Student id:

|  |
| --- |
| 323859629 |

Student full name:

|  |
| --- |
| Orel Arussi |

Describe your project in words:

|  |
| --- |
| This project is a real-time image-to-image translation system using CycleGAN, designed to add and remove sunglasses from human face images. The system allows testing both on static datasets and webcam input, with optional face cropping and live mode. A cycle-consistency loss is used to ensure the model can reconstruct original images after transformation. |

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

|  |
| --- |
| The main method used in this project is CycleGAN (Cycle-Consistent Generative Adversarial Network), which is ideal for unpaired image-to-image translation. CycleGAN was chosen because it enables training on two separate domains (faces with sunglasses and without sunglasses) without requiring paired training examples. This flexibility was essential for our dataset, which includes two unaligned image sets.  In addition to CycleGAN, we used **MediaPipe face detection** to automatically crop the face region in each image before training and inference. This ensured that the model would focus only on the relevant part of the image — the face — leading to better translation quality and more consistent results across different faces and lighting conditions.  We also relied heavily on **Cycle Consistency Loss and Identity Loss** as part of the training objective. Cycle consistency encourages the model to reconstruct the original image after a round-trip translation (A→B→A), while identity loss helps preserve features like colors, lighting, and facial structure when the image already belongs to the target domain. Together, these losses contribute significantly to stable training and realistic outputs. |

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

|  |
| --- |
| The dataset consists of 1,200 training images for each domain (with and without sunglasses) and 60 test images per domain. All images are resized to 256x256. Face cropping is optionally applied using MediaPipe. The dataset was augmented using horizontal flips and resizing. Screenshots are shown below. |

Screenshot depicting one input and its output of your project:

|  |
| --- |
|  |

Screenshots of the summary of all DNNs involved:

|  |
| --- |
|  |

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:

|  |
| --- |
| The model was trained for 110 epochs. The training was stopped at epoch 110 after observing that the generator, discriminator, cycle, and identity losses converged to stable and low values. Visual inspection of generated outputs confirmed high reconstruction quality |

תמונה שמכילה אביזרים לעיניים, סלפי, משקפת, אביזר

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.תמונה שמכילה אביזרים לעיניים, אביזר, סלפי, משקפת

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.תמונה שמכילה אביזרים לעיניים, משקפת, אביזר, סלפי

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.תמונה שמכילה אביזרים לעיניים, אביזר, משקפת, סלפי

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.תמונה שמכילה אביזרים לעיניים, אדם, פני אדם, סלפי

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.

תמונה שמכילה פני אדם, אדם, מצח, אביזרים לעיניים

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.תמונה שמכילה פני אדם, אדם, איש, מצח

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.תמונה שמכילה פני אדם, גבה, מצח, אדם

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.תמונה שמכילה מצח, פני אדם, גבה, אדם

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.תמונה שמכילה אביזרים לעיניים, סלפי, אדם, הצגות

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.

**Summary of Deep Neural Networks (DNNs) Used in the Project:**

**1. Generator G\_A & G\_B (ResnetGenerator):**

* Architecture: ResNet-based generator
* Components:
  + Initial layers: ReflectionPad2D + Conv(7×7) + InstanceNorm2d + ReLU
  + Downsampling: 2 convolutional layers (stride=2)
  + **9 residual blocks (ResnetBlock)** – each with Conv → InstanceNorm → ReLU → Conv → InstanceNorm
  + Upsampling: 2 ConvTranspose2D layers
  + Final layer: Conv(7×7) + Tanh
* Output: Translated image (A→B or B→A)
* Used for both directions: G\_A (adds sunglasses), G\_B (removes sunglasses)

**2. Discriminator D\_A & D\_B (PatchGAN):**

* Architecture: 70×70 PatchGAN
* Structure:
  + Sequential Conv2D layers with increasing channels
  + LeakyReLU activations
  + No fully connected layers
* Output: Patch-wise real/fake classification
* Used to distinguish between real and generated images in each domain.
* **1. Generator G\_A & G\_B — ResNet-based Generator**
* Architecture: ResnetGenerator with 9 residual blocks
* Key layers:
* Reflection padding + Conv2D (7×7)
* Downsampling: 2 Conv layers (stride=2)
* **9 Residual Blocks**: Each with Conv → InstanceNorm → ReLU
* Upsampling: 2 ConvTranspose2D layers
* Output: Tanh() activation to generate image in range [-1, 1]
* Used for domain translation:
* G\_A (adds sunglasses)
* G\_B (removes sunglasses)
* **2. Discriminator D\_A & D\_B — PatchGAN (NLayerDiscriminator)**
* Architecture: NLayerDiscriminator with 5 convolutional layers
* Patch-based discriminator (70×70 receptive field)
* Key layers:
* Conv2D layers with increasing filters: 64 → 128 → 256 → 512
* Activations: LeakyReLU (slope=0.2)
* Normalization: InstanceNorm2D
* Final output layer: Conv2D(512→1) with stride 1
* Output: Discriminates real vs fake per **patch** rather than entire imag
* **SUMMRAY**:

This project uses a CycleGAN architecture with two generators (G\_A and G\_B) and two discriminators (D\_A and D\_B).  
The generators are ResNet-based with 9 residual blocks, enabling high-quality domain translation between faces with and without sunglasses.  
The discriminators are PatchGAN-based (NLayerDiscriminator), classifying small image patches as real or fake.  
This design ensures stable training even without paired datasets and preserves fine facial features in the generated images.

Bibliography (open-source):

[www.chatgpt.com](http://www.chatgpt.com)  
<https://github.com/ekremerakin/RealWorldOccludedFaces> for dataset  
<https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix> for cycleGAN  
[www.stackflow.com](http://www.stackflow.com) for fix bugs and code error