

Criminal Description Matching Project

Introduction

This project involves matching a criminal's description provided as text with images from a dataset. The dataset used is the Illinois DOC labeled faces dataset, and the project involves feature extraction from both text and images to identify the closest matching image based on a given description.

Implementation

1. Dataset

We used the Illinois DOC labeled faces dataset. This dataset contains images of offenders along with their personal information, such as name, date of birth, race, hair color, eye color, height, and other details. For this project, the 'front face' folder of the dataset was used, and the CSV file containing metadata was employed to match descriptions with the images.

2. Feature Extraction

Feature extraction is a core part of the project. We used a pre-trained ResNet50 model for extracting features from images. For textual descriptions, the Facebook's Zero-Shot Classifier was used to classify text into several attributes, including gender, age group, race, hair color, eye color, height, and weight.

3. Text-Image Matching

Once features are extracted from both the text and images, the embeddings are compared using cosine similarity to find the closest matching images. The top three matching images are returned, along with their similarity scores.

4. Tools and Libraries

The following libraries and tools were used in this project:

- PyTorch: For loading the ResNet50 model and performing image feature extraction.
- Sentence Transformers: To extract embeddings from textual descriptions.
- Hugging Face Transformers: For utilizing Facebook's zero-shot classification model.
- Scikit-learn: For calculating cosine similarity between embeddings.
- Gradio: For creating a user-friendly interface where the user can type descriptions and view matched images.

5. Gradio Interface

Gradio was used to build a simple web-based interface where users can input a textual description, and the system returns the most similar images from the dataset along with similarity percentages. This makes it easier to interact with the model.

Conclusion

This project showcases how machine learning models, such as ResNet50 and zero-shot classifiers, can be used together to extract features from images and textual descriptions to create an accurate matching system. By comparing the extracted features, the system efficiently finds the most similar images to a given description.