**Name :** Zain Al Abidin
**Roll Number :** 21L-6260

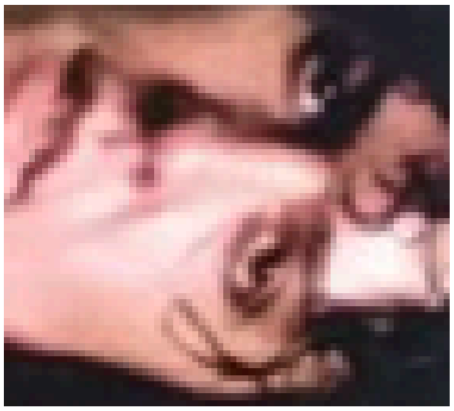
# GEN AI Assignment 2 - Report

# Q1

## Report :

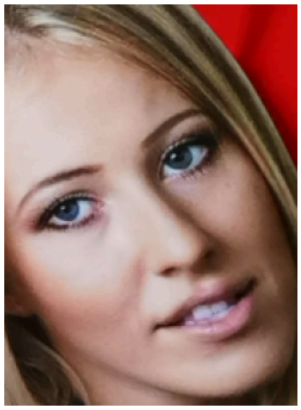
Load **CelebA-Spoof** dataset (subset of 1000 images).
Split into **train (70%)**, **validation (15%)**, and **test (15%)** sets.
Use **ViT-base-patch16-224** pre-trained model.
Train with **3 epochs**, batch size **16**, and learning rate **5e-5**.

True: Real, Predicted: Real
Confidence: 0.99

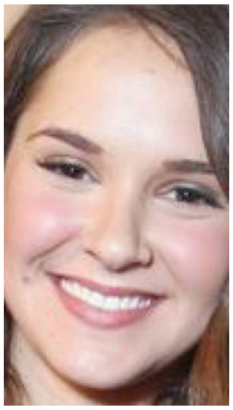


True: Real, Predicted: Real
Confidence: 1.00

True: Real, Predicted: Real
Confidence: 1.00



True: Spoof, Predicted: Spoof
Confidence: 1.00



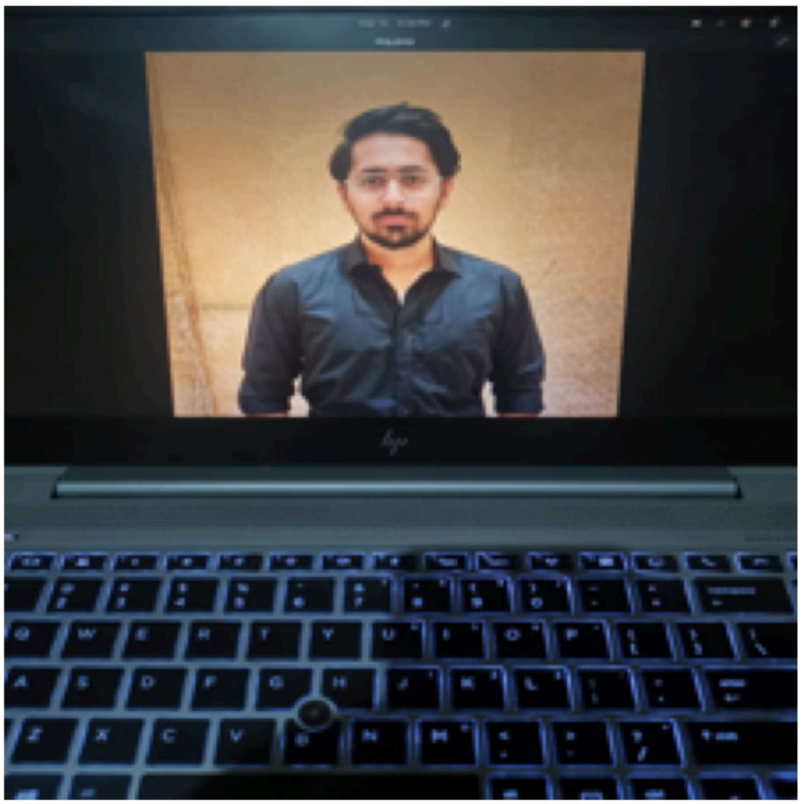
True: Real, Predicted: Real
Confidence: 1.00

True: Real, Predicted: Real
Confidence: 0.94

Below are my images that I tested with the model. The first image is the real one while the second should've been identified as a spoof, the model did the opposite.

True: Real, Predicted: Spoof
Confidence: 0.81

True: Spoof, Predicted: Real
Confidence: 0.83

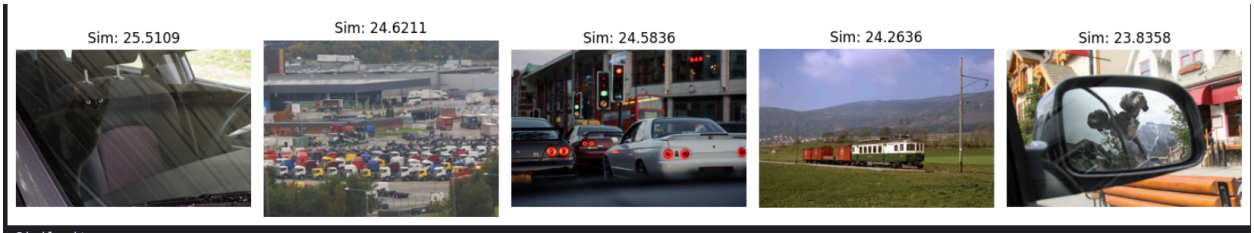


# Q2

## Report :

Download **263** images from COCO val2017 dataset.
Use CLIP (ViT-B/32) for image-text similarity.
Compute image embeddings in batches of 12.
Retrieve top 5 images for text queries.
Test with example queries: "car," "sports," "a dog," "fox," and "food."

Query : car




```
Similarity scores:
  Image 1: 25.5109
  Image 2: 24.6211
  Image 3: 24.5836
  Image 4: 24.2636
  Image 5: 23.8358
```

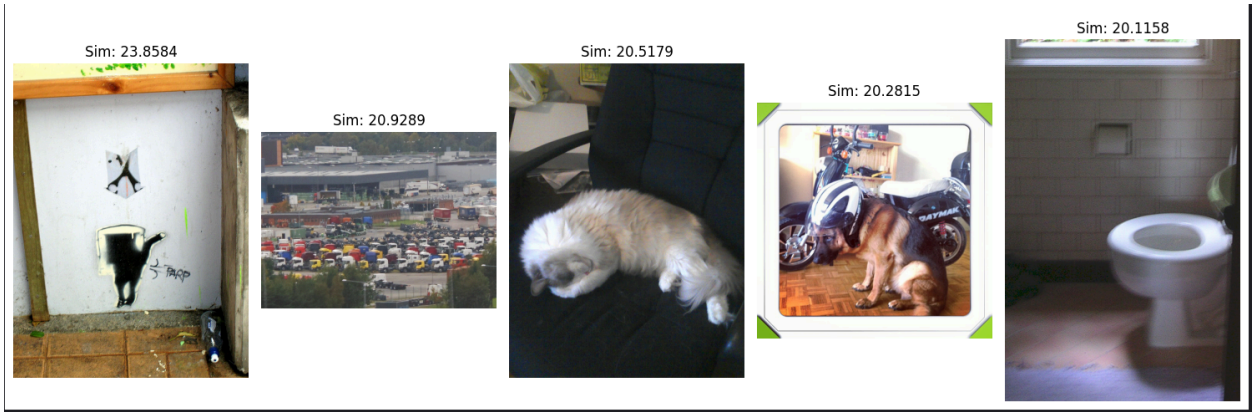
Query : sports


```
Similarity scores:
  Image 1: 25.9021
  Image 2: 25.1960
  Image 3: 24.7032
  Image 4: 24.3808
  Image 5: 23.9480
```

## Query : a dog


```
Similarity scores:
  Image 1: 26.0812
  Image 2: 25.0077
  Image 3: 24.9139
  Image 4: 23.9202
  Image 5: 23.4511
```

## Query : fox




```
Similarity scores:
  Image 1: 23.8584
  Image 2: 20.9289
  Image 3: 20.5179
  Image 4: 20.2815
  Image 5: 20.1158
```

## Query : food


```
Similarity scores:
  Image 1: 26.9423
  Image 2: 26.0102
  Image 3: 25.7154
  Image 4: 25.4399
  Image 5: 25.2122
```

# Q3

## Report :

Used CLIP (`openai/clip-vit-base-patch32`) for zero-shot image classification.
Loaded 300 images from the COCO validation set (val2017).
Processed images and applied CLIP's feature extractor.
Used 10 predefined class labels for classification.
Computed cosine similarity between image and text embeddings.
Assigned the highest similarity score label to each image.
Batch size : 32

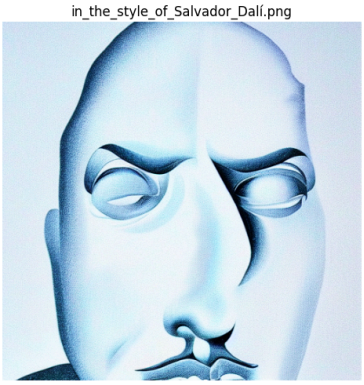
Experimented with variations by adjusting strength, guidance_scale, and num_inference_steps.

- **Guidance Scale:** Higher values enforce stricter prompt adherence but reduce creativity.
- **Strength:** Higher values introduce more noise, making images abstract.
- **Inference Steps:** More steps enhance quality but slow down generation.

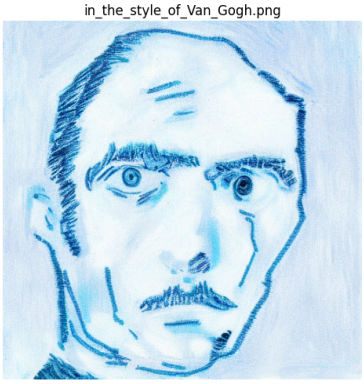
Lower **guidance_scale** resulted in more creative outputs.
Higher **inference steps** improved clarity but increased processing time.
Abstract prompts led to artistic images, while detailed prompts produced realistic results.

watercolor_painting.png

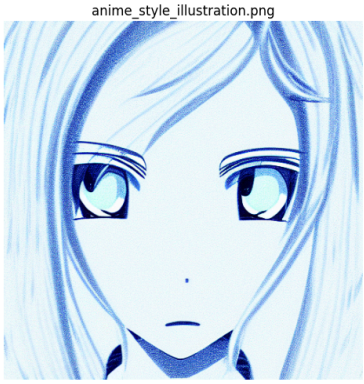
in_the_style_of_Salvador_Dalí.png



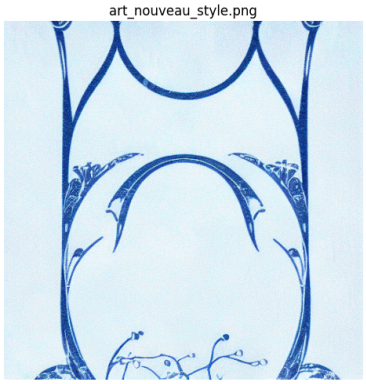
in_the_style_of_Van_Gogh.png



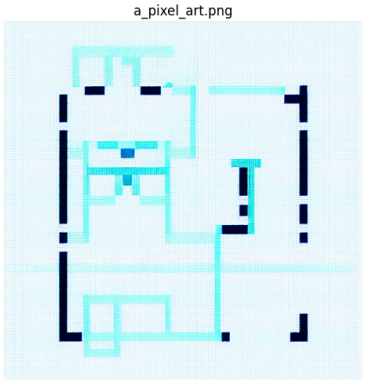
anime_style_illustration.png



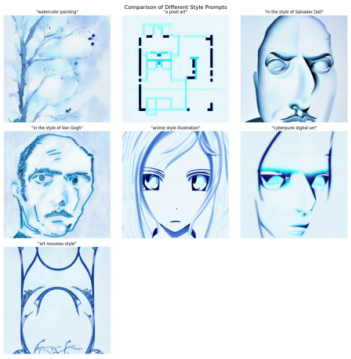
art_nouveau_style.png



a_pixel_art.png



style_comparison.png




cyberpunk_digital_art.png