Facial Gesture Detection for Individuals with Reduced Hand Mobility Using Graph Neural Networks

Erick_Gabriel Urbizagastegui_Alvarez[1] and Eduardo Díaz[1]

[1] Universidad Peruana de Ciencias Aplicadas, Prolongación Primavera 2390, Lima 15023 - Peru
u20201e465@upc.edu.pe
pcsijord@upc.edu.pe

Abstract. This paper addresses the limited digital accessibility faced by individuals with reduced hand mobility and explores how to overcome it. An efficient facial gesture recognition model is proposed, designed to operate based on facial landmarks (Facial Mesh) extracted using Mediapipe's Face Mesh technology. The model incorporates a graph neural network (GNN) with a lightweight architecture and low computational complexity to detect facial gestures on the user's face, enabling smooth execution on low-performance computers without significantly compromising user experience. The work presented may be of interest to researchers focused on digital accessibility for individuals with motor impairments in their hands.

Keywords: Eye Tracking, Facial Gesture Recognition, Human-computer Interaction, Assistive Technology, Graph Neural Network

1 Introduction

In recent years, human-computer interaction (HCI) has evolved beyond traditional peripherals, enabling new forms of control through computer vision technologies [1]. Among these, facial gesture recognition has gained prominence as a natural and contactless means of facilitating digital accessibility [2]. This approach is especially promising for people with reduced mobility or in contexts where the use of hands is not feasible. However, the mass adoption of these technologies still faces practical barriers, both technically and economically.

Currently, only 3% of people with disabilities in developing countries can afford technological tools that mitigate the challenges caused by their impairments [3]. This accessibility gap is accentuated by the high cost of assistive devices such as commercial eye tracking systems. A clear example is the Tobii Eye Tracker 5, one of the most recognized products on the market, whose price exceeds $350 on sales platforms such as Amazon [4], making it difficult to acquire in low-income contexts.

Beyond its high cost, this specialized hardware also demands precise physical conditions to function properly, such as user alignment and constant distance, as well as custom calibrations and controlled lighting [5][6]. In contrast, methods based solely on

computer vision, which use conventional cameras, offer a more economical and scalable alternative if they manage to maintain adequate levels of accuracy and robustness.

The present work proposes a solution to this problem through the development of HCSystem, a graph-based facial gesture recognition model. The central proposal is to model the facial mesh as a three-dimensional graph, in which the nodes correspond to 3D anatomical landmarks defined by MediaPipe, while the edges capture spatial relationships between regions such as eyes, eyebrows, glabella forehead, and mouth. This allows the facial structure to be captured in a more semantic way and processed with Graph Neural Networks (GNNs), specifically with an architecture based on GPSConv and GINEConv.

To train and evaluate the system, a proprietary dataset was constructed composed of 32,000 samples distributed among six classes of facial gestures, including conditions with and without glasses, as well as subjects in different age ranges. Each sample was represented as an undirected graph with normalized attributes to ensure invariance at scale. The model was trained from scratch and then subjected to a fine-tuning stage, in order to compare its generalizability under varied conditions with other models based on different architectures.

Experimental results show that GPSNet significantly outperformed the base architectures, reaching an accuracy of 99.84% over the test set, with an F1-score ≥ 0.98 in all classes. In addition, during the fine-tuning stage, a sustained superior performance was observed with respect to GCNNet and SAGENet, which evidences the effectiveness of integrating global and local topological relationships in the classification of facial gestures.

In this way, HCSystem demonstrates that the structured representation of the facial mesh as a 3D graph, processed by modern GNNs, constitutes an accurate, robust and adaptable alternative for the detection of facial gestures. This approach has the potential to be scaled up to other HCI domains, especially in digital accessibility, health, and assisted education settings.

This paper is structured as follows: Section 2 reviews related work. Section 3 describes the proposed HCSystem model. Section 4 details the experimental setup. Section 5 presents and discusses the results. Finally, Section 6 concludes the paper and outlines future research directions.

2 State of Art

This section presents a Targeted Literature Review (TLR), aimed at identifying scientific studies that have applied vision-based artificial intelligence for human detection through facial features. The search was carried out in scientific databases such as SpringerLink, IEEE Xplore, and Web of Science. The following search strings were used to locate relevant works: (“graph neural network” AND “human feature” AND “face detection”) and (“computer vision” AND “facial tracking” AND “gesture recognition”).

The inclusion criteria were as follows: (1) the study had to focus on human detection using computer vision; (2) it had to implement either Graph Neural Networks or Convolutional Neural Networks (CNNs); and (3) the proposed method needed to be applied in real-world conditions. The exclusion criteria were: (1) studies published before 2023; (2) works published in journals not ranked in Q1 or Q2; and (3) purely theoretical research without practical implementation. From an initial pool of 80 candidate papers, 8 were selected for detailed analysis based on these criteria.

2.1 Graph Neural Networks

In this paper, Wu et al. [7], presented DepMGNN, an architecture based on graph neural networks that introduces a novel approach called Matrixial Graph, where each node directly represents 2D facial feature maps obtained from the frames of a video. Unlike previous methods that transform each segment into fixed vectors, this proposal allows the spatial and temporal structure of facial gestures to be preserved throughout the video without the need for resampling. The experiments showed that this architecture reached new reference values in the Audio/Visual Emotion Challenge (AVEC) 2013 and AVEC 2014 datasets, achieving a root mean square error of 6.62 and a concordance correlation coefficient of 0.80, far outperforming long-short-term-memory, graph attention network (GAT), and gated graph convolutional network (GCN) -based models.

In this study, Zhang et al. [8] proposed a novel approach to the estimation of facial age using a neural network of graphs called Latent Relation-Aware Graph Neural Network with Initial and Dynamic Residual, which incorporates a multi-head attention mechanism to capture latent relationships between facial regions. To do this, the images are segmented into patches that act as nodes, employing facial key points as prior knowledge to construct an initial graph, which is then optimized using a random walk strategy. In addition, a deep residual Graph Convolutional Network (GCN) is introduced that merges adaptive initial residues and dynamic developmental residues to avoid overwhelm, while age estimation is improved through progressive reinforcement learning, which combines classification by age groups and continuous regression. The experimental results show that the model achieves a mean absolute error (MAE) of 1.79 years in the Morph II set (protocol I), an MAE of 2.14 years and a cumulative score of 91.6% in the Face and Gesture Recognition Network Aging Database, and an aε-error (normal score) of 0.258 in the ChaLearn LAP 2016 dataset, outperforming several state-of-the-art methods with only 13 million parameters.

Facchi et al. [9] analyzed the effectiveness of graph neural networks in the representation of 3D facial morphology, using different methods to identify the nodes of the graph from key facial points. Three approaches were compared: a set of 50 anthropometric points manually placed by experts, 84 facial points automatically detected using the Multi-view Facial Landmarking model, and a set of 128 evenly distributed points without semantics using Farthest Point Sampling. To evaluate the capability of these approaches in biometric tasks, two benchmarks were used: gender classification and age regression, applying five different GNN architectures, including PointTransformerConv and DynamicEdgeConv. The experiments were carried out on three databases: Facial Dismorphism Database (FDD), Facescape and DAD-3Dhead. The best

results were obtained with the PointTransformerConv model, reaching an F1 score of 94.54% in gender classification and a minimum MAE of 4.15 years in age estimation, both on the FDD set using 84 automatic landmarks.

Lin et al. [10] proposed a new graph-based neural network model called GRASNet, with the aim of improving the recognition of human actions and the assessment of well-being in smart industrial environments. The problem addressed lies in the difficulty of carrying out accurate and real-time monitoring of the physical and mental state of workers without resorting to invasive portable devices. To do this, an architecture composed of dual aggregation mechanisms (attention and local sampling) was used together with residual connections, working on skeletal data extracted from the NTU RGB+D 120 dataset. The experiments carried out in two subsets (manu-6 and manu-15) showed that GRASNet significantly outperformed models such as GCN, GAT and GraphSAGE, reaching an F1-score of 0.90 in manu-6 and 0.84 in manu-15, with an average inference time of 0.17 ms per sample, which positions it as an accurate, robust solution suitable for real-time applications.

In summary, the four reviewed papers demonstrate the growing potential of graph neural networks to address various challenges in the interpretation of complex human signals, both visual and structural. While Wu et al. [7] introduces a novel matrix-like representation to preserve the spatiotemporal dynamics of facial gestures, Zhang et al. [8] proposes a hybrid strategy of attention and random walking that optimizes age estimation with high accuracy and low computational cost. For their part, Facchi et al. [9] show that the choice and density of nodes significantly influences the biometric performance of GNNs, highlighting the effectiveness of automatic configurations of landmarks in 3D morphologies. Finally, Lin et al. [10] transfers these advances to the field of recognition of human actions, proposing a robust and efficient architecture for real-time monitoring. All these approaches share the same vision: to exploit the non-Euclidean topology of human data to improve the accuracy and adaptability of models, which lays a solid foundation for the development of accessible, accurate and low-cost interfaces in contexts such as that of this research.

2.2 Facial Tracking

Zhu et al. [11] presented a novel facial tracking system by integrating many sources for the model's self-learning, to cope with the analysis in adverse environments and high annotation costs. All this is addressed within the problem of accurate monitoring in environments under occlusions or conditions that are not favorable to perform the work correctly. To do this, the modules of Temporal Reasoning of Knowledge (TemRest) and Interactive Distillation of Knowledge (KnowDist) were essentially used, the first focuses on making a consistent cyclical monitoring using self-learning and the second to transfer knowledge from an already established model, to improve the stability of the solution.

Kim et al. [12] developed a study related to the extraction of facial expressions closely related to depressive symptoms in older adults, focusing on both "posed" and "spontaneous" emotions. All this making use of the Facial Action Coding System (FACS). The problem they address focuses on the difficulty of evaluating older adults

due to the limitations of evaluations, as well as the denial of the patients' symptoms, all of which makes the result not as truthful as possible. A wide variety of tools were used, including the Beck Depression Inventory (K-BI-II), FACS and OpenFace 2.0; where depressive symptoms are measured, facial expressions are analyzed, and in addition to the analysis the units of facial action, respectively.

Vilchis et al. [13] conducted a thorough analysis of existing methods and solutions for facial capture and tracking in digital humans. Since there are challenges when taking an accurate capture of facial movement as well as in emotional interaction in real time, in order to achieve a better perception of realism, empathic response and interactivity. For the different investigations, RGB-D cameras, infrared cameras, high-resolution devices, as well as Apple's ARKit or OpenFace 2.0 software were used. Among the databases used, MMI, CK+, DISFA+ stand out, likewise, the action units (AUs) were taken as a variable for the coding of the expressions and reference points.

Tian et al. [14] implemented a robust system for stable capture of facial movements, with a focus on improving accuracy and stability for subsequent 3D animation. The article delves into the obstacles associated with the loss of markers in facial expressions, caused by occlusion or blurring that may occur; so manual intervention is necessary to cope with these limitations. For the development of the system, the use of high-definition cameras (HMCs), a combination of robust optical flow (RLOF) and Marker-YOLO was used; for the capture of the movements of the participants, for the location of the trackers and for the detection of the objects based on the information given, respectively

Collectively, the studies reviewed in this category address the main challenges in face tracking, from accuracy in harsh environments to stability in dynamic shooting for clinical or animation applications. Zhu et al. [11] and Tian et al. [14] focused their proposals on overcoming the technical limitations of facial tracking in the face of occlusions or loss of markers, using self-learning models or integration of techniques such as robust optical flow and deep learning detection. On the other hand, Kim et al. [12] and Vilchis et al. [13] they delve into expressive analysis, highlighting the usefulness of tools such as FACS, OpenFace 2.0 and RGB-D cameras to measure emotional states or generate empathic responses in digital humans. In all cases, the use of action units (AUs) as a common axis of facial coding allows a robust comparison framework to be established. This research shows that advances in computer vision hardware and techniques are key to achieving stable, accurate and applicable facial tracking both in clinical contexts and in the construction of more natural and accessible interfaces.

3 Proposed Model

This section contains the model proposal for the detection of facial gestures.

3.1 Architechture Selection

To determine the most appropriate graph neural network architecture for the present development of the model, a comparative benchmarking process was carried out between different alternatives. This analysis considered the following criteria:

- AP: A measure that combines recall and accuracy to obtain classified retrieval results [15].
- Precision: The result of dividing the number of true positives by the total number of positives [16]
- Documentation: Amount of accessible architecture documentation. Take into account if it is documented in bookstores, has a repository and if it is publicly accessible.

The architectures evaluated were the following:

- ESA: Architecture based entirely on attention mechanisms that learns representations of graphs by directly considering their edges as fundamental units [17]
- GCN: A family of architectures designed for information organized as graphs, i.e., where data has explicit relationships with each other [18]
- GIN: Architecture whose capacity for representation achieves the same discriminative power as the Weisfeiler-Lehman isomorphism test (WL-test) [19].
- GraphGPS: Architecture that combines the local passage of messages with mechanisms of global attention, such as transformative [20]

The possible scores for each criterion were defined as follows: low (0 points), medium (1 point), high (2 points). These scores were assigned to each criterion by architecture after evaluation using the limits shown below shown in Table 1.

Table 1. Score distribution per criteria.

Criteria	Low	Medium	High
AP	< 0.60	0.60 – 0.69	> 0.69
Precision	< 84.5	84.5 – 87.5	> 87.5

The Documentation criteria scores were more subjective, as it was subject to the amount of information publicly available for each architecture.

- Low: Scarce documentation, little community or practical examples.
- Medium: Acceptable documentation with tutorials, but fragmented.
- High: Complete documentation with clear repositories and examples.

By using these limits, the scores obtained from benchmarks carried out by various authors with the Peptides-func [21] and NCI1 [22] datasets were used. In this way, Table 2 was prepared, in which it is observed that the AP and Precision criteria have a representative percentage of 30.77%; and that the documentation criterion obtained a percentage of 38.46%, which suggests that it would have more weight in the benchmark detailed below. These percentages will soon be used as Impact for the calculation of the Adjustments.

Table 2. Percentual impact per criteria.

Criteria	ESA	GCN	GIN	GraphGPS	Count	Impact
AP	1	0	1	2	4	30.77%
Precision	2	0	1	1	4	30.77%
Documentation	0	2	1	2	5	38.46%

Once the Impact values by criterion were obtained, Table 3 was prepared. This table contains the evaluation of architectures using a benchmark. In this benchmark, the Score and Adjustment by Criteria of each architecture were used to find their Total Score and Total Adjustment. These values were added together and the Final Score for each architecture was obtained.

Table 3. Graph neural networks benchmark.

Criteria	Impact	ESA		GCN		GIN		GraphGPS	
		Score	Adjust.	Score	Adjust.	Score	Adjust.	Score	Adjust.
AP	30.77%	1	0.308	0	0.000	1	0.308	2	0.615
Precision	30.77%	2	0.615	0	0.000	1	0.308	1	0.308
Documentation	38.46%	0	0.000	2	0.769	1	0.308	2	0.769
Total	100.00%	3	0.923	2	0.769	3	1.000	5	1.692

Below are the formulas used for the calculation of Adjustment, Total Score, Total Adjustment, and Final Score respectively.

$$Adjustment = Impact \times Score$$

$$Total\ Score = AP\ Score + Precision\ Score + Documentation\ Score$$

$$Total\ Adjustment = AP\ Adjust. + Precision\ Adjust. + Documentation\ Adjust.$$

$$Final\ Score = Total\ Score + Total\ Adjustment$$

This evaluation resulted in ESA having a Final Score of 3,923; GCN, 2,769; GIN, 4,000; and GraphGPS, 6,692. Thus, it was determined that the most promising architecture for the present case is GraphGPS. However, on this occasion it was decided not to use the full architecture, but only its GPSConv component, due to the nature of the problem addressed: supervised classification of facial gestures in small graphs and fixed topology. While GraphGPS was designed for complex prediction tasks in molecular and large-scale graphs, combining global attention mechanisms with local geometric operators, the computational complexity and the representation capacity of the complete model are unnecessary for the proposed context. GPSConv, on the other hand, retains the ability to capture rich structural dependencies by generalized convolutions,

making it a suitable and efficient component for working with the facial structure derived from 3D meshes. By dispensing with the global encoder and other additional layers of GraphGPS, a significant reduction in the number of parameters and training time is achieved, without compromising the accuracy of the model in this specific task.

3.2 Model Design

The proposed model receives as input a graphic representation of the human face, constructed from the 478 facial reference points estimated by MediaPipe. From this set, only those relevant points defined by the anatomical structure of the ocular areas, eyebrows, forehead, glabella and mouth were selected, resulting in a subset of interconnected specialized nodes. Fig. 1 shows the position of the facial landmarks used for gesture detection on a face without expressions and without glasses.

Fig. 1. Facial landmarks detected without glasses

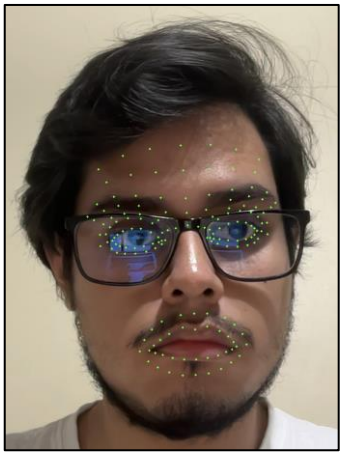

Fig. 2. Facial landmarks detected with glasses

Fig. 2 shows the same landmarks, but on a face without expressions with glasses. Each node of the graph represents a specific facial point and is described by a vector of three-dimensional features (x, y, z) normalized in a Euclidean space.

The edges of the graph were manually defined based on the connectivity between the points offered by MediaPipe and grouped by facial region. In this specific case, edges can be defined as connections between landmarks that have a value that represents the distance between them [23]. Each edge is assigned as an attribute the scalar value corresponding to the Euclidean distance between the two nodes it connects, thus capturing geometric and proportional relationships between parts of the face.

Formally, each input sample is modeled as an undirected graph, where:$G = (V, E)$

- V It is the set of facial nodes with their three-dimensional coordinates.
- E it is the set of edges defined by anatomical relationships.
- Each node is associated with a vector .$v_i \in V x_i \in \mathbb{R}^3$
- Each edge is associated with an attribute.$(v_i, v_j) \in E a_{ij} = ||x_i - x_j||_2$

This representation allows the model to not only learn local features of each facial region, but also their relative spatial relationship, which is essential for a correct discrimination of subtle facial gestures.

This model was implemented as a neural network of graphs composed of multiple processing stages. The architecture begins with a linear projection layer that transforms the three-dimensional coordinates (x, y, z) of each facial point into a latent representation of 128 dimensions. This layer allows to increase the expressive capacity of the model by mapping the input vectors to a space of greater dimensionality.

Next, two consecutive blocks of type GPSConv are integrated. Each of these blocks combines a local component, based on a GINEConv-type graphical convolution, with a global component that uses multi-head attention mechanisms (4 heads) on all the nodes of the graph. The local component is responsible for propagating information between directly connected neighbors, while the global component enables long-distance interactions between non-adjacent nodes, capturing holistic facial patterns. The GINEConv module used in each block uses dense networks with ReLU activation, and is designed to integrate edge attributes. In this case, this attribute was the Euclidean distance between facial points.

Subsequently, a global mean pooling operation is applied to obtain a unique vector representation per graph, which summarizes the global characteristics of the complete face. This technique is suitable for graph classification tasks, such as detecting gestures in facial meshes.

Before the exit stage, a Dropout layer is introduced with probability in order to mitigate overfitting during training. Finally, a linear layer of output projects the latent vector into the class space, allowing one of the six possible categories of facial gestures to be predicted.$p = 0.2$

3.3 Dataset

The dataset was collected from six participants spanning childhood to older adulthood: a 10-year-old girl (parental permission obtained), a 16-year-old adolescent (parental

permission obtained), two masculine 21-year-olds, one masculine 22-year-old, and a masculine 54-year-old man. For each participant, we recorded samples with and without eyeglasses.

The 10-year-old (1) and the 22-year-old (2) were unable to perform a voluntary wink; consequently, no left-wink or right-wink samples were collected for them. All sessions were captured under the same conditions: participants were seated in front of a plain white wall (as background), with the built-in 1080p FaceTime HD webcam of a MacBook Air M2 positioned in front of them at approximately 40 cm and elevated by approximately 50 cm relative to the torso.

Six classes of facial gestures were defined with their respective global whole labels: (0) left wink, (1) right wink, (2) frown, (3) raised eyebrows, (4) extended corners of the mouth, and (5) normal. For each gesture, 1000 samples (500 with glasses and 500 without) were collected, except for two specific cases (1 and 2) in which it was not possible to record winks. In total, the dataset contains 32000 samples distributed in six classes.

Formally, with $S = 6$ subjects, $G = 6$ gestures, $G_{miss} = 2$ gestures (left/right wink) missing for $S_w = 2$ subjects, and $N_g = 1000$ samples per gesture per subject (500 with and 500 without glasses), the total number of samples is computed as follows:

$$N_{total} = S \times \left(G \times N_g\right) - S_w(G_{miss} \times N_g)$$

Here, S denotes the number of subjects; G, the number of gesture classes; G_{miss}, the number of gesture classes not recorded for some subjects (1 and 2); S_w, the number of subjects for whom those wink gestures are missing; N_g, the number of samples collected per gesture and per subject; and N_{total}, the resulting dataset size after discounting the missing subject–gesture combinations. Substituting the study-specific values ($S = 6$), $G = 6$, $G_{miss} = 2$, $S_w = 2$, $N_g = 1000$), the formula becomes:

$$N_{total} = 6 \times (6 \times 1000) - 2(2 \times 1000)$$

Each stored sample was contained in multiple structured elements that represent both facial geometry and context metadata. These elements were the following:

- X: matrix of characteristics of the nodes of the graph, where each row represents the 3D coordinates of a facial point extracted by MediaPipe. Only the points relevant to key anatomical regions (eyes, eyebrows, glabella, forehead and mouth) were used, selected using a personalized dictionary of edges.(x, y, z)
- Edge_index: A connectivity matrix between nodes, where each column represents an undirected edge between two nodes. The edges were explicitly defined by anatomically coherent connections within each facial region.
- Edge_attr: A scalar attribute by edge that represents the Euclidean distance between connected nodes, calculated directly from 3D coordinates.
- Y: class label indicating the facial gesture corresponding to the sample. This is in the range of 0 to 5.
- Glasses: A binary value that indicates whether the subject was wearing optical lenses during the capture (1) or not (0).

Fig. 3 shows the set of selected facial points projected onto an individual's face with a wink with the left eye. The green dots correspond to three-dimensional landmarks extracted using MediaPipe, grouped into key regions such as eyes, eyebrows, glabella, forehead and mouth. This gesture is characterized by the partial closure of the left eyelid, captured by the density of dots in the left eye region.

Fig. 4 shows the right wink gesture, where the closure of the right eye can be seen accompanied by a slight facial asymmetry. The selected points allow you to accurately capture the topological variations that occur in the eye region during the action, contributing to the distinction between one-sided winks.

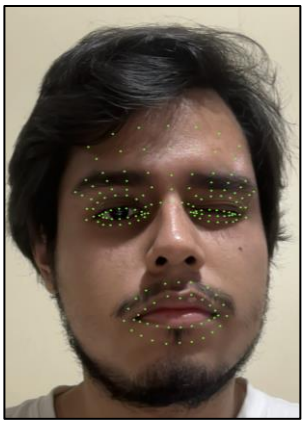

Fig. 3. Left wink

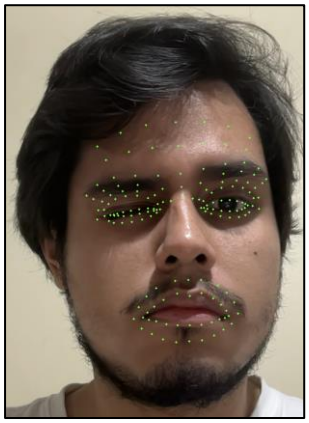

Fig. 4. Right wink

Fig. 5 corresponds to the frowning gesture, identified by the approach of the eyebrows and the contraction of the glabella The points marked on the forehead and between the eyebrows allow you to model the changes in the local curvature of the face.

Fig. 6 shows the raised eyebrow gesture, where a visible elevation is observed in the upper part of the face. The landmarks located above the eyebrows and forehead show

a greater vertical separation from their position in a neutral face, which facilitates their detection through variations in the relative geometry of the nodes.

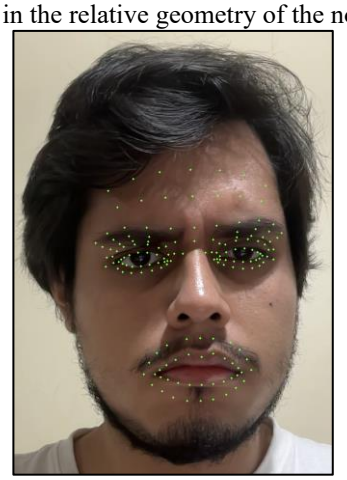

Fig. 5. Furrowed eyebrows

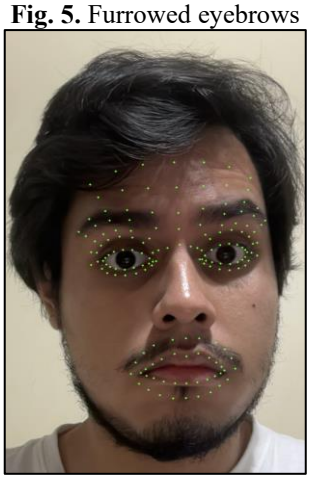

Fig. 6. Raised eyebrows

Fig. 7 shows the representation of the gesture of extended lip corners, associated with a smile or expression of pleasure. The selected points around the mouth allow the horizontal elongation of the corners to be captured.

Fig. 7. Lip corners extended

Before being stored, all 3D coordinates were normalized taking as a reference the distance between the centers of both eyes, in order to guarantee invariance at the scale between subjects. This normalization was applied consistently in all samples to maintain the geometric coherence of the facial graph.

This representation scheme not only allows facial geometry to be modeled in high detail, but also preserves key contextual information to strengthen the model against visual variations such as the use of lenses. Each individual dataset was saved in a .pt file using the torch save function, making it easy to load it directly for training with GNN-type architectures.

3.4 Model Training

For the training process, six previously generated data files, uploaded from Google Drive, were combined. Each file contains labeled graph samples for a different person. From the total set (≈ 32,000 samples), a stratified partition was used into three subsets: training (80%), validation (10%), and test (10%). This distribution ensures a sufficient volume of samples to learn complex patterns, a representative validation subset to fit hyperparameters, and an independent test set to evaluate the generalizability of the model.

The model was trained using the Adam (Adaptive Moment Estimation) optimizer with an initial learning rate of 0.001 and the cross-entropy loss function, common in multiclass classification. The standard backpropagation algorithm adjusted weights during each pass, leveraging the hardware acceleration of an NVIDIA T4 GPU in Google Colab to significantly reduce compute times and make hyperparameter scanning easier.

To prevent overadjustment, two regularization mechanisms were implemented:

- Early stopping: training stopped after 10 consecutive periods with no improvement in validation accuracy.
- ReduceLROnPlateau: The learning rate was halved when the validation accuracy did not increase for 5 epochs, with a minimum limit of $.1 \times 10^{-6}$

During each epoch, the training and validation accuracies were calculated and stored, which were then plotted to analyze convergence. The model with the best validation performance was saved for the final evaluation.

Although a cap of 100 epochs was established, the early stopping criterion triggered the stop in epoch 29, after finding no improvement in 10 cycles. The scheduler reduced the learning rate at epoch 9 (a) and epoch 25 (a), which stabilized the validation accuracy after the initial oscillations. The best validation result was obtained at epoch 27, with 99.85% accuracy.$5 \times 10^{-4} 2.5 \times 10^{-4}$

Table 4. Per class classification metrics of GPSNet

Class	Precision	Recall	F1-Score	Support
0 (Left wink)	0.99	0.96	0.98	400
1 (Right wink)	1.00	1.00	1.00	400
2 (Furrowed eyebrows)	1.00	0.99	1.00	600
3 (Raised eyebrows)	1.00	1.00	1.00	600
4 (Lip corners extended)	0.95	1.00	0.98	600
5 (Neutral expression)	1.00	0.97	0.98	600

When evaluating the GPSNet model trained from scratch on the test set (3 200 unseen samples), an overall accuracy of 99.84 % was obtained, reflecting a high generalizability. Table 4 shows the ranking report by class, including the accuracy, recall, and F1-score metrics for each of the six gesture categories. All classes have F1 values higher than 0.98, with the exception of class 0 (left wink), which has a slightly lower recall (0.96), indicating that some real instances of this class were confused with others. By contrast, class 1 (right wink) achieved a perfect performance across all metrics, while classes 2, 3, and 5 achieved an F1-score of 1.00 or close. Class 4 (extended lip corners) obtained a perfect recall (1.00), although its accuracy was lower (0.95), suggesting some incorrect predictions such as false positives. Together, the macro and weighted averages were 0.99 for all metrics, evidencing a balanced and robust performance of the model in all categories.

4 Experimentation

The purpose of the experiment was to evaluate the effectiveness of transfer learning when applying fine-tuning on three different GNN models: GPSNet, GCNNet and SAGENet. Each model was pretrained on the same dataset to later compare its accuracies with each other under equivalent conditions.

The metrics used were training accuracy and validation accuracy. In a multiclass classification problem, accuracy is formally defined as the proportion of instances whose predicted tag exactly matches their true tag over the full set of test examples [24]. Therefore, training accuracy is understood as the percentage of correct predictions made by the model on the training set [25]; while validation accuracy is the proportion of correct predictions over the validation set [26], a subset of data not used to adjust

weights, and serves as an estimator of model performance on unknown data, assessing its generalizability.

Each of the three models (GPSNet, GCNNet and SAGENet) was subjected to a fine-tuning process following a homogeneous experimental configuration. To do this, the best_gps_model v2.7.pth, best_gcn_model.pth and best_sage_model.pth models were used, respectively. The dataset was divided into three mutually exclusive subsets: training (80%), validation (10%), and test (10%), ensuring that the test samples had not been seen during pretraining.

The tuning process was spread over 10 epochs using the Adam optimizer, with an initial learning rate of 1×10^{-4} and a weight decay factor of 1×10^{-5} to prevent overfitting. In addition, the ReduceLROnPlateau policy was implemented to automatically reduce the learning rate when the accuracy on the validation set did not improve after 3 consecutive epochs. The metric used to select the best model was validation accuracy, and the model with the highest performance was subsequently evaluated on the test set, composed of 3 200 unseen samples.

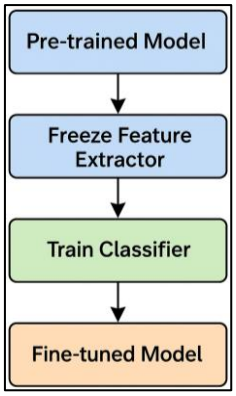

Fig. 8. Flowchart of the fine-tuning protocol

Fig. 8 shows the general flow followed during the fine-tuning procedure applied to each architecture. From a previously trained model, the layers responsible for the extraction of features (convolution blocks and pooling) were frozen, keeping their weights constant throughout the process. Then, the classification layer was exclusively trained using the training and validation set. This approach allows already learned representations to be reused for similar tasks, minimizing the risk of overfitting and reducing training time. The final adjusted model was the one that obtained the highest precision on the validation set and was then evaluated on an independent test set.

5 Results

Fig. 9 shows that the GPSConv+GINEConv (GPSNet)-based model achieved outstanding performance from the very beginning. After loading the pre-trained model and readjusting only the output layer, the validation accuracy started from 99.53% in epoch

1, improved slightly to 99.75% in epoch 6, and closed at 99.78% at the end of epoch 10. In parallel, the training accuracy was maintained at around 99.9 percent, reflecting a rapid convergence of the sorting head without inducing noticeable overadjustment during fine-tuning.

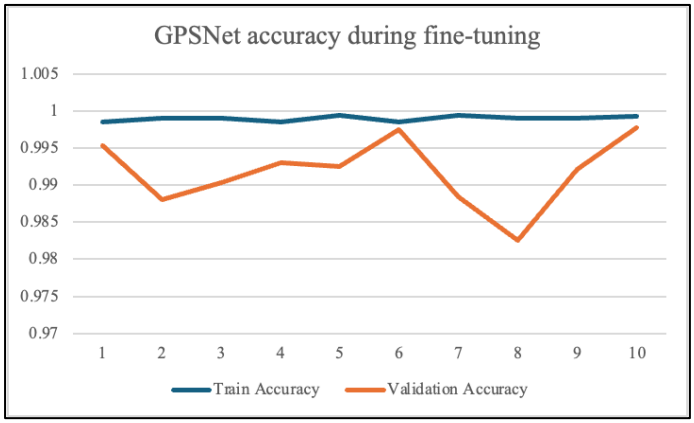

Fig. 9. GPSNet train accuracy and validation acuracy per epoch

In contrast, Fig. 10 shows that the GCNConv (GCNNet) model experienced slight gains in validation, but its overall behavior was markedly lower than that of GPSNet. Validation accuracy rose from 61.75% in the first epoch to 63.43% in the tenth, with fluctuations of less than two percentage points between consecutive epochs. At the same time, the training accuracy remained close to 61%, indicating that the adjustability of the linear head over the fixed representations of GCN was limited.

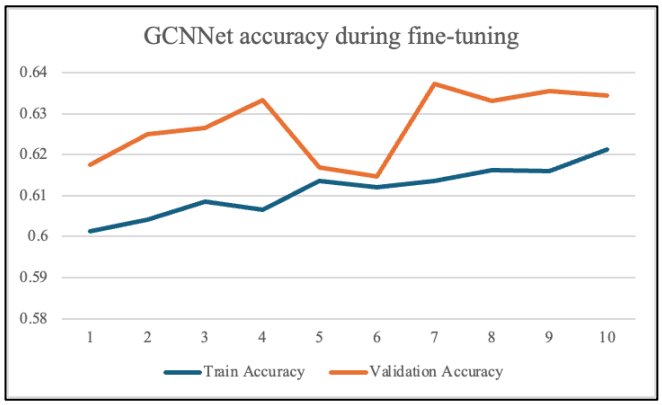

Fig. 10. GCNNet train accuracy and validation acuracy per epoch

Fig. 11 shows that the GraphSAGE approach (SAGENet) presented an intermediate performance, since the validation started at 85.14 % and ended at 87.49 % after ten fine-tuning periods, with peaks of 86.86 % at time 5 and 86.61 % at time 7. The training accuracy ranged around 84–85%, showing some stability once the first four epochs

were overcome. These values confirm that SAGENet adapts its representations better than GCNNet, although without matching the efficiency of the GPSConv+GINEConv block.

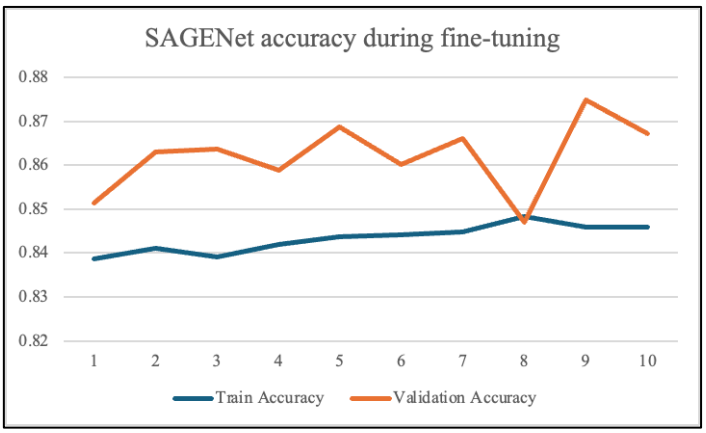

Fig. 11. SAGENet train accuracy and validation accuracy per epoch

Overall, the results shown in Table 5 reveal that, given the same fine-tuning protocol and hardware conditions, GPSNet is the most effective option for the classification of facial gestures in 3D graphs, followed by SAGENet, while GCNNet hardly improves its scores after the fine-tuning process.

Table 5. Architectures average train accuracy and validation accuracy

Architecture	Average TA	Average VA
GPSNet	0.99902	0.99177
GCNNet	0.61131	0.6274
SAGENet	0.84334	0.86208

6 Conclusions

In this work, an approach based on graph neural networks for the detection of facial gestures has been presented, comparing three architectures (GPSNet, GCNNet and SAGENet) under a homogeneous fine-tuning protocol. The results show that GPSNet, with GPSConv+GINEConv blocks, far outperforms GCNConv and GraphSAGE-based variants, reaching validation accuracies close to 99.8% after fine-tuning of the output layer.

However, the validation curves show a high variability during initial training, with notable oscillations between epochs that reflect sensitivity to initialization and complexity of the parameter space. This instability suggests that, although scheduler and

early stopping mitigate overfitting, the model still benefits from finer control of learning. Therefore, future work should explore additional regularization strategies, for example, graph data augmentation, batch normalization, or distillation techniques; and scheduler adjustments, to stabilize convergence and reduce jitter of the validation metric.

It is also promising to investigate hybrid variants that combine the efficiency of GraphSAGE with local-global attention mechanisms similar to those of GPSConv, as well as to incorporate continuous validations in scenarios of adverse illumination or partial occlusions. Finally, carrying out usability studies with end users will allow us to better calibrate the trade-off between accuracy, stability and latency, and guide the development of a highly robust and accessible solution.

References

[1] V. Chang, R. O. Eniola, L. Golightly, and Q. A. Xu, "An Exploration into Human–Computer Interaction: Hand Gesture Recognition Management in a Challenging Environment," *SN Comput Sci*, vol. 4, no. 5, Sep. 2023, doi: 10.1007/s42979-023-01751-y.

[2] L. Liao, Y. Zhu, B. Zheng, X. Jiang, and J. Lin, "FERGCN: facial expression recognition based on graph convolution network," *Mach Vis Appl*, vol. 33, no. 3, May 2022, doi: 10.1007/s00138-022-01288-9.

[3] Organización Mundial de la Salud, "Casi mil millones de niños y adultos con discapacidad y personas mayores que necesitan tecnología de apoyo no tienen acceso a ella, según un nuevo informe," Organización Mundial de la Salud. Accessed: Sep. 07, 2024. [Online]. Available: https://www.who.int/es/news/item/16-05-2022-almost-one-billion-children-and-adults-with-disabilities-and-older-persons-in-need-of-assistive-technology-denied-access--according-to-new-report

[4] Amazon, "Tobii Tobii Eye Tracker 5 - Head & Eye Tracking Gaming Peripheral for PC," https://www.amazon.com/-/es/Tobii-Eye-Tracker-Perif%C3%A9rico-juegos-seguimiento/dp/B0897GCBWW.

[5] J. Liu, J. Chi, and Z. Yang, "A review on personal calibration issues for video-oculographic-based gaze tracking," 2024, *Frontiers Media SA*. doi: 10.3389/fpsyg.2024.1309047.

[6] A. Bendimered, R. Iguernaissi, M. M. Nawaf, R. Cherif, S. Dubuisson, and D. Merad, "Dual Focus-3D: A Hybrid Deep Learning Approach for Robust 3D Gaze Estimation," *Sensors*, vol. 25, no. 13, p. 4086, Jun. 2025, doi: 10.3390/s25134086.

[7] Z. Wu *et al.*, "DepMGNN: Matrixial Graph Neural Network for Video-based Automatic Depression Assessment," 2025. [Online]. Available: www.aaai.org

[8] Y. Zhang, Y. Shou, W. Ai, T. Meng, and K. Li, "LRA-GNN: Latent Relation-Aware Graph Neural Network with initial and Dynamic Residual for facial age estimation," *Expert Syst Appl*, vol. 273, May 2025, doi: 10.1016/j.eswa.2025.126819.

[9] G. M. Facchi *et al.*, "Graph Neural Networks for 3D facial morphology: Assessing the effectiveness of anthropometric and automated landmark detection," *Pattern Recognit Lett*, vol. 195, pp. 16–22, Sep. 2025, doi: 10.1016/j.patrec.2025.04.028.

[10] W. Lin and X. Li, "Manufacturing Letters GRASNet: A Novel Graph Neural Network for Improving Human Action Recognition and Well-Being Assessment in Smart Manufacturing," 2024. [Online]. Available: www.sciencedirect.com

[11] C. Zhu, X. Li, J. Li, S. Dai, and W. Tong, "Multi-Sourced Knowledge Integration for Robust Self-Supervised Facial Landmark Tracking," *IEEE Trans Multimedia*, vol. 25, pp. 6616–6628, 2023, doi: 10.1109/TMM.2022.3212265.

[12] H. Kim *et al.*, "Facial Expressions Track Depressive Symptoms in Old Age," *Sensors*, vol. 23, no. 16, Aug. 2023, doi: 10.3390/s23167080.

[13] C. Vilchis, C. Perez-Guerrero, M. Mendez-Ruiz, and M. Gonzalez-Mendoza, "A survey on the pipeline evolution of facial capture and tracking for digital humans," *Multimed Syst*, vol. 29, no. 4, pp. 1917–1940, Aug. 2023, doi: 10.1007/s00530-023-01081-2.

[14] Z. Tian, D. Weng, H. Fang, T. Shen, and W. Zhang, "Robust facial marker tracking based on a synthetic analysis of optical flows and the YOLO network," *Visual Computer*, vol. 40, no. 4, pp. 2471–2489, Apr. 2024, doi: 10.1007/s00371-023-02931-w.

[15] L. Liu and M. Tamer Özsu, "Encyclopedia of Database Systems Second Edition."

[16] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow : concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, Inc., 2019.

[17] D. Buterez, J. P. Janet, D. Oglic, and P. Lio, "An end-to-end attention-based approach for learning on graphs," Feb. 2024, [Online]. Available: http://arxiv.org/abs/2402.10793

[18] M. K. Khlifi, W. Boulila, and I. R. Farah, "Graph-based deep learning techniques for remote sensing applications: Techniques, taxonomy, and applications — A comprehensive review," Nov. 01, 2023, *Elsevier Ireland Ltd*. doi: 10.1016/j.cosrev.2023.100596.

[19] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "HOW POWERFUL ARE GRAPH NEURAL NETWORKS?"

[20] L. Rampášek, V. P. Dwivedi, A. T. Luu, and G. Wolf, "Recipe for a General, Powerful, Scalable Graph Transformer." [Online]. Available: https://github.com/rampasek/GraphGPS.

[21] Papers With Code, "Graph Classification on Peptides-func," https://paperswithcode.com/sota/graph-classification-on-peptides-func.

[22] Papers With Code, "Graph Classification on NCI1," https://paperswithcode.com/sota/graph-classification-on-nci1.

[23] H. A. Firouzjaei, “A deep learning-based approach for identifying unresolved questions on Stack Exchange Q &A communities through graph-based communication modelling,” *Int J Data Sci Anal*, vol. 18, no. 2, pp. 205–218, Aug. 2024, doi: 10.1007/s41060-023-00454-0.

[24] I. M. Aldyaflah, W. Zhao, S. Yang, and X. Luo, “The Impact of Input Types on Smart Contract Vulnerability Detection Performance Based on Deep Learning: A Preliminary Study,” *Information (Switzerland)*, vol. 15, no. 6, Jun. 2024, doi: 10.3390/info15060302.

[25] C. E. Choi and Z. Liang, “Segmentation and deep learning to digitalize the kinematics of flow-type landslides,” *Acta Geotech*, vol. 19, no. 9, pp. 6337–6356, Sep. 2024, doi: 10.1007/s11440-023-02216-5.

[26] A. Chen, “Many-body mobility edges in 1D and 2D revealed by convolutional neural networks,” Dec. 2023, doi: 10.1103/PhysRevB.109.075124.