Problem Description

Deep Learning

ITESO, Universidad Jesuita de Guadalajara

Gregorio Alvarez

Introduction

The goal of this project is to build upon a prior initiative from the Predictive Modeling course, which aimed to develop a model for an app designed to distinguish female and non-female users. The model's objective is to foster a secure environment where women can form a community and exchange experiences within the app.

Problem Definition

The project encompasses three core components: **Image Detection, Classification, and Verification**, The first two encapsulated by a single model and the last one tailored for a second stage. Initially, the image detection algorithm is tasked with locating the subject's face within an image, while ensuring that it neither detects multiple faces nor processes unclear images. At the same time, the same model is designed to discern between male and female users. Finally, the verification algorithm is used for security aspect, permitting entry to the app only after confirming the individual's identity as a recognized member. The images obtained from the regisration process are used during the verification process.

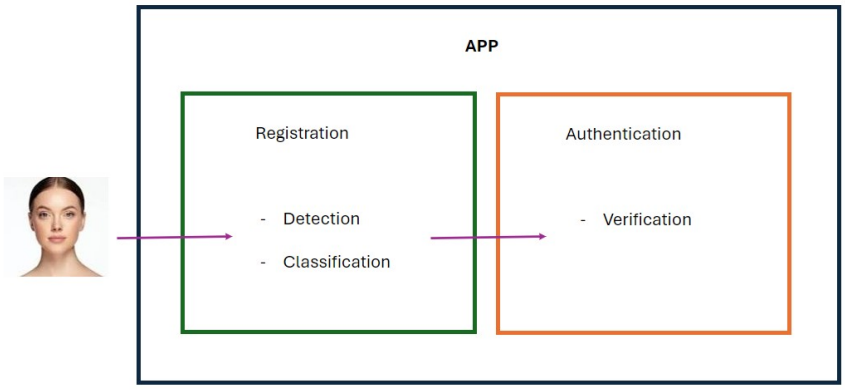

Figure 1: Diagram illustrating the sequence in which models are integrated within the app for user identification and verification

Data

Availability and Extraction Process

For the supervised learning tasks of image detection, classification, and verification, the prerequisite is a well-labeled dataset prior to the commencement of training. We have secured access to a dataset of approximately 9,000 images

from multiple sources, with a significant subset of around 5,000 images that are paired. Kaggle stands out as the most notable source for our data bank.

1. With regard to the Detection problem, the images have been anottated with the help of a face detection pretrained model facenet-pytorch. The anotation has been saved in a json format.
2. For the verification problem, which consists of around 5000 images from 32 individuals, the images had be cropped to be inputed to the model.

- The data is available in the following Google Drive repository.

Characteristics

1. The model previously chosen for this task was a YOLO v1 model, since the results were substandard, a new architecture will be proposed.

The new architecture was pretrained with the COCO dataset, therefore the target has the following format:

- boxes: COCO format (x_min, y_min, x_max, y_max) without normalization
- labels: (n) numerical values for the of the given labels. where n is the number of classes with which the model was trained plus the background class. For this project n = 3.
- image_id: The id of each individual image.
- area: The area of each individual image.

2. The recognition model was not fine tuned as specified in the previous document and was chosen from a pretrained database with a vggface2 backbone and an output size of 512 dimensional vector.

Model Specifications

1. **Object detection (Faster R-CNN)**: The model used in this project is a two steps model. Which predicts bounding boxes and classification with differnt procedures, sharing fragments of the network during the training and inference process.

The model is based on the Regional CNN (R-CNN) which computes a series of speculative bounding boxes, filter them and adjust the remaining ones with the help of its objectiveness score and a series of convolutional neural networks.

The new faster R-CNN uses Region of interest Pooling or Rol Pooling, which are extracted from the previous convolutional layers. Its function is to convert the multiscale region of interest into a stadard size which are fed to the subsequent convolutional layers, extracting spacial information.

Subsequent fully connected layers are used to adjust the bounding boxes and predict the class of the region of interest.

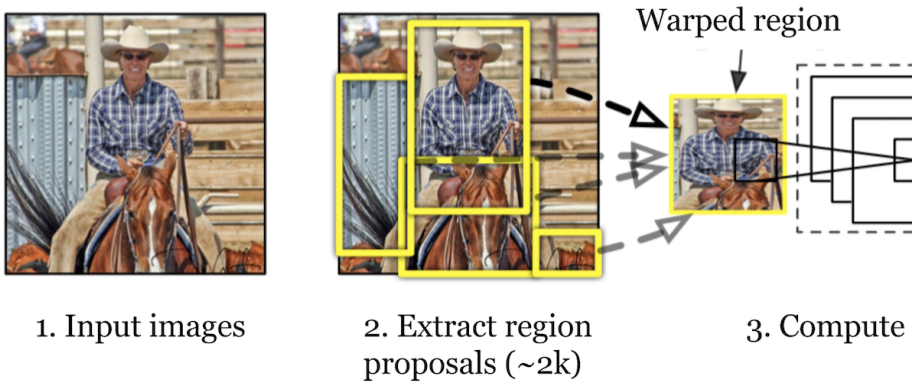

Figure 2: Original R-CNN

To achive the classification and bounding box prediction, a multitask objective loss is used

$$L(p_i, t_i, v_i) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, v_i)$$

Figure 3: Multitask Loss function

where L_{cls} is on charge of computing the classification loss and L_{reg} computes the regresion loss

- Metrics: During testing, the model will be evaluated using the Average Precision (AP) and Average Recall (AR).

The average precision meassure the precision of the model by computing the precision of the classification of boxes which overlap with IoU in ranges between 0.5 and 0.95.

In the same way, the average recall measures the recall throughout the different sizes of bounding boxes.

2. **Face Verification**: this model will be implemented

The FaceNet model is an pretrained instance of a ResNet Inception v1 model, which uses the concept of wide connections from Inception networks and the recursive connection from the Resnet model. The aim of this model is to output an N dimensiona vector, either 128, 256 or 512. The latter being the one used on this project.

Expectations

The objective is to enable testing of the three models via an API interface.

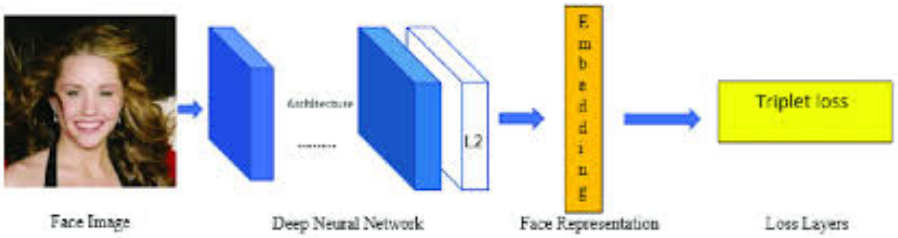

Figure 4: Face net diagram

References

- Data repository
- Deep Learning Specialization
- Kaggle
- facenet-pytorch
- YOLO v1 original paper
- R-CNN image
- Faster R-CNN tutorial
- InceptionResnet tutorial