AI for Cybersecurity

Practical work

Adversarial Attack on Face Recognition

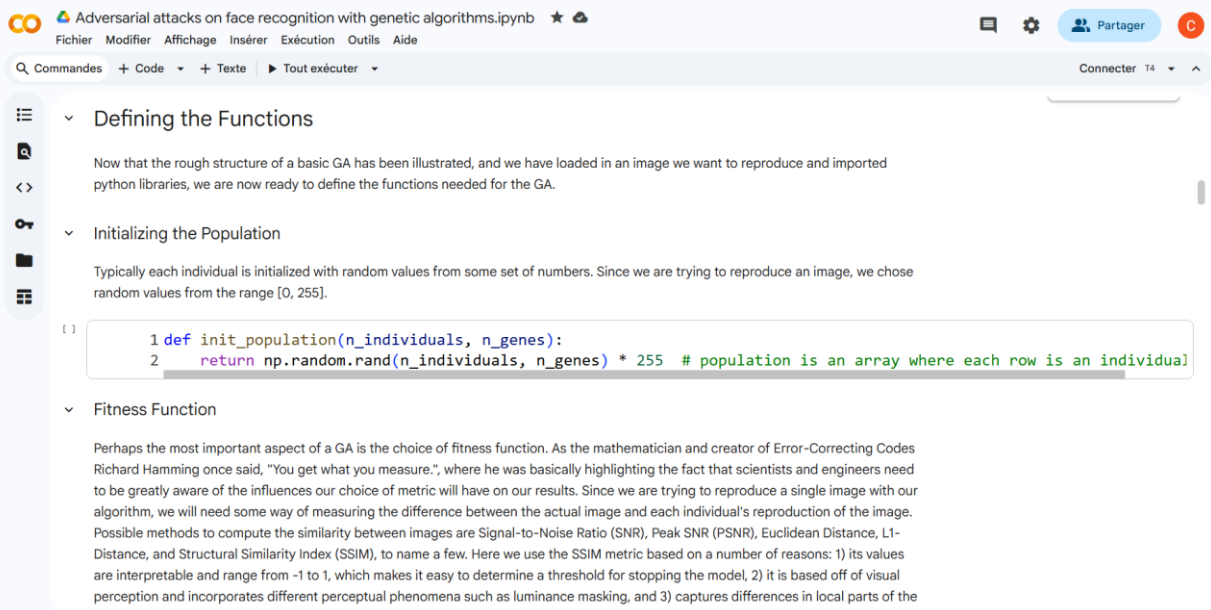
ENSI CAEN

ÉCOLE PUBLIQUE D'INGÉNIEURS
CENTRE DE RECHERCHE

www.ensicaen.fr

1. Google collab tool

Colaboratory, or “Colab” for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. It is often compared to Jupyter. Jupyter runs in your local machine and uses your systems' ram storage and CPU while colab runs in google server and gives you access to free GPU and CPU for faster processing.



Defining the Functions

Now that the rough structure of a basic GA has been illustrated, and we have loaded in an image we want to reproduce and imported python libraries, we are now ready to define the functions needed for the GA.

Initializing the Population

Typically each individual is initialized with random values from some set of numbers. Since we are trying to reproduce an image, we chose random values from the range [0, 255].

```
def init_population(n_individuals, n_genes):
    return np.random.rand(n_individuals, n_genes) * 255  # population is an array where each row is an individual
```

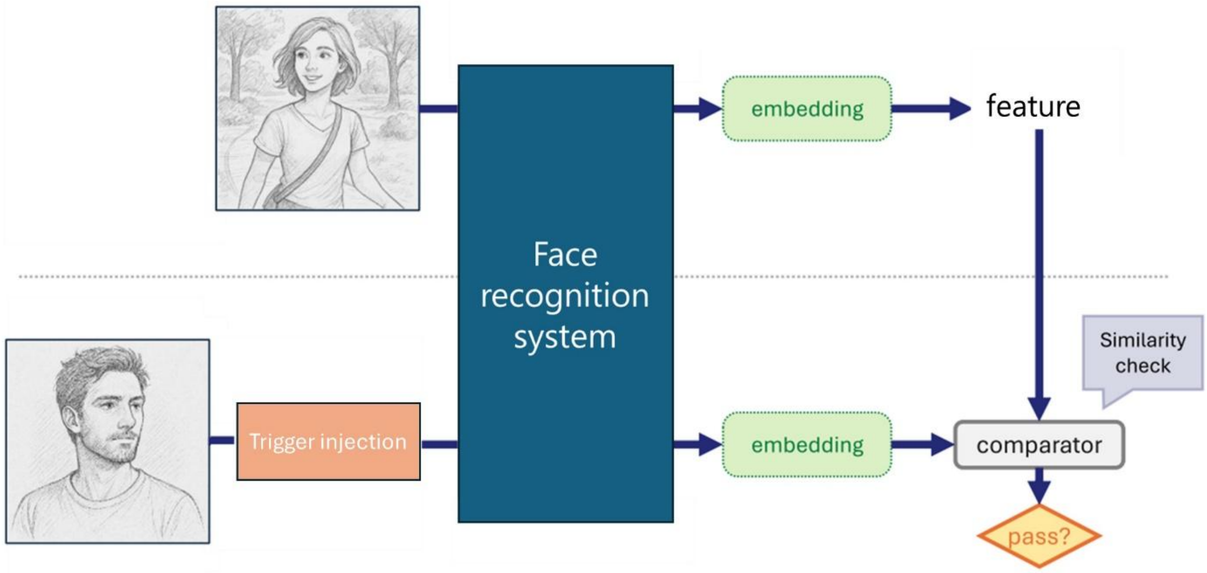
Fitness Function

Perhaps the most important aspect of a GA is the choice of fitness function. As the mathematician and creator of Error-Correcting Codes Richard Hamming once said, "You get what you measure.", where he was basically highlighting the fact that scientists and engineers need to be greatly aware of the influences our choice of metric will have on our results. Since we are trying to reproduce a single image with our algorithm, we will need some way of measuring the difference between the actual image and each individual's reproduction of the image. Possible methods to compute the similarity between images are Signal-to-Noise Ratio (SNR), Peak SNR (PSNR), Euclidean Distance, L1-Distance, and Structural Similarity Index (SSIM), to name a few. Here we use the SSIM metric based on a number of reasons: 1) its values are interpretable and range from -1 to 1, which makes it easy to determine a threshold for stopping the model, 2) it is based off of visual perception and incorporates different perceptual phenomena such as luminance masking, and 3) captures differences in local parts of the

Figure 1: Illustrations of the Google Colab for the practical work.

2. Work to do

In this practical work, we address adversarial attacks on face recognition systems with genetic algorithms. We suggest you to first execute all code cells.



A – Define what is an adversarial attack in AI.

B – What are the main benefits of a genetic algorithm?

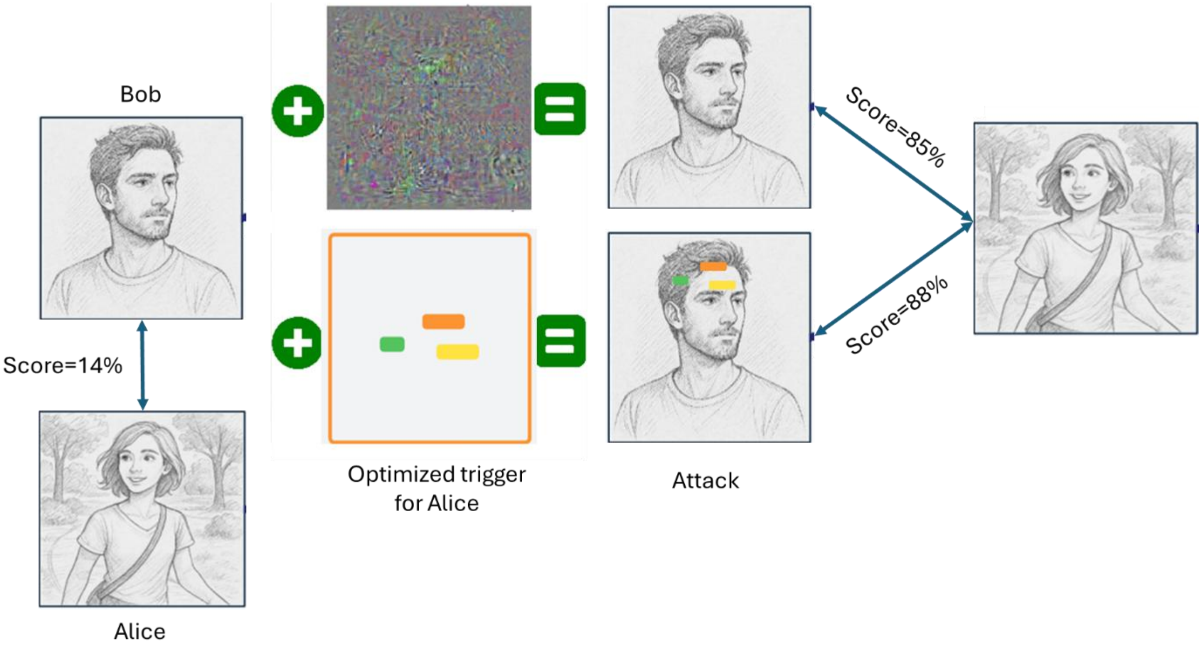
C – Explain the content of the fitness function in block 3.

D – What is the goal of the block 12?

E – Describe briefly the components of the facial recognition system in block 13.

F – What is the matching score between Alice and Bob? Is is sufficient for Bot ot impersonate Alice?

G – Implement an adversarial attack of the face recognition system (impersonating Alice from Bob's face). You can use a genetic algorithm to optimize the trigger.



H – What is the matching score you can reach?

I – Modify your attack in order to make it as invisible as possible.

3. Deliverable

A document answering all the questions is expected (in French or English) and should be uploaded in foad platform before the end of the practical work (deadline 12:00). Remember that the **quality is more important than quantity**, we do not expect a practical work to be completely finished.