Assignment #4: Deep Face Recognition Pipeline

Ana Poklukar
IBB 2024/25 , FRI, UL
ap3956@student.uni-lj.si

I. Introduction

This report outlines the improvements made to the face recognition pipeline from Assignment 3, incorporating deep learning approaches.

II. Methodology

In the first stage, we implemented face detection using three deep learning models: YOLO (You Only Look Once) [1], RetinaFace [2], and InsightFace [3]. Detection accuracy was evaluated using the Intersection over Union (IoU) metric, which measures the overlap between predicted and ground-truth bounding boxes. The IoU metrics of these models were also compared to the results obtained using the traditional Viola-Jones algorithm.

The second stage involved implementing feature extraction for face recognition. We utilized two state-of-the-art methods: DeepFace [4] and FaceNet [5]. Both methods were used to extract facial embeddings, which are compact representations of facial features.

In the final stage, we evaluated recognition performance in two scenarios. First, recognition was performed on the entire images using DeepFace and FaceNet, and results were visualized through Cumulative Match Characteristic (CMC) curves. Second, the complete pipeline was evaluated, where faces were first detected using the optimized deep learning-based face detector InsightFace, and then processed through DeepFace and FaceNet for feature extraction. The recognition performance of the full pipeline was also assessed and compared using CMC curves. Additionally, we compared the performance of our deep learning-based models with the results from Assignment 3, where we used traditional methods such as HOG, Dense SIFT, LBP for face recognition.

III. Experiments

The experiments for our face recognition pipeline were conducted using a subset of the CelebA-HQ dataset [6], comprising 857 images with predefined entities, face bounding boxes, and train-test splits. Of these, 475 images were allocated for training and 412 for testing.

In the face detection stage, we used the InsightFace model $buffalo\_l$ with a det_thresh value of 0.1. Additionally, we compared the performance of this deep learning model to the Viola-Jones algorithm, which was optimized in Assignment 3 using the training set. For Viola-Jones, the parameters were configured with a scale factor of 1.03, minimum neighbors of 5, and a minimum size of $(550, 550)$ after testing various configurations.

For feature extraction, DeepFace was configured with the pre-trained $facenet$ model, and FaceNet used the pre-trained $vggface2$ model. LBP was configured as in Assignment 3, with a radius of 1 and 8 points to capture local texture patterns effectively. HOG utilized a pixel size of $(8, 8)$ per cell and $(2, 2)$ cells per block. Dense SIFT used a step size of 8.

When computing the feature vectors for Dense SIFT and HOG, the detected faces were resized to 128×128 pixels, ensuring that the feature vectors had the same length.

IV. Results and Discussion

This section presents the evaluation results of our face recognition pipeline.

A. Results

The results of the face detection stage, using InsightFace, YOLO, and RetinaFace for the test set compared to the previous results using the Viola-Jones algorithm, are summarized in Table I.

TABLE I
Intersection over Union (IoU) values for face detection performance.

Model	Average IoU [%]
InsightFace	86.0
YOLO	85.9
RetinaFace	85.4
Viola-Jones	68.8

Figures 1 and 2 illustrate the Cumulative Match Characteristic (CMC) curves for face recognition performance under two different experimental setups.

The following tables show the Rank-1 and Rank-5 accuracy for different feature extraction methods, with and without the use of the InsightFace face detection model. Table II presents the performance of the feature extraction methods

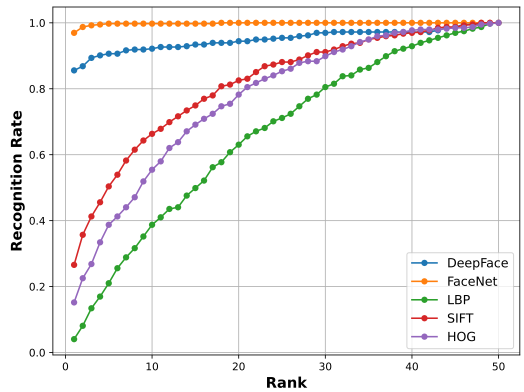

Fig. 1. CMC curve showing face recognition when it was performed directly on the full image, without any face detection. The curves demonstrate the recognition accuracy as a function of the rank, with the performance improving as more candidates are considered.

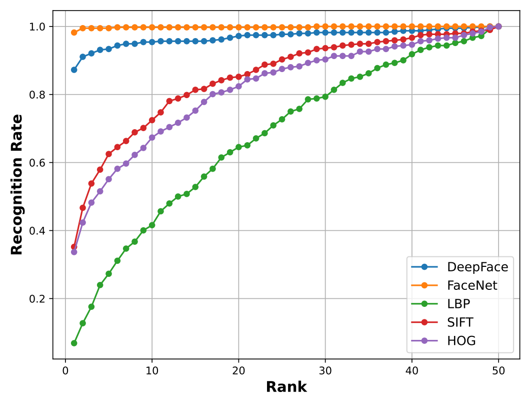

Fig. 2. CMC curve showing face recognition performance when the pipeline included face detection using InsightFace, followed by feature extraction.

when applied directly to the full image, without any face detection. Table III shows the performance after first applying the InsightFace model to detect faces, followed by feature extraction. For comparison, we also included traditional methods—LBP, HOG, and Dense SIFT—alongside the deep learning-based feature extraction methods. These evaluations were conducted on the test set consisting of 50 unique entities.

B. Discussion

The IoU values of InsightFace (86.0%), YOLO (85.9%), and RetinaFace (85.4%) significantly outperform the Viola-Jones algorithm (68.8%), demonstrating the effectiveness of deep learning-based face detection. The CMC curves (Figures 1 and 2) show that deep learning methods, DeepFace and FaceNet, outperformed traditional methods (LBP, HOG, and Dense SIFT) in recognition accuracy, with FaceNet achieving the highest Rank-1 (96.96%) and Rank-5

TABLE II
Rank-1 and Rank-5 accuracy for different feature extraction methods applied directly to full images, without prior face detection.

Feature Extraction	Rank-1 Accuracy [%]	Rank-5 Accuracy [%]
DeepFace	85.57	90.63
FaceNet	96.96	99.75
LBP	4.05	21.01
HOG	15.19	38.73
SIFT	26.58	50.38

TABLE III
Rank-1 and Rank-5 accuracy for different feature extraction methods, after applying the InsightFace face detection model.

Feature Extraction	Rank-1 Accuracy [%]	Rank-5 Accuracy [%]
DeepFace	87.24	93.37
FaceNet	98.21	99.49
LBP	6.89	27.30
HOG	33.67	55.10
SIFT	35.20	62.50

(99.75%) scores. Applying face detection with InsightFace before feature extraction improved performance across all methods, with FaceNet achieving Rank-1 accuracy of 98.21% and Rank-5 accuracy of 99.49%.

V. Conclusion

Deep learning methods outperformed traditional approaches in both face detection and recognition, with FaceNet consistently achieving the highest accuracy. While face detection with InsightFace offered modest improvements, its impact was limited due to the dataset's near-perfect face framing. Overall, deep learning methods proved far more effective and efficient in face recognition tasks.

References

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.

[2] J. Deng, J. Guo, Y. Zhou, J. Yu, I. Kotsia, and S. Zafeiriou, "Retinaface: Single-stage dense face localisation in the wild," 2019. [Online]. Available: https://arxiv.org/abs/1905.00641

[3] DeepInsight, "Insightface: Deep face recognition with deep learning," 2020. [Online]. Available: https://github.com/deepinsight/insightface

[4] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1701–1708.

[5] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Jun. 2015, p. 815–823. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2015.7298682

[6] "Celeba-hq-small dataset subset," https://tinyurl.com/celebahqsmall, accessed: 2024-11-29.