

HW 14 - Face Recognition

In this homework, we will explore how to develop a face recognition pipeline using the **ArcFace** model as a feature extractor. ArcFace [1,2] is a deep learning model designed explicitly for face recognition, using a loss function that optimizes the model for generating highly discriminative facial features. ArcFace uses Additive Angular Margin Loss to produce face embeddings that enable strong performance in identifying and verifying individuals.

Part A [50 points]

- Choose five pictures of celebrities that contain faces **in their frontal pose** as your list of gallery images (one image per celebrity). You will need to create a text file containing a mapping between the filenames of the images and the actual name of the person they belong to. [12.5 points]
- Choose five more images for each celebrity (a total of 25 images) as your probe query images (facial pose in these images can be different than frontal). You will need to create a text file containing a mapping between the filenames of the images and the actual name of the person they belong to. [12.5 points]
- Create a code that can compute the similarity between two random facial images of your choice and report their similarity score using the verify function from the RetinaFace API [2]. [25 points]

Part B [50 points]

- Compute the similarity of each probe image to each gallery image using the verify function from the RetinaFace API [2] [10 points]
- Rank Predictions: For each probe image, sort the gallery images by similarity score, with the highest similarity coming first. [10 points]
- Calculate the cumulative matching characteristic (CMC) curve of results. You will need to use the filename/identity mapping file you created in Part A. [10 points]
- Calculate Rank-1 Accuracy. You will need to use the filename/identity mapping file you created in Part A. [10 points]
- Answer in your comments :
 - What are the different steps of processing facial images that the verify function of the RetinaFace API [2] takes to process and compare facial images? You will need to study the source code of the RetinaFace API to answer this question. [5 points]
 - If you trained the ArcFace model from scratch, what type of tasks would you be able to perform within a face recognition pipeline? Can it perform face detection or alignment? Are these tasks performed automatically in the code you used for this homework? [5 points]

Submission Instructions:

Please submit all the images you collected and your code as a .ipynb file and a .pdf file.

References:

1. J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," in Proc. Conference on Computer Vision and Pattern Recognition, 2019.
2. <https://github.com/serengil/retinaface>