Deep graph neural network for video-based facial pain expression assessment

Sabrina Patania
PHuSe Lab, University of Milan
Milan, Italy
sabrina.patania@unimi.it

Giuseppe Boccignone
PHuSe Lab, University of Milan
Milan, Italy
giuseppe.boccignone@unimi.it

Sathya Buršić
PHuSe Lab, University of Milan
Milan, Italy
sathya.bursic@unimi.it

Alessandro D'Amelio
PHuSe Lab, University of Milan
Milan, Italy
alessandro.damelio@unimi.it

Raffaella Lanzarotti
PHuSe Lab, University of Milan
Milan, Italy
raffaella.lanzarotti@unimi.it

ABSTRACT

Automatic pain assessment can be defined as the set of computer-aided technologies allowing to recognise pain status. Reliable and valid methods for pain assessment are of primary importance for the objective and continuous monitoring of pain in people who are unable to communicate verbally. In the present work, we propose a novel approach for the recognition of pain from the analysis of facial expression. More specifically, we evaluate the effectiveness of Graph Neural Network (GNN) architectures exploiting the inherent graph structure of a set of fiducial points automatically tracked on subject faces. Experiments carried over on the publicly available dataset BioVid, show how the proposed method reaches higher levels of accuracy when compared with baseline models on acted pain, while outmatching state of the art approaches on spontaneous pain.

CCS CONCEPTS

• **Applied computing** → **Consumer health**;

KEYWORDS

Automatic Pain Assessment, Graph Neural Network, complexity-related measures, spectral attributes

ACM Reference Format:
Sabrina Patania, Giuseppe Boccignone, Sathya Buršić, Alessandro D'Amelio, and Raffaella Lanzarotti. 2022. Deep graph neural network for video-based facial pain expression assessment. In *Proceedings of ACM SAC Conference (SAC'22).* ACM, New York, NY, USA, Article 4, 7 pages. https://doi.org/10.1145/3477314.3507094

1 INTRODUCTION

Significant effort has been made in the last two decades in order to gain a better understanding of affect [36]. In particular, the affective computing field has reached relevant achievements in affect recognition exploiting different information channels[10, 31]. Some are easily accessible such as facial expression [9, 13, 18], body gesture [33], prosody [1], while others are hidden to the observer, such as EEG [49], ECG [42], and EDA [46]. Each of these modalities provides insights into human affect analysis with different levels of validity and reliability of the signal and intrusiveness for the user, making them more or less convenient depending on the context. Yet, despite the spread of systems and models for affect detection, most of the research is focused specifically on emotion recognition, while pain has been in general overlooked. Nevertheless, interdisciplinary studies identify valid and reliable indicators of pain, apart from self-reports, starting from the underlying biological process and exploring behavioural evidence.

Among behavioural pain responses, facial expressions are the most investigated [10, 56]; this is due to the well-established prominence of the face as a source of information compared to other channels of nonverbal communication such as paralinguistic vocalization or involuntary and purposeful bodily activity.

Of course, the analysis of facial expression for detecting pain has a common ground with the analysis of affective states through facial mimicry [6]. Indeed, psychological and neurobiological studies highlighted the tight relation between affect and facial movements [15]. In this vein, it has also been claimed that facial pain expression is specific to pain experiencing and can be distinguished from expressions of basic emotions [12, 48, 57].

For instance, the experiments conducted by Prkachin and Solomon, based on a sample of 129 subjects suffering from shoulder pain, identify several facial actions discriminating painful from non-painful movements with high validity and reliability [38]. In this vein, Simon et al. [48] proved the human capacity of discerning prototypical pain expression from other emotional and neutral facial reactions. These experiments argue for consistency of facial pain expression patterns by focusing on the spontaneous arising of specific facial reactions in a painful condition and on the recognition of a painful experience of others by the expression indicator.

Hence, when self-reports (most reliable and valuable methods for pain assessment) are not an option, automatic pain recognition systems based on facial reaction may provide a valuable alternative. In clinical contexts, for example, patients may be unable to communicate verbally and, moreover, the medical staff cannot monitor them continuously [7]. In such a scenario, typical in intensive care

SAC'22, April 25 –April 29, 2022, Brno, Czech Republic

ACM ISBN 978-1-4503-8713-2/22/04…$15.00
https://doi.org/10.1145/3477314.3507094

units (ICU), automatic pain recognition systems based on behaviour and physiological responses could support the clinical routine of pain management.

In this work, we present a novel pain classification system which leverages the natural graph representation of face landmarks [45], and relies on features describing local dynamics evolution. Hence, we take into account the progression of pain expression over time, thus overcoming the inherent limitations of frame-level approaches.

To this end, we employ a Graph Neural Network (GNN) architecture able to capture the expression semantics connecting local motion information from face landmarks to the holistic view coming from the relationships between fiducial points.

In the forthcoming Section (Sec. 2) we briefly investigate the literature concerning automatic pain recognition and Graph Neural Network. In Sec. 3 the proposed approach is presented; results are reported in Sec. 4, while Sec. 5 summarizes the key contributions of the paper and presents some concluding remarks.

2 RELATED WORKS

2.1 Automatic pain recognition

Automatic recognition of pain requires the specification of at least one source of information (modality) as input to the pain recognition system. The pain assessment literature has considered many diverse modalities as useful for the inference of painful states in humans. These can be broadly categorised into two main groups, namely *behavioural* and *physiological* modalities.

The former, considers the bulk of observable behavioural responses typically associated to pain, such as facial expression variations, body movements, vocalisations (such as crying or moaning), and spoken words [34, 47, 60].

On the other hand, the latter has to do with the exploitation of hidden physiological information, typically brain activity, cardiovascular activity, and electro-dermal activity [27].

Oftentimes, more than one source of information has been employed in order to build multi-modal approaches to automatic pain assessment [2, 50, 54].

As a matter of fact [56], the vast majority of pain assessment approaches take advantage of the behavioural responses as recorded from RGB cameras. More specifically, the analysis of facial expressions has been the most adopted technique. This is mostly due to the wide availability of datasets providing such modality (e.g. [28, 52]). Typical approaches involve feature extraction as a critical step that often includes salient points detection (e.g. facial landmarks) for both local and global geometric or appearance analysis [3, 19, 22, 29].

In general, most early works in automatic pain recognition focused on this modality. For instance, in [8] authors propose a machine assessment system for recognising pain from images of neonatal facial displays experiencing the acute pain of a heel lance. They adopt common machine learning methods for classifying pain, namely PCA, LDA, SVMs and NNSOA.

Littlewort *et al.* [26] proposed a pain assessment method based on machine learning techniques for the detection and differentiation of real vs. faked pain, by the analysis on facial Action Units.

In [32] Niese *et al.* presented a method for vision based recognition of pain from facial expressions, using color and gradient information along with a head contour model. A person specific face model is built, from which 3D geometric features are extracted. A support vector machine is then trained on these features.

More recently, [53] proposed a feature set for describing facial actions and their dynamics called *facial activity descriptors* in order to detect pain and estimate its intensity.

In [11] authors combine hand-crafted features (Local Binary Patterns (LBP) and Histogram of Oriented Gradients (HOG)) and features coming from pre-trained Deep CNNs for the assessment of neonatal facial pain.

Finally, in [61] authors propose a Convolutional Neural Network (CNN) trained end-to-end to detect neonatal pain, thus learning the relevant features during training.

2.2 Graph Neural Network

In the last decades Graph Neural Network (GNN) [23] has witnessed a flourish of investigations [58]. In a nutshell, GNNs are suitable to process data in non-Euclidean domains representable as graphs. Node embeddings is learned by exploiting the data structure in order to pass, transform and aggregate node features among neighbours. The high-level information characterising the nodes is then exploited to classify either nodes [17], edges [43] or the whole graphs [44].

While node and edge classification has already achieved resounding success [5, 30], graph classification often results in lower performance. This hurdle is ascribable to the strong compression produced by the *graph pooling step*, that pools together the node features in order to obtain a single embedding for the entire graph. The way this stage is carried out considerably influences the whole performances.

The simplest and fastest approach, though feasible for small graphs only, consists in computing either the *max*, or *min*, or *mean* of all the node embeddings [14]. An enhancement to this solution has been obtained by introducing the attention mechanism to the mean pooling [25].

In [51] the authors propose the Set2Set method, based on Long Short-Term Memory (LSTM) and attention mechanism, aiming at embedding order-dependence information into the graph embedding.

Alternatively, this task can be accomplished taking into account the graph topology by performing graph coarsening (or clustering), that progressively reduces the graph, until obtaining the final graph embedding. In this vein, in [59] a differentiable graph pooling module, namely DiffPool, learns to hierarchically map nodes to a set of clusters on the basis of both node embeddings and graph topology.

In [62] an end-to-end Deep Graph Convolutional Neural Network (DGCNN) is proposed, grounding the graph coarsening on the SortPooling layer: the continuous WL colors [24] are used to sort the nodes, thus conditioning the node order on their structural role within the graph. Sorting step has a twofold benefit: it allows to produce a sorted graph having fixed size (by keeping the first k ordered nodes only). This way, sorted and fixed size graph representation is suitable to be fed into a traditional 1D dense layer for the graph classification.

Similarly, in [63] a graph pooling operator, called HGP-SL, is introduced to sort and select nodes. This is accomplished referring

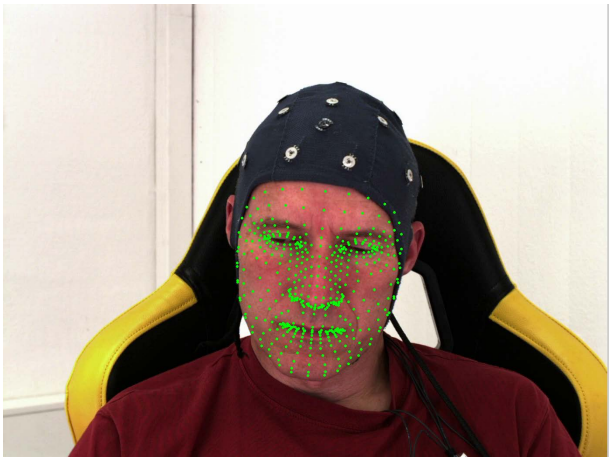

Figure 1: Face mesh based on [20].

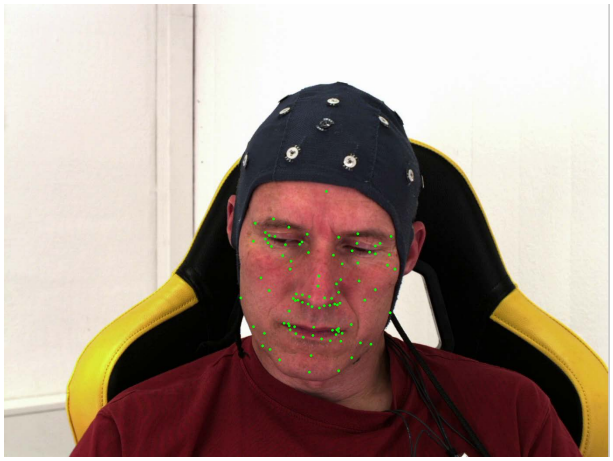

Figure 2: Face mesh following the subsampling.

to the node information score that evaluates the information that each node contains given its neighbourhood.

3 PROPOSED MODEL

The video-based facial pain expression recognition we propose consists of three main steps: graph architecture definition, node-level feature representation and graph processing.

3.1 Graph architecture definition

Given videos of faces, we build graphs such that nodes correspond to salient facial landmarks and edges connect nodes outside the close neighbourhood, according to a thresholded Euclidean distance. A graph configuration is conceived to characterise a short interval of time, thus in case of long videos we split videos in short clips. Each node is associated to a feature vector suitable to characterise the dynamics of that node in that clip.

More specifically, given a video v, if it is longer than f frames, it is split into short clips $v^i, i \in 1...k$, $|v^i| = f$[1]. On each frame in v^i the method extracts a set of fiducial points. In current implementation, we use the method presented in [20] (see Fig. 1), deriving a dense map of fiducial point that we lighten applying a uniform subsampling (see Fig. 2).

Each clip v^i is modelled by a graph G_v^i with nodes corresponding to the n selected landmarks, and edges created connecting nodes outside the close neighbourhood according to the Euclidean distance between each pair of landmarks, using an experimentally fixed threshold (see Fig. 3). This way, the local information can be shared between distant areas, fostering the message passing all over the graph. As detailed in Sec. 3.2, the node characterisation is conceived to produce a feature vector capturing the trajectory followed by the corresponding landmark in the clip at hand.

3.2 Node-level Feature Representation

Each fiducial point f is characterised considering its trajectory as a 2-dimensional stochastic process, which (x_f, y_f) coordinates are

[1]In case of videos with length not multiple of f, the last shortest video clip will be discarded

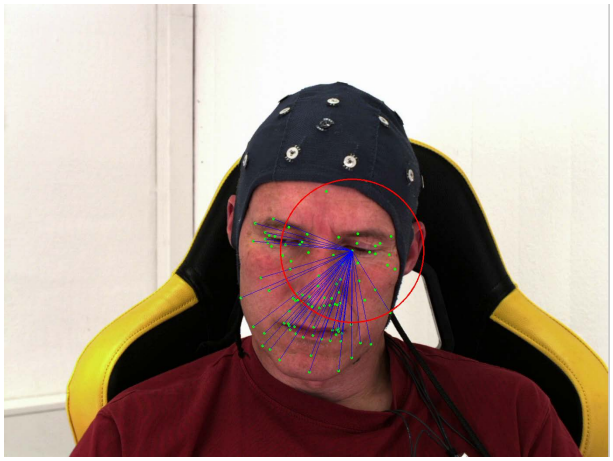

Figure 3: Example of edges for a single node. The radius of the red circle represents the minimum distance for connection.

assumed to be independent from one another. As a consequence, we end up examining $2n$ time-series in total.

Each trajectory is characterised using a set of complexity-related measures, delivering insights concerning the dynamics and predictability of the time series, and spectral attributes summarising the properties of the signals in the frequency domain. In particular, we consider the following features:

3.2.1 Approximate Entropy (ApEn). ApEn [37] is a statistical measure used to quantify the amount of regularity of fluctuations in time-series data. Larger values indicate higher complexity or irregularity in the data. ApEn has been extensively used for the analysis of physiological time-series [39, 41].

3.2.2 Sample Entropy (SampEn). As ApEn, SampEn is another measure of complexity of a signal. Large values indicate high complexity whereas smaller values characterise more self-similar and regular signals. The *SampEn* of a signal x is defined as:

$$SampEn(x, m, r) = -\log \frac{C(m+1, r)}{C(m, r)} \quad (1)$$

where m is the embedding dimension (in our experiments we set $m = 2$) and r is the radius of the neighbourhood (in our case $r = 0.2{*}std(x)$). $C(m{+}1, r)$ and $C(m, r)$ are the number of embedded vectors of length $m + 1$ and m respectively, having a Chebyshev distance inferior to r.

3.2.3 Permutation Entropy (PermEn). The *PermEn* is a complexity measure for time-series first introduced by Bandt and Pompe [4].

Given a signal x, it is defined as:

$$PermEn = -\sum p(\pi) \log_2(\pi) \quad (2)$$

where π is the set of $p!$ permutations of x of order p. In our experiments we set $p = 3$. As with *ApEn* and *SampEn*, the smaller *PermEn* is, the more regular and more deterministic the time series is. Contrarily, higher values of *PermEn*, suggest more noisy and random time series.

3.2.4 SVD Entropy (svdEn). SVD Entropy [40] indicates the number of eigenvectors that are needed for explaining the data. In other words, it measures the dimensionality of the data.

Define an embedding matrix Y of a signal x as:

$$y(i) = \left[x_i, x_{i+\text{delay}}, \ldots, x_{i+\text{ (order-1)*delay}}\right]$$
$$Y = [y(1), y(2), \ldots, y(N - (\text{ order } - 1)) * \text{ delay })]^T$$

where $delay = 1$ and $order = 3$ represent the considered time delay and the length of the embedding dimension, respectively.

The SVD entropy is then obtained as:

$$svdEn = -\sum_{i=1}^{M} \bar{\sigma}_i \log_2(\bar{\sigma}_i) \quad (3)$$

where M is the number of singular values of the embedding matrix Y and σ_i are the normalised singular values of Y. As for the previous measures of Entropy Rate, *svdEn* is lower for simpler time series and higher for more complex ones.

3.2.5 Detrended Fluctuation Analysis (DFA). DFA [35] is a method for determining the statistical self-affinity of a signal. Similarly to the Hurst Exponent, it is useful for analysing the the signal correlation behaviour and it allows the detection of the long-range dependencies. However, differently from Hurst exponent, DFA may also be applied to signals whose underlying statistics (such as mean and variance) or dynamics are non-stationary (changing with time). The computation of DFA, goes as follows. The original signal x on length N is first integrated and its average is subtracted:

$$X = \sum_i (x_i - \langle x \rangle) \quad (4)$$

The resulting cumulative sum X is divided in chunks of length c, within which the linear trend Y is computed. Let Y_i indicate the resulting piece-wise sequence of straight-line fits representing the linear trends estimated via least square fitting in each window. Then, the root-mean-square deviation from the trend (the fluctuation) is calculated as:

$$F(c) = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (X_i - Y_i)^2} \quad (5)$$

Detrending followed by fluctuation measurement is repeated over a range of different window sizes c and a log-log plot of $F(c)$ against c is constructed, upon which a straight line is fitted. The slope of this line represents the scaling exponent α delivering information about the self-affinity of the process.

3.2.6 Higuchi Fractal Dimension (HFD). HFD is a method for approximating the fractal dimension of a time series. HFD measures the rate of increase in the difference of signal amplitude while the signal samples are picked in an increasingly sparse way. HFD is computed as follows. Given a time series x, For each $k \in \{1, \ldots, K\}$ and $m \in \{1, \ldots, k\}$ define the length $L_m(k)$ by:

$$L_m(k) = \frac{N-1}{\left[\frac{N-m}{k}\right] k^2} \sum_{i=1}^{\left[\frac{N-m}{k}\right]} |X_N(m+ik) - X_N(m+(i-1)k)|$$

Total average length $L(k)$ is computed as:

$$L(k) = \frac{1}{k} \sum_{m=1}^{k} L_m(k)$$

The HFD is represented by the slope of the best fitting straight line on the log-log plot of $\frac{1}{k}$ against $L(k)$. In our experiments we set $K = 10$.

3.2.7 Petrosian Fractal Dimension (PFD). PFD represents another method for estimating the fractal dimension of a signal. In particular, the Petrosian fractal dimension of a time-series x is defined as:

$$PFD = \frac{\log_{10}(N)}{\log_{10}(N) + \log_{10}\left(\frac{N}{N+0.4N_s}\right)} \quad (6)$$

where, N is the length of the time series, and N_s is the number of sign changes in the signal derivative.

3.2.8 Katz Fractal Dimension (KFD). Katz [21] proposed yet another method for estimating the fractal dimension of a time-series. Specifically, KFD can be computed as:

$$KFD = \frac{\log_{10}(L/a)}{\log_{10}(d/a)},$$

where L is the sum of distances between successive points, a is their average, and d is the maximum distance between the first point and any other point of the considered signal.

3.2.9 Zero-Crossing Rate (ZCR). The zero-crossing rate (ZCR) is the rate at which a time-series changes from positive to zero to negative, or from negative to zero to positive. Formally, given a signal x of length N, ZCR can be defined as follows:

$$ZCR = \frac{1}{N-1} \sum_{i=1}^{N-1} 1_{\mathbb{R}_{<0}}(x_i x_{i-1})$$

where $1_{\mathbb{R}_{<0}}$ is the indicator function.

3.2.10 Mel Frequency Cepstral Coefficients (MFCCs). Besides considering complexity-related measures of the time domain signals represented by (x, y) coordinates of the facial landmark points, we augment the feature set associated to each node with some spectral features. In particular, we compute the first 13 Mel Frequency Cepstral Coefficients (MFCCs) on each trajectory.

MFCCs are coefficients derived from a representation of the short-term power spectrum of a signal, based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency. They have been extensively used in speech and sound processing as delivering compact and informative summary of the spectral content of a signal. Specifically, MFCC computation is carried out as follows:

- each signal describing the trajectory of a landmark w.r.t. its x or y coordinate, is first transformed to the frequency domain;
- a Mel filter-bank is applied to the spectrum and the energy in each filter su summed
- the logarithm of all filter-bank energies is taken
- the DCT of the log filter-bank energies is computed
- the first 13 coefficients are eventually kept.

Once all the complexity measures and MFCC features are extracted, they are concatenated to form a 44-dimensional feature vector for each node.

3.3 Graph processing

Given a dataset of videos $D = \{v_j, j \in 1..d\}$, each one labelled by $l_v \in \{Pain, notPain\}$, we split D into train and test sets, and for each video v, the corresponding clip graphs G_v^i (possibly 1) are computed as described in Sec. 3.1 and Sec. 3.2.

To solve the binary classification over graphs, we resort to the Deep Graph Convolutional Neural Network (DGCNN) [62]. This is trained on the pairs $\{G_v^i, l_v\}$ in the training set, l_v being the label of the video the clip belongs to.

In testing phase, for each video v we create its graphs G_v^i, and evaluate them by collecting the classifications $C_v = \{\hat{l}_v^i\}$. The video classification $\hat{l}_v$ is finally computed as the median over C_v.

4 EXPERIMENTAL RESULTS

Here we evaluate the performance of our model for pain classification task (neutral vs. pain) in two different scenarios: acted pain and spontaneous pain, induced by a thermal stimulation. This assessment is carried out referring to video materials from the BioVid Heat Pain Database [2] [52].

4.1 BioVid Heat Pain Database

The BioVid (Biopotential and Video) Heat Pain Database collects multimodal reactions from 90 subjects undergoing induced heat pain in four intensities held for 4 seconds, repeated randomly 20 times each, and with a random pose between stimuli. Experiments were conducted both with and without EMG sensors. Since we are interested in analysing the facial expression, we focused our experiments on data acquired without EMG sensors. This part of the dataset (Part A) consists of 8700 samples 5.5 second long, corresponding to 87 subjects covering the 5 intensity classes.

[2] https://www.iikt.ovgu.de/BioVid.html

In the same dataset, the 90 participants posed both basic emotions and pain, bringing to the collection of 630 videos 1 minute long, covering 7 emotions among which pain (Part D).

4.2 Acted pain classification

In the first experiment, we use the Part D of the database, selecting pain and neutral videos for a total of 178 videos. Each video is divided into 200-frame sequences, reaching a total of 1245 samples. This implementation choice is motivated by the very nature of pain expression dynamics, typically discontinuous. Hence, in order to discriminate the presence of pain in a video, we analyse short windows and thence make a global prediction at the video level. Moreover, this approach eases the comparison between the two experimental settings, characterised by videos with a significant difference in duration.

Then, for each sequence we collect the trajectories (in x and y dimensions) of 94 face landmarks, obtained by applying a uniform subsampling to the 468 ones delivered by the MediaPipe Python library [16], and finally we derive the set of 44 features per landmark (see Sec. 3 for details). This information is then associated to each node of the graph, afterwards completed by edges between pairs of landmarks (i.e. nodes) whose distance exceeds an experimental fixed threshold equal to double the distance between the eyes. The number of edges obtained is 4032 on average, given that the amount of edges hinges on the facial configuration in the specific sequence.

Following the feature extraction step, the DGCNN classifier is trained using the Adam optimisation algorithm to minimise the binary cross-entropy loss and evaluated via 5-fold cross-validation where the train/test split is performed to avoid the simultaneous presence of sequences taken from the same video in train and test set. We modified the network structure, making some variations to the model proposed in [62]. First, we add a graph convolutional layers to the original network and double the number of kernels, reaching a total of five graph convolution layers with 64, 64, 64, 64, 1 output channels, respectively. Also, the SortPooling layer is revised to keep the first 40 sorted nodes. Moreover, the hyperbolic tangent activation function is replaced with rectified linear units (ReLU).

The obtained results are presented in Tab. 1 in comparison to a baseline Support Vector Machine (SVM) model created in order to have a benchmark, since, as far as we know, there are no works in literature adopting the acted part of the database for pain classification. In order to have an appropriate data structure for the SVM classifier training, the graph structure was discharged, flattened to a 4324-dimensional feature vector, and then reduced to a 200-dimensional vector using PCA. It is worth noting that the information carried by edges is unavoidably lost in the baseline model.

Further, we compare our DGCNN classifier with an SVM adopting Action Units (AUs) intensities as features. This more standard approach obtains almost the same performance as the SVM baseline, pointing out the importance of the graph structure for learning effectiveness over the concatenation of the feature sets.

In Tab. 1 we report the evaluations of the proposed method (CM+DGCNN), and the baselines (AUs+ SVM and CM+SVM), proving the effectiveness of the adopted features (CM) in combination with the graph structure and learning.

Model	Video-level accuracy
AUs+SVM	0.669 ± 0.146
CM+SVM	0.714 ± 0.102
CM+DGCNN	**0.834** ± 0.116

Table 1: Results on acted pain videos (Part D) using proposed complexity measures (CM) in combination with DGCNN model compared to standard AUs intensity features and SVM.

4.3 Spontaneous pain classification

The spontaneous pain discrimination task is, in general, more worthwhile but also challenging. For this experiment we refer to the short video sequences (5.5 seconds) included in the Part A of the database, taking into account only the sequences labelled as pain-free (0/4) and with maximum pain intensity (4/4). In doing so, we obtain 40 videos per participant (87 subjects altogether), 20 for each label, totalling 3480 videos.

There are no differences in the feature extraction step and the network structure compared to the acted experiment. Although, in this session there is a one-to-one correspondence between videos and graphs motivated by the shortness of video sequences and by the presence of a single painful stimulation per video. For this reason, the video-level accuracy is equivalent to standard accuracy of the DGCNN.

For this experiment we evaluate our approach, CM+DGCNN, and compare it to both the baseline method AUs+SVM, and the results reported in [53] and [55]. As shown in Tab. 2, our results are slightly above the state of the art on this Database.

Model	Accuracy
AUs+SVM	0.648 ± 0.068
Werner at al. (2016)	0.700
Normalized Werner at al. (2016)	0.724
Werner et al. (2017)	0.718
CM+DGCNN	**0.732** ± 0.139

Table 2: Results on spontaneous pain videos (Part A) in comparison with the state of the art and a plain AUs-based classifier. Werner at al. (2016) reports two results. The second adding a feature standardization per subject.

5 CONCLUSION

In this paper, we presented a novel approach to pain expression recognition that harnesses the local dynamics of facial movements along with geometric properties of the face to train a GNN for video-based pain classification. The proposed method proves its effectiveness in comparison to the state-of-the-art models on the BioVid Heat Pain Database.

The promising results and the flexibility of the GNN-based approach open up to many chances for future works. First of all an insight into the DGCNN would highlight which nodes are selected as more relavant for the classification by the DGCNN SortPooling layer, and in which order. This way, a face map of relevance to pain would be produced. Second, the adoption of a multimodal strategy, by the inclusion of physiological signals, may lead to further improvements, also regarding the reliability of the pain recognition system. To this end, the graph structure would enable many embedding strategies for different sources of data. Moreover, pain intensity levels could be taken into account to achieve finest predictions. Last, the specificity of the exploited complexity-related measures and spectral attributes of facial points trajectories in relation to pain could be evaluated by testing the performance on pain vs. emotions task.

ACKNOWLEDGMENTS

This work was part of the project n. 2018-0858 title "Stairway to elders: bridging space, time and emotions in their social environment for wellbeing" supported by Fondazione CARIPLO.

REFERENCES

[1] Samuel Albanie, Arsha Nagrani, Andrea Vedaldi, and Andrew Zisserman. 2018. Emotion recognition in speech using cross-modal transfer in the wild. In *Proceedings of the 26th ACM international conference on Multimedia*. 292–301.
[2] Mohammadreza Amirian, Markus Kächele, and Friedhelm Schwenker. 2016. Using radial basis function neural networks for continuous and discrete pain estimation from bio-physiological signals. In *IAPR Workshop on Artificial Neural Networks in Pattern Recognition*. Springer, 269–284.
[3] Ahmed Bilal Ashraf, Simon Lucey, Jeffrey F Cohn, Tsuhan Chen, Zara Ambadar, Kenneth M Prkachin, and Patricia E Solomon. 2009. The painful face–pain expression recognition using active appearance models. *Image and vision computing* 27, 12 (2009), 1788–1796.
[4] Christoph Bandt and Bernd Pompe. 2002. Permutation entropy: a natural complexity measure for time series. *Physical review letters* 88, 17 (2002), 174102.
[5] Smriti Bhagat, Graham Cormode, and S Muthukrishnan. 2011. Node classification in social networks. In *Social network data analytics*. Springer, 115–148.
[6] G. Boccignone, D. Conte, V. Cuculo, A. D'Amelio, G. Grossi, and R. Lanzarotti. 2018. Deep Construction of an Affective Latent Space via Multimodal Enactment. *IEEE Transactions on Cognitive and Developmental Systems* 10, 4 (Dec 2018), 865–880. https://doi.org/10.1109/TCDS.2017.2788820
[7] Giuseppe Boccignone, Claudio de'Sperati, Marco Granato, Giuliano Grossi, Raffaella Lanzarotti, Nicoletta Noceti, and Francesca Odone. 2020. Stairway to Elders: Bridging Space, Time and Emotions in Their Social Environment for Wellbeing.. In *ICPRAM*. 548–554.
[8] Sheryl Brahnam, Chao-Fa Chuang, Randall S Sexton, and Frank Y Shih. 2007. Machine assessment of neonatal facial expressions of acute pain. *Decision Support Systems* 43, 4 (2007), 1242–1254.
[9] Sathya Bursic, Giuseppe Boccignone, Alfio Ferrara, Alessandro D'Amelio, and Raffaella Lanzarotti. 2020. Improving the Accuracy of Automatic Facial Expression Recognition in Speaking Subjects with Deep Learning. *Applied Sciences* 10, 11 (2020), 4002.
[10] Rafael A Calvo and Sidney D'Mello. 2010. Affect detection: An interdisciplinary review of models, methods, and their applications. *IEEE Transactions on affective computing* 1, 1 (2010), 18–37.
[11] Luigi Celona and Luca Manoni. 2017. Neonatal facial pain assessment combining hand-crafted and deep features. In *International Conference on Image Analysis and Processing*. Springer, 197–204.
[12] Kenneth D Craig, Kenneth M Prkachin, and Ruth E Grunau. 2011. The facial expression of pain. (2011).
[13] Vittorio Cuculo and Alessandro D'Amelio. 2019. OpenFACS: an open source FACS-based 3D face animation system. In *International Conference on Image and Graphics*. Springer, 232–242.
[14] David Duvenaud, Dougal Maclaurin, Jorge Aguilera-Iparraguirre, Rafael Gómez-Bombarelli, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. 2015. Convolutional networks on graphs for learning molecular fingerprints. *arXiv preprint arXiv:1509.09292* (2015).
[15] Paul Ekman, Wallace V Friesen, and Phoebe Ellsworth. 2013. *Emotion in the human face: Guidelines for research and an integration of findings*. Vol. 11. Elsevier.

[16] Google. 2021. MediaPipe Face Mesh. https://google.github.io/mediapipe/solutions/face_mesh

[17] William L Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 1025–1035.

[18] Neha Jain, Shishir Kumar, Amit Kumar, Pourya Shamsolmoali, and Masoumeh Zareapoor. 2018. Hybrid deep neural networks for face emotion recognition. *Pattern Recognition Letters* 115 (2018), 101–106.

[19] Sebastian Kaltwang, Ognjen Rudovic, and Maja Pantic. 2012. Continuous pain intensity estimation from facial expressions. In *International Symposium on Visual Computing*. Springer, 368–377.

[20] Yury Kartynnik, Artsiom Ablavatski, Ivan Grishchenko, and Matthias Grundmann. 2019. Real-time facial surface geometry from monocular video on mobile GPUs. *arXiv preprint arXiv:1907.06724* (2019).

[21] Michael J Katz. 1988. Fractals and the analysis of waveforms. *Computers in biology and medicine* 18, 3 (1988), 145–156.

[22] Rizwan Ahmed Khan, Alexandre Meyer, Hubert Konik, and Saida Bouakaz. 2013. Pain detection through shape and appearance features. In *2013 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 1–6.

[23] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[24] AA Leman and B Weisfeiler. 1968. A reduction of a graph to a canonical form and an algebra arising during this reduction. *Nauchno-Technicheskaya Informatsiya* 2, 9 (1968), 12–16.

[25] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. 2015. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493* (2015).

[26] Gwen C Littlewort, Marian Stewart Bartlett, and Kang Lee. 2009. Automatic coding of facial expressions displayed during posed and genuine pain. *Image and Vision Computing* 27, 12 (2009), 1797–1803.

[27] Daniel Lopez-Martinez and Rosalind Picard. 2017. Multi-task neural networks for personalized pain recognition from physiological signals. In *2017 Seventh International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*. IEEE, 181–184.

[28] Patrick Lucey, Jeffrey F Cohn, Kenneth M Prkachin, Patricia E Solomon, and Iain Matthews. 2011. Painful data: The UNBC-McMaster shoulder pain expression archive database. In *2011 IEEE International Conference on Automatic Face & Gesture Recognition (FG)*. IEEE, 57–64.

[29] Hongying Meng and Nadia Bianchi-Berthouze. 2013. Affective state level recognition in naturalistic facial and vocal expressions. *IEEE Transactions on Cybernetics* 44, 3 (2013), 315–328.

[30] Bonaventure Molokwu, Shaon Bhatta Shuvo, Narayan C Kar, and Ziad Kobti. 2020. Node Classification and Link Prediction in Social Graphs using RLVECN. In *32nd International Conference on Scientific and Statistical Database Management*. 1–10.

[31] Dung Nguyen, Kien Nguyen, Sridha Sridharan, David Dean, and Clinton Fookes. 2018. Deep spatio-temporal feature fusion with compact bilinear pooling for multimodal emotion recognition. *Computer Vision and Image Understanding* 174 (2018), 33–42.

[32] Robert Niese, Ayoub Al-Hamadi, Axel Panning, Dominik Brammen, Uwe Ebmeyer, and Bernd Michaelis. 2009. Towards pain recognition in post-operative phases using 3d-based features from video and support vector machines. *International Journal of Digital Content Technology and its Applications* 3, 4 (2009), 21–31.

[33] Fatemeh Noroozi, Dorota Kaminska, Ciprian Corneanu, Tomasz Sapinski, Sergio Escalera, and Gholamreza Anbarjafari. 2018. Survey on emotional body gesture recognition. *IEEE transactions on affective computing* (2018).

[34] Yaniv Oshrat, Ayala Bloch, Anat Lerner, Azaria Cohen, Mireille Avigal, and Gabi Zeilig. 2016. Speech prosody as a biosignal for physical pain detection. In *Conf Proc 8th Speech Prosody*. 420–24.

[35] C-K Peng, Shlomo Havlin, H Eugene Stanley, and Ary L Goldberger. 1995. Quantification of scaling exponents and crossover phenomena in nonstationary heartbeat time series. *Chaos: an interdisciplinary journal of nonlinear science* 5, 1 (1995), 82–87.

[36] Rosalind W Picard. 2000. *Affective computing*. MIT press.

[37] Steven M Pincus. 1991. Approximate entropy as a measure of system complexity. *Proceedings of the National Academy of Sciences* 88, 6 (1991), 2297–2301.

[38] Kenneth M Prkachin and Patricia E Solomon. 2008. The structure, reliability and validity of pain expression: Evidence from patients with shoulder pain. *Pain* 139, 2 (2008), 267–274.

[39] Joshua S Richman and J Randall Moorman. 2000. Physiological time-series analysis using approximate entropy and sample entropy. *American Journal of Physiology-Heart and Circulatory Physiology* 278, 6 (2000), H2039–H2049.

[40] Stephen J Roberts, William Penny, and Iead Rezek. 1999. Temporal and spatial complexity measures for electroencephalogram based brain-computer interfacing. *Medical & biological engineering & computing* 37, 1 (1999), 93–98.

[41] Malihe Sabeti, Serajeddin Katebi, and Reza Boostani. 2009. Entropy and complexity measures for EEG signal classification of schizophrenic and control participants. *Artificial intelligence in medicine* 47, 3 (2009), 263–274.

[42] Pritam Sarkar and Ali Etemad. 2020. Self-supervised ECG Representation Learning for Emotion Recognition. *IEEE Transactions on Affective Computing* (2020), 1–1. https://doi.org/10.1109/TAFFC.2020.3014842

[43] Kristof T Schütt, Pieter-Jan Kindermans, Huziel E Sauceda, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. 2017. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *arXiv preprint arXiv:1706.08566* (2017).

[44] Nino Shervashidze, Pascal Schweitzer, Erik Jan Van Leeuwen, Kurt Mehlhorn, and Karsten M Borgwardt. 2011. Weisfeiler-Lehman graph kernels. *Journal of Machine Learning Research* 12, 9 (2011).

[45] Jiazheng Shi, Ashok Samal, and David Marx. 2006. How effective are landmarks and their geometry for face recognition? *Computer vision and image understanding* 102, 2 (2006), 117–133.

[46] Jainendra Shukla, Miguel Barreda-Angeles, Joan Oliver, G. C. Nandi, and Domenec Puig. 2019. Feature Extraction and Selection for Emotion Recognition from Electrodermal Activity. *IEEE Transactions on Affective Computing* (2019), 1–1. https://doi.org/10.1109/TAFFC.2019.2901673

[47] Karan Sikka, Alex A Ahmed, Damaris Diaz, Matthew S Goodwin, Kenneth D Craig, Marian S Bartlett, and Jeannie S Huang. 2015. Automated assessment of children's postoperative pain using computer vision. *Pediatrics* 136, 1 (2015), e124–e131.

[48] Daniela Simon, Kenneth D Craig, Frederic Gosselin, Pascal Belin, and Pierre Rainville. 2008. Recognition and discrimination of prototypical dynamic expressions of pain and emotions. *PAIN®* 135, 1-2 (2008), 55–64.

[49] Tengfei Song, Wenming Zheng, Peng Song, and Zhen Cui. 2018. EEG emotion recognition using dynamical graph convolutional neural networks. *IEEE Transactions on Affective Computing* 11, 3 (2018), 532–541.

[50] Patrick Thiam and Friedhelm Schwenker. 2017. Multi-modal data fusion for pain intensity assessment and classification. In *2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA)*. IEEE, 1–6.

[51] Oriol Vinyals, Samy Bengio, and Manjunath Kudlur. 2015. Order matters: Sequence to sequence for sets. *arXiv preprint arXiv:1511.06391* (2015).

[52] Steffen Walter, Sascha Gruss, Hagen Ehleiter, Junwen Tan, Harald C Traue, Philipp Werner, Ayoub Al-Hamadi, Stephen Crawcour, Adriano O Andrade, and Gustavo Moreira da Silva. 2013. The biovid heat pain database data for the advancement and systematic validation of an automated pain recognition system. In *2013 IEEE international conference on cybernetics (CYBCO)*. IEEE, 128–131.

[53] Philipp Werner, Ayoub Al-Hamadi, Kerstin Limbrecht-Ecklundt, Steffen Walter, Sascha Gruss, and Harald C Traue. 2016. Automatic pain assessment with facial activity descriptors. *IEEE Transactions on Affective Computing* 8, 3 (2016), 286–299.

[54] Philipp Werner, Ayoub Al-Hamadi, Robert Niese, Steffen Walter, Sascha Gruss, and Harald C Traue. 2014. Automatic pain recognition from video and biomedical signals. In *2014 22nd International Conference on Pattern Recognition*. IEEE, 4582–4587.

[55] Philipp Werner, Ayoub Al-Hamadi, and Steffen Walter. 2017. Analysis of facial expressiveness during experimentally induced heat pain. In *2017 Seventh International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*. IEEE, 176–180.

[56] Philipp Werner, Daniel Lopez-Martinez, Steffen Walter, Ayoub Al-Hamadi, Sascha Gruss, and Rosalind Picard. 2019. Automatic recognition methods supporting pain assessment: A survey. *IEEE Transactions on Affective Computing* (2019).

[57] Amanda C de C Williams. 2002. Facial expression of pain: an evolutionary account. *Behavioral and brain sciences* 25, 4 (2002), 439–455.

[58] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. 2020. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems* 32, 1 (2020), 4–24.

[59] Rex Ying, Jiaxuan You, Christopher Morris, Xiang Ren, William L Hamilton, and Jure Leskovec. 2018. Hierarchical graph representation learning with differentiable pooling. *arXiv preprint arXiv:1806.08804* (2018).

[60] Ghada Zamzmi, Chih-Yun Pai, Dmitry Goldgof, Rangachar Kasturi, Yu Sun, and Terri Ashmeade. 2017. Automated pain assessment in neonates. In *Scandinavian Conference on Image Analysis*. Springer, 350–361.

[61] Ghada Zamzmi, Rahul Paul, Dmitry Goldgof, Rangachar Kasturi, and Yu Sun. 2019. Pain assessment from facial expression: Neonatal convolutional neural network (N-CNN). In *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–7.

[62] Muhan Zhang, Zhicheng Cui, Marion Neumann, and Yixin Chen. 2018. An end-to-end deep learning architecture for graph classification. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[63] Zhen Zhang, Jiajun Bu, Martin Ester, Jianfeng Zhang, Chengwei Yao, Zhi Yu, and Can Wang. 2019. Hierarchical graph pooling with structure learning. *arXiv preprint arXiv:1911.05954* (2019).