Student id:

305152969

Student full name:

Tzach Fleischer

Describe your project in words:

I used Cycle-GAN PIX2PIX to generate an image of my face with sunglasses from an image of me without sunglasses

Method used to achieve project goals and why did you choose this method:

The model uses 4 models: 2 generators, 2 discriminators
The generators are Resnets
The optimizers are ADAM

I found the model on Kaggle, it references this research:
https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix

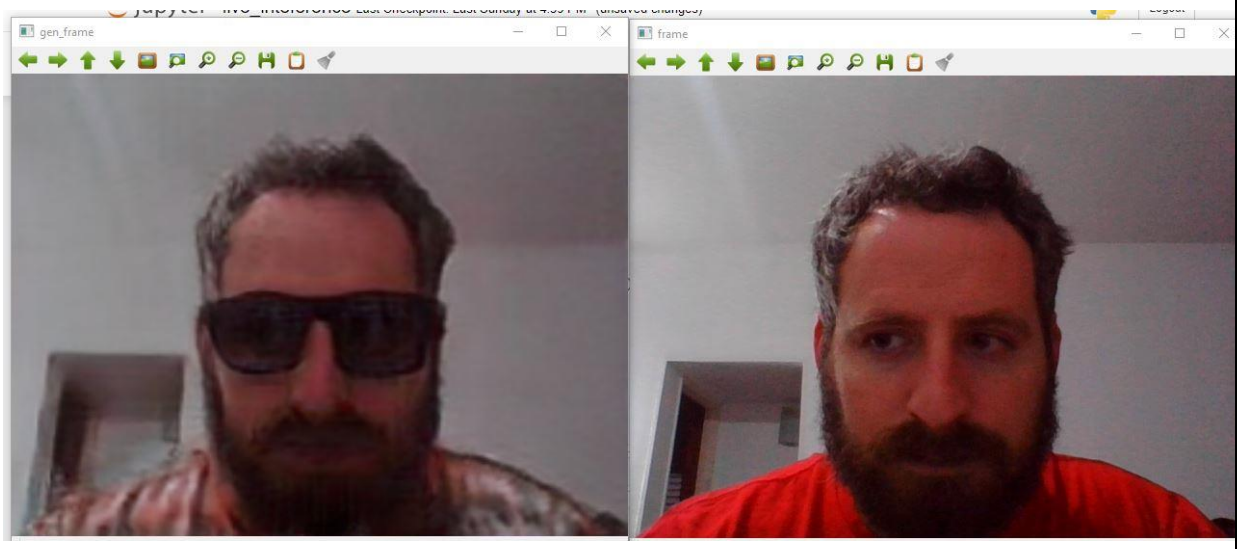
Describe your data-set (size, resolution, show 10 screenshots of train and 10 of test images):

1698 training images
1691 test images
Resolution is : 240*200

Train:

Test:

Screenshot depicting one input and its output of your project:



Screenshots of the summary of all DNNs involved:

Discriminator:

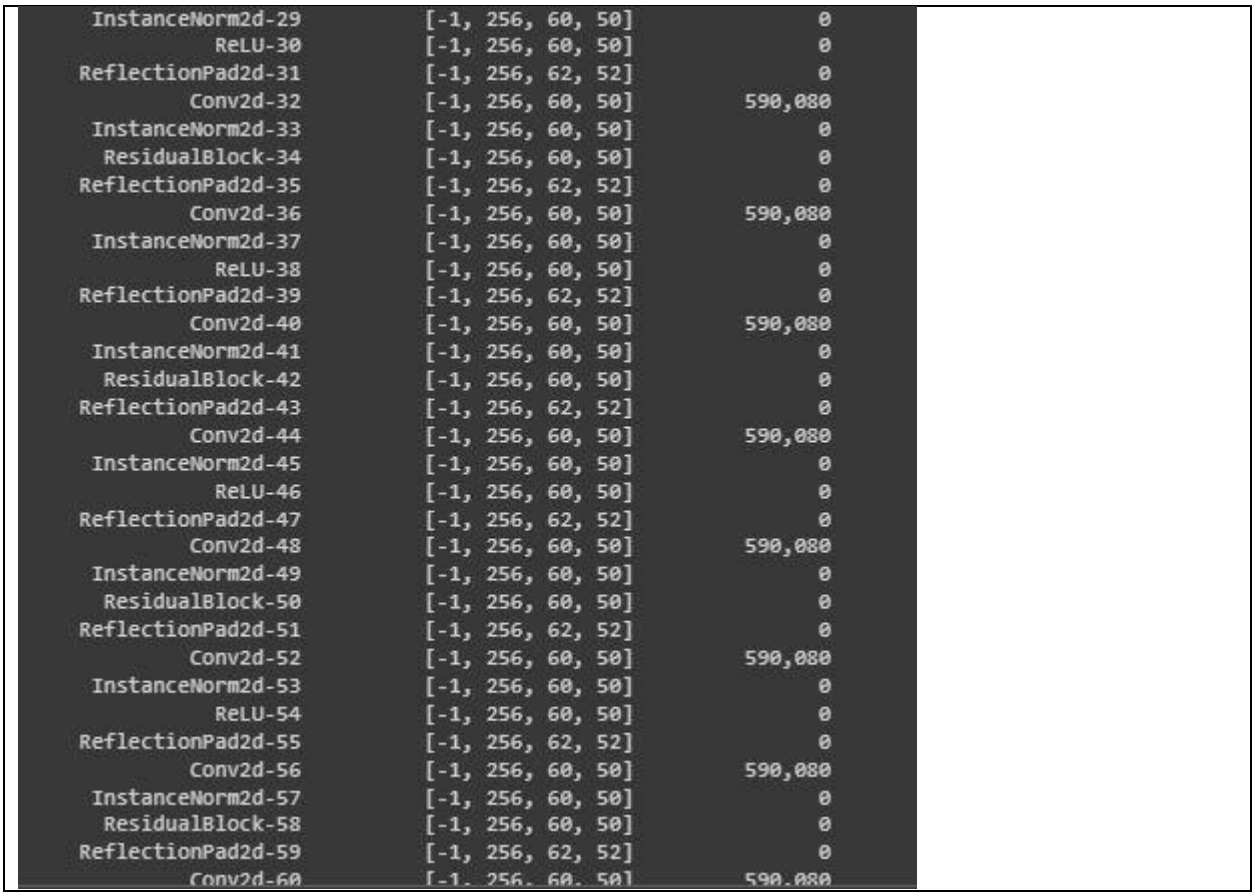
```
[27] summary(D_A, (3, 240, 200))

----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1         [-1, 64, 120, 100]           3,136
         LeakyReLU-2         [-1, 64, 120, 100]               0
            Conv2d-3          [-1, 128, 60, 50]         131,200
    InstanceNorm2d-4          [-1, 128, 60, 50]               0
         LeakyReLU-5          [-1, 128, 60, 50]               0
            Conv2d-6          [-1, 256, 30, 25]         524,544
    InstanceNorm2d-7          [-1, 256, 30, 25]               0
         LeakyReLU-8          [-1, 256, 30, 25]               0
            Conv2d-9          [-1, 512, 15, 12]       2,097,664
   InstanceNorm2d-10          [-1, 512, 15, 12]               0
        LeakyReLU-11          [-1, 512, 15, 12]               0
       ZeroPad2d-12          [-1, 512, 16, 13]               0
           Conv2d-13            [-1, 1, 15, 12]           8,193
================================================================
Total params: 2,764,737
Trainable params: 2,764,737
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.55
Forward/backward pass size (MB): 27.83
Params size (MB): 10.55
Estimated Total Size (MB): 38.92
----------------------------------------------------------------
```

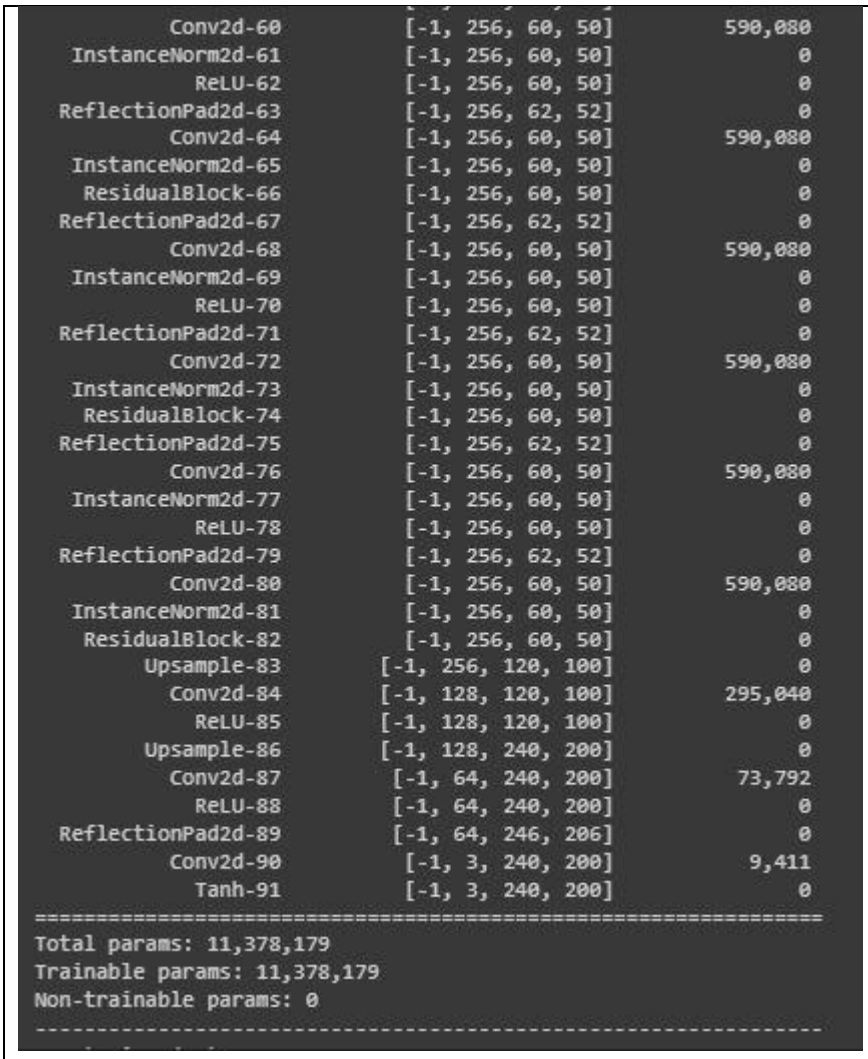
Generator:

```
from torchsummary import summary
summary(G_AB, (3, 240, 200))
```

```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
   ReflectionPad2d-1          [-1, 3, 246, 206]               0
            Conv2d-2         [-1, 64, 240, 200]           9,472
    InstanceNorm2d-3         [-1, 64, 240, 200]               0
              ReLU-4         [-1, 64, 240, 200]               0
            Conv2d-5        [-1, 128, 120, 100]          73,856
    InstanceNorm2d-6        [-1, 128, 120, 100]               0
              ReLU-7        [-1, 128, 120, 100]               0
            Conv2d-8          [-1, 256, 60, 50]         295,168
    InstanceNorm2d-9          [-1, 256, 60, 50]               0
             ReLU-10          [-1, 256, 60, 50]               0
  ReflectionPad2d-11          [-1, 256, 62, 52]               0
           Conv2d-12          [-1, 256, 60, 50]         590,080
   InstanceNorm2d-13          [-1, 256, 60, 50]               0
             ReLU-14          [-1, 256, 60, 50]               0
  ReflectionPad2d-15          [-1, 256, 62, 52]               0
           Conv2d-16          [-1, 256, 60, 50]         590,080
   InstanceNorm2d-17          [-1, 256, 60, 50]               0
    ResidualBlock-18          [-1, 256, 60, 50]               0
  ReflectionPad2d-19          [-1, 256, 62, 52]               0
           Conv2d-20          [-1, 256, 60, 50]         590,080
   InstanceNorm2d-21          [-1, 256, 60, 50]               0
             ReLU-22          [-1, 256, 60, 50]               0
  ReflectionPad2d-23          [-1, 256, 62, 52]               0
           Conv2d-24          [-1, 256, 60, 50]         590,080
   InstanceNorm2d-25          [-1, 256, 60, 50]               0
    ResidualBlock-26          [-1, 256, 60, 50]               0
  ReflectionPad2d-27          [-1, 256, 62, 52]               0
           Conv2d-28          [-1, 256, 60, 50]         590,080
```

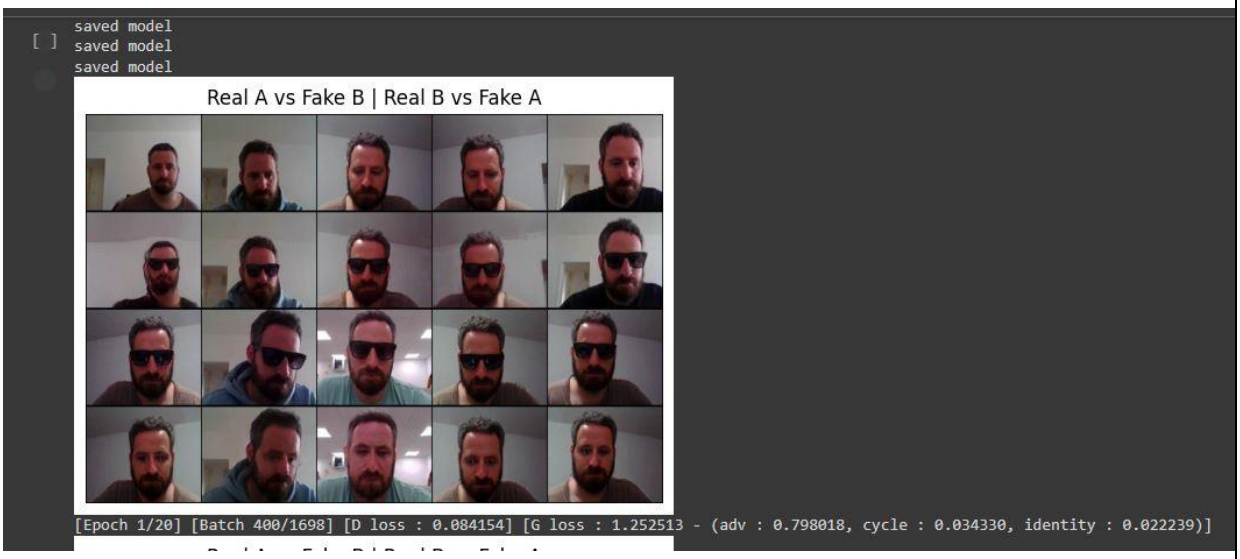


InstanceNorm2d-29	[-1, 256, 60, 50]	0
ReLU-30	[-1, 256, 60, 50]	0
ReflectionPad2d-31	[-1, 256, 62, 52]	0
Conv2d-32	[-1, 256, 60, 50]	590,080
InstanceNorm2d-33	[-1, 256, 60, 50]	0
ResidualBlock-34	[-1, 256, 60, 50]	0
ReflectionPad2d-35	[-1, 256, 62, 52]	0
Conv2d-36	[-1, 256, 60, 50]	590,080
InstanceNorm2d-37	[-1, 256, 60, 50]	0
ReLU-38	[-1, 256, 60, 50]	0
ReflectionPad2d-39	[-1, 256, 62, 52]	0
Conv2d-40	[-1, 256, 60, 50]	590,080
InstanceNorm2d-41	[-1, 256, 60, 50]	0
ResidualBlock-42	[-1, 256, 60, 50]	0
ReflectionPad2d-43	[-1, 256, 62, 52]	0
Conv2d-44	[-1, 256, 60, 50]	590,080
InstanceNorm2d-45	[-1, 256, 60, 50]	0
ReLU-46	[-1, 256, 60, 50]	0
ReflectionPad2d-47	[-1, 256, 62, 52]	0
Conv2d-48	[-1, 256, 60, 50]	590,080
InstanceNorm2d-49	[-1, 256, 60, 50]	0
ResidualBlock-50	[-1, 256, 60, 50]	0
ReflectionPad2d-51	[-1, 256, 62, 52]	0
Conv2d-52	[-1, 256, 60, 50]	590,080
InstanceNorm2d-53	[-1, 256, 60, 50]	0
ReLU-54	[-1, 256, 60, 50]	0
ReflectionPad2d-55	[-1, 256, 62, 52]	0
Conv2d-56	[-1, 256, 60, 50]	590,080
InstanceNorm2d-57	[-1, 256, 60, 50]	0
ResidualBlock-58	[-1, 256, 60, 50]	0
ReflectionPad2d-59	[-1, 256, 62, 52]	0
Conv2d-60	[-1, 256, 60, 50]	590,080



```
          Conv2d-60          [-1, 256, 60, 50]         590,080
    InstanceNorm2d-61          [-1, 256, 60, 50]               0
              ReLU-62          [-1, 256, 60, 50]               0
   ReflectionPad2d-63          [-1, 256, 62, 52]               0
          Conv2d-64          [-1, 256, 60, 50]         590,080
    InstanceNorm2d-65          [-1, 256, 60, 50]               0
     ResidualBlock-66          [-1, 256, 60, 50]               0
   ReflectionPad2d-67          [-1, 256, 62, 52]               0
          Conv2d-68          [-1, 256, 60, 50]         590,080
    InstanceNorm2d-69          [-1, 256, 60, 50]               0
              ReLU-70          [-1, 256, 60, 50]               0
   ReflectionPad2d-71          [-1, 256, 62, 52]               0
          Conv2d-72          [-1, 256, 60, 50]         590,080
    InstanceNorm2d-73          [-1, 256, 60, 50]               0
     ResidualBlock-74          [-1, 256, 60, 50]               0
   ReflectionPad2d-75          [-1, 256, 62, 52]               0
          Conv2d-76          [-1, 256, 60, 50]         590,080
    InstanceNorm2d-77          [-1, 256, 60, 50]               0
              ReLU-78          [-1, 256, 60, 50]               0
   ReflectionPad2d-79          [-1, 256, 62, 52]               0
          Conv2d-80          [-1, 256, 60, 50]         590,080
    InstanceNorm2d-81          [-1, 256, 60, 50]               0
     ResidualBlock-82          [-1, 256, 60, 50]               0
        Upsample-83        [-1, 256, 120, 100]               0
          Conv2d-84        [-1, 128, 120, 100]         295,040
            ReLU-85        [-1, 128, 120, 100]               0
        Upsample-86        [-1, 128, 240, 200]               0
          Conv2d-87         [-1, 64, 240, 200]          73,792
            ReLU-88         [-1, 64, 240, 200]               0
   ReflectionPad2d-89         [-1, 64, 246, 206]               0
          Conv2d-90          [-1, 3, 240, 200]           9,411
            Tanh-91          [-1, 3, 240, 200]               0
================================================================
Total params: 11,378,179
Trainable params: 11,378,179
Non-trainable params: 0
----------------------------------------------------------------
```

Screenshots of the training process per epoch (all loss values):



Describe how many epochs did you train, and how and why did you decide to stop:

I trained 150 epochs, around this number of epochs, the generated images started failing, and when the image wasn't easy to interpret the generated image didn’t look good, so I decided to stop

Screenshots of how training progress on a specific image (10 images overall):