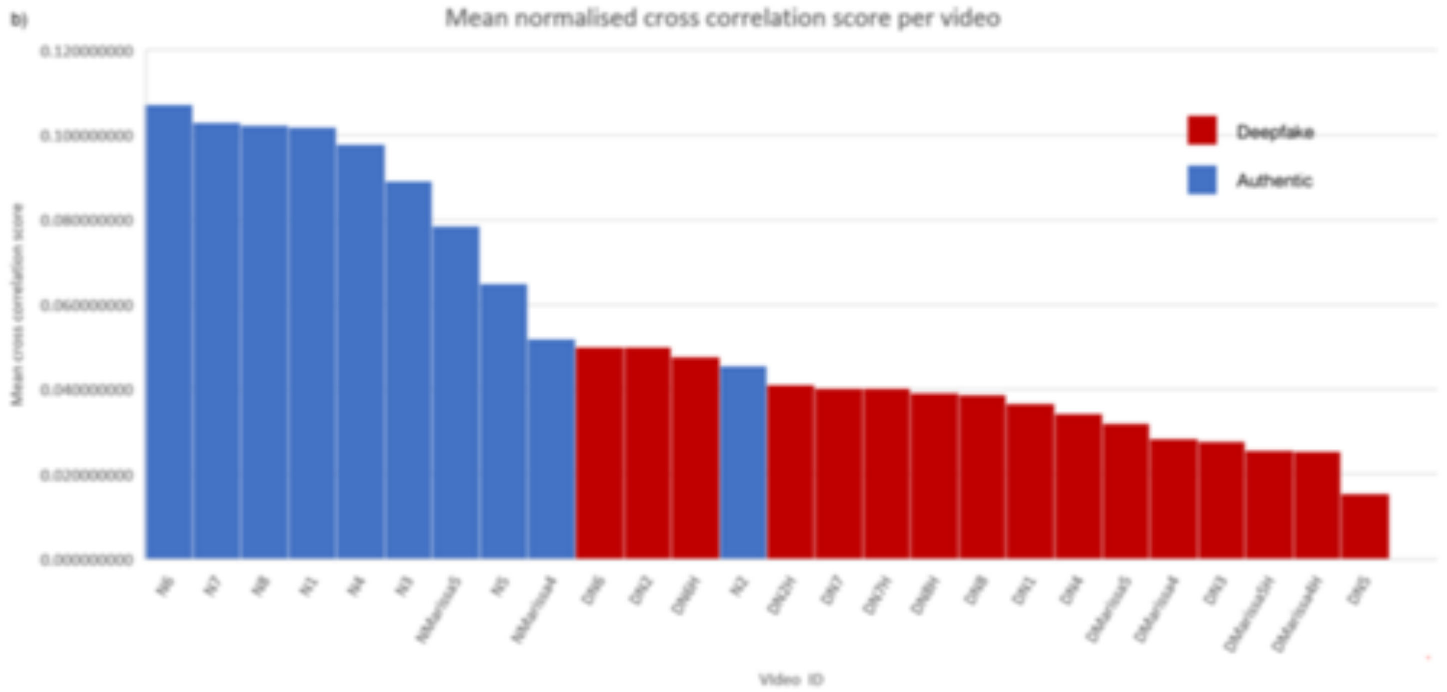
Y. Li, M.-C. Chang, and S. Lyu, “In Ictu Oculi: Exposing AI Generated Fake Face Videos by Detecting Eye Blinking,” arXiv, Jun. 2018.

Photo Response Non-Uniformity

The PRNU (photo response non-uniformity) pattern of a digital image is a noise pattern created by small factory defects in the light sensitive sensors of a camera

- Highly individualized - referred to as the digital image fingerprint

Idea is that the manipulation of the facial area will affect the local PRNU pattern

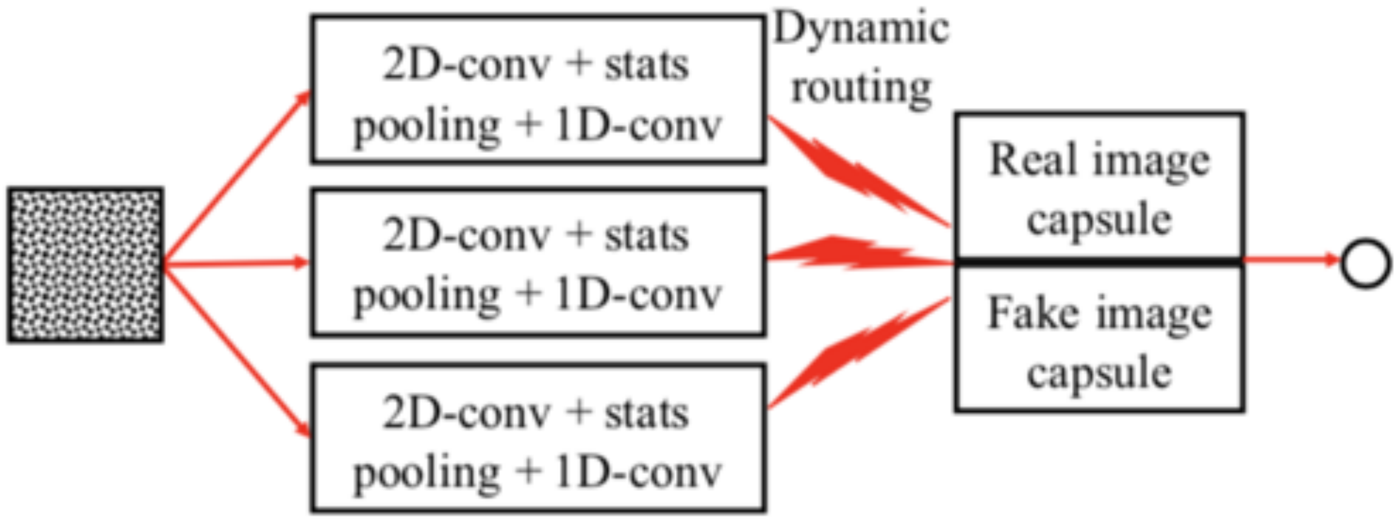


M. Koopman, Z. Geradts, and A. Rodriguez, “Detection of Deepfake Video Manipulation,” IMVIP, Aug. 2018.

Novel Neural Networks

MesoNet

- Novel contribution is the use of a deep learning network with only a small number of layers (4)
- Basic idea is that a “microscopic” approach degrades as image resolution decreases and that at a high level human’s have issues detecting forgeries
  - Therefore a neural network that operates in-between these two levels will produce the best detections


2D-conv + stats pooling + 1D-conv
2D-conv + stats pooling + 1D-conv
2D-conv + stats pooling + 1D-conv
Dynamic routing
Real image capsule
Fake image capsule

Capsule-Forensics

- Used VGG-19 network for input to the capsule network
- 3 primary capsules and 2 output capsules

H. Nguyen, J. Yamagishi, and I. Echizen, “Capsule-Forensics: Using Capsule Networks to Detect Forged Images and Videos,” arXiv, Oct. 2018.
D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, “MesoNet: a Compact Facial Video Forgery Detection Network,” arXiv, Sep. 2018.

Recurrent Neural Networks

Image Classifier + LSTM

Exploit the fact that deepfakes contain intra-frame and temporal inconsistencies

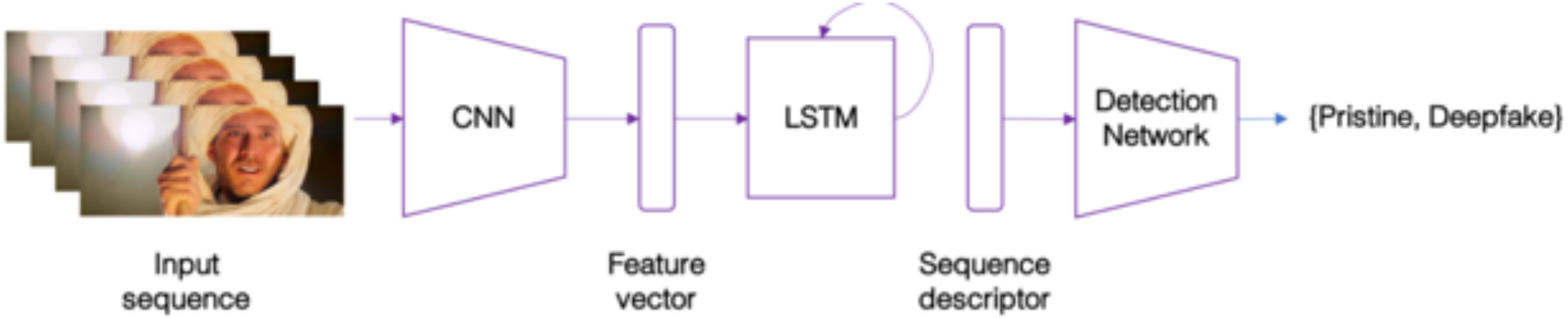
Forgery Detection Methods + RNN

Combines image forgery detection methods into a RNN model.

Takes advantage of the temporal forensic information

Why does this work?

- Deepfake autoencoder training process and differences in training image scenes
- Encoder is not aware of skin or other scene information which leads to boundary effects
- Lack of temporal awareness within the autoencoder



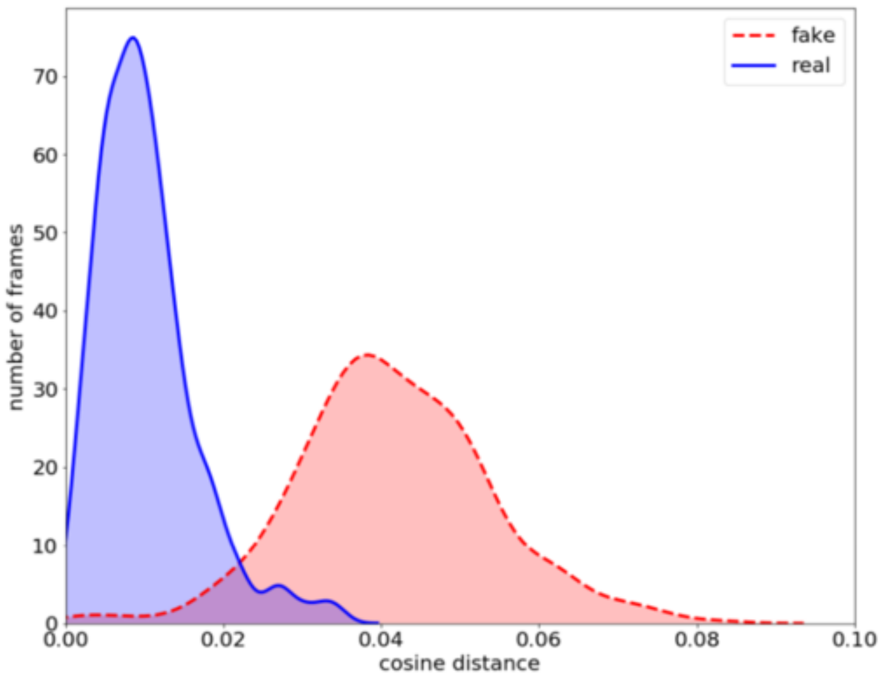
E. Sabir, J. Cheng, A. Jaiswal, W. AbdAlmageed, I. Masi, and P. Natarajan, “Recurrent Convolutional Strategies for Face Manipulation Detection in Videos,” arXiv, May 2019.
D. Guera and E. Delp, “Deepfake Video Detection Using Recurrent Neural Networks,” 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Nov. 2018.

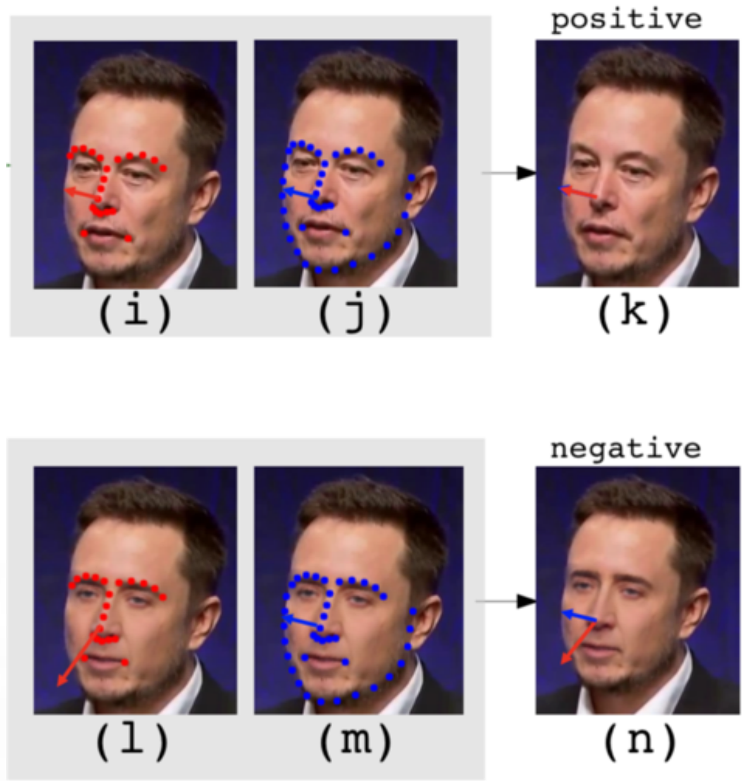
Head Poses

- Insight is that the face replacement of the target person will result in mismatched facial landmarks
- Compare head poses using all facial landmarks to just the central face landmarks

Why does this work?

- Expect "real" videos to contain head poses that are similar
- Deepfakes will result in larger differences in estimated head pose





X. Yang, Y. Li, and S. Lyu, "Exposing Deep Fakes Using Inconsistent Head Poses," arXiv, Nov. 2018.

Deepfake Artifacts

Face warping artifacts

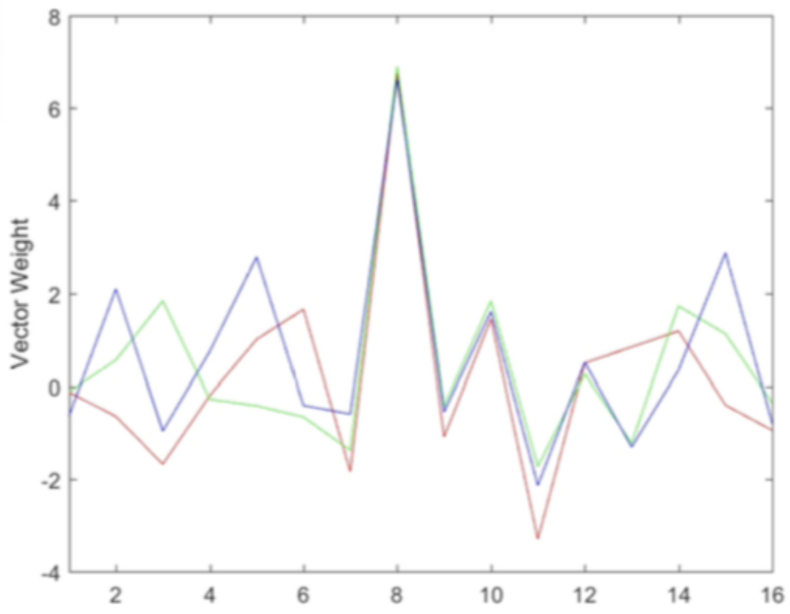
- Deepfake algorithms operate on images of a fixed size
- Faces must go through affine warping process

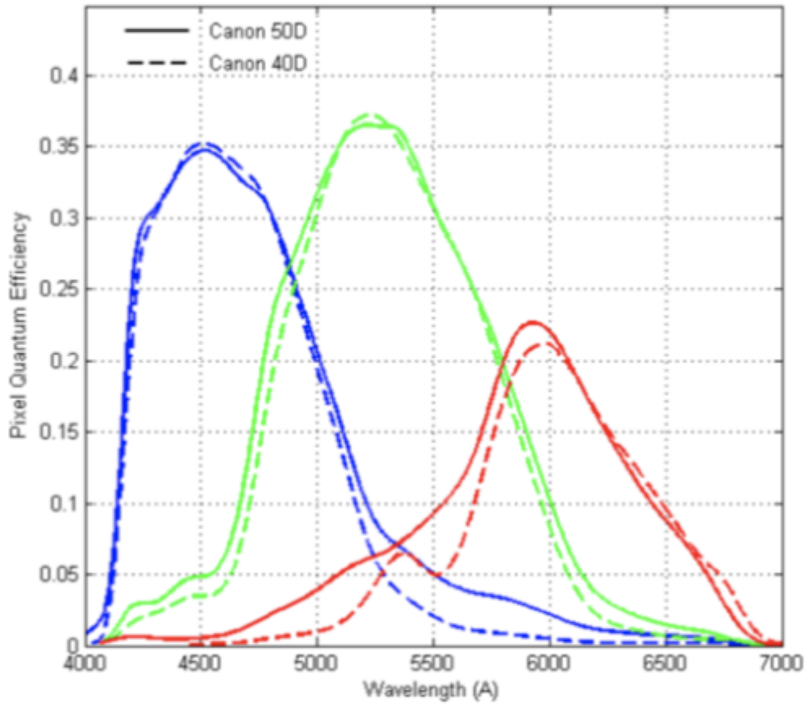
GAN generated image artifacts

- GAN generated images leave behind key artifacts
- Color image forensics
- Saturation-based forensics

Why does this work?

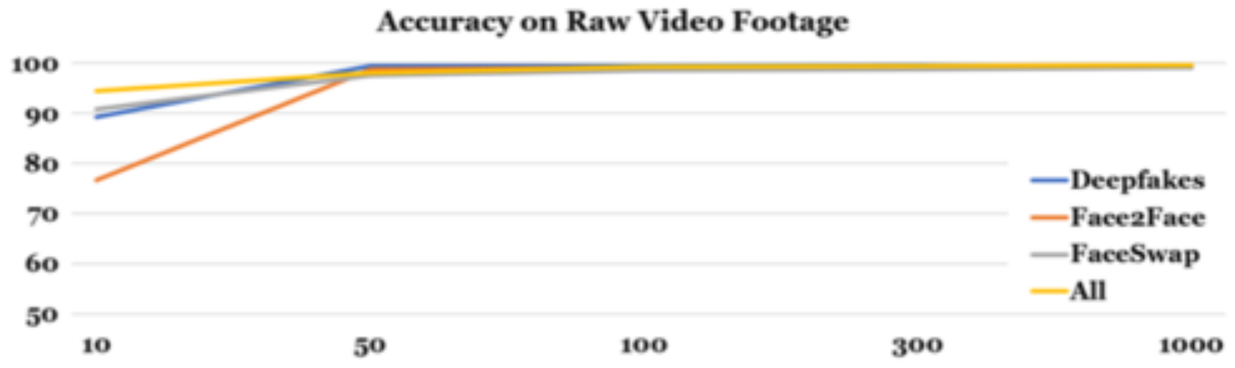
- Limitations in current deepfake creation methods
- Looking “correct” is not “is pixel-wise correct”

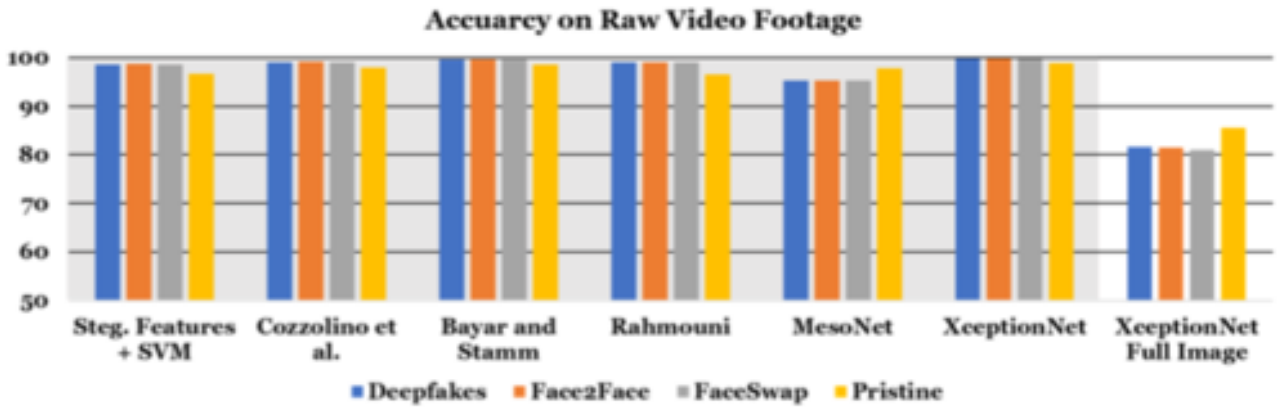


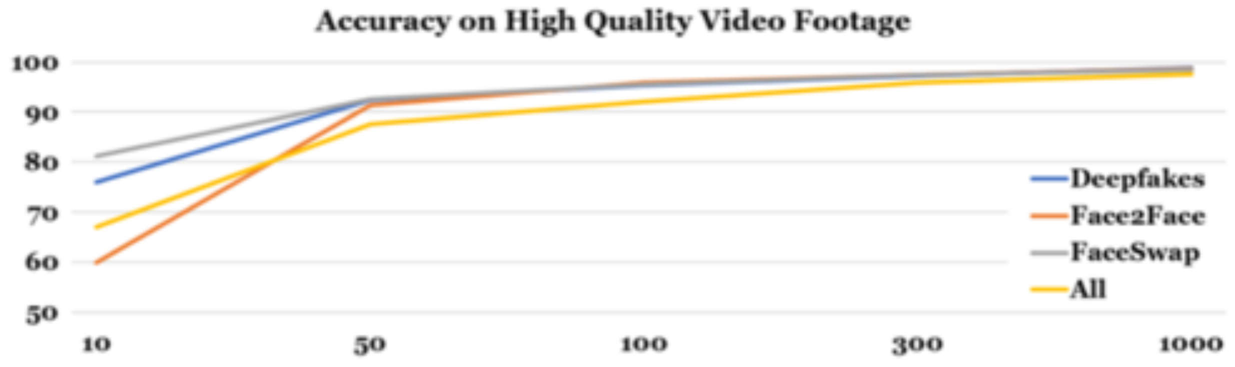


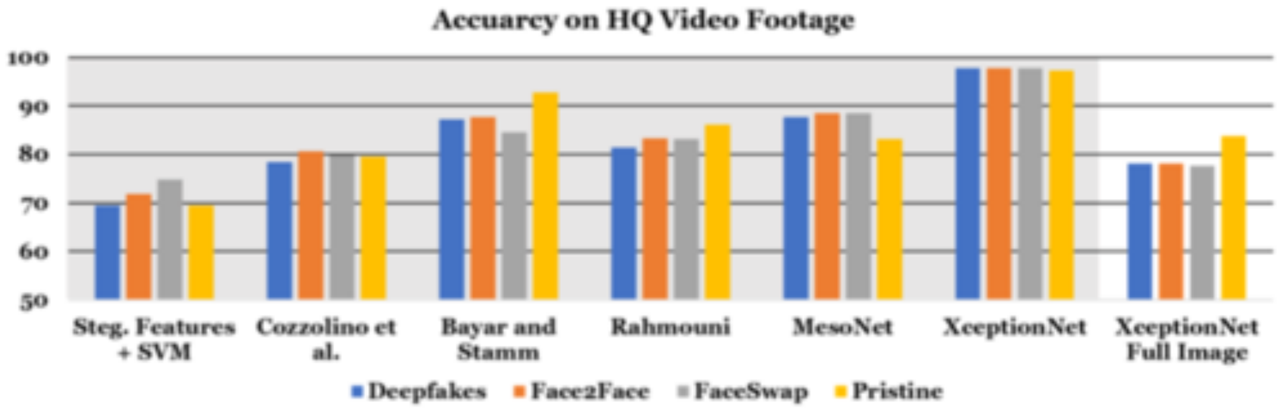
S. McCloskey and M. Albright, “Detecting GAN-generated Imagery using Color Cues,” arXiv, Dec. 2018.
Y. Li and S. Lyu, “Exposing Deepfake Videos By Detecting Face Warping Artifacts,” arXiv, Nov. 2018.

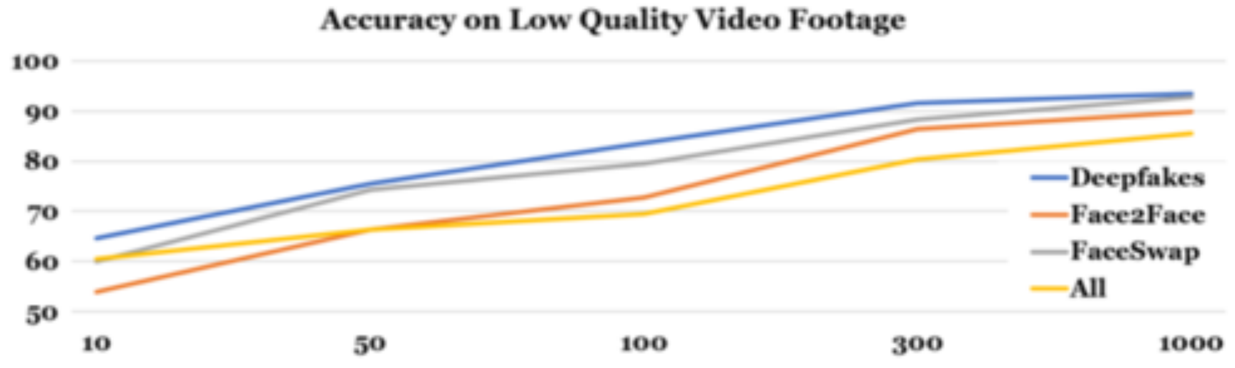
Generalized Detection Method Results

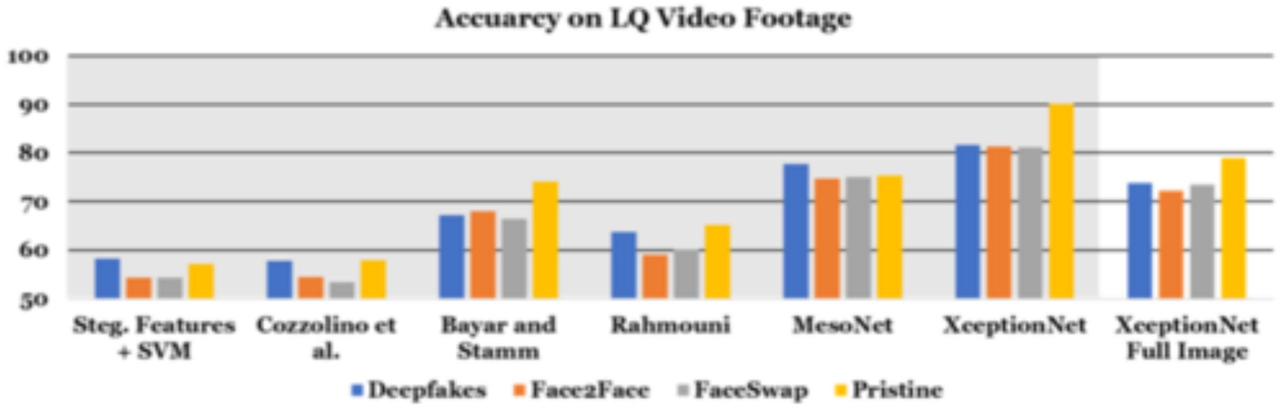












A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, “FaceForensics : Learning to Detect Manipulated Facial Images,” arXiv, Apr. 2019.

Individualized Detection Methods

- Observed that people have distinct facial and head movements which deepfake methods disrupt
- Used a 10 sec clip from a video to generate a feature vector containing 190 individual features
- Training only requires real videos of individual
- Model: one-class (real or not) SVM to determine the authenticity of the video

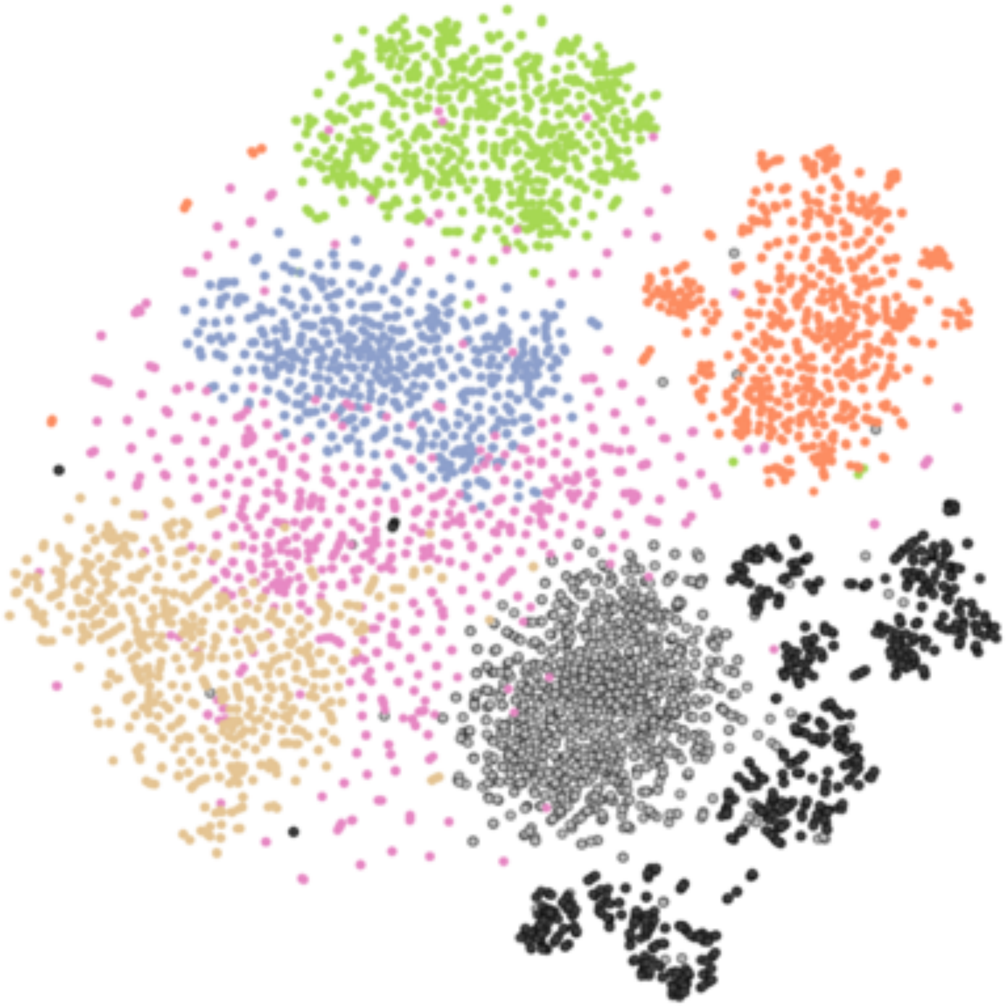


Figure 3. Shown is a 2-D visualization of the 190-D features for Hillary Clinton (brown), Barack Obama (light gray with a black border), Bernie Sanders (green), Donald Trump (orange), Elizabeth Warren (blue), random people [23] (pink), and lip-sync deep fake of Barack Obama (dark gray with a black border).

S. Agarwal and H. Farid, “Protecting World Leaders Against Deep Fakes,” CVPR Workshop, pp. 38–45, Jun. 2019.

Open Detection Method Issues

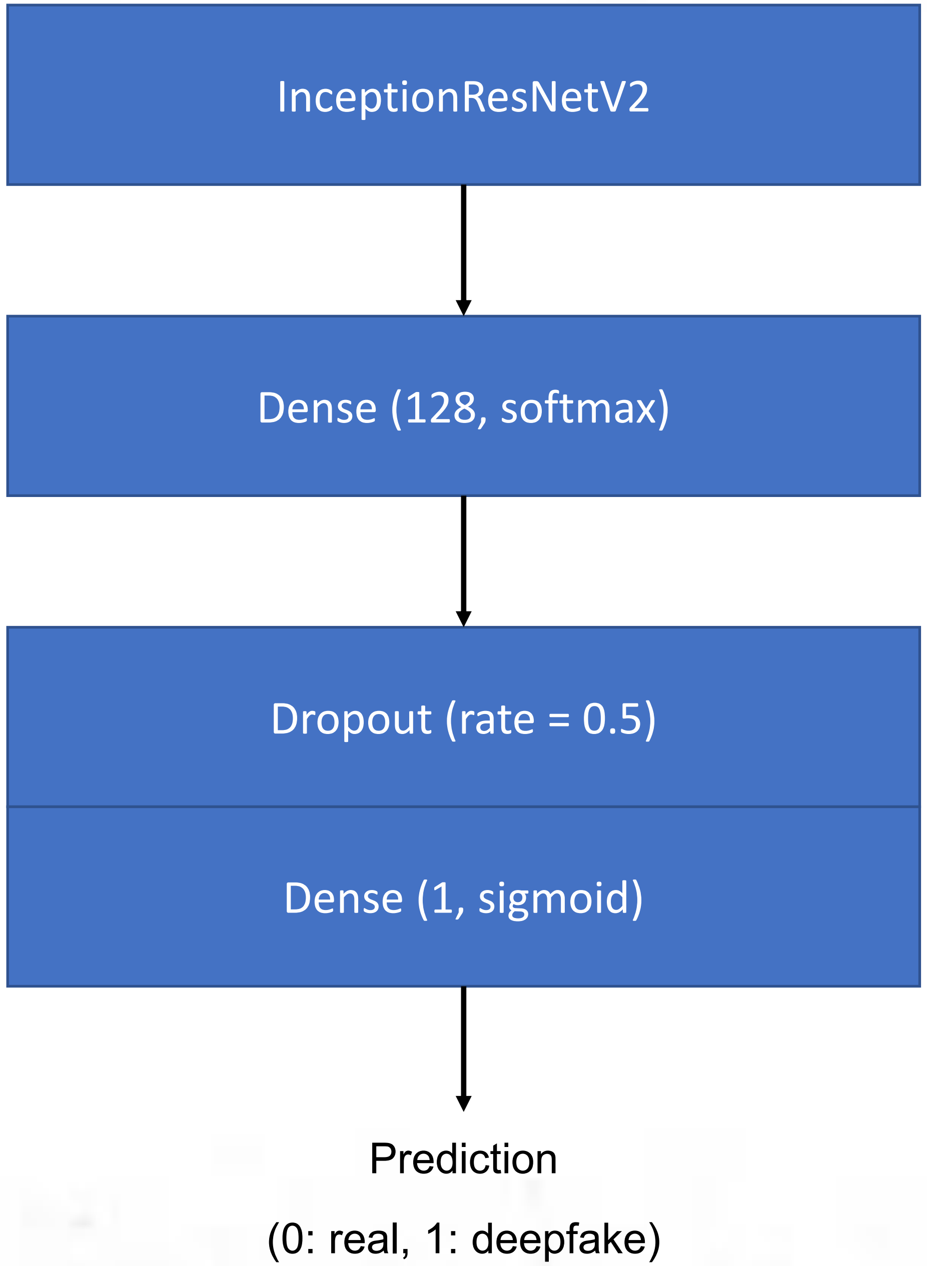
- Video compression
- Video resolution
- Rapid improvement of deepfake generation capabilities
- Ability for one model to detect all 3 deepfaking methods
- Selective deepfaking
  - Only small portions of the video is deepfaked
  - Multiple people in video with only some individuals deepfaked
- Availability of deepfake datasets

Our research into solutions to offensive use of deepfakes

How It Works

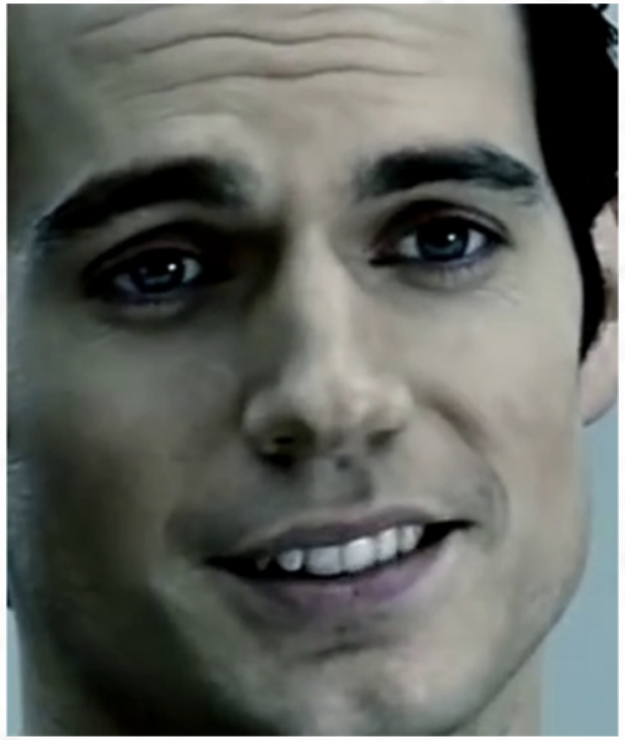
- Image classification
- Mouth detector
  - Attempting to detect all 3 types of deepfakes
  - By definition, deepfakes alter the target's mouth
- Limited videos to only some online platforms

Dataset	# of Real	# of Deepfake	# of Faces Extracted
Train	84	111	23,492
Validation	21	27	5,850
Test	83	17	9,500

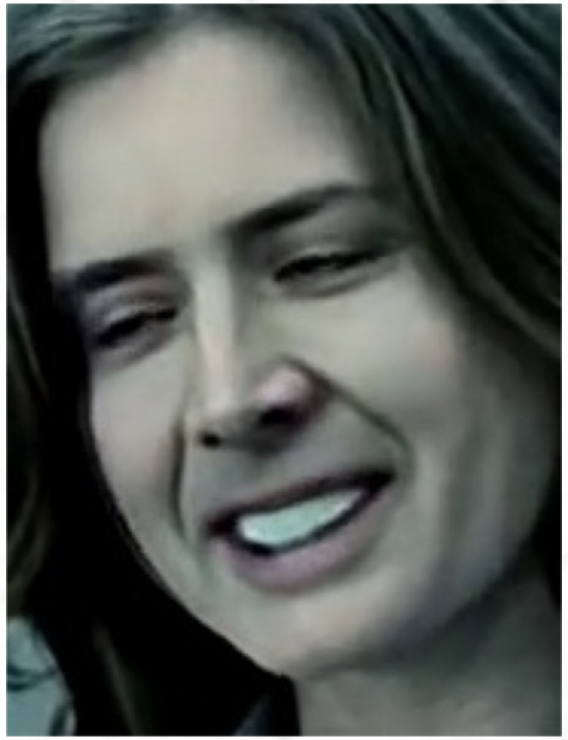


Intuition

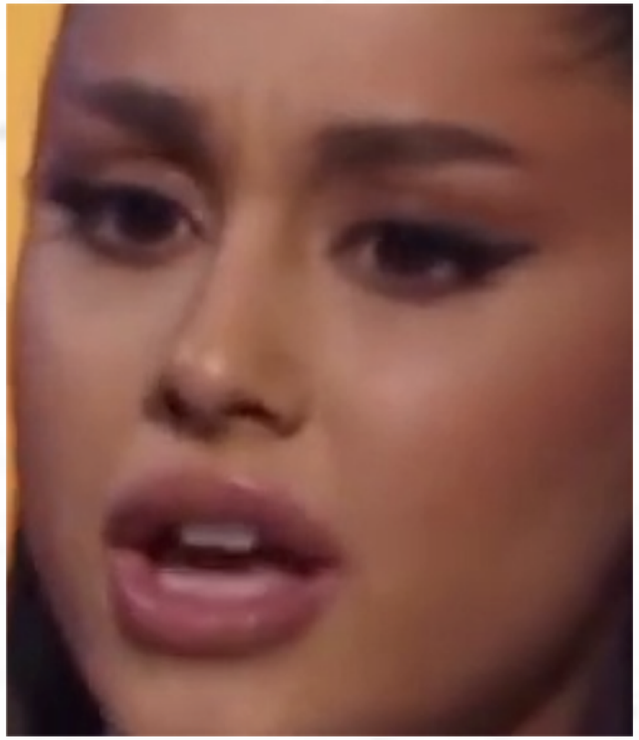
Real



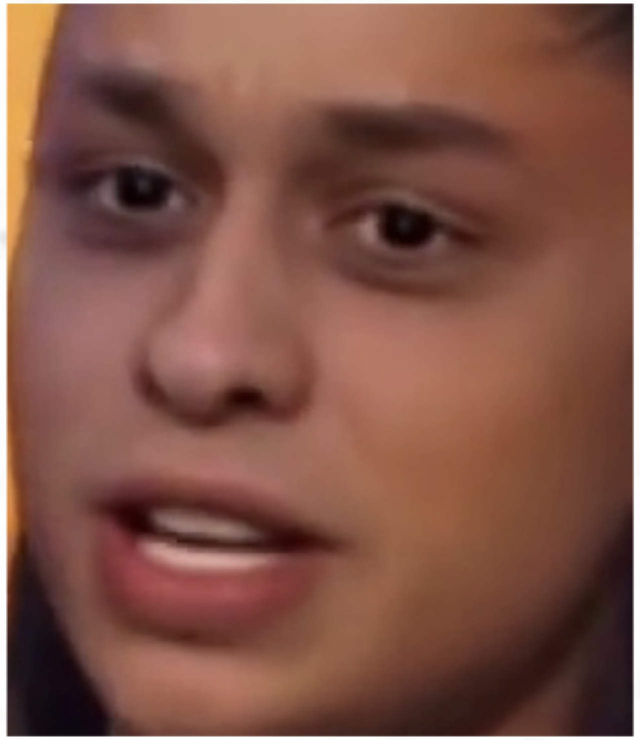
Fake



Real



Fake



Results

- Image classification validation accuracy: 83%
- MesoNet (faces) validation accuracy: 72%
- Test results on 100 videos:
  - Detection method
    - Extract first 100 frames with faces
    - Average(predictions) > threshold == deepfake
  - 7/17 (41%) deepfakes detected
  - 8/83 (10%) real videos misclassified as deepfakes

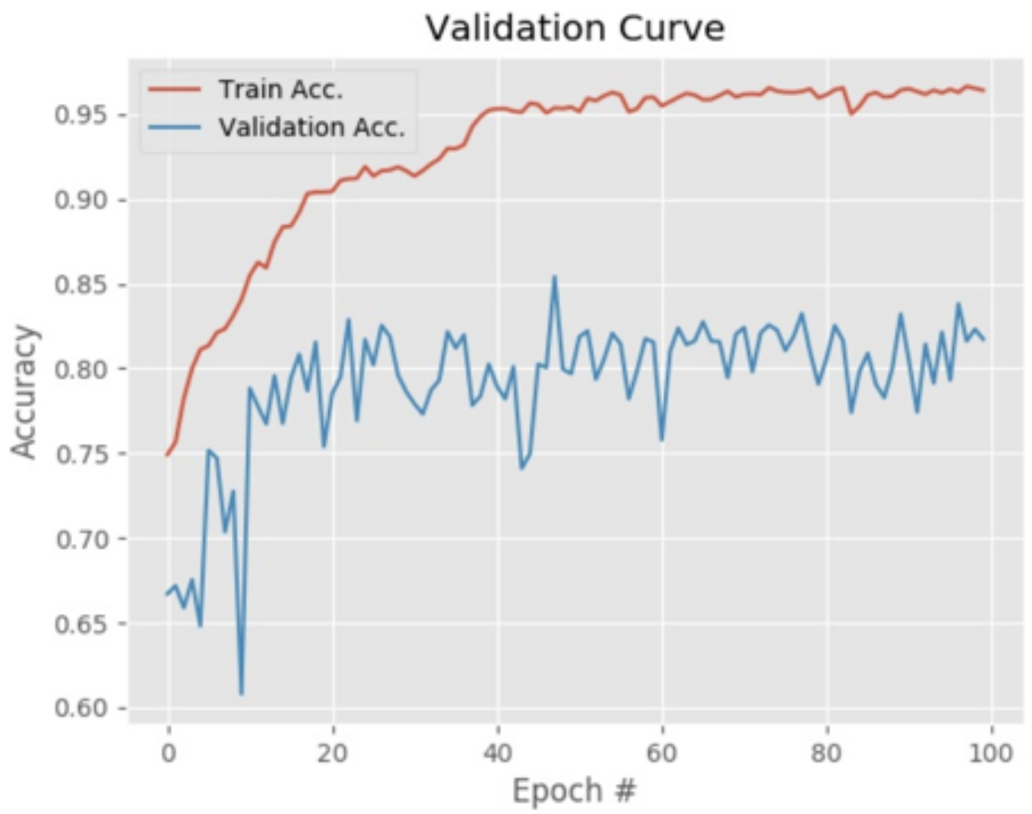
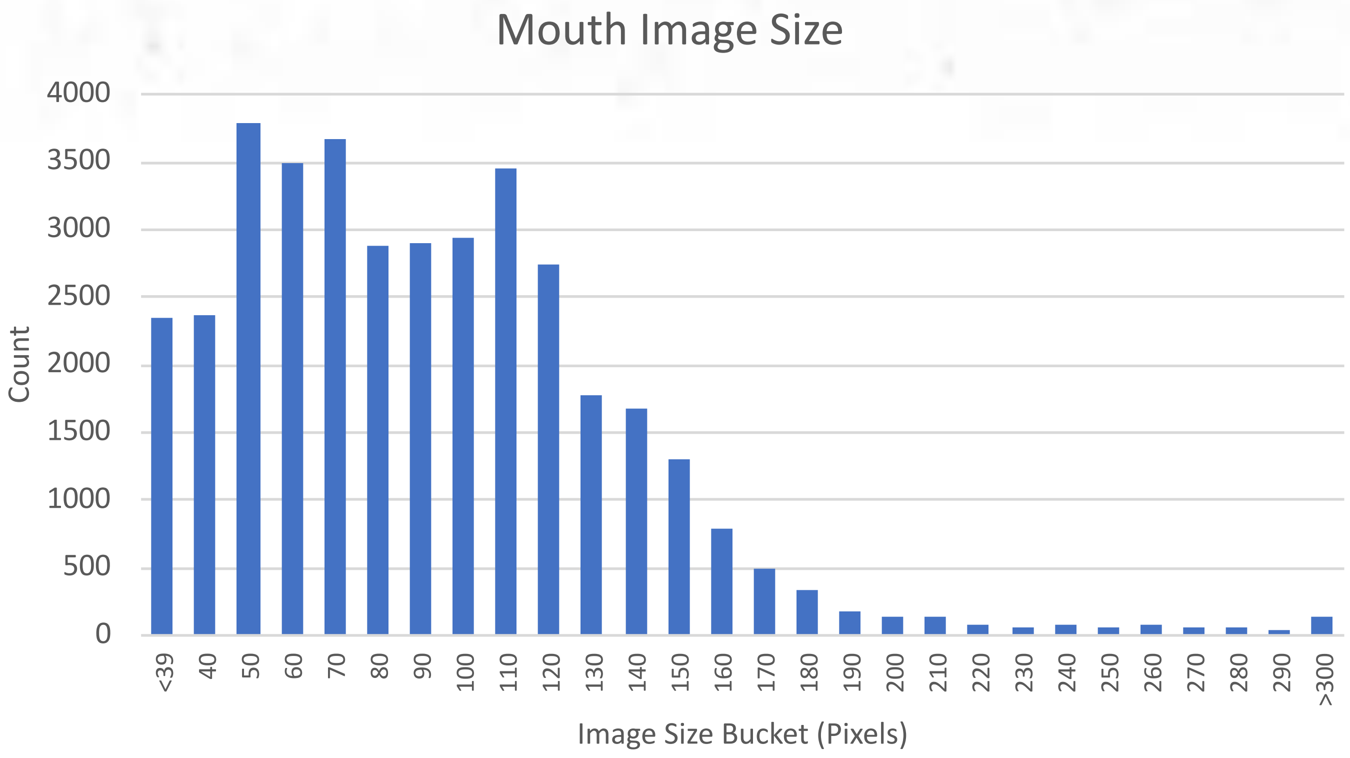
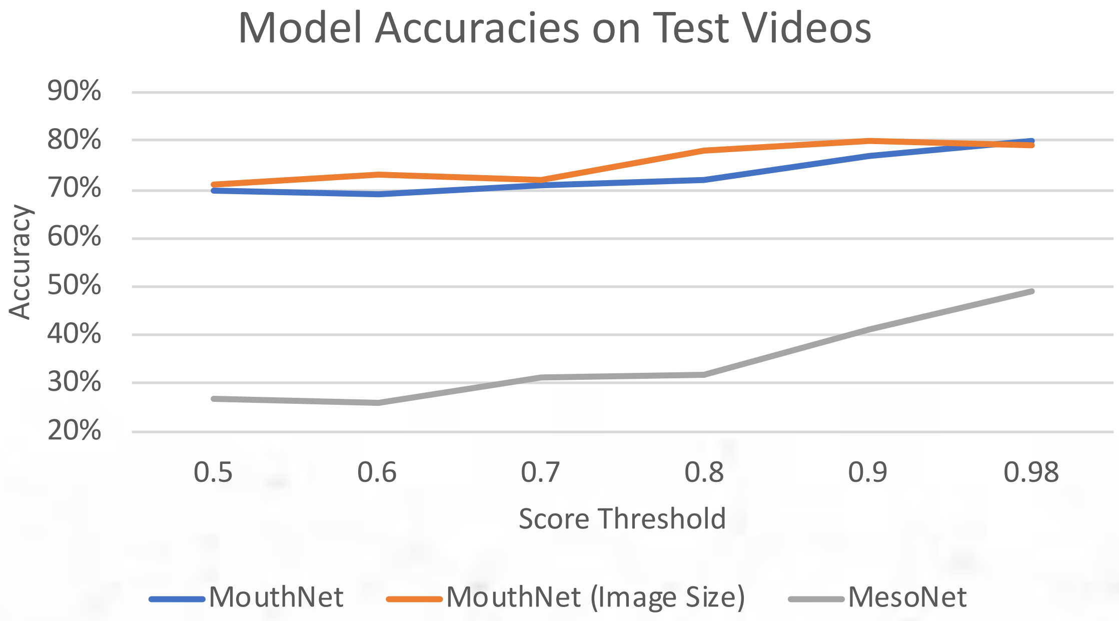


Image Size

- Created 4 models for image size buckets
- Image classification validation accuracy
  - 60-89 pixels: 81%
  - 90-119 pixels: 86%
  - 120-149 pixels: 88%
  - 150-180 pixels: 76%
- Test results on 100 videos:
  - 9/17 (53%) deepfakes detected
  - 20/78 (26%) real videos misclassified as deepfakes





Video Misclassifications

- Low resolution videos
  - High and low resolution models?
  - More samples (retrieving high and low resolution versions of the videos)
- Movie trailers and baby videos
  - Suspect movie trailers may be digitally altered triggering the model
  - Need to include baby videos in the dataset
- Multiple people within a deepfake
  - Clustering faces by similarity
- Face extraction
  - Eliminating “background” faces and other noise

Deepstar

Deepstar

- We built deepstar to help streamline our own research into deepfake defense
- It’s helpful with curation of data sets useful for detection of deepfakes
- It’s additionally helpful with research into performance of particular steps in the defensive process
- It’s additionally helpful for testing new detection algorithms
- The source code will be made available this week under a BSD-like license and at the following URL: https://github.com/zerofox-oss/deepstar
- We hope to see detection plugin contributions from the community!

Questions

ZEROFOX®

Thank You