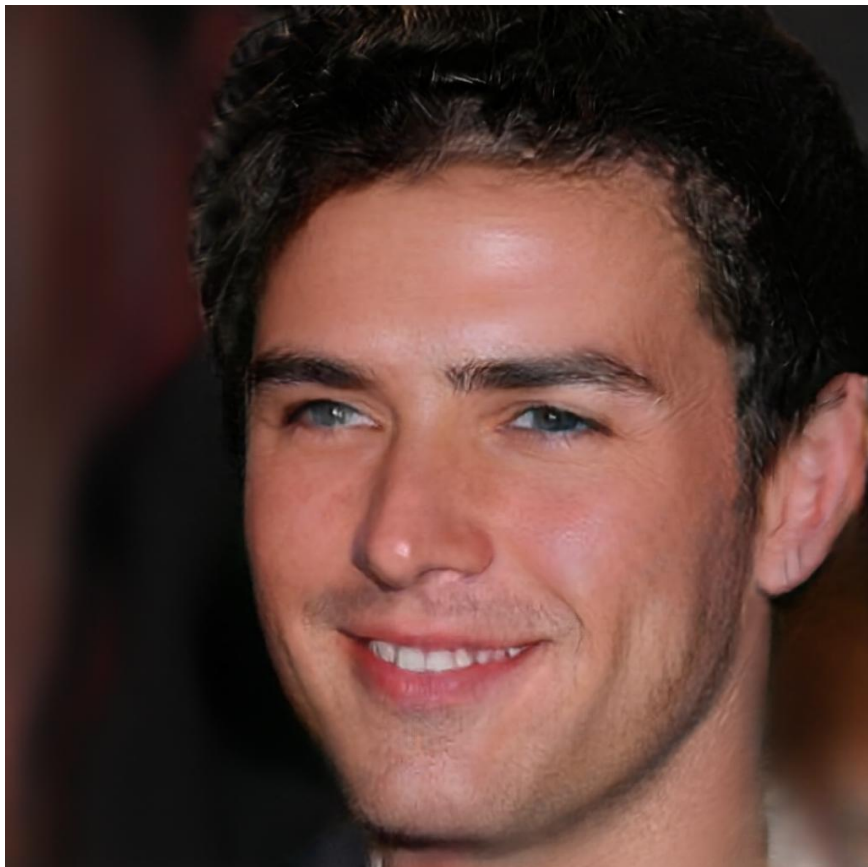**NVIDIA**

MIXED PRECISION TRAINING FOR PIX2PIXHD

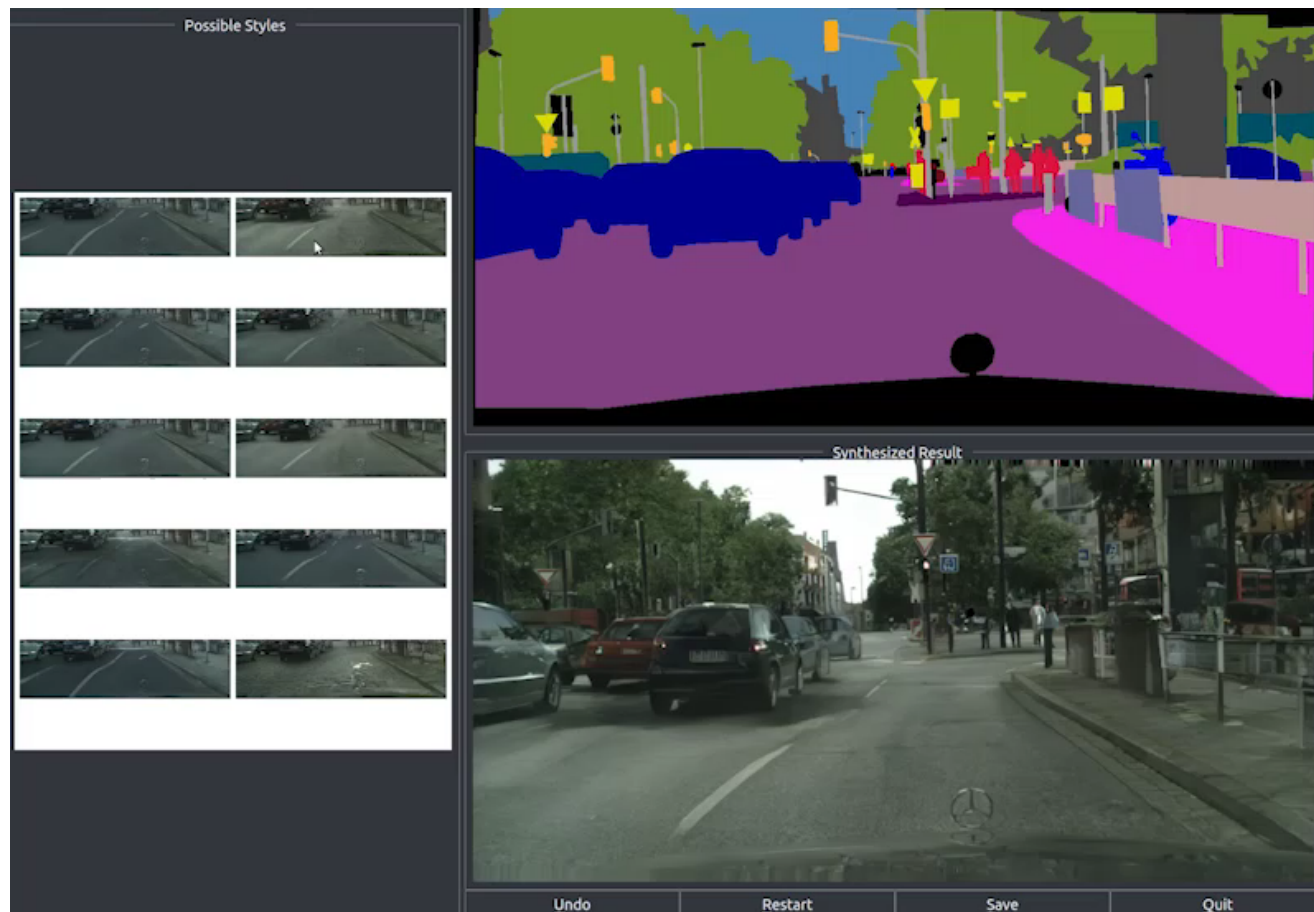Ting-Chun Wang
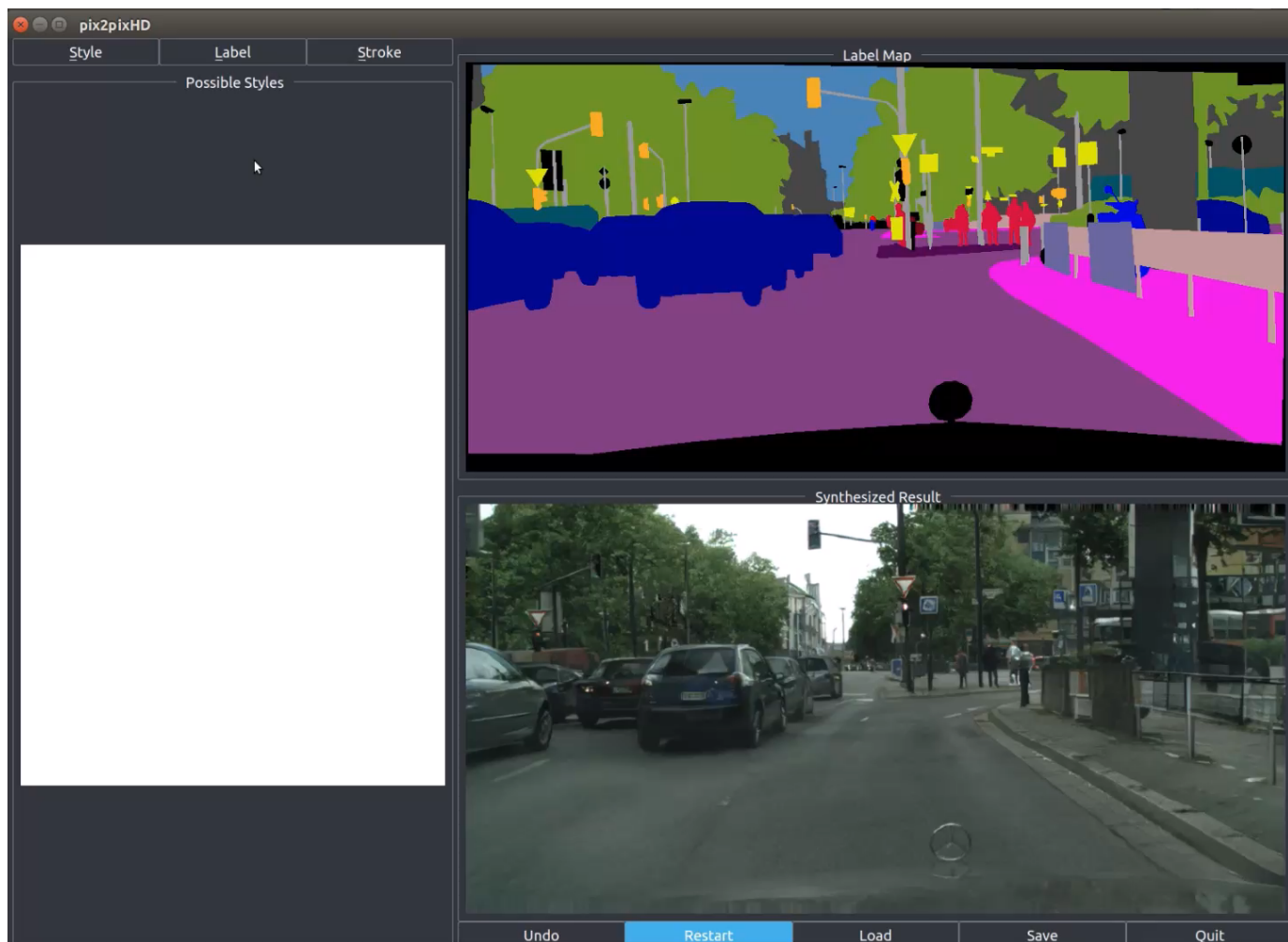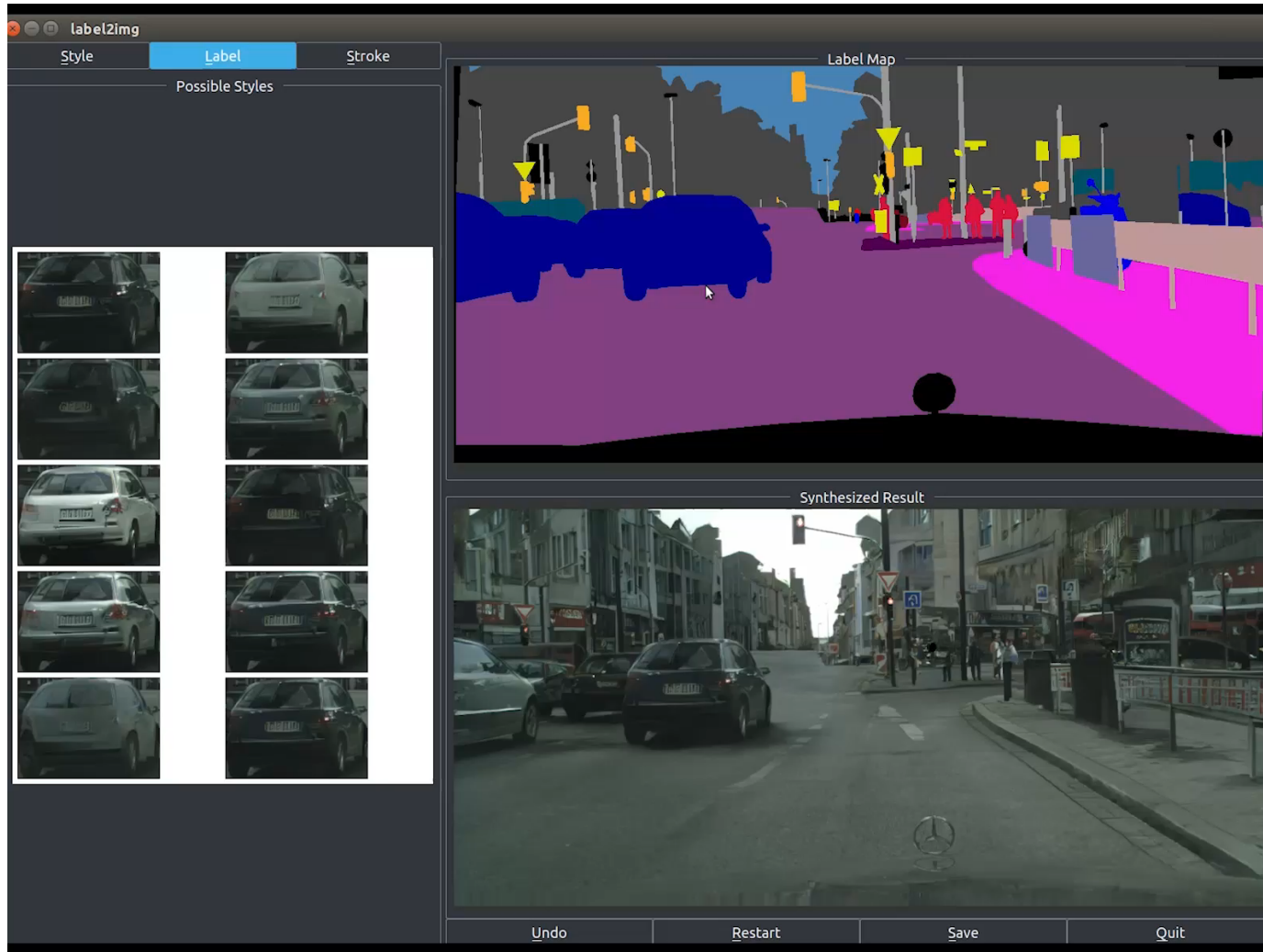
INTRODUCTION

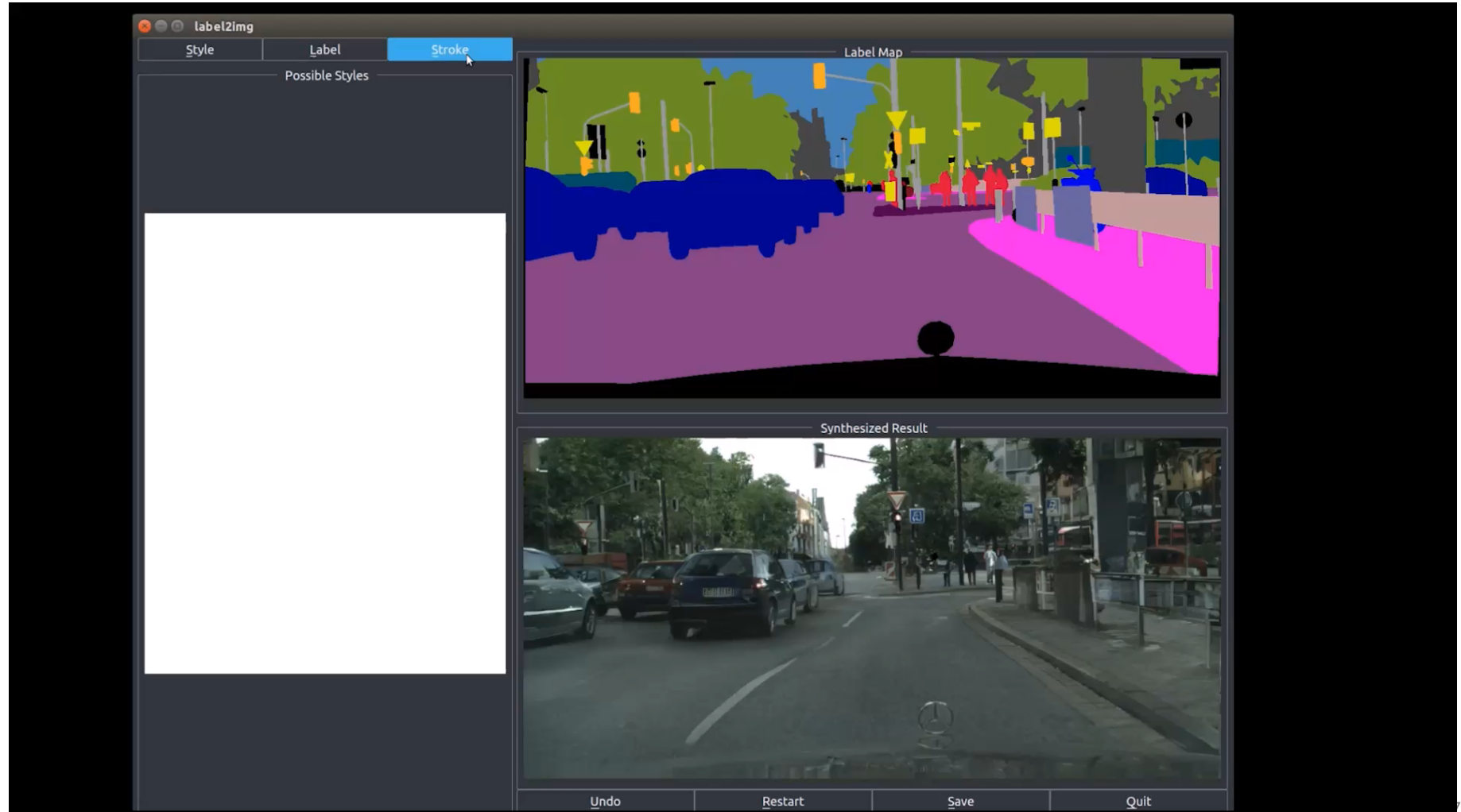# INTRODUCTION

APPLICATIONS: STYLE CHANGING
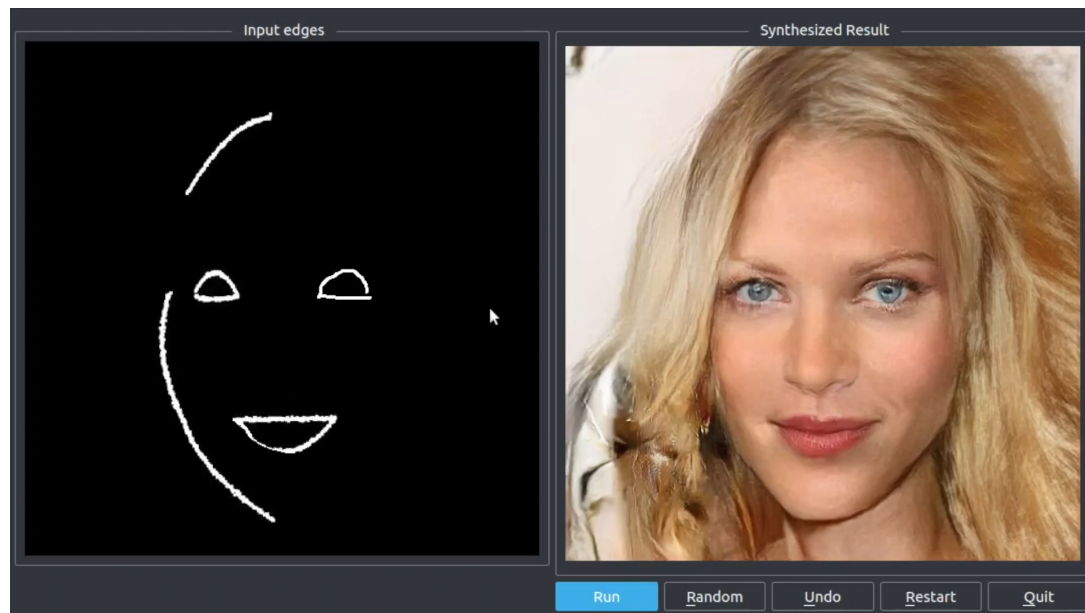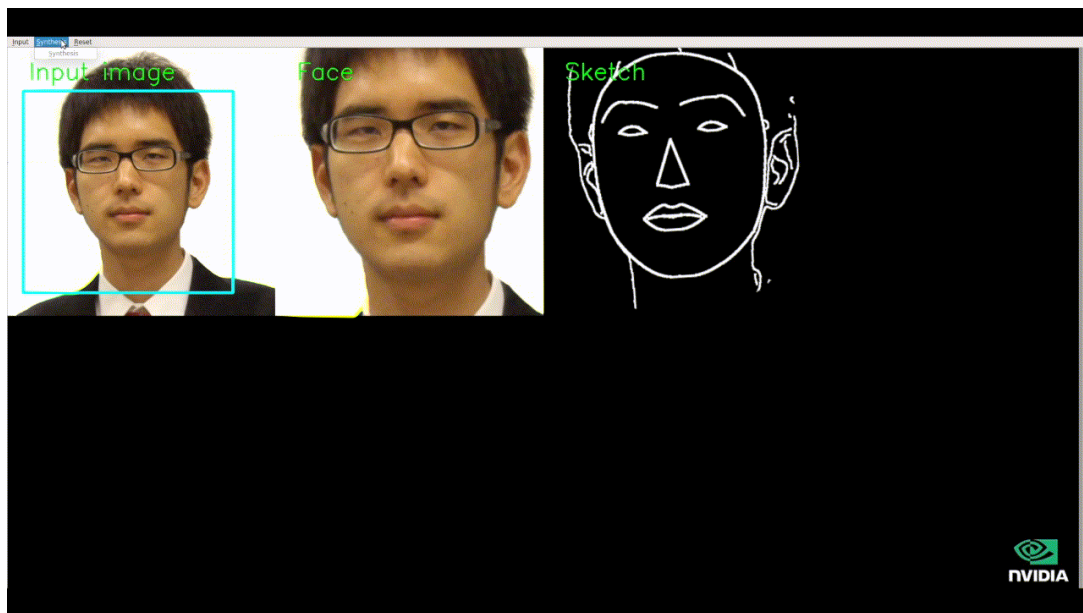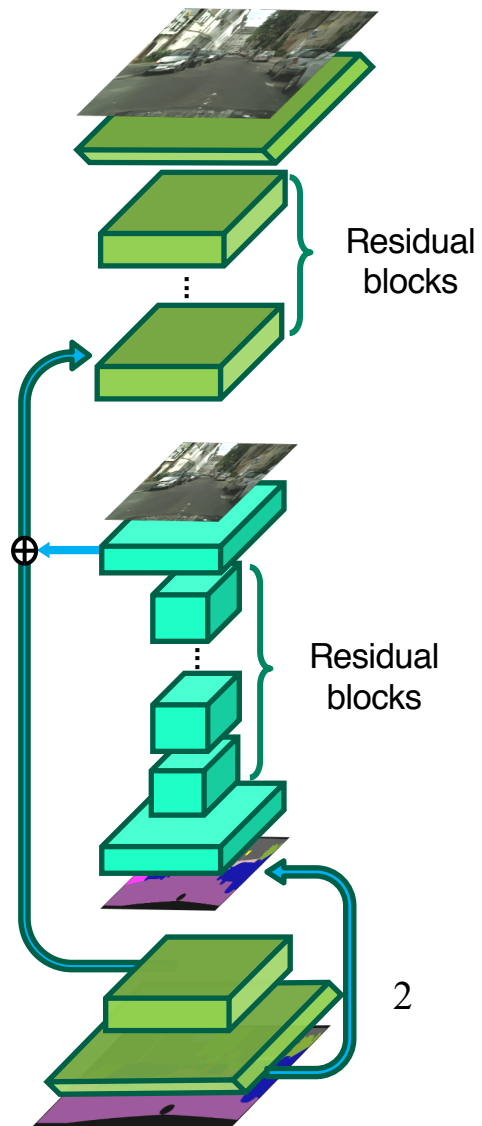
APPLICATIONS: LABEL CHANGING

APPLICATIONS: ADDING OBJECTS

APPLICATIONS: ADDING STROKES

APPLICATIONS: FACE CHANGING

Residual
blocks

Residual
blocks

2

D_3

D_2

D_1

real   synthesized

D_i

D_i

real

match

match

real   synthesized

Coarse-to-fine Generator

Residual blocks

Residual blocks

2

Multi-scale Discriminators

D_3

D_2

D_1

real

synthesized

Robust Objective

D_i

D_i
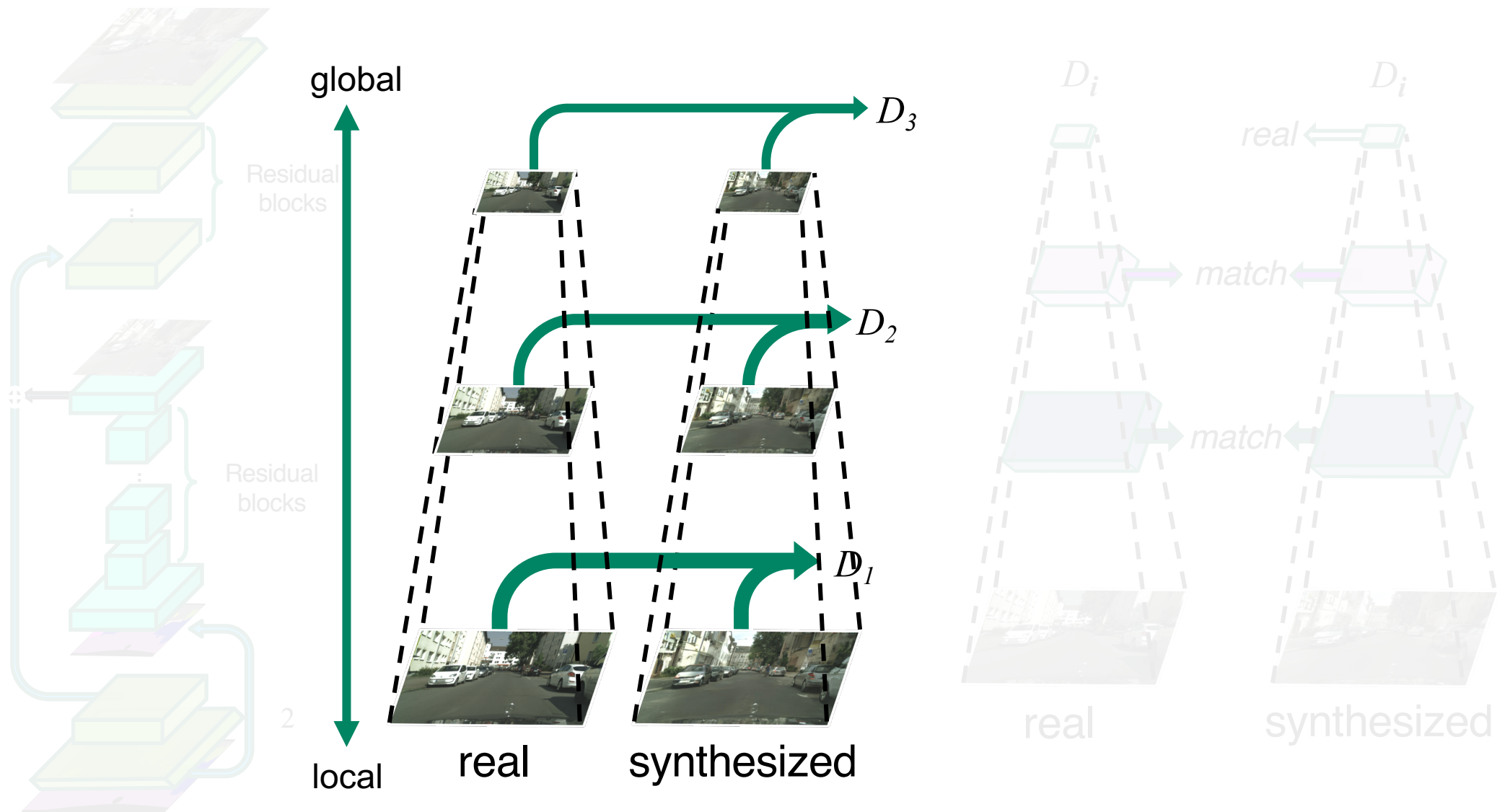
real

match

match

real

synthesized

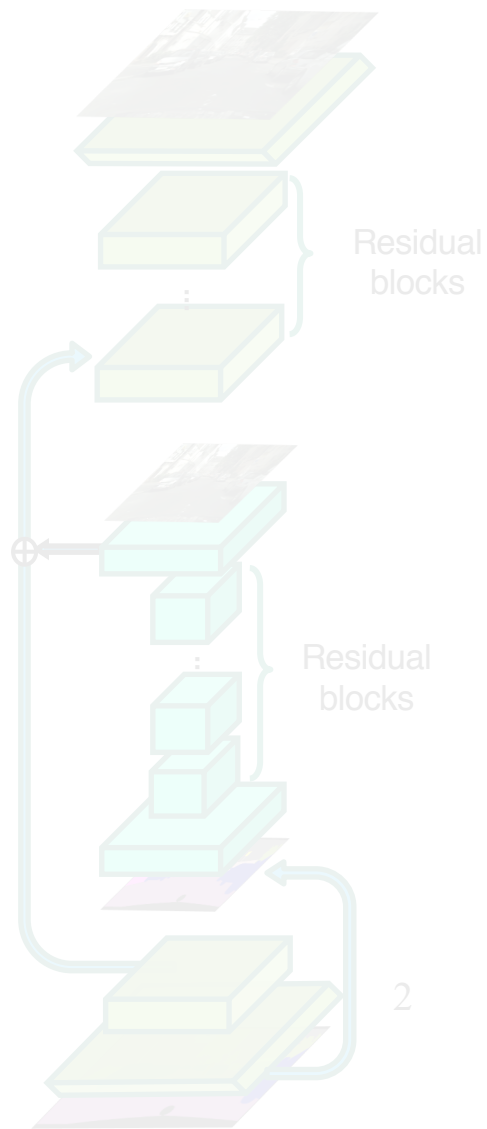*Similar ideas in Denton et al. 2015, Huang et al. 2017, Chen et al. 2017, Zhang et al. 2017

Multi-scale Discriminators

*Similar ideas in Durugkar et al. 2016, Iizuka et al. 2017, Zhang et al. 2017

Coarse-to-fine Generator   *Multi-scale Discriminators*   **Robust Objective**

Residual blocks

Residual blocks

D_3

D_2

D_1

real   synthesized

D_i D_i

real

match

match

real   synthesized

*Similar ideas in Larsen et al. 2016

CHANGES REQUIRED

```python
from apex import amp
```

```python
model, [optimizer_G, optimizer_D] = \
    amp.initialize(model, [model.optimizer_G, model.optimizer_D],
                   opt_level=cfg.amp)
```
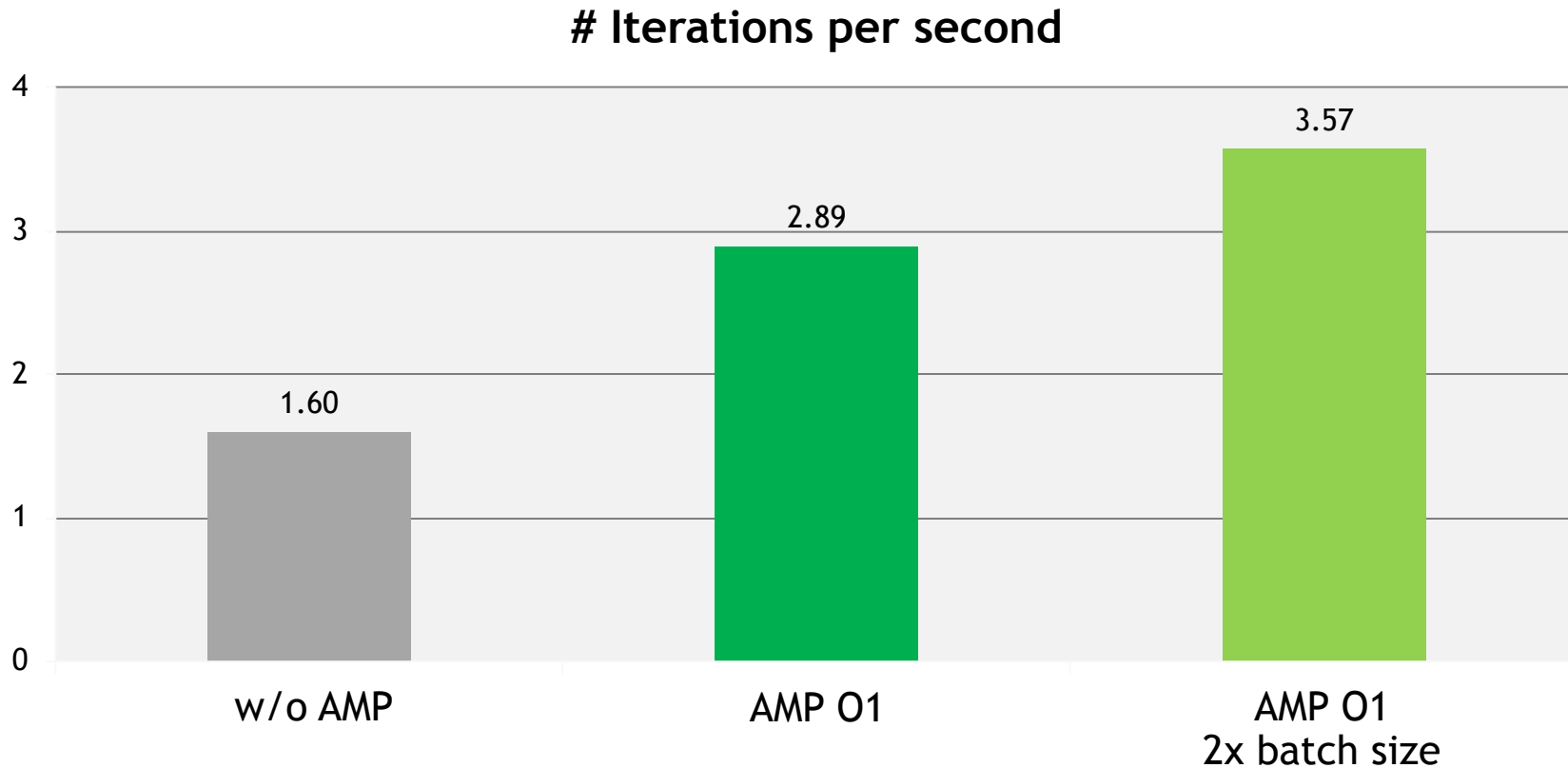
```python
with amp.scale_loss(loss_G, optimizer_G) as scaled_loss:
    scaled_loss.backward()
```

```python
with amp.scale_loss(loss_D, optimizer_D) as scaled_loss:
    scaled_loss.backward()
```

NVIDIA.

OBTAINED SPEEDUPS
1.8x faster (2.25x faster with doubled batch size)
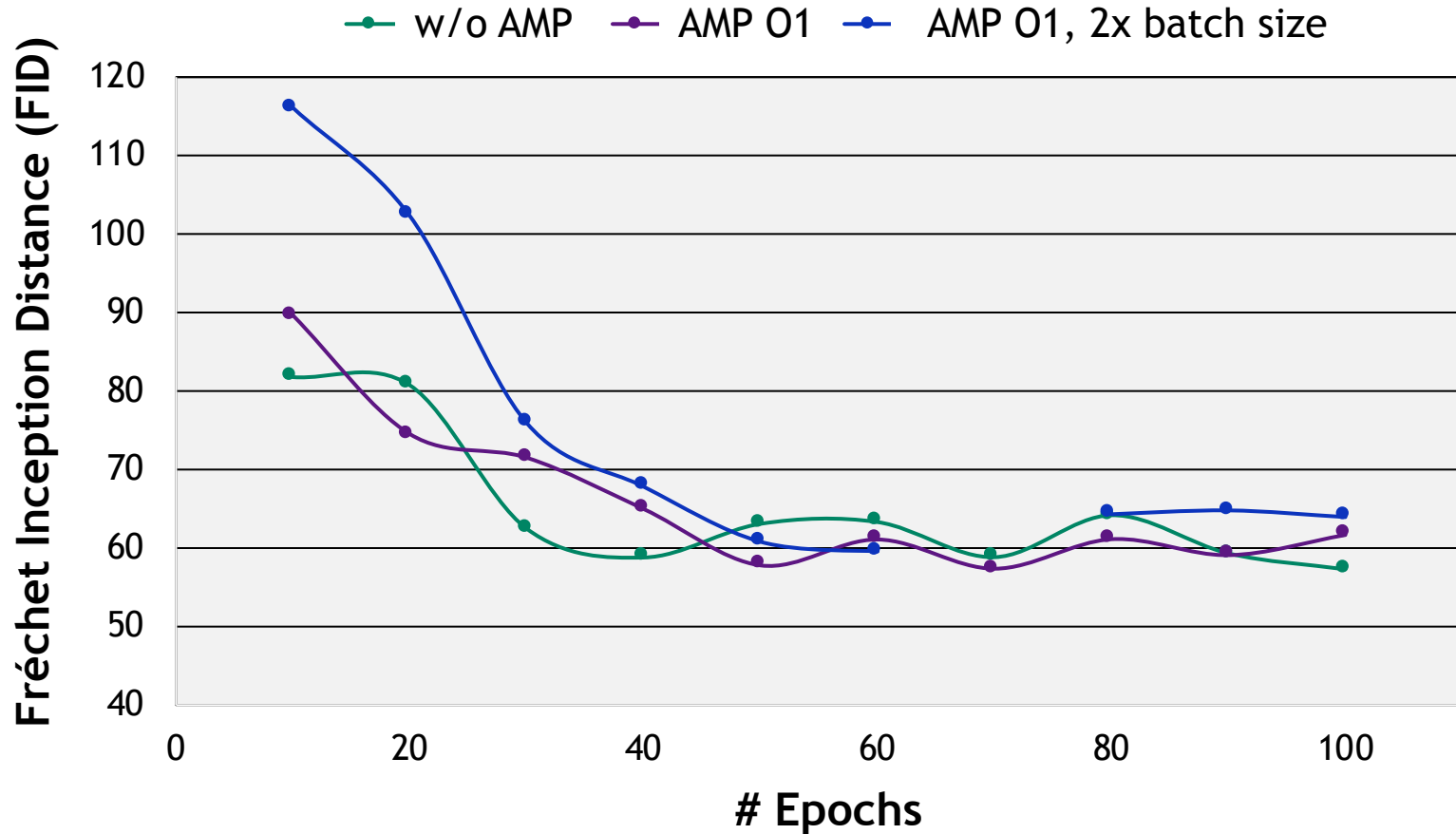
 # Iterations per second

EVALUATION CRITERIA
Frechet Inception Distance

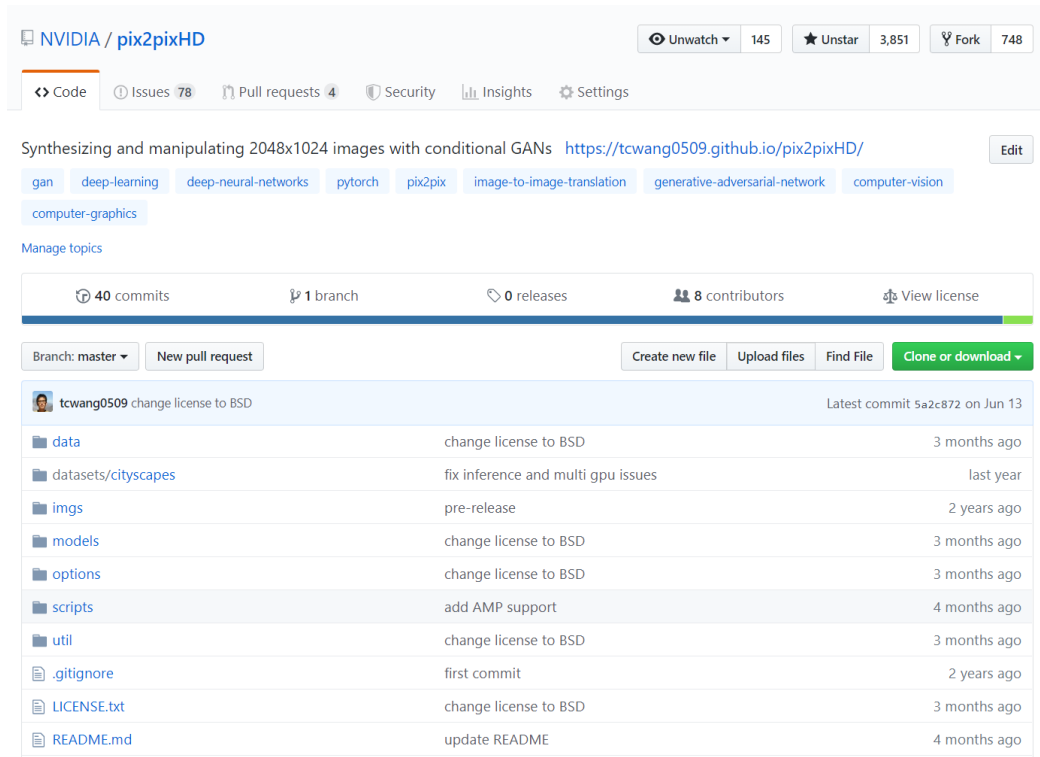$$\text{FID} = ||\mu_r - \mu_g||^2 + \text{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2}),$$

NVIDIA.

EFFECT ON METRICS

- In general, the differences are within the error range

- Usually cannot be differentiated by human eyes

READY TO USE!

https://github.com/NVIDIA/pix2pixHD

Training with Automatic Mixed Precision (AMP) for faster speed

- To train with mixed precision support, please first install apex from: https://github.com/NVIDIA/apex
- You can then train the model by adding `--fp16`. For example,

```
#!./scripts/train_512p_fp16.sh
python -m torch.distributed.launch train.py --name label2city_512p --fp16
```

In our test case, it trains about 80% faster with AMP on a Volta machine.

CONCLUSION

Mixed precision training is useful for pix2pixHD training

1. Drop-in replacement when utilizing NVIDIA APEX AMP library

    Only need to change 4 lines of code!

2. Training is much faster for the same model and batch size

    1.8x speed up

3. Consumes less memory → may train with larger batch size

    2.25x speed up using the same model and 2x batch size

NVIDIA®