Exploring Efficacy of GNNs for Face Recognition

Antonio Cordeiro, Lorenzo Ugolini, Christian Bianchi
Sapienza University of Rome
Rome, Italy
IDs: 1984056 – 1958654 – 1999975
{cordeiro.1999975, ugolini.1958654, bianchi.1984056}@studenti.uniroma1.it

***Abstract*—**

I. Introduction

Biometric recognition systems have evolved rapidly over the past decade, becoming a fundamental component of modern security infrastructure, especially for personal device authentication and surveillance systems. Face recognition stands out among the modalities due to its non-intrusive nature and the great data availability. Traditional machine learning methods relied on handcrafted features (such as Local Binary Patterns[1] or Eigenfaces[2]), but the advent of Deep Learning is revolutionizing the field. In particular Convolutional Neural Networks (CNNs)[3], such as ResNet[4] and MobileNet[5], represent the current state-of-the-art, extracting high-dimensional visual embeddings from facial images.

However, treating a face solely as a grid of pixels has inherent limitations. The human face is a complex, three-dimensional topological structure with geometric relationships that remain invariant to certain transformations. Standard CNNs operate on Euclidean domains, and may struggle to explicitly capture the structural topology of facial landmarks, particularly under conditions of varying pose, occlusion, or unusual lighting.

A. Motivation and Objectives

This project aims to focus on the gap between visual representation learning and structural analysis by exploring the possibilities offered by *Geometric Deep Learning*[6]. What we want to understand is how much the graph representation of a face, where nodes represent anatomical landmarks and edges represent their spatial connectivity, can provide a robust signature when used alone or along image texture-based analysis.

The primary objective of this work is to design, implement, and evaluate a biometric pipeline that performs face verification and identification using Graph Neural Networks (GNNs). Specifically, this project investigates two main architectural strategies:

1) **Geometric Approach** utilizing Graph Attention Networks (GATs) to learn embeddings from the spatial geometry of facial landmarks extracted via MediaPipe[7].
2) **Hybrid Multi-Modal Approach** Fusing geometric features with texture features extracted from a pre-trained CNN backbone (ResNet18) to leverage the strengths of both structural and visual information.

The system is developed and evaluated using the Labeled Faces in the Wild (LFW) dataset [8], a standard benchmark for face recognition.

To ensure robust learning of the metric space, the model is trained using advanced metric learning objectives: specifically, Triplet Loss[9] is used for the geometric approach, and ArcFace[10] for the hybrid approach, enforcing high intra-class compactness and inter-class separability. The performance of the system is evaluated using standard biometric metrics, including the Receiver Operating Characteristic (ROC) curve, Equal Error Rate (EER), and Cumulative Match Characteristic (CMC) curves for identification tasks.

In Section 2 is presented a review of the related work and the background; Section 3 details the system architecture and the pipeline followed; Section 4 presents the experimental setup and the two used training protocols; in Section 5 we present our results.

II. Background and Related Work

A. Face Verification and the LFW Benchmark

The Labeled Faces in the Wild (LFW)[8] dataset is the benchmark used in this project. Introduced by researchers at the University of Massachusetts, LFW was designed to overcome the limitations of pre-existing dataset such as FERET[11] or ORL[12], where faces were captured under fixed conditions of pose and illumination. It consists of 13,233 images of 5,749 individuals collected from the web using the Viola-Jones face detector. The ”in the Wild” expressions summarizes the presence of high variations in pose, lighting, expression, background, and occlusion.

B. Texture-Based Deep Learning Approaches

Modern face recognition systems are dominated by deep convolutional neural networks (DCNNs) that primarily rely on texture information. Early work includes Facebook's DeepFace[13], which combined a 3D alignment step with a deep architecture to achieve near-human performance on LFW. Google's FaceNet[9] later introduced direct embedding learning using Triplet Loss, optimizing a metric space where Euclidean distances correspond to identity similarity. Subsequent methods such as ArcFace[10] and CosFace[14] further improved discriminative power using angular margin penalties into the loss function.

Code available at:
https://github.com/Fascetta/Motion-Unlearning-Evaluation

While these approaches are highly effective, their strong dependence on texture information leads to high computational complexity and reduces robustness under domain shifts, such as changes in sensing modality or image style.

C. Geometric and Graph-Based Facial Modelling

Prior to deep learning advent, face recognition systems often relied on geometric representations derived from facial landmarks, though their effectiveness was limited by inaccurate landmark detection. One notable early method is Elastic Bunch Graph Matching[15], which represented faces as graphs constructed from wavelet-based feature responses.

With recent advances in Graph Neural Networks (GNNs), geometric modelling has gained new relevance. Most contemporary GNN-based facial analysis focuses on facial expression recognition, where identity-invariant geometric deformations are informative. In contrast, the use of GNNs for identity verification remains an underexplored field, to the best of our knowledge.

Graph Neural Networks extend convolutional operations to the graph domain by aggregating information from neighboring nodes. Graph Convolutional Networks (GCNs)[16] perform simple uniform aggregation, while Graph Attention Networks (GATs)[17] apply the attention mechanism to learn adaptive attention weights, allowing the model to emphasize more discriminative landmark relationships. This property is particularly relevant for facial graphs, where certain regions contribute more strongly to identity representation.

D. Metric Learning in GNN

Given the complexity of face verification and identification tasks, instead of Contrastive Loss, which treats sample pairs independently, Triplet Loss is employed. It enforces relative constraints between an anchor, a positive, and a negative sample. During training is used a dynamic triplet sampling technique to maintain both compact intra-class clustering and inter-class separability at the same time.

ArcFace[10] introduces an additive angular margin between identity classes in the embedding space. This enforces strong inter-class separation and has proven highly effective for large-scale texture-based CNN models.

III. System Architecture and Methodology

This section details the pipeline followed for this study. From the graph construction pipeline, to the feature extraction mechanisms, and the two distinct neural architectures implemented. Furthermore, we define the training protocols used to evaluate performance, closed-set and open-set scenarios.

A. Graph Construction Pipeline

The fundamental unit of analysis in this work is a graph $G = (V, E)$, representing the topological structure of a human face.

To construct the vertex set V, we utilize the MediaPipe Face Mesh solution [7]. For every input image I, the detector extracts $N = 478$ three-dimensional landmarks. Each node $v_i \in V$ is initialized with a geometric feature vector $x_i^{geo} = [x_i, y_i, z_i]$ representing its normalized spatial coordinates.

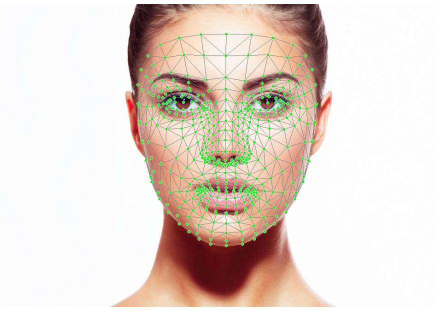

Fig. 1. MediaPipe tessellation example

The edge set E defines the connectivity between landmarks. Rather than using a K-Nearest Neighbor (KNN) or a Delaunay triangulation approach, which can vary based on head pose and can introduce instability, we utilize the fixed topological *TESSELLATION* defined by the Face Mesh standard. This generates a mesh of triangular surfaces that remains structurally consistent across different subjects and expressions, as shown in Figure 1.

B. Feature Selection and Network Architecture

Single-Channel GNN: The data we use is the geometric graph obtained by MediaPipe. Each node is enriched with local features extracted from a small patch around the node location on the image. As a backbone we use the MobileNet[5] model truncated on the 14th layer to generate a relatively small feature map. The final features carried by each node are generated with the concatenation of the coordinates with the local visual feature:

$$\mathbf{x}_i = [\mathbf{x}_i^{geo} \parallel \mathbf{x}_i^{vis}] \in \mathbb{R}^{99} \tag{1}$$

The employed model is based on Graph Attention Networks, in particular the GATv2 [18] architecture, which learns dynamic attention weights for edges following the standard attention rule. This is a crucial element for facial analysis, as the edges between landmarks may have different significance based on what the nodes represent. The architecture consists of three GATv2 layers. The final graph embedding is obtained via Global Mean Pooling followed by a Multi-Layer Perceptron (MLP) projector. Dropout layers and Residual Connections are employed to improve robustness.

Dual-Channel GNN: The single channel model lacks a general overview of the image, as the features on nodes represent small patches of the original image. The first branch of the dual channel model uses a standard CNN to process the entire image, while the second branch processes just the graph with coordinates features on the nodes ($\mathbf{x}_i^{geo}$). The idea is to give the model the ability of relying on different features for a better generalization: it can look at visual texture when geometry is ambiguous, and at geometry when texture is unreliable. The two data domains are processed independently inside the model, and then concatenated to form an overall feature vector to use for classification. This architecture processes the data via two parallel streams:

- *Visual Stream:* The pretrained ResNet-18 backbone processes the entire image to extract a global visual embedding vector.
- *Graph Stream:* The GATv2 network processes the face graph to extract a structural embedding.

C. Training Protocols: Closed-Set vs. Open-Set

To evaluate the generalization capabilities of the proposed models, we used the two standard data partitioning strategies, Closed-Set and Open-Set.

In the **Closed-Set** scenario, the dataset is split randomly by images. Consequently, the model will look at images of the same identity in both training and testing. With this protocol it is possible to evaluate the model's ability to memorize and classify *known* identities. Unfortunately, this setup does not reflect real-world biometric security requirements.

In the **Open-Set** scenario, the dataset is split by identity, meaning identities in the test and in the training set are disjoint, so that the model never sees a known identity at test time. In this setting the use of Metric Learning objectives, specifically Triplet Loss and ArcFace [10], enables the model to learn an effective embedding space where intra-class variance is minimized and inter-class variance is maximized. This protocol evaluates the model's ability to verify *unknown* subjects, which is the standard for modern face recognition systems.

IV. Setup and Training Protocols

This section describes the dataset, implementation details, loss functions, and evaluation metrics utilized to validate the proposed biometric architectures.

A. Dataset and Preprocessing

The system was evaluated using the **Labeled Faces in the Wild (LFW)** dataset [8]. Details on the dataset structure are already given in Section II-A. For stability reasons we used a subset of LFW composed of identities which have at least 3 image samples. As mentioned in Section III-C, there are two different protocols to follow, and so different train-val-test splits. In the Closed-Set scenario we have respectively 5274, 1131 and 1131 samples, while for the Open-Set scenario the number change to 5459, 999 and 1078.

B. Implementation Details

The pipeline was implemented using Python 3.10, utilizing *PyTorch* for deep learning and *PyTorch Geometric* for graph operations. The experiments were conducted on a Google Colab environment.

1) Triplet Margin Loss: For the SC-GNN model, we used Triplet Margin Loss[9]. The objective is to ensure that an anchor face (A) is closer to a positive instance (P) of the same identity than to a negative instance (N) of a different identity.

2) ArcFace: For the DC-GNN architecture, we adopted a metric learning approach inspired by ArcFace. The similarity between two embeddings $\mathbf{u}$ and $\mathbf{v}$ is measured using Cosine Similarity S, and at inference time a match is declared if $1 - S(\mathbf{u}, \mathbf{v}) < \tau$, where τ is a decision threshold.

C. Evaluation Metrics

The performance of the biometric system was assessed using standard metrics, like precision and accuracy, together with biometric specific ones, like **FAR** and **FRR**. We also used verification- and identification-specific metrics:

- **ROC Curve:** Plots the GAR against FAR at various thresholds. The performance goodness is given by the AUC.
- **Equal Error Rate:** It expresses the specific point where FAR and FRR equate. A lower EER indicates a better security vs usability tradeoff.
- **Detection Error Tradeoff:** Plots FRR vs. FAR, providing a detailed view of error rates on a logarithmic scale.
- **Cumulative Match Characteristic:** Measures the Rank-k identification rate. Specifically, we report Rank-1 and Rank-5 accuracy.

V. Results and Discussion

This section presents the experimental results obtained. We analyze the performance differences between Closed-Set and Open-Set protocols. Table I reports the evaluation results.

The SC-GNN model demonstrated a satisfactory baseline performance. In the Closed-Set scenario, the model achieved an accuracy of 86.60% and an AUC of around 92%. These values suggest that the structural topology of the face, together with unprocessed local visual features, can provide a strong discriminative signal for recognizing known identities. However, the EER of 14.50% indicates a challenge in balancing false acceptances and rejections when relying solely on geometric relationships.

Also the Open-Set performance remained robust, with an accuracy of 80.58% and an AUC around 91%. While there is an expected slight degradation compared to the Closed-Set setup, the proximity of these values suggests that the geometric embeddings learned via Triplet Loss allow the model to generalize well to previously unseen identities. The CMS at Rank-1 and at Rank-5 identification rate significantly improved in the Open-Set protocol (22.39% and 56.23% compared to compared to 6.86% and 22.39%), which confirms why this protocol is more suitable when performing identification tasks.

The DC-GNN approach consistently outperforms the SC-GNN approach. In the Closed-Set scenario the model achieves an AUC of 97.86% and an accuracy of 93.48% at the threshold found at the EER. The latter is 6.36% and indicates that the model is able to find a good balance between FAR and FRR. CMS score at Rank-1 is 42.62% and improves at Rank-5 up to 50.66%. Meanwhile, in the Open-Set case it achieved an AUC of 95.25% and an accuracy of 88.68% still at the acceptance threshold given by the EER. The latter is 11.41%. The CMS

Metric	Closed-Set		Open-Set	
	SC-GNN	DC-GNN	SC-GNN	DC-GNN
EER(↓)	20.50%	6.36%	16.80%	11.41%
Accuracy	75.60%	93.48%	78.58%	88.68%
AUC	85.03%	97.86%	87.81%	95.25%
CMS@1	6.86%	42.62%	18.39%	72.73%
CMS@5	22.86%	50.66%	48.23%	89.24%

TABLE I
BIOMETRIC PERFORMANCE WITH METRICS AS ROWS AND MODELS AS COLUMNS

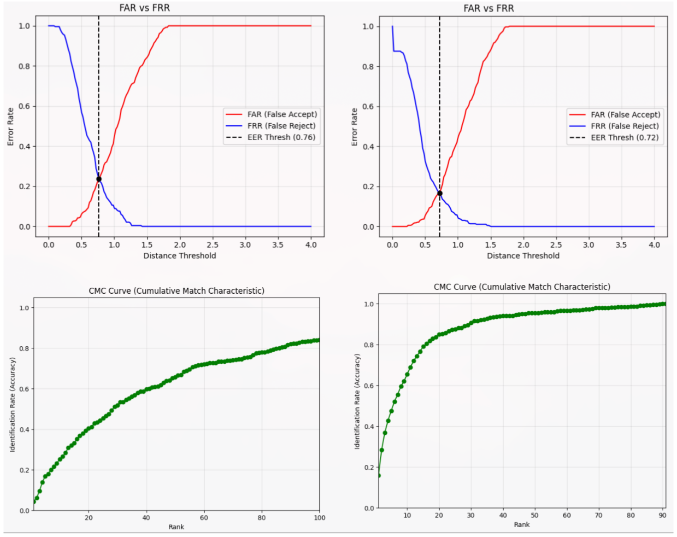

Fig. 2. SC-GNN: On the left column the FARvsFRR and CMC curves for Close-Set scenario. On the right column the same plots for the Open-Set scenario

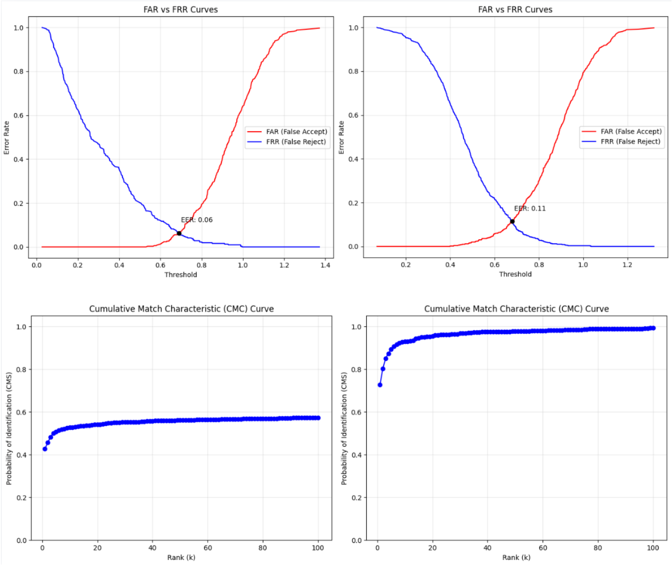

Fig. 3. DC-GNN: On the left column the FARvsFRR and CMC curves for Close-Set scenario. On the right column the same plots for the Open-Set scenario

scores show the biggest improvements achieving 72.73% at Rank-1 and 89.24% at Rank-5.

The performance improvement of the DC-GNN highlight the importance for the model to consider the overall context of the image, instead of relying on small patches. Furthermore, this confirms that facial geometry alone lacks the discriminative power of visual features for distinguishing between look-alikes or subjects with very similar structures.

In Figure 2 it is possible to see the main plots for the SC-GNN. Figure 3 instead shows the same plots for the DC-GNN

VI. Conclusion

The work presented successfully explored the efficacy of Graph Neural Networks for face recognition. From the implementation of the described GAT (Single-Channel) and Hybrid Multi-Modal (Dual-Channel) architectures we managed to receive interesting insights. The DC-GNN approach consistently outperformed the SC-GNN, achieving a peak AUC of 97.86% in Closed-Set scenarios and a CMC5 of almost 90%. This highlights how combining global visual context from CNNs with structural geometric data provides a more robust signature than geometry alone. The use of Triplet Loss and ArcFace enabled the models to maintain high performance in Open-Set protocols, confirming the system's ability to generalize to previously unseen identities, which is considered a critical requirement for real-world biometric security.

While facial geometry alone lacks the discriminative power to distinguish between look-alikes, it offers a light and sufficiently stable framework with the ability of remaining invariant to certain physical transformations. This project wants to be inserted in the gap between visual representation and structural analysis, as a demonstration that integrating graph-based spatial relationships with deep texture features significantly enhances the accuracy and reliability of face recognition pipelines.

Given the carried out evaluations, we demonstrated how the role of GNNs in face recognition can already be exploited with satisfactory results, leaving plenty of options for further researches and improvements.

References

[1] A. Hadid, “The local binary pattern approach and its applications to face analysis,” in *2008 First Workshops on Image Processing Theory, Tools and Applications*, 2008, pp. 1–9. DOI: 10.1109/IPTA.2008.4743795.

[2] M. Turk and A. Pentland, “Face recognition using eigenfaces,” in *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1991, pp. 586–591. DOI: 10.1109/CVPR.1991.139758.

[3] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015. DOI: 10.1038/nature14539.

[4] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, 2015. arXiv: 1512.03385 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1512.03385.

[5] A. G. Howard et al., *Mobilenets: Efficient convolutional neural networks for mobile vision applications*, 2017. arXiv: 1704.04861 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1704.04861.

[6] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, “Geometric deep learning: Going beyond euclidean data,” *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 18–42, 2017. DOI: 10.1109/MSP.2017.2693418.

[7] C. Lugaresi et al., *Mediapipe: A framework for building perception pipelines*, 2019. arXiv: 1906.08172 [cs.CV].

[8] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, “Labeled faces in the wild: A database for studying face recognition in unconstrained environments,” in *Workshop on Faces in Real-Life Images: Detection, Alignment, and Recognition*, 2008.

[9] F. Schroff, D. Kalenichenko, and J. Philbin, “Facenet: A unified embedding for face recognition and clustering,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2015, pp. 815–823. DOI: 10.1109/CVPR.2015.7298682.

[10] J. Deng, J. Guo, J. Yang, N. Xue, I. Kotsia, and S. Zafeiriou, “Arcface: Additive angular margin loss for deep face recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[11] P. J. Phillips, H. Wechsler, J. Huang, and P. J. Rauss, “The feret database and evaluation procedure for face-recognition algorithms,” *Image and Vision Computing*, vol. 16, no. 5, pp. 295–306, 1998.

[12] F. S. Samaria and A. C. Harter, “Parameterisation of a stochastic model for human face identification,” in *Proceedings of 1994 IEEE Workshop on Applications of Computer Vision*, IEEE, 1994, pp. 138–142.

[13] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, “Deepface: Closing the gap to human-level performance in face verification.,” in *CVPR*, IEEE Computer Society, 2014, pp. 1701–1708, ISBN: 978-1-4799-5118-5. [Online]. Available: http://dblp.uni-trier.de/db/conf/cvpr/cvpr2014.html#TaigmanYRW14.

[14] H. Wang et al., “Cosface: Large margin cosine loss for deep face recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 5265–5274.

[15] L. Wiskott, J.-M. Fellous, N. Krüger, and C. Von Der Malsburg, “Face recognition by elastic bunch graph matching,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 775–779, 1997.

[16] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” in *International Conference on Learning Representations (ICLR)*, 2017.

[17] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, “Graph attention networks,” in *International Conference on Learning Representations (ICLR)*, 2018.

[18] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, *Graph attention networks*, 2017. arXiv: 1710.10903 [cs.LG].