



PLAAN: Pain Level Assessment with Anomaly-detection based Network

Yi Li¹ · Shreya Ghosh¹ · Jyoti Joshi¹

Received: 9 June 2020 / Accepted: 17 November 2020 / Published online: 6 January 2021
© The Author(s), under exclusive licence to Springer Nature Switzerland AG part of Springer Nature 2021

Abstract

Automatic chronic pain assessment and pain intensity estimation has been attracting growing attention due to its widespread applications. One of the prevalent issues in automatic pain analysis is inadequate balanced expert-labelled data for pain estimation. This work proposes an anomaly detection based network addressing one of the existing limitations of automatic pain assessment. The evaluation of the network is performed on pain intensity estimation and protective behaviour estimation tasks from body movements in the EmoPain Challenge dataset. The EmoPain dataset consists of body part based sensor data for both the tasks. The proposed network, PLAAN (Pain Level Assessment with Anomaly-detection based Network), is a lightweight LSTM-DNN network which considers features based on sensor data as the input and predicts intensity level of pain and presence or absence of protective behaviour in chronic low back pain patients. Joint training considering body movement patterns, such as exercise type, corresponding to pain exhibition as a label improves the performance of the network. However, contrary to perception, protective behaviour rather exists sporadically alongside pain in the EmoPain dataset. This induces yet another complication in accurate estimation of protective behaviour. This problem is resolved by incorporating anomaly detection in the network. A detailed comparison of different networks with varied features is outlined in the paper, presenting a significant improvement with the final proposed anomaly detection based network.

Keywords Automatic Pain Assessment · Protective behaviour · Emopain challenge · Neural network · Anomaly detection

1 Introduction

The International Association for the Study of Pain defines chronic pain as “an unpleasant sensory and emotional experience associated with actual or potential tissue damage” which lasts for a long period of time [1]. Chronic pain can influence an individual’s emotional and mental well-being. It can bring changes to one’s attitude, beliefs, personality and thus, adversely affecting important activities of daily living such as in workplace or social life [2]. For some chronic pain patients, even standing for five minutes could bring a lot of pain, which would not allow them to continue what they are going to do next. Hence, correct pain intensity as well as protective behaviour assessment is necessary to assist patients in their pain management. Mostly, pain intensity and protective behaviour is measured by self-report via clinical interview

of the patient [1]. Thus, correct assessment of occurrence of pain and its intensity is a challenging problem in itself due to idiosyncrasy and subjective biases [3]. Additionally, the treatment and management of pain not only depends on the occurrence of pain but also heavily relies on the correct estimation of pain intensity.

There are different treatments available to manage and ease the pain for chronic patients [4]. Generally, people with chronic pain go through rehabilitation for their pain management in proper clinical settings. Trained physiotherapists treat chronic pain patients via psychological support, movement and demolish patients’ ‘movement behaviour fear’ [5]. The most common treatment is called ‘cognitive behavioural therapy’ which requires an experienced therapist to help the patients to set up a personalized solution for their daily activities. The main difficulty in this treatment is that a patient could not verbally express his/her reaction to the pain explicitly. Due to the aforementioned difficulty, sometimes the cognitive behavioural therapy fails to heal the patient properly. Furthermore, when patients are not available for self-reporting the pain, direct behaviour observation done by clinicians could be an alternative. While interact-

Yi Li and Shreya Ghosh have equal first authors.

✉ Shreya Ghosh
shreya.ghosh@monash.edu

¹ Human Centered AI at Monash University, Melbourne, Australia

ing with professional practitioners, patients are prone to avoid pain-related uncomfortable behaviours which they are experiencing [6]. Therefore, physicians can assess the pain severity by observing the magnitude of patients' behaviour. Even so, there are some drawbacks of direct observation. The main drawback occurs when patients tend to control their spontaneous behaviour. Either restraining or exaggerating the expressions could lead to inaccurate observed results. On the other hand, biased observed outcomes could exist between different observers [6].

Nowadays clinicians are shifting from physical rehabilitation to self-management, in which ubiquitous technology-based tools provide huge support to the physiotherapists [7]. The 'self-care' based therapy is mainly based on patients' better understanding of their own pain level which could be effective in managing their pain. On the other hand, it is obvious that the patient does not have sufficient knowledge to select the appropriate exercises/movements for them. The physiotherapists guide them regarding the same.

With advances in deep learning and ubiquitous computing, the automatic monitoring of chronic pain patients in the rehabilitation center is drawing an increasing attention [8–12]. These studies have addressed the existing challenges in the domain of automatic pain assessment and pain intensity estimation. Similar to pain, protective behaviour can cause reduced participation in social life [13,14] which further leads to depression. According to a study [9], there are five types of protective behaviours (hesitation, guarding, stiffness, bracing and support). The performance of PLAAN is evaluated on the EmoPain Challenge dataset [9]. In this dataset, all of the above-mentioned behaviours are treated as one class (protective behaviour) [11].

In the present work, deep learning-based light-weight networks are proposed which predict pain intensity and protective behaviour of the chronic pain patients. Additionally, the concept of anomaly detection in the context of automatic pain assessment is also proposed. Anomaly detection is the identification of outliers in a given data. This paper is an extension of our earlier work [15], which was part of the EmoPain 2020 challenge. The key-additions are: (a) pain intensity and protective behavior are modelled as anomaly detection; (b) pain estimation is further explored as a hierarchical classification problem; (c) pain intensity estimation is proposed by protective behaviour only, and (d) extensive literature review is performed. The main *contributions* of this paper are as follows:

- A deep learning-based light-weight network, PLAAN (Pain Level Assessment with Anomaly detection based Network), is proposed to predict pain intensity and protective behaviour.

- To enhance the performance of PLAAN, joint-training of the networks are performed considering exercise type as an additional label.
- In order to handle data imbalance, anomaly detection strategy is adopted which further improves class bias.

The rest of the paper is organized as follows: Sect. 2 describes the prior work in this area. Section 3 contains the details of the proposed method. Section 4 is about the experimental details. Section 5 contains the experimental result description and ablation study. The last section describes conclusion and future directions of this study.

2 Background

This paper attempts to cover several subtopics from the literature which form essential parts of the proposed PLAAN architecture. The related work is presented in the following order—Existing techniques for automatic pain behaviour analysis, Sect. 2.1; Proposed algorithm literature for segregating outliers by means of anomaly detection in sequential inputs, Sect. 2.2; Persisting issues and management of insufficient and imbalanced datasets in automatic pain assessment, Section 2.3.

2.1 Pain behaviour analysis

The development of automatic pain detection can be chased back to one of the early works done by Ashraf et al. [16]. In the study, shape and appearance characteristics from video sequences of facial expression were extracted by Active Appearance Models (AAM) [17]. Further, the refined features were piped into Support Vector Machines (SVM) for classifying whether the pain event exists. The main finding of this study proves the feasibility of detecting pain expression automatically in video sequences. However, to relax the pressure on memory, Ashraf et al. [16] compressed the temporal information during pain detection, which is considered having a negative influence on model accuracy as reported in a study by Lucey et al. [18]. The study conducted a comparison between the performance of two different compressed signals on the AAM model followed by SVM classification similar to Ashraf et al.'s work [16]. This experiments proved the importance of temporal information with respect to pain estimation in video clips.

In another work, Lucey et al. [19] collected UNBC-Mac-Master Shoulder Pain Expression Archive Database. To evaluate the dataset effectiveness in pain behaviour detection, they used the same system in the previous study [18] as a baseline network and then compared it with the other two experiments: detection of pain-related action units at the sequence-level and frame-level. The results imply that the

accuracy of a pain estimation network should be considered in the context of the system application, whether a frame-level detection result is required. The study by Sikka et al. [20] proposed a novel framework to address the issues in automatic pain recognition in a video sequence when the frame-wise pain level is absent, and the specific timing of the occurrence of pain behaviour is unknown. The proposed network was the first implementation of Multiple Instance Learning (MIL) in video-based pain detection, and it treats every video sequence as a bag and generates multiple segments from it for further weakly supervised learning. In present work, a similar MIL based framework to differentiate multiclass pain intensity has been explored, and it is further used to make a comparison with the best-performed network.

In one of the earlier works in pain intensity estimation, Hammal and Cohn [21] used Prkachin and Solomon Pain Intensity (PSPI) to classify frame-wise pain level of video clips from the UNBC-McMaster shoulder pain dataset, and trained an SVM classifier for each pain intensity individually. This study shows the reliability of using facial expression in pain level recognition. In another study, instead of treating pain intensity as multiclass classification, Kaltwang et al. [22] considered it as a regression problem and developed a framework of Relevance Vector Regression (RVR) to estimate the continuous pain intensity. The study suggests that the regression network outperforms the calculation done by AUs in terms of pain intensity in a static image.

In another study of pain recognition from facial images, Bellantonio et al. [23] figured out three essential factors that will affect the pain detection outcome. One of them is the static information from a single video frame, and another one is the dynamic information regarding continuous facial expression along the entire video sequence. Furthermore, they emphasise that including deep temporal information can enhance the network's discrimination ability between different pain levels. This idea is also confirmed later in the work of Rodriguez et al. [24], where they proposed a CNN to estimate the binary pain recognition on facial images and further improved the performance by piping the CNN output features to LSTM. Similarly, Zhou et al. [10] developed a novel framework by embedding multiple RNN layers after a standard CNN layer. This network effectively predicts the continuous pain intensity of facial videos from UNBC-McMaster shoulder pain dataset. Recently, Bargshady et al. [25] design a framework that feeds VGG features to a two-stream DNN and conclude that this joint training approach can improve the performance of pain level multi-classification problem.

While the field of pain estimation draws significant attention, some of the studies also focus on the detection of protective behaviour since it is essential for pain recognition. Wang et al. [26] explored the approach of implementing stacked LSTM and dual LSTM for detecting protective

behaviour from MoCap and sEMG data. The performance of different LSTM networks was also evaluated for detecting the protective behaviour in videos from patients with Chronic Low Back Pain (CLBP) [12]. Wang et al. [11] investigated further on the usage of an attention-based deep learning algorithm in the detection of protective behaviour and also proposed a framework called BodyAttentionNet that can learn more informative information to improve the detection accuracy.

Kaltwang et al. [22] discovered that the combination of facial landmarks and appearance features can adequately estimate the pain severity compared with applying only one type of features using images. In another work related to severity estimation of CLBP, Olugbade et al. [27] explored the feasibility of investigating body motion and muscle activity patterns while the participants carry out an action or perform an activity that involves them in reaching forward. One key finding is that considering electromyographic data from muscle activities individually for the task of pain detection shows excellent performance. Moreover, another significant result is that the combination of body motion and muscle activity achieved an optimal result in the estimation of levels of CLBP. Later, they also confirmed that the modality is also valuable when the participants are undertaking full trunk flexion and sit-to-stand exercises [28].

Some studies have also emphasised that the detection of protective behaviour is critical in the context of chronic pain recognition [8]. When people with chronic pain try to prevent themselves from getting hurt in some intensive physical activities, their body will naturally react avoidance. Therefore, bodily expression as protective behaviour is a salient representation of chronic pain. After confirming the effectiveness of body movement and muscle energies in the context of automatic pain detection, Aung et al. [9] curated the EmoPain dataset. The baseline of this dataset was based on a random forest on the 13 joint angles, joint energies and sEMG data. The work presented in this paper utilises this dataset and also makes a comparison with results from other available studies using same data. The performance comparison is listed in Tables 5, 6 and 7.

Another study [29] focuses on manipulating the input data where a multimodal and multilevel network is constructed that will treat the existence of protective behaviour as an early sign of future pain event, and perform pain recognition in several stages. The proposed network detects the appearance of pain first, then feeds the output and previous features to the second level to detect the occurrence of protective behaviour. Next, the outcome of the second level will be concatenated with the previous input to estimate the pain intensity at the last stage. The main finding of this work is that the successful detection of the existence of protective behaviour can improve the result of pain estimation. A similar network architecture is evaluated in this paper for comparison

with other proposed approaches. Furthermore, the proposed network, PLAAN, is trained to discriminate the undertaking exercise type and the associated pain intensity/protective behaviour at the same time.

Another work of EmoPain Challenge by Yuan and Mahmoud [30] focuses on building an effective neural network. They proposed an Autoencoder-LSTM-Attention-Net (ALANet), which was designed for extracting the most expressive temporal information. In comparison with the baseline network of EmoPain dataset, ALANet exceeds the baseline accuracy with only one-twentieth training time. However, this network cannot discriminate well between low pain and high pain targets. This issue is occurring due to the class-wise imbalance in the EmoPain dataset. PLAAN is developed especially to address this concern by applying anomaly detection.

Similar to Wang et al. [11], similar terminology has been used in this paper: ‘sample’ refers to a single data point at every single timestep; ‘segment’ or ‘frame’ refers to a small data chunk having several samples within; ‘instance’ is at participant-level data which contains full data sequence of all the activities during one trial.

2.2 Anomaly detection

The proportion of frames containing an event of occurrence of pain or protective behaviour in the EmoPain dataset is significantly less. Statistically, the number of frames having pain or protective behaviour labels are in range 16–20 per 100 frames. This motivate us to follow the approach of anomaly detection. In an interesting work, Ravanbaksh et al. [31] proposed a video based flow detection of an event via an adversarial approach for abnormal activity detection. Ribeiro et al. [32] proposed a convolutional auto encoder based technique for anomaly detection in videos. The reconstruction error of each frame is considered as an anomaly score. The final score for a video is predicted by aggregating high-level spatial and temporal features with the input frames. Meheta et al. [33] used an adversarial network comprised of two channel 3D convolutional auto encoders. Both channels deals with video sequences and optical flow which reconstruct thermal data and optical flow input sequences. Chalapathy et al. [34] conducted a survey on deep learning techniques for abnormal activity detection. Most of the existing research on this topic focus on a application of specific research area where the learning is based on an auto-encoder for one class learning.

2.3 Handling imbalanced data

A useful technique for handling imbalanced data is required to improve the network performance, given the huge imbalance in the EmoPain Challenge data. The implementation

of balancing techniques are generally at the input level. The training dataset can be manually balanced by over-sampling the minority class or under sampling the majority class. Furthermore, Synthesizing new data for the minority class can also balance the class distribution. Chawla et al. [35] develop the Synthetic Minority Over-sampling Technique (SMOTE) that can enhance the discrimination between minority and majority classes. SMOTE is based on the idea of the KNN algorithm. For any randomly picked instances from the minority class, SMOTE will choose its K nearest neighbours and compute the distances between it and its neighbours, then synthesize new data points with only half of the previous distance. Inspired by the above mentioned literature, we use SMOTE data balancing technique in our experiments.

Except for the manipulation of data records, an alternative way to modify the input data would be feature engineering. In the study of continuous pain estimation on UNBC-McMaster pain database, Egede et al. [36] imply that a combination of handcrafted features and deep learned features could combat the limitation of the imbalanced data, since the significant features can improve the network performance.

3 Proposed approach

3.1 Preliminaries

This section introduces the deep learning frameworks, which are trained to classify the above mentioned tasks. The Long Short Term Memory (LSTM) [37], Bidirectional-LSTM [38], Attention-LSTM and LSTM-DNN networks are trained and evaluated. For all cases, layers configurations are chosen empirically.

LSTM We use a simple LSTM network as a baseline. LSTM is an improved version of RNN, which is designed to solve the long dependency issue. LSTM’s main learning mechanism called ‘gate’, which allows the network itself to identify the part having less/more information. It acts accordingly to ignore and pay more attention that specific part of the sequential input. Thus, LSTM can learn and capture the dependency within the sequence better and provide higher performance on time series data. The network has three LSTM layers: 1024, 512, 256 for pain intensity estimation and 128, 64, 128 for protective behaviour respectively.

Bi-directional LSTM Based on traditional LSTMs, Bidirectional LSTMs were developed to improve the estimation quality by two way sequence analysis. One way to compare LSTM with Bidirectional LSTM is that in the latter, the first LSTM layer is duplicated. Then two LSTM layers are concatenated side by side so that the output of the first layer will be reversed and fed to the second layer. The network in our case, contains one 1024 dimensional bi-LSTM layer

followed by two 512 and 256-dimensional LSTM layers for pain intensity estimation.

Attention-LSTM The attention-LSTM architecture used in protective behaviour estimation task and contains three layer LSTMs (128, 64, 128 dimensional) followed by a dense attention layer (64 dimensional) and a dense layer (64 dimensional) before joint estimation for the same task.

LSTM-DNN The LSTM-DNN architecture has three-layer LSTMs (128, 64, 128 dimensional) followed by three dense layers having 512, 1024, 128 nodes.

3.2 Joint training with exercise type

In this work, both pain intensity and protective behaviour are estimated on the basis of body part movement. The movement in the body parts occurs when the participants performs the instructed exercises. The hypothesis behind joint training of the network considering exercise type as an additional label is that there is a correlation between exercise labels and tasks performed. Based on our assumption, we calculate the correlation coefficient and plot a graph between both tasks and exercise labels (Fig. 1). The plots in Fig. 1 show the Spearman’s Rank-Order correlation between the types of exercise undertaken and pain intensity estimation, protective behaviour detection, respectively. From Table 1, it is observed that while the participants perform exercise (for example one leg stand, bend etc.) they are feeling low/high pain and protective behaviour. Thus, there is a positive correlation of these exercises with LP, HP, P. The exception happens for standing still and walking. The reason behind this is that during these two activities the chronic back pain does not occur. There is another category in the data, termed as other, where no activity is performed. Thus, in this case

also, pain usually does not occur. Additionally, please note that the correlation results also include the statistics from healthy patients as well where ideally there is no correlation between any exercise and pain levels, protective behaviour.

Further, we jointly trained all the networks with exercise type along with ground truth labels which enhances the performance for both the tasks. More information on the dataset and labels follows in the Sect. 4.1.

3.3 Anomaly detection framework

As occurrence of either chronic pain or protective behaviour is infrequent event with respect to total number of frames, it is difficult to extract the patterns from the imbalanced data. To handle this problem, presence of exhibition of pain and protective behaviour is considered as an anomaly detection problem in a LSTM-DNN framework. For simplicity, we consider non-protective and protective as ‘normal’ and ‘abnormal’ activities, respectively. Similarly for the pain estimation, we consider no-pain and pain categories as ‘normal’ and ‘abnormal’ activity, respectively. Further following steps are taken:

1. A LSTM-DNN framework is jointly trained for a binary classification task.
2. The anomaly score is calculated for each frames in a segment. Here, the overall loss of the LSTM-DNN network is considered as the anomaly score.
3. The average of the anomaly scores over a segment is calculated to get segment wise anomaly score.
4. An empirical threshold is applied on the anomaly score to predict abnormal activity.

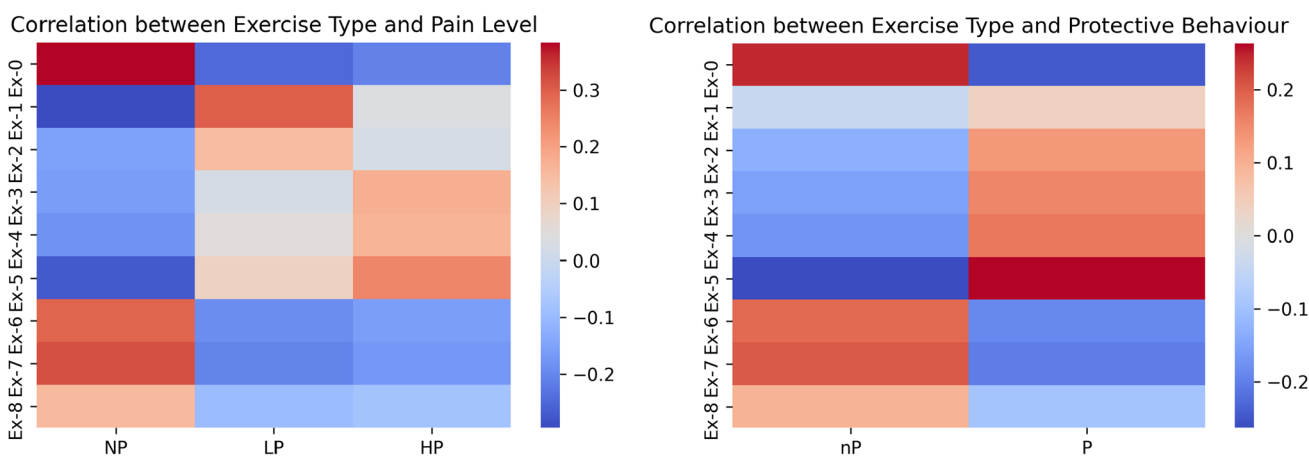


Fig. 1 Correlation plots representing the correlation between exercise and the task. The left plot represents the correlation between exercise and pain intensity level. The right plot represents the correlation between exercise and protective behaviour. Here, Ex-0, Ex-1, Ex-2, Ex-

3, Ex-4, Ex-5, Ex-6, Ex-7, Ex-8 represent different exercises including others, one-leg-stand, reach-forward, bend, sit-to-stand, stand-to-sit, sitting still, standing still, and walking. NP, LP, HP, nP and P represents no pain, low pain, high pain, non-protective and protective, respectively

Table 1 The correlation and corresponding p values between exercise type and pain behaviour labels

Exercise type	Correlation matrix					p values				
	NP	LP	HP	nP	P	NP	LP	HP	nP	P
One-leg-stand	-0.2945	0.3014	0.0370	-0.0404	0.0404	0.002988	<0.00001	<0.00001	0.00692	<0.00001
Reach forward	-0.1520	0.1479	0.0276	-0.1337	0.1337	<0.00001	<0.00001	<0.00001	<0.00001	<0.00001
Bend	-0.1602	0.0236	0.1753	-0.1561	0.1561	0.005978	<0.00001	<0.00001	<0.00001	<0.00001
Sit-to-stand	-0.1784	0.0527	0.1661	-0.1714	0.1714	<0.00001	<0.00001	<0.00001	0.00053	<0.00001
Stand-to-sit	-0.2726	0.0897	0.2435	-0.2633	0.2633	<0.00001	<0.00001	<0.00001	<0.00001	<0.00001
Sitting still	0.2930	-0.1901	-0.1583	0.1875	-0.1875	<0.00001	<0.00001	<0.00001	<0.00001	0.00173
Standing still	0.3178	-0.2063	-0.1716	0.2033	-0.2033	<0.00001	<0.00001	<0.00001	<0.00001	<0.00001
Walking	0.1510	-0.0980	-0.082	0.0966	-0.0966	<0.00001	<0.00001	<0.00001	<0.00001	<0.00001
Other	0.3837	-0.2489	-0.2073	0.2455	-0.2455	<0.00001	<0.00001	<0.00001	<0.00001	<0.00001

Here, NP, LP, HP, nP and P represents no pain, low pain, high pain, non-protective and protective, respectively

3.4 Hierarchical classification for pain estimation

A hierarchical pain intensity estimation method is employed. In the first step, ‘no-pain’ and ‘pain’ classes are classified via anomaly detection on the basis of LSTM-DNN’s loss prediction. Further, estimation of low and high pain is performed via a LSTM-DNN based pain classifier where the pain classifier is trained on the low pain and high pain labeled data. Further, we treated this activity as anomaly detection problem. The overview of this network is shown in Fig. 2.

3.4.1 Protective to pain hierarchical classification

Patients having CLBP are mostly protective in nature. Based on this hypothesis, we propose a two stage network architecture, which predicts protective behaviour in first stage followed by pain intensity estimation in the second stage. The overview of the proposed PLAAN network for both pain intensity and protective behaviour prediction is shown in Fig. 2.

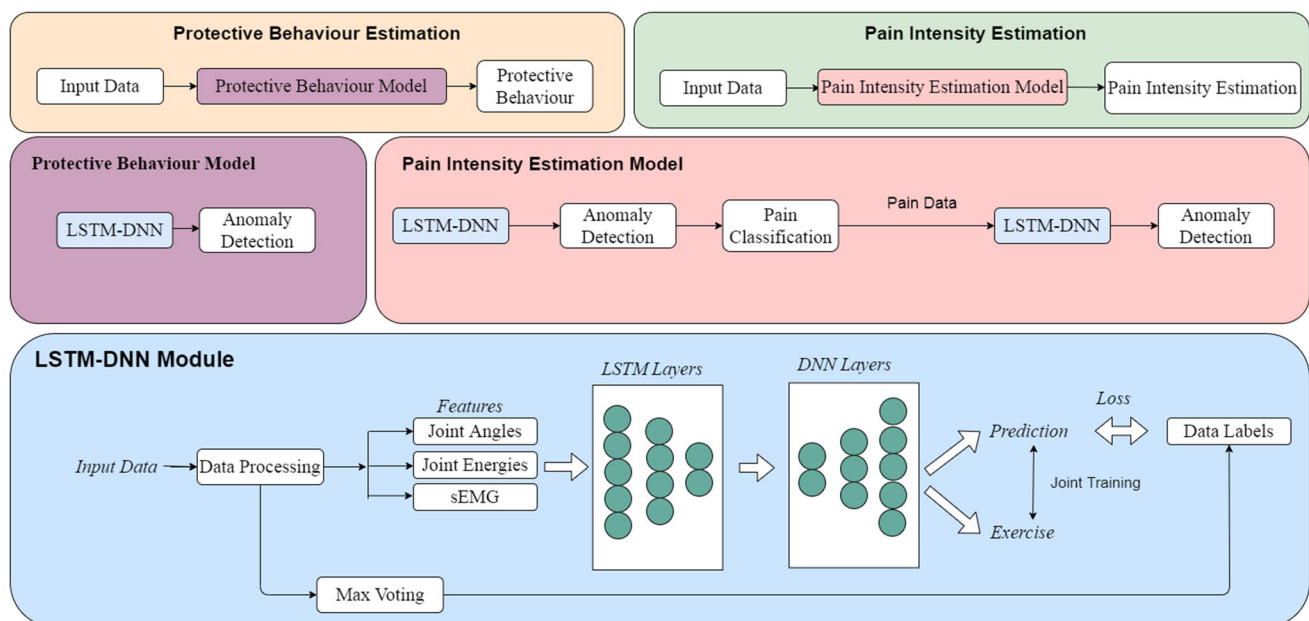


Fig. 2 Overview of the proposed PLAAN network. Here, data processing refers to the window based temporal segment selection process. For the estimation task, either pain intensity or protective behaviour is used

4 Experiments

4.1 Emopain dataset

Aung et al. [9] curated the Emopain dataset for research purposes and released the dataset through the movement challenge [39]. The dataset comprises of 30 participants whose data is partitioned randomly into three sets: training set (10 CLBP participants, 6 healthy participants), validation set (4 CLBP participants, 3 healthy participants) and test set (4 CLBP participants, 3 healthy participants). This dataset has full-body motion capture (MoCap) data captured with 18 microelectromechanical (MEMS) based Inertial Measurement Units (IMU). The placement of 18 sensors is as follow: twelve sensors were distributed on 4 limbs evenly; one on the hip, one on the centre of the torso, a pair on both ends of the shoulder, one around the neck and the last one is on the head. Each sensor was connected with Velcro attachment straps to minimize subject’s uncomfortable that could bring inaccuracy to the motion data. While the participant was performing the required exercises, each sensor recorded a sequence of 3D Euler angles, which was used in calculating the postural information of 26 anatomical joints in 3D space. The surface electromyography (sEMG) data were collected from 4 locations on the back, two of them were at the upper back, while the other two were at the lower back. Each video recorded the body movement data taken from a participant instance doing a series of exercises for one specific difficulty.

Recorded activities included: *One-leg-stand*, *Stand-to-sit*, *Sit-to-stand*, *Reach forward* and *Bend*—typical everyday activities that are generally challenging for subjects with CLBP. This paper focuses mainly on two tasks corresponding to the challenge. The first task is **pain intensity estimation**; that is to determine if a participant has chronic pain along with the estimation of the intensity of the pain (i.e. low-level pain or high-level pain) or if s(h)e is a healthy control participant. The second task is **protective behaviour identification**; based on the exercise performance of a participant, presence or absence of any protective behaviour has to be identified.

Dataset statistics In Emopain dataset, for body movement data, the training and validation sets ratio is approximately 2:1. There are 23 video clips in the training set and 12 video clips in the validation set. Please note that few participants

have two video clips. The ground truth labels for the task of pain intensity estimation has three classes; No pain, low pain and high pain and the task of protective behaviour estimation is a binary class problem. Given that the ground truth of pain intensity, exercise type, and the existence of protective behaviour are all frame-wise, the label distributions are as reported in Table 2.

4.2 Data pre-processing

We perform the following pre-processing steps before training: as the pain recognition network needs to conduct an exercise-wise protocol, we re-organize the movement dataset based on the exercise types and ignore the unlabeled frames. We split the data in segments of length n , where n is the number of frames in each segment (in our experiments, $n = 180$). In total, there are 141 exercise instances in the training set and 108 instances in the validation set. The dataset has frame wise labels (Pain labels—0: Healthy, 1: Low-level pain, 2: High-level pain, -1: Not reported (only for the patients). Protective behavior labels—0: Not protective, 1: Protective.) and to label a segment, we compute majority voting over the labels of the frames in the segment. Similar to Wang et al. [11], only the motion capture data of the EmoPain dataset is used. From the protective behaviour-experiments, a sliding window of length=3s and overlapping ratio=75% is used for each activity type in the data instance. Zero-padding is applied, when the window is beyond the end of a given activity type. All of the activities are considered irrespective of the pain level. Majority voting technique is applied to pool the labels at the window level. If there is greater than 50% voting belongs to a class, we assign the window label with that.

Imbalanced data After the pre-processing of data, we noticed that for exercise-wise segments, the label distribution between classes is biased towards to non-CLBP subjects.

In the segment-level, 61 segments have no pain; 44 pain level 1; 36 pain level 2 in the training set, and 73 pain level 0; 30 pain level 1; 5 pain level 2 in the validation set.

In the labels for protective behaviour, we have 10,370 non-protective behaviour labels; 6280 protective behaviour labels in the training set, and 5100 non-protective behaviour labels; 3517 protective behaviour labels in the validation set. Given above, it is reasonable for us to choose evaluation metrics

Table 2 Class-wise frame distribution in Emopain Dataset for training and validation set

Class-wise frame distribution	Pain intensity estimation			Protective behaviour	
	No Pain	Low pain	High pain	Non protective	Protective
Train (in %)	24.02	11.32	8.53	81.30	18.70
Validation (in %)	83.78	13.55	2.67	93.23	6.77

For pain intensity estimation, a large amount of frames (approx 56.13%) are not annotated

Table 3 The results of proposed method on validation set where LSTM-DNN is the best performing network

Pain estimation				Protective behaviour			
Networks	Accuracy (in %)	MCC	F1 Score	Networks	Accuracy (in %)	MCC	F1 score
LSTM	67.59	0.51	NP: 0.66 LP: 0.04 HP: 0.00	LSTM	92.77	0.86	nP: 0.92 P: 0.00
bi-LSTM	70.37	0.51	NP: 0.72 LP: 0.02 HP: 0.00	Attention-LSTM	93.33	0.87	nP: 0.93 P: 0.00
LSTM-DNN	80.00	0.69	NP: 0.79 LP: 0.05 HP: 0.00	LSTM-DNN	94.08	0.87	nP: 0.93 P: 0.00
LSTM-DNN (SMOTE)	61.11	0.03	NP: 0.74 LP: 0.26 HP: 0.00	LSTM-DNN (SMOTE)	9.00	0.00	nP: 0.17 P: 0.00
LSTM-DNN (Under-sampling)	55.56	0.00	NP: 0.74 LP: 0.13 HP: 0.00	LSTM-DNN (Under-sampling)	90.74	0.82	nP: 0.90 P: 0.00
LSTM-DNN (MIL)	78.15	0.67	NP: 0.76 LP: 0.23 HP: 0.01	LSTM-DNN (MIL)	91.70	0.86	nP: 0.87 P: 0.09
LSTM-DNN (Anomaly)	85.26	0.73	NP: 0.89 LP: 0.41 HP: 0.29	LSTM-DNN (Anomaly)	87.00	0.90	nP: 0.93 P: 0.50

NP no pain, *LP* low pain, *HP* high pain, *nP* non-protective, *P* protective

that can reflect the network's performance among all classes, which we will introduce in the later section.

4.3 Experimental setup

For experimental purpose, we use the Keras deep learning library with the TensorFlow backend. With the help of joint training, our networks' ability to discriminate all training samples gets enhanced. We performed the following experiments on the pain data. (1) Basic LSTM, (2) bi-LSTM, (3) Attention-LSTM, (4) LSTM-DNN, (5) Joint training of task with respect to exercise performed, (6) Pain and Protective Behaviour estimation as a anomaly detection task, (7) Hierarchical pain intensity estimation, (8) Cascaded estimation of protective behaviour and pain intensity estimation.

The comparison between joint training and single training label for pain intensity estimation is reported in Table 4. All trained networks' performances are reported in Table 3.

4.4 Evaluation metric

In the network's training process, we specified the exercise type as an additional label. Next, we jointly trained the PLAAN network with pain level label for pain intensity estimation, and protective behaviour label for protective behaviour detection, respectively. Even so, we only evaluated networks based on the output for either pain level or protective behaviour.

In this work, we use accuracy to evaluate networks' overall performance. Additionally, given that the dataset is imbalanced and our PLAAN network effectively discriminate the difference between control subjects and people with CLBP, we also implement F1 score and Matthews Correlation Coefficient (MCC) in measuring the predictive results among all classes.

4.5 Implementation details

Training the basic LSTM network During experiments, different network's output were recorded to analyze gradual improvements. We first used traditional LSTM with only one hidden layer, however the result was not significant. Then two more LSTM layers were added to create a stacked LSTM. An LSTM layer can return a sequence output then pipe it to the next LSTM layer. With more hidden layers, an in-depth abstraction of the learned representation can be achieved as stated by Hermans et al. [40]. Besides, we used Adam optimizer [41] with a learning rate of 0.001 instead of SGD [42]. Adam optimization algorithm as the improved combination of RMSprop [43] and Momentum optimization algorithm [44] is less affected when the gradient is re-scaling. We have used categorical cross-entropy as loss function for this clas-

sification problem. Furthermore, we trained the network for 100 epochs with batch size 128.

Training the bidirectional LSTM network Similar to the basic LSTM network, we used Adam optimizer with a learning rate of 0.001 and also used categorical cross-entropy as loss function. We trained the network for 100 epochs with batch size 128. To improve the performance of basic LSTM network, bi-LSTM layers are used.

Training the attention-LSTM network For training attention-LSTM, we use SGD optimizer with learning rate 0.01 with momentum 0.9 and learning rate decay $1e-6$ per epoch. We used categorical cross-entropy as loss function. We trained the network for 100 epochs with batch size = 32.

Training the LSTM-DNN network Similar to the attention-LSTM network, we use the SGD optimizer with learning rate 0.01 with momentum 0.9 and learning rate decay $1e-6$ per epoch. We used the categorical cross-entropy as loss function. We trained the network for 100 epochs having batch size 32. We implemented LSTM-DNN network architecture (Fig. 2) for both pain recognition and protective behaviour classification.

5 Results

5.1 Network performance

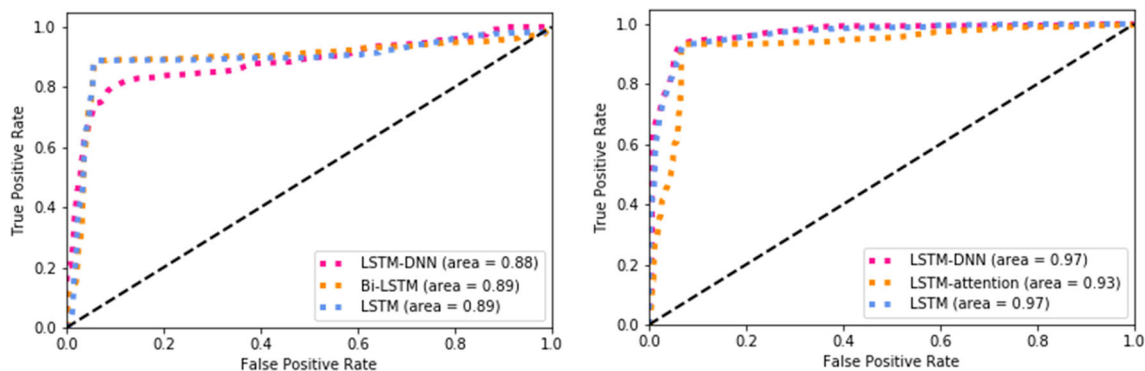
(1) *Experiment with LSTM Variants* We experimented with different variants of LSTM network architectures to get an overview of the performance. For pain estimation task, we achieved 67.6%, 70.36% and 80.00% accuracy on the validation set for LSTM, bi-LSTM and LSTM-DNN respectively. bi-LSTM improved the performance by approximately 3%. Further with LSTM-DNN network the accuracy improved from 70.36 to 80.00%.

Similarly, for protective behaviour estimation, we achieved 92.77%, 93.33% and 94.08% accuracy on the validation set for LSTM, Attention-LSTM and LSTM-DNN respectively. Attention-LSTM improved the LSTM's performance approximately 1%. Further with LSTM-DNN network the accuracy is improved slightly from 93.33 to 94.08%. LSTM-DNN attained an accuracy of 80.00% in the former task and the accuracy of 94.08% in the later task. The relative performance comparison of different networks for both tasks is depicted in Fig. 3 and Table 3.

(2) *Performance on test set* The results on the test set indicate that our network is biased toward healthy patients. For pain intensity estimation, the test accuracy is 45.45% (F1 score of healthy, low pain and high pain patients are 0.64, 0 and 0 respectively) which indicates that the presented network needs to address the unbalance in the data. Our networks perform better than the given baselines 35% with KNN and 7% with SVM. On the other hand, the accuracy for pro-

Table 4 Comparison of the task specific training with joint training in terms of accuracy and Matthews correlation coefficient

Pain estimation			Protective behaviour		
LSTM	Accuracy (in %)	MCC	LSTM	Accuracy (in %)	MCC
Pain	64.82	0.45	Protective	89.00	0.83
Pain + Exercise	67.59	0.51	Protective + Exercise	92.77	0.86

**Fig. 3** ROC curves of DNN-LSTM networks for pain intensity (left) and protective behaviour estimation (right), respectively

tective behaviour estimation with the proposed network is 89.18% (F1 score of non-protective and protective classes are 0.94 and 0.0053 respectively). Our network performs better than the baseline network 82.80% with stacked-LSTM. *Please note that we can not estimate the test results of the anomaly detection framework as the submission protocol is different.* In test phase, our anomaly detection framework follow the following steps: we train an LSTM-DNN based encoder-decoder module with MSE as reconstruction loss and cross entropy for joint training. For test results, we compute the MSE between the original and reconstructed features.

(3) *Data balancing techniques* As the data is largely biased on negative classes, we experimented with two data sampling techniques: k-mean clustering based SMOTE and manual data under-sampling.

In case of k-mean based SMOTE [46], the minor class data interpolation is performed on the basis of three steps: clustering, filtering, and oversampling. In the clustering step, k-means clustering is performed to identify different classes. The filtering step selects the classes for which oversampling is required. In the next step, the synthetic samples are generated and sparsely distributed throughout the selected region. The other technique that we experimented with was under-sampling the major class by random sampling method.

We observe that the results of the minority classes slightly improve for pain intensity estimation. However, since the validation set lacks protective behaviour instances, the balancing techniques do not work as expected in protective behaviour detection. The results are shown in Table 3.

(4) *Multiple instance learning* Further, to get the advantage of both skewed and balanced data-trained networks, we

conduct an experiment with a three channels based multiple instance learning framework. Each of the three channels consist of a LSTM-DNN network trained on positively skewed, negatively skewed and balanced data. We generate positive and negative labeled data by organizing the positive and negative class ratio 3:1 (both cases). We use the SMOTE technique to balance the data. Further, these three channel features are concatenated to get the overall performance. The results are shown in Table 3. From the table, it is observed that the overall accuracy increases but class-wise F1 score does not improve significantly.

(5) *Anomaly detection* The main problem with the pain estimation and protective behaviour estimation tasks are imbalanced data. Although we attempted to remove this issue via MIL and dataset balancing techniques, the results improved slightly. Thus, we use anomaly detection framework to handle the issue. The results are shown in Table 3. Figure 4 provide the probability distribution of the anomaly detection network where the red part represents the outliers. Further, one way ANOVA test is performed on the models is to calculate the statistical significance of the models. The p-values of the LSTM-DNN anomaly detection models for pain level and protective behaviour estimation are 0.003 and 0.001, respectively. The p-values of the models are <0.05, which indicates that the results are statistically significant.

(6) *Hierarchical classification* We estimate pain intensity of a patient via hierarchical classification. In the first stage, we estimate whether the participant have pain or not. In the second stage, we estimate the low or high pain corresponding to the ‘pain’ participants (detected in first stage). The results are shown in Table 3.

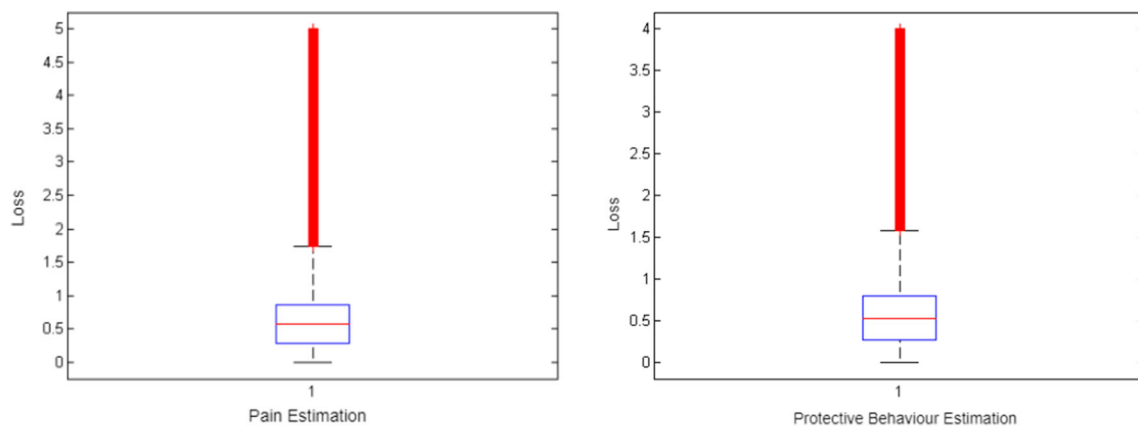


Fig. 4 Box-plot representing the probability distribution of the LSTM-DNN network estimation for pain estimation and protective behaviour detection. The red part represents outliers. Left and right plot represents the pain level and protective behaviour probabilities

Table 5 Performance comparison with state-of-the-art methods for pain intensity estimation

Pain intensity estimation				
Method	Accuracy (in %)	F1-NP	F1-LP	F1-HP
Baseline [39]	7.00	0.00	0.14	0.00
Yuan et al. [30]	45.61	0.40	0.00	0.00
Uddin et al. [29]	95.00	1.00	0.87	0.49
Haider et al. [45]	43.00	0.45	0.41	0.42
PLAAN	85.26	0.89	0.41	0.29

Here, F1-NP, F1-LP and F1-HP refer to F1 score corresponding to the no pain, low pain and high pain categories, respectively

Table 6 Performance comparison with state-of-the-art methods for protective behaviour estimation

Protective behaviour estimation			
Method	Accuracy (in %)	F1-nP	F1-P
Baseline [39]	46.36	0.96	–
Uddin et al. [29]	93.00	0.96	0.48
Yuan et al. [30]	85.00	0.91	0.26
Haider et al. [45]	61.21	0.96	0.25
PLAAN	87.00	0.93	0.50

Here, F1-nP and F1-P refer to F1 score corresponding to the non protective and protective behaviour respectively

Table 7 The results of Leave One Subject Out protocol on the validation set

Validation accuracy (in %)			
Method	KNN [39]	SVM [39]	Ours
LOSO	37.00	44.00	54.60

(7) *Leave one subject out* We also computed the results with leave one subject out cross validation protocol to further validate the proposed method. Segment-wise average score for each subjects were computed. The LOSO method gives 54.6% validation accuracy for pain estimation as compared to the SVM (44%) and KNN (37%). The results are compared in Table 7.

(8) *Comparison with state-of-the-art method* We compare with state-of-the-art methods for pain intensity estimation (Table 5) and protective behaviour (Table 6). From Tables 5 and 6, our method performs better than baseline [30,39]. With our PLAAN framework, class-wise F1-score improves significantly. In these tables, we compare our results with baseline [29,30,39]. For pain intensity estimation, the baseline method [39] mainly use hand-crafted features (e.g. range of joint angle, max/min/mean speed, and range of muscle activity) to capture the dynamics of each data instances. Further, these features are classified via a support vector machine having gaussian kernel. Yuan et al. [30] uses data augmentation, including normalized gaussian noise and creating new data instances by random selection to balance the training set. An autoencoder LSTM is used to decrease the dimension of the raw data, while an attention mechanism is used to extract more discriminative features for pain intensity estimation. Uddin et al. [29] uses the protective behaviour probability and the it's feature set to estimate the pain level. This method fuses three random forest-based models and two XGBoost of different feature subsets at the decision stage to balance the performance in three pain level classes. Our proposed model performs better than baseline [30,39]. *Our proposed method have comparable performance with respect to [29] considering it's model's computational complexity. Five models are fused to infer the pain intensity as compared to our anomaly detection procedure. Additionally, the inference time for pain behaviour prediction with PLAAN will be relatively lower as compared to Uddin et al. [29].*

Table 8 Comparison of performance when pain is estimated as a consequence of protective behaviour

Network	Accuracy (in %)	MCC	F1 Score		
			No pain	Low pain	High pain
LSTM-DNN	80.00	0.69	0.79	0.05	0.00
LSTM-DNN (Protective + Pain)	81.26	0.71	0.84	0.40	0.19

Table 9 Comparison of performance when different segment duration is chosen

Segment length	Accuracy(in %)
100	67.59
200	67.59
300	67.59
400	67.59
500	68.52
600	67.59
700	67.59
800	68.52
900	67.59
1000	67.59

5.2 Ablation study

(1) *Input segment duration* As the training is performed segment-wise, we conduct an ablation study regarding the trade-off between input segment duration and overall accuracy. For this experiment, we use the LSTM-DNN network for pain intensity estimation on a subset of data to observe the network performance. The results are depicted in Table 9. From the Table 9, it is observed that segment duration does not affect the performance except 500 and 800.

(2) *Protective behaviour to pain estimation* We also conducted an experiment to observe the effect of protective behaviour estimation followed by pain estimation. The rationale behind this is that—‘patients having chronic lower back pain show protective behaviour’. For this study, we train a LSTM-DNN network on the protective behaviour data and use this as pain/no-pain classification. Quantitatively, 78.3% frames having high / low pain are considered in this classification. This indicates pain can be inferred from the protective behaviour statistics. Further, we classified low and high pain levels. The results are described in Table 8. We compared this experiment with LSTM-DNN network trained on the pain dataset. It is observed that the increment in overall accuracy is 1.26%. However, both the MCC and class-wise F1 score improved significantly.

(3) *Visualization of loss values* We plotted a boxplot (Fig. 4) with the loss values where red line represents the outlier frames. This boxplot indicates that we can treat this problem as anomaly detection problem.

6 Conclusion and future work

This paper presents a deep learning-based approach for chronic pain intensity and protective behaviour estimation from movement data. We explore the use of joint training with LSTM, Bi-LSTM, Attention-LSTM and LSTM-DNN networks. The overall experimental outcomes indicate that LSTM-DNN network performs better than the aforementioned networks. The baseline accuracy of pain intensity estimation provided by Egede et al. [39] is 37% based on leave-one-subject-out cross-validation and for protective behaviour classification, the baseline accuracy is 46.36% based on the hold-out validation. Our experiments show large improvement over the baseline methods and outperform the baseline on the validation set by an accuracy gap of 35.00% for pain intensity estimation and 47.72% for protective behaviour estimation, respectively. A possible future work would be experimenting with transfer learning techniques for sharing subject movements through neural network weights from MOCAP datasets to EmoPain dataset. The features are extracted from the body sensors which have inherent fixed structure due to the kinematics constraints of the human body. Therefore, it would be interesting to explore graph convolution network for the task.

Acknowledgements We are grateful to all the brave frontline workers who are working hard during this difficult COVID-19 situation. We also thank to Emopain challenge organizers for sharing the dataset with us. We also thank to anonymous reviewers for their insightful comments and helpful suggestions to improve the quality of this paper.

References

1. Harold M (1994) Part iii pain terms, a current list with definitions and notes on usage. In: Classification of chronic pain-descriptions of chronic pain syndromes and definitions of pain terms, pp 207–214
2. Breivik H, Collett B, Ventafridda V, Cohen R, Gallacher D (2006) Survey of chronic pain in Europe: prevalence, impact on daily life, and treatment. *Eur J Pain* 10(4):287
3. Turk DC, Melzack R (2011) Handbook of pain assessment. Guilford Press, New York
4. Gatchel RJ, McGeary DD, McGeary CA, Lippe B (2014) Interdisciplinary chronic pain management: past, present, and future. *Am Psychol* 69(2):119
5. Singh A, Piana S, Pollarolo D, Volpe G, Varni G, Tajadura-Jiménez A, Williams ACC, Camurri A, Bianchi-Berthouze N (2016) Go-with-the-flow: tracking, analysis and sonification of movement and

- breathing to build confidence in activity despite chronic pain. *Hum Comput Interact* 31(3–4):335–383
6. Keefe FJ, Smith S (2002) The assessment of pain behavior: implications for applied psychophysiology and future research directions. *Appl Psychophysiol Biofeedback* 27(2):117–127
 7. Singh A, Bianchi-Berthouze N, Williams ACC (2017) Supporting everyday function in chronic pain using wearable technology. In: *Proceedings of the 2017 CHI Conference on human factors in computing systems*, pp 3903–3915
 8. Hane Aung MS, Bianchi-Berthouze N, Watson P, Williams ACC (2014) Automatic recognition of fear-avoidance behavior in chronic pain physical rehabilitation. In: *Proceedings of the 8th international conference on pervasive computing technologies for healthcare*. ICST, pp 158–161
 9. Aung MSH, Kaltwang S, Romera-Paredes B, Martinez B, Singh A, Cella M, Valstar M, Meng H, Kemp A, Shafizadeh M et al (2015) The automatic detection of chronic pain-related expression: requirements, challenges and the multimodal emopain dataset. *IEEE Trans Affect Comput* 7(4):435–451
 10. Zhou J, Hong X, Su F, Zhao G (2016) Recurrent convolutional neural network regression for continuous pain intensity estimation in video. In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp 84–92
 11. Wang C, Peng M, Olugbade TA, Lane ND, Williams ACC, Bianchi-Berthouze N (2019) Learning temporal and bodily attention in protective movement behavior detection. In: *International conference on affective computing and intelligent interaction workshops and demos*, pp 324–330
 12. Wang C, Olugbade TA, Mathur A, Williams ACC, Lane ND, Bianchi-Berthouze N (2019) Automatic detection of protective behavior in chronic pain physical rehabilitation: A recurrent neural network approach. *arXiv preprint arXiv:1902.08990*
 13. Vlaeyen JWS, Morley S, Crombez G (2016) The experimental analysis of the interruptive, interfering, and identity-distorting effects of chronic pain. *Behav Res Ther* 86:23–34
 14. Keefe FJ, Block AR (1982) Development of an observation method for assessing pain behavior in chronic low back pain patients. *Behav Ther*
 15. Li Y, Ghosh S, Joshi J, Oviatt S (2020) Lstm-dnn based approach for pain intensity and protective behaviour prediction. 2020 15th IEEE international conference on automatic face and gesture recognition (FG 2020) (FG). IEEE Computer Society, Los Alamitos, pp 392–396
 16. Ashraf AB, Lucey S, Cohn JF, Chen T, Ambadar Z, Prkachin KM, Solomon PE (2009) The painful face-pain expression recognition using active appearance models. *Image Vis Comput* 27(12):1788–1796
 17. Cootes TF, Edwards GJ, Taylor CJ (2001) Active appearance models. *IEEE Trans Pattern Anal Mach Intell* 23(6):681–685
 18. Lucey P, Howlett J, Cohn J, Lucey S, Sridharan S, Ambadar Z (2008) Improving pain recognition through better utilisation of temporal information. In: *International conference on auditory-visual speech processing*, vol 2008. NIH Public Access, p 167
 19. Lucey P, Cohn JF, Prkachin KM, Solomon PE, Matthews I (2011) Painful data: the unbc-mcmaster shoulder pain expression archive database. In: *Face and gesture 2011*. IEEE, pp 57–64
 20. Sikka K, Dhall A, Bartlett M (2013) Weakly supervised pain localization using multiple instance learning. In: 2013 10th IEEE international conference and workshops on automatic face and gesture recognition (FG). IEEE, pp 1–8
 21. Hammal Z, Cohn JF (2012) Automatic detection of pain intensity. In: *Proceedings of the 14th ACM international conference on Multimodal interaction*, pp 47–52
 22. Kaltwang S, Rudovic O, Pantic M (2012) Continuous pain intensity estimation from facial expressions. In: *International symposium on visual computing*. Springer, pp 368–377
 23. Bellantonio M, Haque MA, Rodriguez P, Nasrollahi K, Telve T, Escalera S, Gonzalez J, Moeslund TB, Rasti P, Anbarjafari G (2016) Spatio-temporal pain recognition in CNN-based super-resolved facial images. In: *Video analytics. Face and facial expression recognition and audience measurement*. Springer, pp 151–162
 24. Rodriguez P, Cucurull G, Gonzalez J, Gonfaus JM, Nasrollahi K, Moeslund TB, Roca FX (2017) Deep pain: exploiting long short-term memory networks for facial expression classification. *IEEE Trans Cybern*
 25. Bargshady G, Soar J, Zhou X, Deo RC, Whittaker F, Wang H (2019) A joint deep neural network model for pain recognition from face. In: 2019 IEEE 4th international conference on computer and communication systems (ICCCS). IEEE, pp 52–56
 26. Wang C, Olugbade TA, Mathur A, Williams ACC, Lane ND, Bianchi-Berthouze N (2019) Recurrent network based automatic detection of chronic pain protective behavior using mocap and semg data. In: *Proceedings of the 23rd international symposium on wearable computers*, pp 225–230
 27. Olugbade TA, Hane Aung MS, Bianchi-Berthouze N, Marquardt N, Williams AC (2014) Bi-modal detection of painful reaching for chronic pain rehabilitation systems. In: *Proceedings of the 16th international conference on multimodal interaction*, pp 455–458
 28. Olugbade TA, Bianchi-Berthouze N, Marquardt N, Williams AC (2015) Pain level recognition using kinematics and muscle activity for physical rehabilitation in chronic pain. In: 2015 international conference on affective computing and intelligent interaction (ACII). IEEE, pp 243–249
 29. Uddin MT, Canavan S (2020) Multimodal multilevel fusion for sequential protective behavior detection and pain estimation
 30. Yuan X, Mahmoud M (2020) Alanet: Autoencoder-lstm for pain and protective behaviour detection
 31. Ravanbakhsh M, Nabi M, Sangineto E, Marcenaro L, Regazzoni C, Sebe N (2017) Abnormal event detection in videos using generative adversarial nets. In: 2017 IEEE international conference on image processing (ICIP). IEEE, pp 1577–1581
 32. Ribeiro M, Lazzaretti AE, Lopes HS (2018) A study of deep convolutional auto-encoders for anomaly detection in videos. *Pattern Recognit Lett* 105:13–22
 33. Mehta V, Dhall A, Pal S, Khan S (2020) Motion and region aware adversarial learning for fall detection with thermal imaging. *arXiv preprint arXiv:2004.08352*
 34. Chalapathy R, Chawla S (2019) Deep learning for anomaly detection: a survey. *arXiv preprint arXiv:1901.03407*
 35. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) SMOTE: synthetic minority over-sampling technique. *J Artif Intell Res* 16:321–357
 36. Egede J, Valstar M, Martinez B (2017) Fusing deep learned and hand-crafted features of appearance, shape, and dynamics for automatic pain estimation. In: 2017 12th IEEE international conference on automatic face & gesture recognition (FG 2017) IEEE, pp 689–696
 37. Greff K, Srivastava RK, Koutník J, Steunebrink BR, Schmidhuber J (2016) LSTM: a search space odyssey. *IEEE Trans Neural Netw Learn Syst* 28(10):2222–2232
 38. Graves A, Schmidhuber J (2005) Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Netw* 18(5–6):602–610
 39. Egede J, Olugbade T, Wang C, Song S, Berthouze N, Valstar M, Williams A, Meng H, Aung M, Lane N (2020) Emopain challenge 2020: multimodal pain evaluation from facial and bodily expressions. In: *IEEE international conference on automatic face and gesture recognition workshop*
 40. Hermans M, Schrauwen B (2013) Training and analysing deep recurrent neural networks. In: *Advances in neural information processing systems*, pp 190–198

41. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv preprint [arXiv:1412.6980](https://arxiv.org/abs/1412.6980)
42. Bottou L (2010) Large-scale machine learning with stochastic gradient descent. In: Proceedings of COMPSTAT'2010. Springer, pp 177–186
43. Tieleman T, Hinton G (2012) Lecture 6.5-rmsprop: divide the gradient by a running average of its recent magnitude. COURSERA Neural Netw Mach Learn 4(2):26–31
44. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. *Nature* 323(6088):533–536
45. Haider F, Albert P, Luz S (2020) Automatic recognition of low-back chronic pain level and protective movement behaviour using physical and muscle activity information. In: 2020 15th IEEE international conference on automatic face and gesture recognition (FG 2020) (FG). IEEE Computer Society, Los Alamitos, pp 415–419
46. F Last, G Douzas, Bação F (2017) Oversampling for imbalanced learning based on k-means and SMOTE. CoRR, abs/1711.00837

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.